ElectricFlow 9.0
Installation Guide

Electric Cloud, Inc.
125 South Market Street, Suite 400
San Jose, CA 95113
www.electric-cloud.com

Electric Cloud

ElectricFlow version 9.0

Copyright © 2002–2019 Electric Cloud, Inc. All rights reserved.

Published 8/30/2019 2:05:28 PM

Electric Cloud® believes the information in this publication is accurate as of its publication date. The information is subject to change without notice and does not represent a commitment from the vendor.

THE INFORMATION IN THIS PUBLICATION IS PROVIDED "AS IS." ELECTRIC CLOUD, INCORPORATED MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WITH RESPECT TO THE INFORMATION IN THIS PUBLICATION AND SPECIFICALLY DISCLAIMS IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Use, copying, and distribution of any ELECTRIC CLOUD software described in this publication requires an applicable software license.

Copyright protection includes all forms and matters of copyrightable material and information now allowed by statutory or judicial law or hereinafter granted, including without limitation, material generated from software programs displayed on the screen such as icons and screen display appearance.

The software and/or databases described in this document are furnished under a license agreement or nondisclosure agreement. The software and/or databases may be used or copied only in accordance with terms of the agreement. It is against the law to copy the software on any medium except as specifically allowed in the license or nondisclosure agreement.

Trademarks

Electric Cloud, ElectricAccelerator, ElectricAccelerator Huddle, ElectricCommander, ElectricFlow, ElectricFlow Deploy, ElectricFlow DevOps Foresight, ElectricFlow DevOps Insight, ElectricFlow Release, ElectricInsight, and Electric Make are registered trademarks or trademarks of Electric Cloud, Incorporated.

Most Electric Cloud products—ElectricAccelerator, ElectricAccelerator Huddle, ElectricCommander, ElectricFlow, ElectricFlow Deploy, ElectricFlow DevOps Foresight, ElectricFlow Release, ElectricInsight, and Electric Make—are commonly referred to by their "short names"—Accelerator, Huddle, Commander, Flow, Deploy, Foresight, Release, Insight, and eMake—throughout various types of Electric Cloud product-specific documentation.

All other trademarks used herein are the property of their respective owners.

Contents

Chapter 1: Introduction to ElectricFlow 1-1

Unique Functionality 1-1
Challenges Solved by ElectricFlow 1-3
Architecture 1-4
Local Configuration 1-4
Remote Database Configuration 1-5
Remote DevOps Insight Server Configuration 1-6
Remote Web Server Configuration 1-6
Clustered Configuration 1-7

Chapter 2: System Requirements and Supported Platforms 2-1

Supported Server Platforms 2-1
Supported Agent Platforms 2-2
"Pure" Agent Platforms 2-3
Proxy Agents for Other Platforms 2-7
Server and Agent Compatibility 2-8
Diffie-Hellman Key Size Incompatibility 2-8
Server and DevOps Insight Server Compatibility 2-8
Hardware Requirements 2-9
Browser Requirements 2-9
Java Requirements 2-9
Port Usage 2-10
Default Server Ports 2-10
Default ElectricFlow Services Ports 2-10
Default DevOps Insight Server Ports 2-11
Avoiding Port Conflicts 2-11
Database Requirements 2-11
Built-In Database 2-11
Supported Alternate Databases 2-12
Alternate Database Requirements 2-12
Database Sizing 2-12
Disk Usage 2-12
Server 2-12
Agents 2-12
Sizing Artifact Cache Directory Space on Resources 2-13
Repository Server 2-13

Sizing the Repository Backingstore 2-13
Logs 2-13
DevOps Insight Server 2-14
Memory Settings 2-15
Modifying Memory Settings for an ElectricFlow Server 2-15
Modifying Memory Settings for an ElectricFlow Agent 2-16
Modifying Memory Settings for a Containerized ElectricFlow Server or ElectricFlow Repository Server 2-16
Checksum Utility 2-17
Linux 2-17
Windows 2-17
macOS 2-17
Software Licenses 2-18

Chapter 3: Installing ElectricFlow 3-1

ElectricFlow Installer Files 3-1
Availability of Installers with a Non-Root/Non-sudo or Non-Administrator Mode 3-2
Choosing the Correct Installation Interface and Installer Option 3-3
User Interface Installation Process 3-3
Interactive Command-Line Installation Process (Linux Only) 3-5
Silent Unattended Installation 3-7
Non-Server Platform Agent Interface 3-7
Before You Install ElectricFlow 3-7
Linux and Windows ElectricFlow Installations 3-7
Linux ElectricFlow Installations 3-9
Remote ElectricFlow Web Server Installation Prerequisites 3-11
Requirements for Non-Root or Docker DevOps Insight Installations on Linux Platforms 3-12
Default Installation Directories 3-14
Graphical User Interface Installation Methods 3-14
Running an Express Server Graphical User Interface Installation 3-14
Running an Advanced Graphical User Interface Installation 3-23
Running an Express Agent Graphical User Interface Installation 3-40
Running an Express Agent Graphical User Interface Installation (Agent-Only Installer) 3-49
Running an Advanced Agent Graphical User Interface Installation (Agent-Only Installer) 3-57
Running a DevOps Insight Server Graphical User Interface Installation 3-67
Interactive Command-Line Installation Methods 3-87
Running an Express Server Command-Line Installation 3-87
Running an Advanced Command-Line Installation 3-89
Running an Express Agent Command-Line Installation 3-94
Running an Express Agent Command-Line Installation (Agent-Only Installer) 3-97
Running an Advanced Agent Command-Line Installation (Agent-Only Installer) 3-99
Running an Express Agent Command-Line Installation (Agent-Only Installer) When the Server Uses Registered and Concurrent Licenses 3-102
Running a DevOps Insight Server Interactive Command-Line Installation 3-104
Silent Unattended Installation Method 3-112

Running a Silent Install 3-112
Silent Install Arguments 3-113
Linux Silent Installation Examples 3-122
Windows Silent Installation Examples 3-126
Linux Repository Server Silent Installation 3-129
Windows Repository Server Silent Installation Example 3-129
Windows or Linux DevOps Insight Server Silent Unattended Installation Example 3-130
Non-Server Platform Installation Method for UNIX Agents 3-136
Interactive Command-Line Installation Method for UNIX or macOS Agents 3-136
Unattended (Silent) Installation Method for UNIX or macOS Agents 3-141
Installing or Upgrading Remote Agents 3-143
Prerequisites 3-143
Permissions for Installing or Upgrading Remote Agents 3-144
Installing Remote Agents Using the Web Interface 3-145
Installing Remote Agents Using the API 3-157
Upgrading Remote Agents Using the Web Interface 3-168
Upgrading Remote Agents Using the API 3-173
Moving the Artifact Repository in Linux 3-175
Moving the Artifact Repository in Windows 3-177
Connecting ElectricFlow to a Microsoft SQL Server Named Instance 3-179
Installing the MySQL JDBC Driver 3-181
Logging Into the ElectricFlow Web Interface 3-181

Chapter 4: Creating a Server Cluster for ElectricFlow or DevOps Insight 4-1

Benefits from Clustering 4-1
Architecture of an ElectricFlow Cluster 4-1
Resource, Agent, and Procedure Configuration Considerations 4-2
Default Local Resource Use 4-2
Unsupported Host 4-3
Separate Local Agents For Improved Performance 4-3
Pool Local Agents For Improved Reliability 4-3
Procedure Strategies 4-3
Agent Resource Strategies 4-3
Database Restriction 4-4
broker-data Directory Restriction 4-4
Software for Clustering 4-4
Apache Zookeeper 4-4
Load Balancer 4-4
Dependencies for Clustering 4-5
Configuring Clustering 4-5
Separating Agents from ElectricFlow Servers 4-6
Preparing Your Cluster Resources 4-6
Installing and Configuring a Load Balancer 4-7
ElectricFlow Server Load Balancer 4-7
Web Server Load Balancer 4-8

Installing ZooKeeper 4-9
Running ZooKeeper as a Service on Linux 4-10
Running ZooKeeper as a Service on Windows 4-10
Ensuring that ZooKeeper Can Locate Java 4-11
Verifying that ZooKeeper is in Standalone Mode 4-11
Verifying that ZooKeeper is Running 4-12
Exhibitor Software 4-12
Configuring a Multi-ZooKeeper Cluster 4-13
ZooKeeper Requires a Majority of Nodes to Be Up 4-13
Installing ElectricFlow Software 4-13
New ElectricFlow Installation for Reliability 4-14
New ElectricFlow Installation for Performance 4-15
Converting an Existing ElectricFlow Installation for Reliability 4-16
Converting an Existing ElectricFlow Installation for Performance 4-16
Configuring Repository Servers 4-17
Overall Steps for Configuring Repository Servers 4-17
Re-Creating a Deleted DATA_DIR/tmp Directory on an ElectricFlow Web Server 4-21
Running a Cluster in Clustered Mode 4-22
Running a Cluster in Single-Server Mode 4-24
Adding the Configuration to ZooKeeper 4-25
Uploading Configuration Files to ZooKeeper 4-25
Prerequisites 4-25
Location 4-26
Command 4-26
Import Files 4-27
Next Steps 4-27
Getting information on the ElectricFlow Server Cluster from ZooKeeper 4-28
Prerequisites 4-28
Locations 4-28
Command 4-28
Output 4-28
Configuring Web Server Properties 4-29
Configuring ElectricFlow Agents 4-30
Configuring the Cluster Workspace 4-31
Adding Trusted Agents to Clusters 4-31
Preparing Your Cluster Environment 4-32
Method 1 4-33
Method 2 4-34
Separating Agents from ElectricFlow Servers 4-34
Verifying ElectricFlow Services 4-35
Accessing ElectricFlow with Clustering 4-35
Health Check for the ElectricFlow Cluster 4-35
Additional Ways to Improve an ElectricFlow Cluster 4-35
Third-Party Software 4-36

ElectricFlow Components 4-36
Creating a DevOps Insight Server Cluster 4-36
Overall Steps for Creating a Typical DevOps Insight Cluster 4-36
DNS Issue and Publish Host Setting 4-38
Ensuring a Healthy Cluster Before Upgrade or Reconfiguration Operations 4-39
Upgrading from Version 8.4 and Earlier 4-40
Changing the Password for Secure Access to a DevOps Insight Cluster 4-40

Chapter 5: Configuring ElectricFlow 5-1

The Default Zone and Gateways to Remote Zones 5-1
Applying an Enterprise License Key 5-1
External Database Configuration 5-2
Database Interactions 5-2
Database User 5-3
Default Database Ports 5-3
MySQL Prerequisites 5-3
Configuring ElectricFlow to Use an Alternate Database 5-4
Setting the Database with the Web Interface 5-4
Setting the Database from a Command Line 5-5
Configuring Services Autostart for Non-Root/Non-sudo Linux Installations 5-11
Setting Up Services Autostart 5-11
Disabling Services Autostart 5-18
Universal Access to the Plugins Directory 5-21
Configuring Universal Access for a Network Location 5-21
Replicating the Plugin Directory on Remote Systems 5-24
Network Plugins Shares for High-Availability ElectricFlow Components 5-24
Configuring Single Sign-On 5-27
Prerequisites 5-27
Example Data for Kerberos Configuration 5-28
Configuring Kerberos with Active Directory 5-28
Configuring Web Browsers for Single Sign-On 5-31
Configuring ElectricFlow for Single Sign-On 5-39
End-User Login Flow for Single Sign-On 5-39
Configuring the Web and Repository Servers to Use IPv4 5-40
Configuring ElectricFlow Server-to-Agent Communication to Use Trusted HTTPS 5-41
Environment Proxy Server Configuration 5-45
Configuring Proxy Settings for Servers 5-45
Testing Server Proxy Settings 5-46
Configuring Proxy Agents 5-46
Increasing File Descriptors for Linux and Linux Docker Containers 5-47
Adjusting Swappiness on Linux 5-48

Chapter 6: Roadmap for Upgrading ElectricFlow 6-1

Single-Site Architecture 6-3
Remote Web Server Configuration 6-4

Clustered Configuration 6-5
Server Components in ElectricFlow 6-6

Chapter 7: Upgrading from ElectricFlow 5.x, 6.x, 7.x, or 8.x to 9.x 7-1

Configuration Settings Preserved After an Upgrade 7-1
Agent Configuration Settings 7-1
ElectricFlow Server Configuration Settings 7-3
Repository Server Configuration Settings 7-5
Web Server Configuration Settings 7-6
Built-In Database Configuration Settings 7-6
Use Cases 7-6
Preparing for Your Upgrade 7-7
Upgrade Testing 7-7
Backing Up Your Existing ElectricFlow Data 7-7
Upgrade Installer Preservation 7-8
MySQL Upgrades 7-8
Choosing the Correct Upgrade Method 7-8
User Interface Upgrade Method 7-10
Interactive Command-Line Upgrade Method 7-14
Silent (Unattended) Upgrade Method 7-15
Copying Repository Contents 7-16

Chapter 8: Upgrading from ElectricFlow 5.x, 6.x, 7.x, or 8.x to 9.x in a Clustered Environment 8-1

Configuration Settings Preserved After an Upgrade 8-1
Agent Configuration Settings 8-1
ElectricFlow Server Configuration Settings 8-3
Repository Server Configuration Settings 8-5
Web Server Configuration Settings 8-6
Built-In Database Configuration Settings 8-6
Use Cases 8-6
Upgrading Clusters 8-7
Preparing for Your Upgrade 8-8
Upgrade Testing 8-8
Backing Up Your Existing ElectricFlow Data 8-8
Upgrade Installer Preservation 8-10
MySQL Upgrades 8-10
Choosing the Correct Upgrade Method 8-10
User Interface Upgrade Method 8-12
Interactive Command-Line Upgrade Method 8-16
Silent (Unattended) Upgrade Method 8-17
Copying Repository Contents 8-18
Uploading Configuration Files to ZooKeeper if Needed 8-18

Chapter 9: Upgrading the ElectricFlow DevOps Insight Server 9-1

Before You Upgrade 9-1

Upgrading the ElectricFlow Server 9-1
Upgrading the DevOps Insight Server on a System with Other ElectricFlow Components 9-1
Preserving Non-DevOps Insight Custom Settings in the elasticsearch.yml File 9-1
User Interface Upgrade Method 9-1
Interactive Command-Line Upgrade Method 9-16
Silent (Unattended) Upgrade Method 9-22
Reconfiguring the DevOps Insight Server After the Upgrade 9-23
Configuring the DevOps Insight Server on the ElectricFlow Server 9-23
Checking the DevOps Insight Server Configuration on the ElectricFlow Server 9-23
Testing Connectivity and Authentication Between the DevOps Insight Server and the ElectricFlow Server 9-24

Chapter 10: Uninstalling ElectricFlow 10-1

Uninstalling ElectricFlow on Linux, UNIX, or macOS 10-1
Uninstalling ElectricFlow Using root or an Account with sudo Privileges 10-1
Uninstalling UNIX or macOS ElectricFlow Agents Using a Non-root Account 10-1
Uninstalling ElectricFlow on Windows 10-3
Uninstalling on Windows 2008 or Windows 7 10-3
Uninstalling the ElectricFlow DevOps Insight Server on Linux 10-3
Uninstalling the ElectricFlow DevOps Insight Server on Windows 10-4

Chapter 11: Configuring Disaster Recovery and Recovering from a Disaster 11-1

Disaster Recovery Environment Setup 11-1
Configurations and Settings 11-1
Disaster Recovery Environment Setup for DevOps Insight Server 11-7
Steps to Perform During a Disaster Recovery Failover 11-8
Disaster Recovery Failover Steps for a DevOps Insight Server 11-8
Server Maintenance 11-8

Chapter 12: Maintaining ElectricFlow 12-1

ElectricFlow Server Backups 12-1
Data Backup Methods 12-1
Preparing for a Backup 12-2
Backing Up an ElectricFlow Server 12-2
ElectricFlow Server Restores 12-3
Preparing for a Restore 12-3
Restoring Your ElectricFlow Server 12-3
Switching to an Alternate Database from the Built-In Database 12-10
Preventing Database Changes During the Export 12-10
Exporting and Importing Your Data 12-10
Switching from an Alternate Database to the Built-In Database 12-11
Maintaining DevOps Insight Server Data 12-11
Backing Up DevOps Insight Server Elasticsearch Data 12-11
Removing Old DevOps Insight Elasticsearch Data 12-12
Removing Incorrect DevOps Insight Elasticsearch Data 12-14
Apache Web Server or Agent Certificates 12-16

Generating a CA Request 12-16
Sending the CA Request 12-17
Installing the Signed Certificate 12-17
Using chkconfig 12-18
Starting and Stopping Servers and Agents Manually 12-18
Stopping the ElectricFlow Agent Service 12-19
Stopping All ElectricFlow Server Services 12-19
Stopping All DevOps Insight Services 12-20
Starting the ElectricFlow Agent Service 12-20
Starting All ElectricFlow Server Services 12-21
Starting All DevOps Insight Services 12-21
Collecting ElectricFlow Logs 12-22
Prerequisites and Limitations for ElectricFlow Log Collection 12-22
Collecting Logs by Using the Logs Collection Self-Service Catalog Item 12-22
Collecting Logs by Running the EC-FlowLogCollector Plugin Procedure Directly 12-28
Collecting Logs Manually 12-31
Web Interface Online Help System 12-32

Chapter 13: Troubleshooting an ElectricFlow Installation 13-1

ElectricFlow Server is Unresponsive and Displays an OutOfMemory Error 13-1
Description 13-1
Workaround 13-1
Windows PHP Does Not Handle Time Zones Correctly 13-2
Description 13-2
Workaround 13-2
ElectricFlow Self-Signed Server Certificate Fails Security Scan 13-2
Description 13-2
Workaround 13-3
ElectricFlow CA or Intermediate CA Server Certificate Expires 13-4
Linux Upgrade Breaks Symbolic Links 13-5
Description 13-5
Workaround 13-5

Chapter 14: Performing Agent-Only Installations 14-1

Graphical User Interface Installation Methods 14-1
Running an Express Agent Graphical User Interface Installation (Agent-Only Installer) 14-1
Running an Advanced Agent Graphical User Interface Installation (Agent-Only Installer) 14-8
Interactive Command-Line Installation Methods 14-18
Running an Express Agent Command-Line Installation 14-19
Running an Express Agent Command-Line Installation (Agent-Only Installer) 14-22
Running an Advanced Agent Command-Line Installation (Agent-Only Installer) 14-24
Running an Express Agent Command-Line Installation (Agent-Only Installer) When the Server Uses Registered and Concurrent Licenses 14-27
Non-Server Platform Installation Method for UNIX Agents 14-30
Interactive Command-Line Installation Method for UNIX or macOS Agents 14-30

Unattended (Silent) Installation Method for UNIX or macOS Agents 14-34

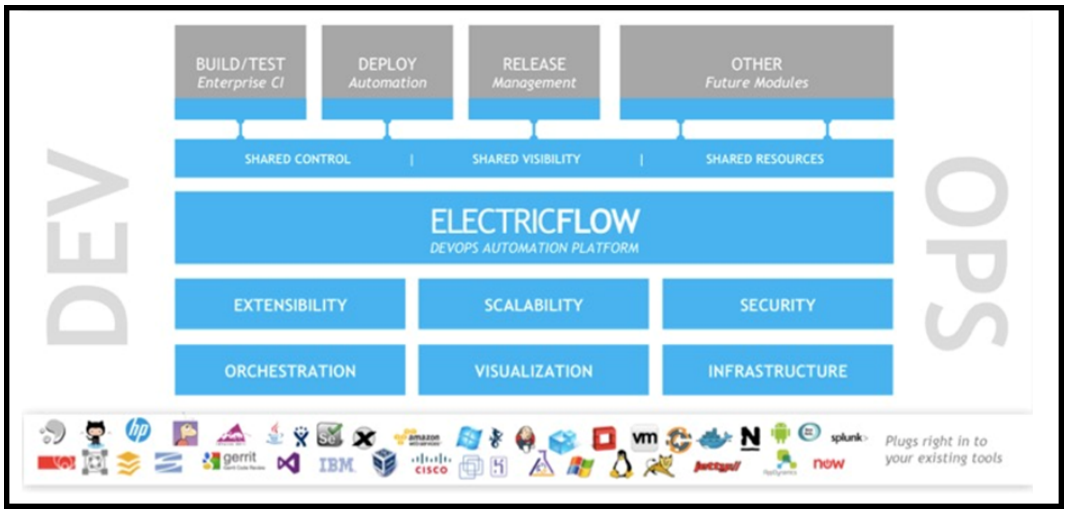
Chapter 1: Introduction to ElectricFlow

ElectricFlow® (including the ElectricFlow Platform, formerly known as ElectricCommander) accelerates the continuous delivery of software and makes software delivery processes more repeatable, visible, scalable, and efficient. It provides domain-specific capabilities to automate the build, test, package, deploy, and release processes across many delivery pipelines.

ElectricFlow is built on a powerful proven automation platform that natively integrates domain-specific capabilities for enterprise-level continuous delivery. The automation platform gives distributed DevOps teams shared control and visibility into infrastructure, tool chains, and processes. It accelerates and automates the software delivery process and enables agility, availability, predictability, and security across many build-test, deployment, and release pipelines.

The following diagram shows how ElectricFlow provides build/test, deploy, and release automation.

- ElectricFlow provides automation, management, and visibility of the build, test, deploy, and release processes by
    - Automating any workflows and pipelines.
    - Modeling and deploying one application for more than one use case.
    - Deploying all, some, or specific versions of artifacts in an application.
    - Keeping track of changes to tracked objects including applications, artifacts, jobs, resources, and workflows, referred to as Change Tracking.
    - Optimizing how resources are used in dynamic environments.
- ElectricFlow uses a process model to connect applications to environments.
- You create and manage resources, artifacts, projects, workflows, and procedures to support deployment and pipeline automation.



Unique Functionality

ElectricFlow is the most scalable solution on the market. Only ElectricFlow provides enterprise-class scalability for build and release management. It is easy to install and use on a simple build, yet scales to

support the largest and most complex build and test processes. The ElectricFlow multi-threaded Java server provides efficient synchronization even under high job volume.

Facilities provided by ElectricFlow:

- **Complete end-to-end software deployment solution**

  Automates standard deployment processes across your enterprise. You can select the components of the working applications in your software environment.

- **Workflow functionality**

  Use Workflows to design and manage processes at a higher level than individual jobs. Workflows allow you to combine procedures into processes to create build-test-deploy lifecycles (for example). A workflow contains states and transitions you define to provide complete control over your workflow process. The ElectricFlow Workflow feature allows you to define an unlimited range of large or small lifecycle combinations to meet your needs.

- **Continuous Integration Manager** (CI Manager)

  This feature provides a front-end user interface for creating, managing, and monitoring continuous integration builds. The CI Manager dashboard provides:

  - Visually see your running builds, build progress, build status, and historical build outcomes.
  - Easily accessed "Actions" to configure a continuous integration build.
  - Quick configuration of your preferred SCM system.
  - A project can contain any number of continuous integration builds, depending on the work you have already set up for your procedures/steps to perform.

- **Resource management**

  If a resource is over committed, ElectricFlow delays some jobs until others have finished with the resource. You can define pools of equivalent resources and ElectricFlow balances the load across the pool.

- **Access control**

  Users log into the system and ElectricFlow uses their information to control activities. Privileges can be set for individuals or groups to ensure the security you need.

- **Preflight Build functionality**

  Developers can build and test code changes in isolation on their local machines before those changes are committed to a production build.

- **Search, sort, and filter functions**

  Minimize the display of information that is of no interest to you, and quickly retrieve the information you need.

- **Detailed job information**

  ElectricFlow records a variety of information about each job. You can view jobs and see step run times, successes, and failures.

- **Email notifications**

  Get important information or data to individuals or groups immediately and on a regular basis for a particular job or a specific job aspect.

- **Powerful and flexible reporting facilities**

  Various statistics such as the number of compiles or test errors are collected after each step and recorded in the ElectricFlow database. A variety of reports can be generated from this information.

- **Artifact management**

  Use artifacts to improve performance across builds, provide better reusability of components, and improve cross-team collaboration with greater traceability. For example, instead of each developer repeatedly downloading third-party packages from external source, these components can be published and versioned as an artifact. A developer then simply retrieves a specific artifact version from a local repository, guaranteeing a consistent package from build to build.

- **ElectricFlow command-line tool**

  All ElectricFlow features are available from a command-line tool (ectool), a Perl API (ec-perl), and a web interface.

- **Plugin capability**

  ElectricFlow has an extensible UI which enables easy development of plugins. You can integrate with other tools, use custom dashboards, and create unique user experiences based on roles.

- **Workspaces**

  ElectricFlow creates a workspace for each job. A workspace is a disk area jobs can use for storage.

- **Data models based on properties**

  Properties are used to store job input data such as the source code branch to use for the build, to collect data during a job (such as number of errors or warnings), and to annotate the job after it completes (for example, a build has passed QA).

- **Zones and gateways**

  A zone (or top-level network) is a way to partition a collection of agents to secure them from use by other groups. A gateway is a secured connection between two zones when you want to share or transfer information to another zone. For example, you might want a developers zone and a test zone. The ElectricFlow server is a member of the default zone, created during ElectricFlow installation.

  > **Note:** The ElectricFlow server is a member of the `default` zone (created during ElectricFlow installation) and must be able to reach every remote zone via a gateway or a gateway chain. To ensure that the `default` zone can reach remote zones, do not rename it.

Challenges Solved by ElectricFlow

Traditional software build processes face the following challenges:

- Wasted time on script-intensive, manual, home-grown systems

  These systems are error prone, do not scale well, and have little or no management visibility or reporting.

- Multiple, disconnected build and test systems across locations

  Disconnected build and test systems result in redundant work and the inability to share/reuse code files across teams, making it painful to manage build and test data.

- Slow overall build and release cycles

  Slow cycle times directly impact release predictability and time-to-market.

ElectricFlow addresses these problems with a three-tier architecture, AJAX-powered web interface, and first-of-its-kind build and release analytic capabilities for reporting and compliance. With this solution, your developers, release engineers, build managers, QA teams, and managers gain:

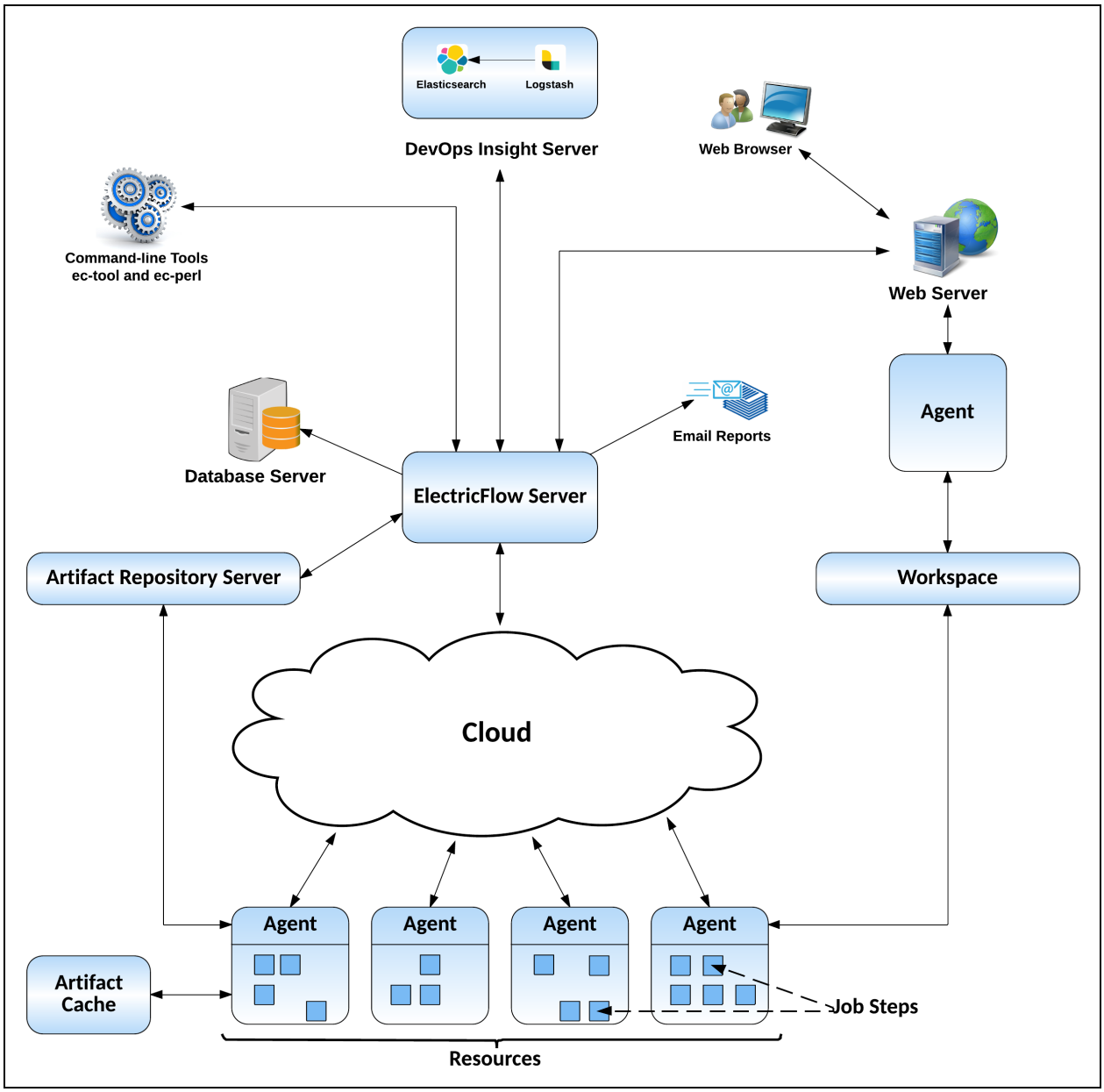
- A shared platform for disseminating best practices and reusing common procedures
- The ability to support geographically distributed teams
- Continuous integration and greater agility
- Faster throughput and more efficient hardware utilization
- Visibility and reporting for better project predictability
- Better software quality by integrating and validating against all target platforms and configurations

Architecture

ElectricFlow supports enterprise-scale software production. Based on a three-tier architecture, ElectricFlow scales to handle large, complex environments. ElectricFlow's multithreaded Java server provides efficient synchronization even under high job volume.

Local Configuration

The following diagram shows an ElectricFlow architecture configuration at a single site.



In the local configuration:

- The ElectricFlow server manages resources, issues commands, generates reports.
- An underlying database stores commands and metadata.
- Agents execute commands, monitor status, and collect results, in parallel across a cluster of servers for rapid throughput.

Remote Database Configuration

For a production environment, Electric Cloud recommends that you install the database on a separate machine from the ElectricFlow server to prevent performance issues. It is acceptable for the ElectricFlow server, web server, and repository server to reside on the same machine in a local

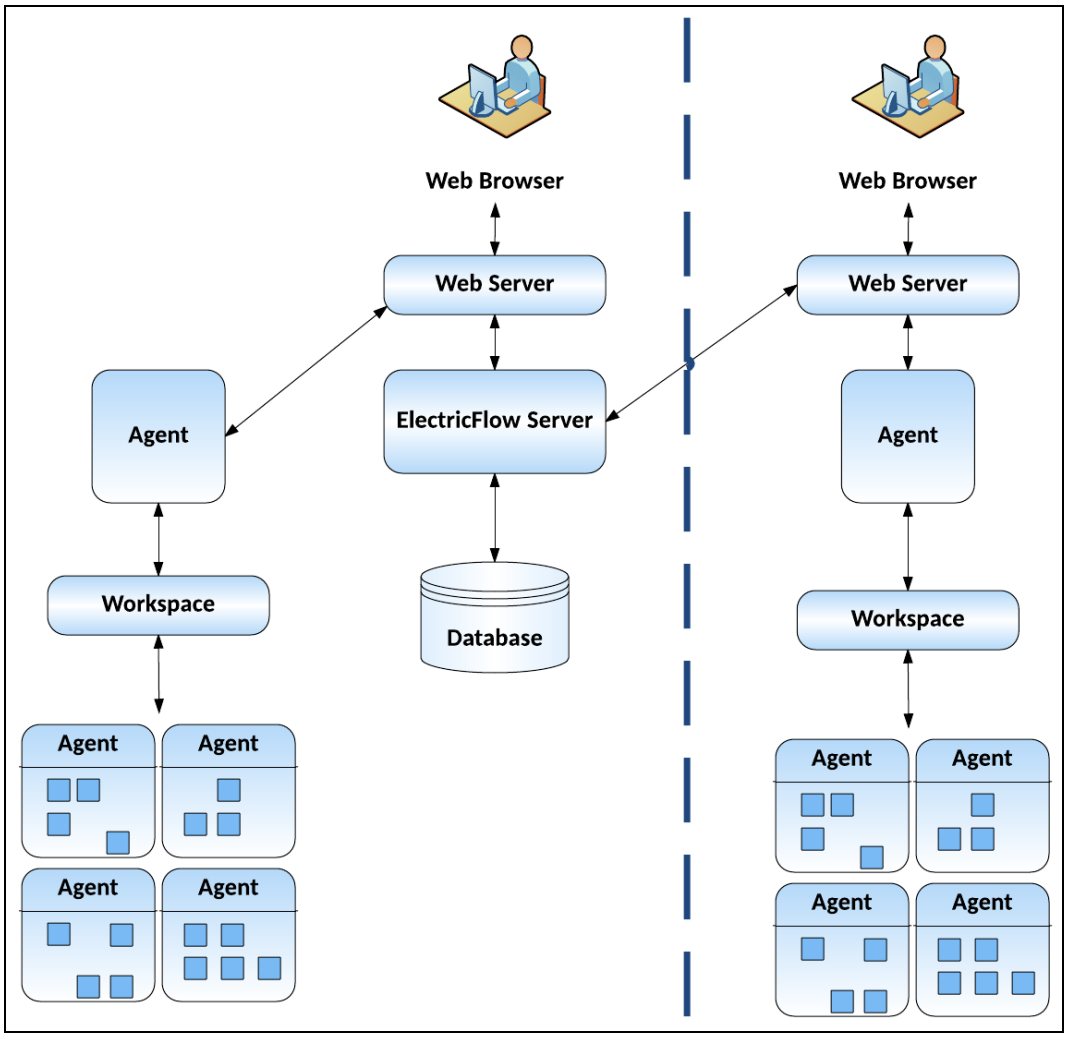
configuration, but not required. If you are only evaluating ElectricFlow, ElectricFlow, the database, the ElectricFlow server, the web server, and the repository server can reside on the same machine.

Remote DevOps Insight Server Configuration

For a production environment, Electric Cloud recommends that you install the DevOps Insight server on a system other than systems running other ElectricFlow components (such as the ElectricFlow server, web server, repository server, or agent). If you must install it on the same system (such as for testing or other nonproduction or trial-basis situations only), see Running the DevOps Insight Server on a System with Other ElectricFlow Components on page 3-8 for instructions.

Remote Web Server Configuration

The following diagram shows an example of a remote web server architecture configuration.



In this example remote web server configuration:

- There are web servers at each site
- The database and ElectricFlow server is located at your headquarters

- Proxy resources exist at each site

Benefits of a Remote Web Server Configuration

A remote web server configuration helps prevent network latency. If you have multiple sites, ElectricFlow can be configured in numerous ways to help you work more efficiently.

Central Web Server and a Remote Web Server at Each Site

You should consider installing multiple web servers for different locations in your organization to help handle user web traffic . ElectricFlow supports multiple workspaces, including those co-located on agents that use them. In this architecture, step log files are created locally so even the largest log files can be captured without a performance penalty.

You can view the step log files remotely from the web UI, but performance decreases if the files must be retrieved across the WAN. This means that remote users will experience the penalty when the web server retrieves the step log file contents and when the contents are sent back across the WAN to the browser.

To minimize these performance issues, install one central ElectricFlow server, and then install an ElectricFlow web server at each remote site. The remote web servers should be co-located with the remote agents and workspaces so remote users can log in through their local web server. Any operations initiated from the remote location, including running jobs, are completed by the central ElectricFlow server.

In this configuration, job data is retrieved from the central server when a remote user views the Job Details page. If the job is using a workspace at the remote user's site, the links to all step log files will refer to local paths.

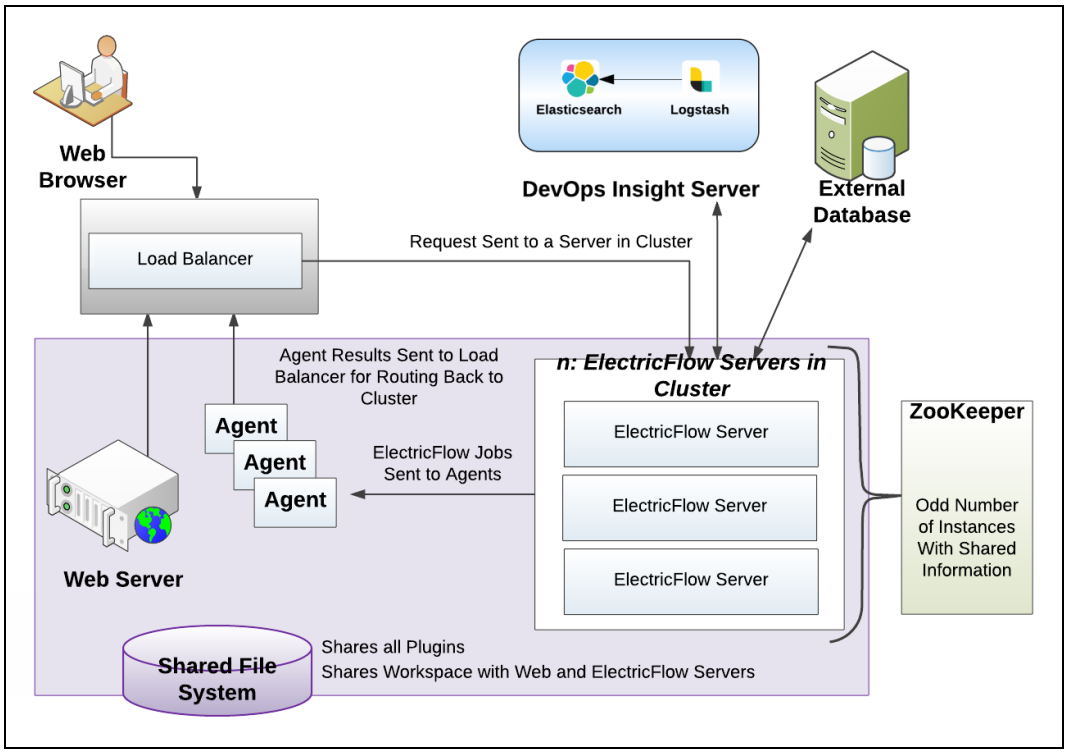
Also, in this configuration, the log files are accessed only by the remote web server's agent and not the ElectricFlow server. This eliminates both trips across the WAN, which improves performance. The ElectricFlow web server reads the log file locally (via its local agent) and then displays the page to the user whose browser is also on the same side of the WAN.

Prerequisites for Installing Remote Web Servers

For details about the remote web server prerequisites such as memory, agents, and centralized plugin directory access, see Remote ElectricFlow Web Server Installation Prerequisites on page 3-11.

Clustered Configuration

The following diagram shows an ElectricFlow clustered configuration.



You can also add horizontal scalability and high availability to your ElectricFlow environment by adding additional machines to create a clustered ElectricFlow configuration.

Benefits of a Clustered Configuration

A clustered ElectricFlow configuration has the following benefits:

- Add fault tolerance by re-routing jobs to running ElectricFlow servers
- Increase the supported number of simultaneous jobs and corresponding API requests
- Expand capacity over time by adding additional ElectricFlow servers
- Distribute API requests across multiple ElectricFlow servers
- Distribute ElectricFlow requests across multiple web servers

Required Additional Software Components for Clustered Machines

A clustered ElectricFlow configuration requires two additional software components:

- A centralized service for maintaining and synchronizing group services in cluster
- A load balancer for routing work to machines in the cluster

Plugins Directory Accessibility Requirement for Clustered Machines

Electric Cloud strongly recommends that all server machines in a clustered server configuration be able to access a common plugins directory. This avoids the overhead of managing multiple plugins directories. For details, see Configuring Universal Access for a Network Location on page 5-21

See Creating a Server Cluster for ElectricFlow or DevOps Insight on page 4-1 for additional details and clustered configuration set up procedures.

Chapter 2: System Requirements and Supported Platforms

This section describes hardware and software specifications and configurations for installing and running ElectricFlow on Windows or UNIX systems. All version requirements for operating systems and databases are routinely tested and fully supported by Electric Cloud. Contact Electric Cloud technical support if you have any questions regarding newer software versions.

Supported Server Platforms

This section describes the supported platforms for the ElectricFlow, web, repository, and DevOps Insight servers.

Windows Platforms

The following table lists all supported Microsoft Windows server platforms.

Platform	Notes
Windows 10 (64-bit)	–
Windows 8.1 (64-bit)	–
Windows 7 (64-bit)	• Service Pack 1 is recommended. • An administrator might need to disable User Account Control (UAC). If the installer runs under account *x*, but services will run under account *y*, installation directories (both program and data) will probably have permissions that prevent *y*'s access. This applies particularly to data directories.
Windows Server 2016 (64-bit) Windows Server 2012 R2 (64-bit) Windows Server 2012 (64-bit)	• An administrator might need to disable User Account Control (UAC). If the installer runs under account *x*, but services will run under account *y*, installation directories (both program and data) will probably have permissions that prevent *y*'s access. This applies particularly to data directories.

Linux Platforms

The following table lists all supported Linux server platforms.

Platform	Notes
CentOS 7 (64-bit)	The following installation prerequisites apply to all ElectricFlow installers. Do not choose “nobody” for the CentOS user. CentOS does not allow a command such as `su - nobody -c foo.sh`, because it is not a shell account.
Red Hat Enterprise Linux 7 (64-bit) Red Hat Enterprise Linux 6 (64-bit)	The following installation prerequisites apply to all ElectricFlow installers. Do not choose “nobody” for the RHEL user. RHEL does not allow a command such as `su - nobody -c foo.sh`, because it is not a shell account.

Ubuntu 18.04 (64-bit)
Ubuntu 16.04 (64-bit)
Ubuntu 14.04 (64-bit)
	The following installation prerequisites apply to all ElectricFlow installers.
	Choosing the Ubuntu User
	Do not choose “nobody” for the Ubuntu user. Ubuntu does not allow a command such as `su - nobody -c foo.sh`, because it is not a shell account.
	Adding the bin Directory to the PATH Environment Variable
	Update `/etc/environment` to include the ElectricCommander `bin` directory in the PATH environment variable. Steps running with impersonation on Ubuntu use PATH that is set in `/etc/environment`. As a side-effect, the ElectricCommander `bin` directory is not in PATH in the impersonation context, so calls to tools such as ectool and postp fail with a “not found” error.
	Fixing the "raise ValueError, 'need a file or string" Error
	If you receive an error during installation similar to the following:
	`File "/usr/lib/lsb/install_initd", line 3, in <module>`
	`import sys, re, os, initdutils`
	`File "/usr/lib/lsb/initdutils.py", line 18`
	`raise ValueError, 'need a file or string'`
	`^`
	`SyntaxError: invalid syntax`
	run the following command:
	`sudo sed -i "s/python3/python/" /usr/lib/lsb/install_initd`
	This error is a known Ubuntu bug.

Supported Agent Platforms

“Pure” Agent Platforms on page 2-3

Proxy Agents for Other Platforms on page 2-7

This section lists all agent platforms supported by ElectricFlow. You can drive automation on target machines by either installing agents natively or by running them remotely using proxy agents.

"Pure" Agent Platforms

Platform	Notes
Platforms supported by the ElectricFlow server	See Supported Server Platforms on page 2-1.

Platform	Notes
AIX 7.1	• AIX agents are not compatible with ElectricCommander server versions earlier than 5.4 over HTTPS connections.

Platform	Notes

	• If you require interaction between the agent and the repository, make sure that IBM Java 1.8.0 or newer is installed on each agent machine. To do so:
	  • Enter `java -version` from the command line to check the current Java version. For example:
	  `$ java -version`
	  `java version "1.8.0"`
	  `Java(TM) SE Runtime Environment (build pap6480sr3fp22-20161213`
	  `_02 (SR3 FP22))`
	  `IBM J9 VM (build 2.8, JRE 1.8.0 AIX ppc64-64 Compressed References 20161209_329148 (JIT enabled, AOT enabled)`
	  `J9VM - R28_20161209_1345_B329148`
	  `JIT - tr.r14.java.green_20161207_128946`
	  `GC - R28_20161209_1345_B329148_CMPRSS`
	  `J9CL - 20161209_329148)`
	  `JCL - 20161213_01 based on Oracle jdk8u111-b14`
	  • Make sure that the `PATH` environment variable is updated to point to the current Java version in all applicable files (including /etc/environment/ and /etc/profile). For example, change:
	  `PATH=/usr/bin:/etc:/usr/sbin:/usr/ucb:/usr/bin/X11:/sbin: /usr/java5/jre/bin: /usr/java5/bin`
	  to
	  `PATH=/usr/bin:/etc:/usr/sbin:/usr/ucb:/usr/bin/X11:/sbin: /usr/java8_64/jre/bin:/usr/java8_64/bin`
	  • If you installed IBM Java after installing the agent on a machine, restart its agent service.
	An incorrect Java version can cause errors such as the following:
	`host01[/opt/electriccloud/ec/wkspc] > ectool publishArtifactVersion --artifactName "myGroup:myKey" --version "1.0.1" --fromDirectory /tmp/artifacts_test --repositoryName repo_for_aix`
	`Exit code 1: The java class could not be loaded. java.lang. ClassFormatError: (com/electriccloud/repo/client/ PublishArtifactVersionClient) unknown constant pool entry tag at offset=253`
	For more information about installing and configuring IBM Java, see:
	  • IBM Java for AIX HowTo: Install, Upgrade, or Downgrade IBM Java
	  • IBM Java for AIX FAQ: Identifying the Java versions and Java installation locations for an AIX system
	  • Setting up and checking your AIX environment

Platform	Notes
HP-UX 11i v1 (11.11) or later (PA-RISC 2.0 architecture)	• Make sure that patches PHKL_29243 and PHSS_39077 (or patches superseding these patches) are installed. • HP-UX Secure Shell requires a random number generator on the system. It searches for `/dev/urandom` and then `/dev/random` and uses the first device it finds. If it fails to find them, it uses its own internal random number generator. By default, HP-UX 11i v2 systems includes these random number devices. You can obtain them for HP-UX 11i v1 by downloading and installing the HP-UX Strong Random Number Generator from http://software.hp.com. HP recommends that Secure Shell users on HP-UX 11i v1 systems install the Strong Random Number Generator, because it significantly speeds up program initialization and execution for some commands.
macOS X 10.4 (Tiger) or later (Intel architecture)	-
Oracle Solaris 10 (SPARC and Intel x86 architectures)	If you require interaction between the agent and the repository, make sure that Java 1.8.0 or newer is installed on each agent machine. For more information, see How do I download and install Java for Solaris?. If you install Java after installing the agent on a machine, you must restart its agent service.
Oracle Solaris 9 (SPARC architecture)	If you require interaction between the agent and the repository, make sure that Java 1.8.0 or newer is installed on each agent machine. For more information, see How do I download and install Java for Solaris?. If you install Java after installing the agent on a machine, you must restart its agent service.
SUSE Linux Enterprise Server 12.3 (32- and 64-bit)	Run `zypper install libstdc++6-32bit` before installing agents on a 64-bit machine. This command installs the SUSE 32-bit libraries required by the ElectricFlow executable file.
SUSE Linux Enterprise Server 11.4 (32 and 64-bit)	-

Proxy Agents for Other Platforms

A proxy agent is an ElectricFlow agent that channels to a proxy target, which lets you drive automation in an agentless fashion. A proxy agent is an agent on a supported Windows or Linux platform that you use to take actions on any platform that is not listed above. For example, you can use a proxy agent to automate actions on an IBM z Systems mainframe running z/OS or Linux OS.

You can use a proxy agent to communicate with any target platform that can run commands via an SSH protocol. For details, see the "Environment Proxy Server Configuration" section in the "Configuration" chapter of the *ElectricFlow Installation Guide* at http://docs.electric-cloud.com/eflow_doc/FlowIndex.html.

Server and Agent Compatibility

Not all combinations of server version and agent version are supported. This is because of an ectool/Perl API communication incompatibility as well as a Diffie-Hellman key size incompatibility.

Diffie-Hellman Key Size Incompatibility

To enable the ElectricFlow server versions 7.0 or newer to configure Diffie-Hellman cipher suites properly, ElectricFlow uses OpenSSL-1.0.1T or newer versions with SSLv2 enabled. Because of OpenSSL and JRE changes, the minimum Diffie-Hellman key size requirement is increased to 1024 bits (from 768 bits) as of version 7.0.

Server versions 7.0 or newer use Jetty (a Java HTTP server), which listens on the 8000 (unsecure) and 8443 (secure) ports. Server versions 7.0 or newer use Java 1.8.0_66, in which the ephemeral DH key size defaults to 1024 bits during SSL/TLS handshaking in the SunJSSE provider.

For details on the increase of the key size requirement as of Java 1.6-u101, see the Java release note at http://www.oracle.com/technetwork/java/javase/overview-156328.html#6u101-b31. For details as of Java 1.7-u85, see the Java release note at http://www.oracle.com/technetwork/java/javase/7u85-relnotes-2587591.html.

Because their minimum key size is 1024 bits, agent versions 7.0 or newer can connect only to:

- Server versions 5.4, 6.0.1, or 6.5 or higher via ectool
- External applications that require SSL with a minimum key size of 1024 bits

However, ElectricCommander agents of versions 5.0.6, 5.3, or 5.4 and ElectricFlow agent versions 6.0.1 or 6.5 or newer can connect to all ElectricFlow server versions (including 7.0 or newer) via ectool and ec-perl.

ElectricCommander server versions 5.0.6 or 5.3 or newer can run jobs using all agent versions (including 7.0 or newer). ElectricFlow server versions 7.0 or newer can run jobs using ElectricCommander agent versions 5.0.6 or 5.3 or newer.

Server and DevOps Insight Server Compatibility

Not all combinations of ElectricFlow server version and DevOps Insight server version are supported. The following matrix lists the compatibility for server and DevOps Insight server versions.

	ElectricFlow version							
	7.3	8.0	8.1	8.2	8.3	8.4	8.5	9.0

DevOps Insight server version	7.3	✓	-	-	-	-	-	-	-
	8.0	-	✓	-	-	-	-	-	-
	8.1	-	-	✓	✓	✓	✓	-	-
	8.2	-	-	-	✓	✓	✓	-	-
	8.3	-	-	-	-	✓	✓	-	-
	8.4	-	-	-	-	-	✓	-	-
	8.5	-	-	-	-	-	-	✓	✓
	9.0	-	-	-	-	-	-	-	✓

Hardware Requirements

This section lists the minimum requirements for any Windows or Linux machines installed with the ElectricFlow server software.

- Processor clock rate: 1.5 GHz or higher
- Memory: 4 GB available RAM or more (16 GB recommended for small to medium deployments, for the DevOps Insight server)
- Processors: 2 or more (4 processors recommended for small to medium deployments)

Browser Requirements

ElectricFlow supports the following web browsers:

- Microsoft Internet Explorer 11
- Microsoft Edge
- Mozilla Firefox
- Google Chrome

> **Note:** Web browser extensions such as AdBlock Plus for Google Chrome can interfere with the display of ElectricFlow web pages. You should disable any ad-blocking browser extensions or add an exclusion for ElectricFlow web pages.

Java Requirements

ElectricFlow uses the web server SSL cipher suite for intermediate compatibility as described in the Mozilla Security/Server Side TLS wiki page. To comply with this cipher suite, you must use Java version 1.7 or newer.

Port Usage

ElectricFlow uses certain ports by default. Make sure your firewall is open for these ports. This section contains the default port values and information about avoiding port conflicts.

> **Note:** Transport Layer Security (TLS) has replaced Secure Sockets Layer version 3.0 (SSLv3) on the ElectricFlow server and the ElectricFlow web server.

Default Server Ports

ElectricFlow servers use the following ports:

Port	Used by
8000	ElectricFlow server
8443	ElectricFlow server (SSL port)
80	ElectricFlow web server
7080	ElectricFlow web server when installed on Linux platforms without root privileges
443	ElectricFlow web server (SSL port)
7443	ElectricFlow web server (SSL port) when installed on Linux platforms without root privileges
6800	Port used by the ElectricFlow agent for HTTP communication on the localhost network interface
7800	ElectricFlow agents (by default, this is an HTTPS port)
61613	Preflight file transfer port, other file transfer, event notifications, or other messaging
8200	Artifact repository server (by default, this is an HTTPS port)
8900	ElectricFlow built-in (default) database. You can change this port number by using the `ecconfigure --databasePort <port_number>` option or the `--databasePort` installer argument.

Default ElectricFlow Services Ports

The Java Service Wrapper uses the following ports to communicate with a Java virtual machine (JVM):

Port	Used by
127.0.0.1:32000	ElectricFlow agents
127.0.0.1:32001	ElectricFlow server
127.0.0.1:32002	Artifact repository

Default DevOps Insight Server Ports

The ElectricFlow DevOps Insight server uses the following ports:

Port	Used by
9200	DevOps Insight server to retrieve data from Elasticsearch
9300	Used by the Elasticsearch service for internal communication between nodes within the Elasticsearch cluster
9500	Logstash to receive data from ElectricFlow
9600	Used by the Logstash service for the Logstash monitoring APIs

Avoiding Port Conflicts

If you are installing an ElectricFlow server and your web server or other application uses the same ports as the ElectricFlow host, you must take one of the following actions:

- Select different web server or application ports
- Uninstall the existing web server or application
- Disable the existing web server or application
- Reconfigure the existing web server or application to use another port

Database Requirements

You cannot log into ElectricFlow until a database is configured. During the ElectricFlow server installation, you can select the built-in (default) ElectricFlow database (MariaDB) or an alternate database.

> **Note:** If you are using two different ElectricFlow servers in a non-HA configuration, they cannot point to the same database.

Built-In Database

ElectricFlow ships with a "demo" license, which limits the software to two concurrent job steps and the ElectricFlow-provided built-in database. Running ElectricFlow on a single machine with the demo license is generally *not* recommended for a production environment. Also, the built-in database is not supported in a clustered ElectricFlow configuration.

ElectricFlow should connect to an alternate, external database in a typical production configuration. If ElectricFlow was installed with the built-in database, you can reconfigure it to use an alternate external database at any time. For a list of alternate databases supported by ElectricFlow, see Supported Alternate Databases on page 2-12. For more information and configuration instructions, see External Database Configuration on page 5-2.

Using an alternate database requires an ElectricFlow enterprise license. You must configure an alternate database at the same time as you install your enterprise license to prevent error messages about an unsupported configuration or a license requirement.

Supported Alternate Databases

ElectricFlow supports the following alternate databases:

- MySQL 5.5.12, 5.6, 5.7 or later
  - Clean installations of the ElectricFlow server require the MySQL JDBC driver. See Installing the MySQL JDBC Driver on page 3-181.
  - For upgrades, additions to my.cnf/my.ini are required. See Installing the MySQL JDBC Driver on page 3-181.
- MS SQL Server2012 (2012 R4 is recommended), 2014, 2016, and 2017
- Oracle 12c, 18c, and 19c

Alternate Database Requirements

Alternate databases must be configured to use UTF-8 encoding and configured to allow up to 200 open connections.

In an Oracle database, set the `OPEN_CURSORS` parameter to at least 1000 to prevent ElectricFlow from running out of open cursors. But depending on your ElectricFlow server usage, the `OPEN_CURSORS` value of 1000 might not be sufficient, and you might see `java.sql.SQLException: ORA-01000: maximum open cursors exceeded`" in the *<DATA_DIR>*`/logs/commander.log` file. In this case, you must increase the value of `OPEN_CURSORS` to one that is optimal depending on your usage.

Database Sizing

Expected database growth over time can be correlated with the number of job steps created. Database growth is NOT correlated with build log or build artifact sizes.

To create a reasonable database growth estimate per period:

1. Estimate the number of jobs per period.
2. Multiply the "estimated number of jobs" by the number of steps estimated per job. This will determine the estimated number of steps per period.
3. Multiply the "estimated number of steps per period" by 10 to determine the disk size (in Kbytes) required per period.

For example, if you run 500 jobs per day with an average of 200 steps per job, you would run 100K steps per day. This means your database would grow about 1 GB per day or 90 GB per quarter. Using this example, if you prune jobs older than 30 days, database size could be maintained at about 30 GB.

Disk Usage

Disk space usage varies and depends on the quantity and size of the jobs you run. We recommend starting with the following free space recommendations:

Server

10 GB is recommended.

Agents

5 GB each is recommended.

Sizing Artifact Cache Directory Space on Resources

By default, artifacts are retrieved into the `<DATA_DIRECTORY>/artifact-cache` directory of the agent installation. You can modify the `agent.conf` file to change the location, or you can specify the cache directory location on each resource known to ElectricFlow.

Determining how much free space the cache partition needs to accommodate all of your artifact versions can be difficult. One approach is that for each artifact, estimate how large you think each version will be and how many versions you plan to keep. Compute the total required space to be the sum of version-size * numVersions for each artifact. Add a buffer of 50%. Using your end result, allocate a disk/partition of that size and configure the cache as a directory on that disk/partition.

Repository Server

If using Artifact Management functionality, the repository server might need 20-30 GB.

Although a server install includes an artifact repository, we recommend that production repository servers be installed on different machines than the ElectricFlow server. The repository server might do a very large amount of disk and network I/O when transferring artifact versions to and from requesters, and this might adversely affect ElectricFlow server performance.

Sizing the Repository Backingstore

For a repository installation, by default, the repository backingstore is the `<DATA_DIRECTORY>/repository-data` directory. You can modify the `<DATA_DIRECTORY>/conf/repository/server.properties` file or use `ecconfigure` to update the backingstore location. Determining exactly how much free space the backingstore disk/partition needs to accommodate your artifact versions can be difficult. Here is one approach to approximate the disk size you need:

For each artifact, estimate how large you think each version will be and how many versions you plan to keep. Compute the total required space to be the sum of version-size * numVersions for each artifact. Add a buffer of 50%. Using your end result, allocate a disk/partition that size and configure the repository backingstore as a directory on that disk/partition.

Logs

You can set the following properties as Java system properties in `wrapper.conf`:

- `ec.logRoot` controls the location of the log output. The default location is the logs/commander.log directory.
- `ec.logHistory` controls the number of days of log history that is kept. The default is 30 days.
- `ec.logSize` controls the size of each log file before it is zipped up and a new log file started. The default is 100 MB, but each log rotation will zip the file, so that only about 6-7 MB of space are being taken.

  Production systems generate multiple log files per day – an average system can generate 50-100 log files. This means that the daily requirement for space (under this type of load) is 300-700 MB. Retaining 1 months' worth of logs requires 9-21 GB of space, so adjusting the `ec.logHistory` value to something lower might be appropriate, if you want to allot less space for this logging.

To limit the amount of disk space for logging, the most effective approach is to use a lower `ec.logHistory` value.

DevOps Insight Server

Determining the amount of disk space required for the DevOps Insight Server depends on the shape and size of data that you will store on the DevOps server. This data is used by Elasticsearch, which is the underlying analytics store and search engine. Following are general guidelines based on Electric Cloud performance and scalability tests.

ElectricFlow sends all deployment events, pipeline runs, and release data to the DevOps Insight server. The following table shows the average size for each data set:

Data set	Amount	Documents in corresponding Elasticsearch index	Average index size
Deployments	100 deployments	73443 (The number of documents per deployment depends on the number of deployment events in your deployment process)	18.5 MB
Pipeline runs	4 pipeline runs	24 (The number of documents per pipeline run depends on your pipeline definition)	285 KB
Releases	5 releases	5	52 KB

Based on the above table, if you will run 10 deployments a week, 2 pipeline runs a month, and 1 release per month, then over one year including weekends and holidays, you will need about 97 MB (95 MB + ~1.7 MB + ~0 MB) of disk space to store deployment events, pipeline runs, and release data in the Elasticsearch server backing the DevOps Insight server.

If you have also set up the plugins to collect and send data for the Release Command Center to the DevOps Insight server, then you will need additional disk space, which can be determined as follows. The following table shows the average size for each data set:

Data set	Amount	Documents in corresponding Elasticsearch index	Average index size
Features (stories)	400 features	1200 (Assuming that each feature underwent three updates from the point it was first sent to Elasticsearch)	256 KB
Builds	40 builds	40	248 KB
Quality (aggregated test results)	140	140	347 KB
Incidents	90 incidents	270 (Assuming that each incident underwent three updates from the point it was first sent to Elasticsearch)	164 KB

Based on the above table, if you will run 10 builds a day, 50 aggregated test results a day, 50 features (stories) per month, and 5 incidents per month, then over one year including weekends and holidays, you will need about 240 MB (22 MB + 44 MB + 64 MB + 109 MB) of additional disk space to store data collected by the plugins.

The total disk space for a year would be about 340 MB (97 MB + 240 MB) based on the above metrics. You should apply the required adjustments to calculate your disk space requirements for the DevOps Insight server based on your data-generation patterns.

Memory Settings

Memory usage varies depending on whether or not the ElectricFlow server is a dedicated machine.

- An ElectricFlow server running on a dedicated machine has a default minimum heap memory allocation of 20% and a maximum heap memory allocation of 40%. This applies to either a 32 or 64-bit system.
- In general, an ElectricFlow agent has a default minimum memory usage of 16 MB and a maximum memory usage of 64 MB. However, agents for REPO-server, Web-Server and Proxy agents needing higher settings; for details, see the *KBEC-00248 - Agent Memory Configuration* Knowledge Base article.

Modifying Memory Settings for an ElectricFlow Server

There are two ways you can adjust the amount of memory for the ElectricFlow server.

- Modify the `wrapper.java.initmemory.percent` and `wrapper.java.maxmemory.percent` lines in `wrapper.conf`

  Use the following table to determine the correct directory path.

Server Type	System	Path
Non-repository	Windows 2008	c:\ProgramData\Electric Cloud\ElectricCommander\conf\wrapper.conf
	Windows 7	
	Linux	/opt/electriccloud/electriccommander/conf/wrapper.conf
Repository	Windows 2008	c:\ProgramData\Electric Cloud\ElectricCommander\conf\repository\wrapper.conf
	Windows 7	
	Linux	/opt/electriccloud/electriccommander/conf/repository/wrapper.conf

- Use ecconfigure to set the initial and maximum memory settings.

  For example, to set the ElectricFlow Server initial memory percentage to 21% and the maximum memory percentage to 31%, enter the following command:

  ```
  ecconfigure --serverInitMemory 21 --serverMaxMemory 31
  ```

Modifying Memory Settings for an ElectricFlow Agent

To adjust the amount of memory for the ElectricFlow agent, modify the `wrapper.java.initmemory.percent` and `wrapper.java.maxmemory.percent` lines in wrapper.conf for the agent. Use the appropriate directory path:

- Windows: C:\ProgramData\Electric Cloud\ElectricCommander\conf\agent\wrapper.conf
- Linux: /opt/electriccloud/electriccommander/conf/agent/wrapper.conf

Modifying Memory Settings for a Containerized ElectricFlow Server or ElectricFlow Repository Server

By default, the initial memory and maximum memory for the ElectricFlow server and repository server JVMs are configured as percentages of the total system memory. However, if these servers are running in a container, their JVMs cannot see the container's total system memory.

To fix this problem, you can either:

- Make the JVM aware that it is running in a docker container and observe the container memory limits.
- Modify the settings in the `/opt/electriccloud/electriccommander/conf/wrapper.conf` file for the ElectricFlow server and ElectricFlow repository server to use absolute values (in MB) instead of using the `wrapper.java.initmemory.percent` and `wrapper.java.maxmemory.percent` settings.

Making the ElectricFlow Server JVM Aware of Docker Container Memory Limits

As of Java SE 8u131, and in JDK 9, you can transparently set a maximum Java heap for Docker memory limits. To make the JVM aware of these limits if you do not set a maximum Java heap via `-Xmx`, you must use two experimental JVM command line options:

```
-XX:+UnlockExperimentalVMOptions
```

```
-XX:+UseCGroupMemoryLimitForHeap
```

For more information, see the *KBEC-00376 - Making the ElectricFlow Server JVM Aware of Docker Container Memory Limits* knowledge base article. ElectricFlow 8.0.1 and later versions include JRE build 1.8.0_131-b11 to provide this capability.

Configuring Initial and Maximum Memory Settings for a Containerized ElectricFlow Server or ElectricFlow Repository Server

To configure the ElectricFlow server and repository server Java processes to use absolute values:

1. Open a Bash session in the container by entering:

   ```
   docker exec -it <container_name> bash
   ```

   where *<container_name>* is the name of your ElectricFlow server or repository server container. For example, enter:

   ```
   docker exec -it efserver bash
   ```

2. Enter the following command:

   ```
   ecconfigure --serverInitMemoryMB=<megabytes> --serverMaxMemoryMB=<megabytes>
   ```

   or

   ```
   ecconfigure --repositoryInitMemoryMB=<megabytes> --
   repositoryMaxMemoryMB=<megabytes>
   ```

   For example, enter:

   ```
   ecconfigure --serverInitMemoryMB=4096 --serverMaxMemoryMB=6144
   ```

   or

   ```
   ecconfigure --repositoryInitMemoryMB=512 --repositoryMaxMemoryMB=1024
   ```

   Select the maximum values based on your usage requirements. The server service restarts and begins using the new settings.

For more information, see the *KBEC-00387 - Configuring Initial and Maximum Memory Settings for a Containerized ElectricFlow Server or ElectricFlow Repository Server* knowledge base article. For information about using `ecconfigure`, see the "ElectricFlow Installed Tools" section in the "Automation Platform" chapter of the *ElectricFlow User Guide* at http://docs.electric-cloud.com/eflow_doc/FlowIndex.html.

Checksum Utility

An MD5 checksum file is available for each installer file on the Electric Cloud ShareFile and FTP sites. To verify that ElectricFlow files are intact and unaltered after you download them, download the corresponding MD5 checksum file also. MD5 utilities are available for Windows, Linux, and macOS operating systems.

Linux

On Linux, verify with:

```
md5sum --check ElectricFlow-<version>.md5
```

Most Linux installations provide an `md5sum` command for calculating MD5 prompt digests.

Windows

You can download an MD5 utility for Windows at http://fourmilab.ch/md5/.

macOS

To use the MD5 checksum utility on macOS:

1. In Finder, browse to `/Applications/Utilities`.
2. Double-click the Terminal icon.

   A terminal window appears.
3. In the terminal window, type: "`md5`" (followed by a space).
4. Drag the downloaded file from the Finder into the Terminal window.

5. Click in the Terminal window and press `Return`.
6. Compare the checksum displayed on the screen to the one on the download page.

Software Licenses

To see your software usage entitlements, go to the **Licenses** page in the Automation Platform web UI. To do so, browse to https://<*ElectricFlow_server*>/commander/, and then click **Administration** > **Licenses**.

For information about how to import licenses, delete licenses, and view license usage statistics, see the "Licenses" section in the "Automation Platform" chapter of the *ElectricFlow User Guide* at http://docs.electric-cloud.com/eflow_doc/FlowIndex.html. The section also discusses the various types of licensing, which is based on concurrent steps, concurrent hosts, concurrent users, proxied hosts, registered hosts and users, and creation of applications and microservices.

Chapter 3: Installing ElectricFlow

This section describes the methods for installing ElectricFlow in a new environment. To upgrade ElectricCommander or ElectricFlow, see Roadmap for Upgrading ElectricFlow on page 6-1.

Important: The following situation might occur when the workspace files are in a directory other than the default *workspace* directory and the ElectricFlow configuration links to it. When you install a new version, ElectricFlow creates a workspace directory in the default location. It does not recognize the preconfigured workspace link in the previous configuration.

When configuring ElectricFlow after an upgrade, you cannot use ecconfigure to move the workspace directory to the preconfigured network location. You must manually specify the link to the workspace directory in the new configuration.

Not all combinations of server version and agent version are supported. This is because of an incompatibility involving ectool/Perl API communication with the ElectricFlow server and a Diffie-Hellman key size incompatibility. For details, see Server and Agent Compatibility on page 2-8.

Note: Although the ElectricFlow installer is in `$INSTALL_DIRECTORY/src`, do not launch it from there.

ElectricFlow Installer Files

The following installer files are available for the Windows and Linux platforms.

Type	Platform	What the Installer Does	Filename
"Pseudo" 64-bit full	Windows	Installs all components, including the ElectricFlow server, built-in database, web server, repository server, agents, and ElectricFlow tools. The "pseudo" 64-bit agent is installed.	`ElectricFlow-<version>.exe` Example: `ElectricFlow-8.5.0.12345.exe`
"Pure" 64-bit full	Linux	Installs all components, including the ElectricFlow server, web server, repository server, agents, and ElectricFlow tools. The "pure" 64-bit agent is installed. Has an option for installation by a non-root user or a user without `sudo` privileges.	`ElectricFlow-x64-<version>` Example: `ElectricFlow-x64-8.5.0.12345`

Type	Platform	What the Installer Does	Filename
32-bit agent only	Windows	Installs the 32-bit agent.	`ElectricFlowAgent-x86-<version>.exe` Example: `ElectricFlowAgent-x86-8.5.0.12345.exe`
"Pseudo" 64-bit agent only	Windows	Installs the "pseudo" 64-bit agent.	`ElectricFlowAgent-x64-<version>.exe` Example: `ElectricFlowAgent-x64-8.5.0.12345.exe`
"Pure" 64-bit agent only	Linux	Installs the "pure" 64-bit agent. Has an option for installation by a non-root user or a user without `sudo` privileges.	`ElectricFlowAgent-x64-<version>` Example: `ElectricFlowAgent-x64-8.5.0.12345`
"Pure" 64-bit DevOps Insight	Windows and Linux	Installs ElectricFlow DevOps Insight. Has an option for installation by a non-root user or a user without `sudo` privileges.	`ElectricFlowDevOpsInsightServer-x64-<version>` Example: `ElectricFlowDevOpsInsightServer-x64-8.5.0.132129`
"Pure" 64-bit DevOps Foresight	Windows and Linux	Installs ElectricFlow DevOps Foresight. Requires installation by root or a user with `sudo` privileges.	`ElectricFlowDevOpsForesightServer-x64-<version>` Example: `ElectricFlowDevOpsForesightServer-x64-8.5.0.132129`

Availability of Installers with a Non-Root/Non-sudo or Non-Administrator Mode

Certain ElectricFlow installers allow you to perform installations as a non-root/non-Administrator user or a user without `sudo` privileges. The following table shows whether a particular installer has an option to run in this mode.

Platform	Server	32-Bit Agent-Only	"Pseudo" 64-Bit Agent-Only	"Pure" 64-Bit Agent-Only
Linux	Yes	No	Yes	Yes
Windows	No	No	No	No
AIX	No installer	No installer	No installer	Yes

Platform	Server	32-Bit Agent-Only	"Pseudo" 64-Bit Agent-Only	"Pure" 64-Bit Agent-Only
HP-UX	No installer	Yes	No installer	No installer
MacOS	No installer	Yes	No installer	No installer
Solaris	No installer	Yes	No installer	No installer

Note: For server installations, you cannot specify different users for the agent service and for the other services (ElectricFlow server, web server, and repository server) during the same installer session—The user who launched the installer will be the owner for all services.

Choosing the Correct Installation Interface and Installer Option

This section describes the various installation interfaces and available options for specific platform types.

For information about supported server platforms and non-server platforms, see Supported Server Platforms on page 2-1 and Supported Agent Platforms on page 2-2.

User Interface Installation Process

This process provides an installation Wizard for installing ElectricFlow on a supported server platform. The following installation options are generally preferred by Windows users, but they are also supported on Linux platforms with the X Window System installed.

The installation options are:

- **Express Server**

  This option installs the ElectricFlow server, built-in database, web server, and repository server on one machine. The default ElectricFlow server settings are used. A local agent (required for running jobs), and ElectricFlow tools are also installed.

  This option is available via a "full" installer file (see ElectricFlow Installer Files on page 3-1). This option is best for quickly installing the ElectricFlow software for evaluation purposes.

  Important:
  ElectricFlow ships with a "demo" license, which limits the software to two concurrent job steps and the ElectricFlow-provided built-in database. Running ElectricFlow on a single machine with the demo license is generally *not* recommended for a production environment. Also, the built-in database is not supported in a clustered ElectricFlow configuration.

ElectricFlow should connect to an alternate, external database in a typical production configuration. If ElectricFlow was installed with the built-in database, you can reconfigure it to use an alternate external database at any time. For a list of alternate databases supported by ElectricFlow, see Supported Alternate Databases on page 2-12. For more information and configuration instructions, see External Database Configuration on page 5-2.

Using an alternate database requires an ElectricFlow enterprise license. You must configure an alternate database at the same time as you install your enterprise license to prevent error messages about an unsupported configuration or a license requirement.

- **Express Agent**

  This option installs an ElectricFlow agent and ElectricFlow tools. This option is available via a "full" installer file (see ElectricFlow Installer Files on page 3-1). Use this option for managed hosts where you want to run job steps.

  This option is useful for installing a single agent. To install agents on multiple machines, you should use Silent Unattended Installation on page 3-7.

- **Advanced**

  This option installs individual components, directories, or ports of your choice. This option is available via one of the "full" installer files (see ElectricFlow Installer Files on page 3-1).

  You use this option to install any combination of your choice among the ElectricFlow server, built-in database, web server, and repository server. (A local agent and ElectricFlow tools are required and are automatically installed.)

- **DevOps Insight Server**

  Installs the ElectricFlow DevOps Insight server. This option requires the DevOps Insight-only installer file (see ElectricFlow Installer Files on page 3-1).

  This option includes the ability to add DevOps Insight servers to a DevOps Insight cluster. For details, see Creating a DevOps Insight Server Cluster on page 4-36.

- **DevOps Foresight Server**

  Installs the ElectricFlow DevOps Foresight server. This option requires the DevOps Foresight-only installer file (see ElectricFlow Installer Files on page 3-1).

- **32-Bit Agent-Only (Windows Only)**

  Installs a 32-bit ElectricFlow agent. This option is available via the 32-bit agent-only installer file (see ElectricFlow Installer Files on page 3-1). Use this option for managed hosts where you want to run job steps.

  This option is useful for installing a single agent. To install multiple agents, you should use Silent Unattended Installation on page 3-7.

- **"Pure" 64-Bit Agent-Only (Linux Only)**

  Installs a "pure" 64-bit ElectricFlow agent. This option is available via the "pure" 64-bit agent-only Linux installer file (see ElectricFlow Installer Files on page 3-1).

  Use this option for managed hosts where you want to run job steps. This option is for installing a single agent. To install multiple agents, you should use Silent Unattended Installation on page 3-7.

- **"Pseudo" 64-Bit Agent-Only (Windows Only)**

  Installs a "pseudo" 64-bit ElectricFlow agent. This option is available via the "pseudo" 64-bit agent-only Windows installer file (see ElectricFlow Installer Files on page 3-1). Use this option for managed hosts where you want to run job steps.

  This option is useful for installing a single agent. To install multiple agents, you should use Silent Unattended Installation on page 3-7.

Interactive Command-Line Installation Process (Linux Only)

These installation options provide an interactive command line for installing ElectricFlow on a supported server platform. These installation methods are available only for Linux platforms.

The installation options are:

- **Express Server**

  This option installs the ElectricFlow server, built-in database, web server, and repository server on one machine. The default ElectricFlow server settings are used. A local agent (required for running jobs), and ElectricFlow tools are also installed.

  This option is available via a "full" installer file (see ElectricFlow Installer Files on page 3-1). This option is best for quickly installing the ElectricFlow software for evaluation purposes.

  Important:
  ElectricFlow ships with a "demo" license, which limits the software to two concurrent job steps and the ElectricFlow-provided built-in database. Running ElectricFlow on a single machine with the demo license is generally *not* recommended for a production environment. Also, the built-in database is not supported in a clustered ElectricFlow configuration.

  ElectricFlow should connect to an alternate, external database in a typical production configuration. If ElectricFlow was installed with the built-in database, you can reconfigure it to use an alternate external database at any time. For a list of alternate databases supported by ElectricFlow, see Supported Alternate Databases on page 2-12. For more information and configuration instructions, see External Database Configuration on page 5-2.

  Using an alternate database requires an ElectricFlow enterprise license. You must configure an alternate database at the same time as you install your enterprise

license to prevent error messages about an unsupported configuration or a license requirement.

- **Express Agent**

  This option installs an ElectricFlow agent and ElectricFlow tools. This option is available via a "full" installer file (see ElectricFlow Installer Files on page 3-1). Use this option for managed hosts where you want to run job steps.

  This option is useful for installing a single agent. To install agents on multiple machines, you should use Silent Unattended Installation on page 3-7.

- **Advanced**

  This option installs individual components, directories, or ports of your choice. This option is available via one of the "full" installer files (see ElectricFlow Installer Files on page 3-1).

  You use this option to install any combination of your choice among the ElectricFlow server, built-in database, web server, and repository server. (A local agent and ElectricFlow tools are required and are automatically installed.)

- **DevOps Insight Server**

  Installs the ElectricFlow DevOps Insight server. This option requires the DevOps Insight-only installer file (see ElectricFlow Installer Files on page 3-1).

  This option includes the ability to add DevOps Insight servers to a DevOps Insight cluster. For details, see Creating a DevOps Insight Server Cluster on page 4-36.

- **DevOps Foresight Server**

  Installs the ElectricFlow DevOps Foresight server. This option requires the DevOps Foresight-only installer file (see ElectricFlow Installer Files on page 3-1).

- **32-Bit Agent-Only (Windows Only)**

  Installs a 32-bit ElectricFlow agent. This option is available via the 32-bit agent-only installer file (see ElectricFlow Installer Files on page 3-1). Use this option for managed hosts where you want to run job steps.

  This option is useful for installing a single agent. To install multiple agents, you should use Silent Unattended Installation on page 3-7.

- **"Pure" 64-Bit Agent-Only (Linux Only)**

  Installs a "pure" 64-bit ElectricFlow agent. This option is available via the "pure" 64-bit agent-only Linux installer file (see ElectricFlow Installer Files on page 3-1).

  Use this option for managed hosts where you want to run job steps. This option is for installing a single agent. To install multiple agents, you should use Silent Unattended Installation on page 3-7.

- **"Pseudo" 64-Bit Agent-Only (Windows Only)**

  Installs a "pseudo" 64-bit ElectricFlow agent. This option is available via the "pseudo" 64-bit agent-only Windows installer file (see ElectricFlow Installer Files on page 3-1). Use this option for managed hosts where you want to run job steps.

  This option is useful for installing a single agent. To install multiple agents, you should use Silent Unattended Installation on page 3-7.

Silent Unattended Installation

These installation options provide a non-interactive command-line installation for supported server platforms. For a list of these options and the installers required for them, see ElectricFlow Installer Files on page 3-1.

You might find this installation process preferable for installing multiple remote agents, servers, or DevOps Insight servers. This installation includes the ability to add DevOps Insight servers to a DevOps Insight cluster.

The installation options are:

- **Windows**

  This option is only for Windows platforms.

- **Linux**

  This option is only for Linux platforms.

Non-Server Platform Agent Interface

This is a command line interface for installing the ElectricFlow agent and tool software only on supported non-server platforms.

The installations options are:

- **Command-Line Agent**

  Installs an agent from a UNIX command-line installer.

- **Silent Agent**

  Runs unattended (silent) installations with the UNIX installer.

Before You Install ElectricFlow

Review the following information before attempting to install any ElectricFlow software.

Linux and Windows ElectricFlow Installations

Platform Setup Prerequisite

Make sure you have completed any prerequisite platform setup. For details, see Supported Server Platforms on page 2-1 and Supported Agent Platforms on page 2-2.

Local Drive Requirement

You must install ElectricFlow on a local drive. Electric Cloud does not support installing the ElectricFlow server on a network volume.

Installation Order

Electric Cloud recommends installing the ElectricFlow server before installing remote agents or web servers.

Built-In Database Versus Alternate Databases

ElectricFlow ships with a "demo" license, which limits the software to two concurrent job steps and the ElectricFlow-provided built-in database. Running ElectricFlow on a single machine with the demo license is generally *not* recommended for a production environment. Also, the built-in database is not supported in a clustered ElectricFlow configuration.

ElectricFlow should connect to an alternate, external database in a typical production configuration. If ElectricFlow was installed with the built-in database, you can reconfigure it to use an alternate external database at any time. For a list of alternate databases supported by ElectricFlow, see Supported Alternate Databases on page 2-12. For more information and configuration instructions, see External Database Configuration on page 5-2.

Using an alternate database requires an ElectricFlow enterprise license. You must configure an alternate database at the same time as you install your enterprise license to prevent error messages about an unsupported configuration or a license requirement.

Java Runtime Environment Bitness

When you install a 64-bit machine, the 64-bit version of the Java Runtime Environment is installed automatically.

Specifying a Remote ElectricFlow Server

When installing an agent, repository server, or web server, you can enter information for a remote ElectricFlow server. That information is used to discover the server's plugins directory and set it so that the local installation is in sync with the remote ElectricFlow server.

During an agent installation, you can create a resource object on the server automatically. During a repository installation, you can create a repository object on the server automatically.

Clustered ElectricFlow Configurations

If you plan to use a clustered ElectricFlow configuration, see Creating a Server Cluster for ElectricFlow or DevOps Insight on page 4-1 for additional requirements and considerations.

Clustered DevOps Insight Server Configurations

For details about the overall steps for installing DevOps Insight on a group of servers to create a DevOps Insight server cluster, see Creating a DevOps Insight Server Cluster on page 4-36.

Running the DevOps Insight Server on a System with Other ElectricFlow Components

For a production environment, Electric Cloud recommends that you install the DevOps Insight server on a system other than systems running other ElectricFlow components (such as the ElectricFlow server, web server, repository server, or agent). If you must install it on the same system (such as for testing or other non-production or trial-basis situations), use one of the following installation processes.

If you have *not yet* installed the ElectricFlow DevOps Insight server on a system:

1. Install the other ElectricFlow components on the system as needed.
2. Install the ElectricFlow DevOps Insight server on the system.

If you have *already* installed the DevOps Insight server on a system:

1. Uninstall the ElectricFlow DevOps Insight server from the system.
2. Clean up data, logs, and any configuration files from the ElectricFlow data directory on the system.
3. Install the other ElectricFlow components on the system as needed.
4. Reinstall the ElectricFlow DevOps Insight server on the system.

Linux ElectricFlow Installations

Review the following information before installing ElectricFlow on a Linux machine.

umask and File Permission Requirements

The ElectricFlow installer sets the required umask and permissions on all ElectricFlow directories and files as follows:

- umask: 0022
- Permissions for owner of Electric Cloud files: 0644
- Executable file permissions: 0755

To avoid unexpected errors in functionality, do not change these values.

Installation Mode Without the X Window System

If the X Window System is not running or not available, the Linux user interface installer runs in interactive command-line mode.

"Pseudo" 64-bit Agent-Only Installers (Linux)

The 32-bit agent installer is a 32-bit executable that does not check if the machine has the required 32-bit compatibility libraries during the installation session.

Important:
When installing ElectricFlow on RHEL 6.x: For the 32-bit agent-only installer or the "pseudo" 64-bit agent-only installer on unsupported platforms or without an internet connection, you must install certain 32-bit libraries that were omitted by Red Hat. Otherwise, the installer exits because it cannot find those libraries. No error prompt is displayed, and the log file does not contain the error. To install the libraries, run the following commands:

- `yum install libstdc++.i686`—Without this command, the ElectricFlow Apache server will not start, and the installer silently fails for any type of ElectricFlow installation.
- `yum install libuuid.i686`—Required if your ElectricFlow installation includes an Apache server. If you are installing ElectricFlow agents only without Apache, you do not need this command on agent machines.
- `yum install nss-pam-ldapd*.i686`—Installs 32-bit NSS packages if using an LDAP account for ownership of the server, web, and repository services. Without this command, the ElectricFlow Apache server fails to start.

32-bit libraries are not required for the "pure" 64-bit full installer or the "pure" 64-bit agent-only installer.

Important:
When installing ElectricFlow on Ubuntu versions listed below: For the 32-bit agent-only installer or the "pseudo" 64-bit agent-only installer on unsupported platforms or without an internet connection, you must install certain 32-bit libraries that were omitted by Ubuntu. Otherwise, the installer exits because it cannot find those libraries. No error prompt is displayed, and the log file does not contain the error. To install the libraries, run the following commands:

- On Ubuntu 14.04, enter:
```
sudo apt-get update
sudo dpkg --add-architecture i386
sudo apt-get update
sudo apt-get install lib32bz2-1.0
sudo apt-get update
sudo apt-get install libuuid1:i386
```
- On Ubuntu 18.04 or 16.04, enter these commands:
```
sudo apt-get update
sudo dpkg --add-architecture i386
sudo apt-get update
sudo apt-get install libbz2-1.0:i386
sudo apt-get update
sudo apt-get install libuuid1:i386
```
- If you will use an LDAP account for ownership of the server, web, and repository services with 64-bit Ubuntu, you must run `sudo apt-get update && sudo apt-get install libnss-ldap:i386`. This command installs 32-bit NSS packages, which ensures that the ElectricFlow Apache server starts.

32-bit libraries are not required for the "pure" 64-bit full installer or the "pure" 64-bit agent-only installer.

Unsupported Linux Platforms

For platforms such as Debian, CentOS, or Fedora, install the following 32-bit libraries before installing ElectricFlow. They are required by the ElectricFlow installation executable file. Electric Cloud recommends installing *all* of these libraries on your 64-bit machines.

1. libstdc++.i686: If you do not install this, the ElectricFlow Apache server will not start, and the installer silently fails for any type of ElectricFlow installation.
2. libuuid.i686:Install this if you are performing an ElectricFlow installation that includes an Apache server. If you are installing ElectricFlow agents only, without a web server, you do not need to run this command on each agent machine.
3. nss-pam-ldapd*.i686: Install the 32-bit NSS packages if you are using an LDAP account for ownership of the server, web, or repository services. If you do not run this command, the ElectricFlow Apache server fails to start.

Installing or Uninstalling Without Root/sudo or Administrator Privileges

Certain ElectricFlow installers allow you to perform installations as a non-root user or a user without `sudo` privileges. To determine whether a particular installer has an option to run in this mode, see Availability of Installers with a Non-Root/Non-sudo or Non-Administrator Mode on page 3-2.

The installer writes installation data to the home directory of the user who invoked the installer. By default, the installer checks whether the `HOME` environment variable is defined and points to a writeable directory. The installer will read this data during subsequent upgrades or uninstallations.

Therefore, if you anticipate a same-system future upgrade (or uninstallation), you must ensure that you have a home directory before invoking the installer. If you do *not* plan to upgrade, you must use the `--skipCheckUserHomeDirectory` installer argument to ensure that the installer finishes successfully.

Remote ElectricFlow Web Server Installation Prerequisites

A remote web server configuration helps prevent network latency. If you have multiple sites, ElectricFlow can be configured in numerous ways to help you work more efficiently. For details about the architecture for this configuration as well as a discussion of the benefits of using a central web server and web servers at each remote site, see Remote Web Server Configuration on page 1-6.

Web Server Platform and Memory Requirements

You can install an ElectricFlow web server on any Windows or Linux platform suitable for installing the ElectricFlow server. For platform requirements, see Supported Server Platforms on page 2-1.

The memory settings for the agent on each web server machine must be higher than the default agent settings. More memory is typically needed for streaming large log files and so on. For agent memory requirements and instructions for configuring agent memory, see the *KBEC-00248 - Agent Memory Configuration* KB article.

Local Agent Installation Requirement for Web Server Machines

Every local or remote web server requires a local agent (that is, an agent on that machine) to be present to enable communication with the ElectricFlow server or other agents. Whenever any web server is installed, a local agent is also installed because:

- Each web server delegates all requests to the ElectricFlow server to its local agent, which then knows how to forward the request to the ElectricFlow server.
- If a web server must render the step log from a remote agent to the browser, it delegates the request to its local agent. The local agent then asks the ElectricFlow server for a route to reach the remote agent and the location of the step log, so that the step log can then be streamed from the remote agent.

Note: You should not use these local agents to run jobs.

Plugins Directory Accessibility Requirement for Web Server Machines

A plugin is a collection of one or more features or a third-party integration or tool that can be added to ElectricFlow. The ElectricFlow server installs all plugins into a configurable location named the plugins directory. This directory must be readable by the web server and any agents that need access to the content of one or more plugins.

There are two ways to make the plugins directory readable by the web server and any agents. You can configure the ElectricFlow server, agents, and web servers to point to a central network location, or you can replicate the contents of the plugins directory on remote agents and web servers.

Electric Cloud strongly recommends that all server machines in a remote web server configuration be able to access a common plugins directory in a central network location. This avoids the overhead of managing multiple plugins directories. For details, see Configuring Universal Access for a Network Location on page 5-21

Requirements for Non-Root or Docker DevOps Insight Installations on Linux Platforms

You typically perform a) installation as root, which gives the installer all the required permissions to change certain operating system settingsas needed. If you will be performing an installation as a non-root user or in a Docker environment, you must change these settings manually.

Checking the Virtual Memory Areas Setting

1. If the value displayed is less than 262144, a
2. by entering the following command
3. by entering

Run the following command as the user to be used for non-root installation:

```
$ /sbin/sysctl vm.max_map_count

vm.max_map_count = 262144
```

If the retrieved value is less than 262144, then this environment is not compatible with ElecteicFlow DevOps Insight Server. The setting must be increased at least to this value. It can be done by these steps:

Add this line to the `/etc/sysctl.conf` file:

```
vm.max_map_count = 262144
```

Apply the settings by this command using the root account:

```
$ sudo /sbin/sysctl -p
```

Verify that the following variable has the required value:

```
$ /sbin/sysctl vm.max_map_count

vm.max_map_count = 262144
```

For more information, see https://www.elastic.co/guide/en/elasticsearch/reference/current/vm-max-map-count.html.

Checking the Maximum Number of Open Files Descriptors Setting

1. If the value displayed is less than 65536, a
2. by entering the following command

Run the following command as the user that will be used for the non-root installation:

```
$ ulimit -n

65536
```

If the retrieved value is less than 65536, then this environment is not compatible with the DevOps Insight server. The setting must be increased at least to this value. It can be done by these steps:

Add the following lines to the `/etc/security/limits.conf` file:

```
* soft nofile 65536

* hard nofile 65536
```

These settings will change the values for all users in system. To change the settings only for the user to be used for ElectricFlow DevOps Insight server installation, replace the asterisks in the above lines by that username.

Log back into the system.

Verify that the setting has the required value:

```
$ ulimit -n

65536
```

For more information, see https://www.elastic.co/guide/en/elasticsearch/reference/current/file-descriptors.html.https://www.elastic.co/guide/en/elasticsearch/reference/current/file-descriptors.html

Checking the Number of Threads Setting

1. by entering the following command
2. If the value displayed is less than 4096, a

   s
3. by entering the following command

Run the following command as the user to be used for non-root installation:

```
$ ulimit -u

4096
```

If the retrieved value is less than 4096, then this environment is not compatible with the DevOps Insight server. The setting must be increased at least to this value. It can be done by these steps:

Add the following lines to the `/etc/security/limits.conf` file:

```
* soft nproc 4096

* hard nproc 4096
```

These settings will change the value for all users in the system. To change the settings only for one user which will be used for ElectricFlow DevOps Insight Server installation replace asterisks in the above lines by needed username.

Log back into the system.

Verify that the setting has the required value:

```
$ ulimit -u

4096
```

> **Note:** Note: If the login shell is the dash shell, then you must use the `ulimit -p` command to check this setting.

For more information, see https://www.elastic.co/guide/en/elasticsearch/reference/current/max-number-of-threads.html.

Default Installation Directories

ElectricFlow uses the following default installation directories:

Platform	Data Type	Default Path
Windows	Program files	C:\Program Files\Electric Cloud\ElectricCommander
	Data (database, logs, configuration files)	Windows 2008 or Windows 7: C:\ProgramData\Electric Cloud\ElectricCommander
UNIX and macOS	All program files and data	/opt/electriccloud/electriccommander

Note: You can change the installation directories when you install the ElectricFlow software.

Graphical User Interface Installation Methods

The graphical user interface installation methods are supported by Windows platforms and Linux platforms running the X Window System.

Running an Express Server Graphical User Interface Installation

The express server installation installs the ElectricFlow server, including the web server, built-in database, agent (for running jobs), and ElectricFlow tools. Review Before You Install ElectricFlow on page 3-7 before performing this procedure.

The built-in database is not supported in a clustered ElectricFlow configuration.

1. (Linux only) Enter the following command to make the installer file executable:

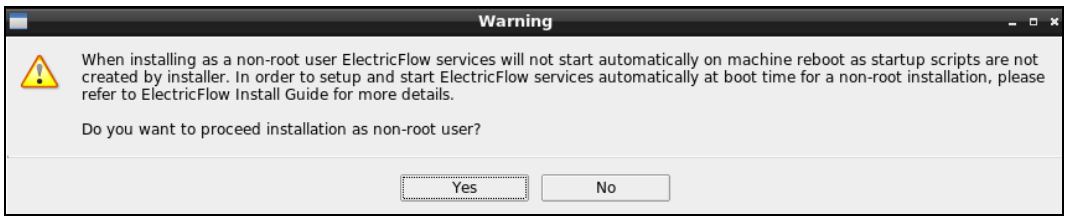
   ```
   chmod +x ./ElectricFlow-<version>
   ```

2. Do one of the following to start the installation:

   - For Linux with root or `sudo` privileges or for Windows installations, double-click the installer file.
   - For Linux non-root/non-`sudo` installations, enter:

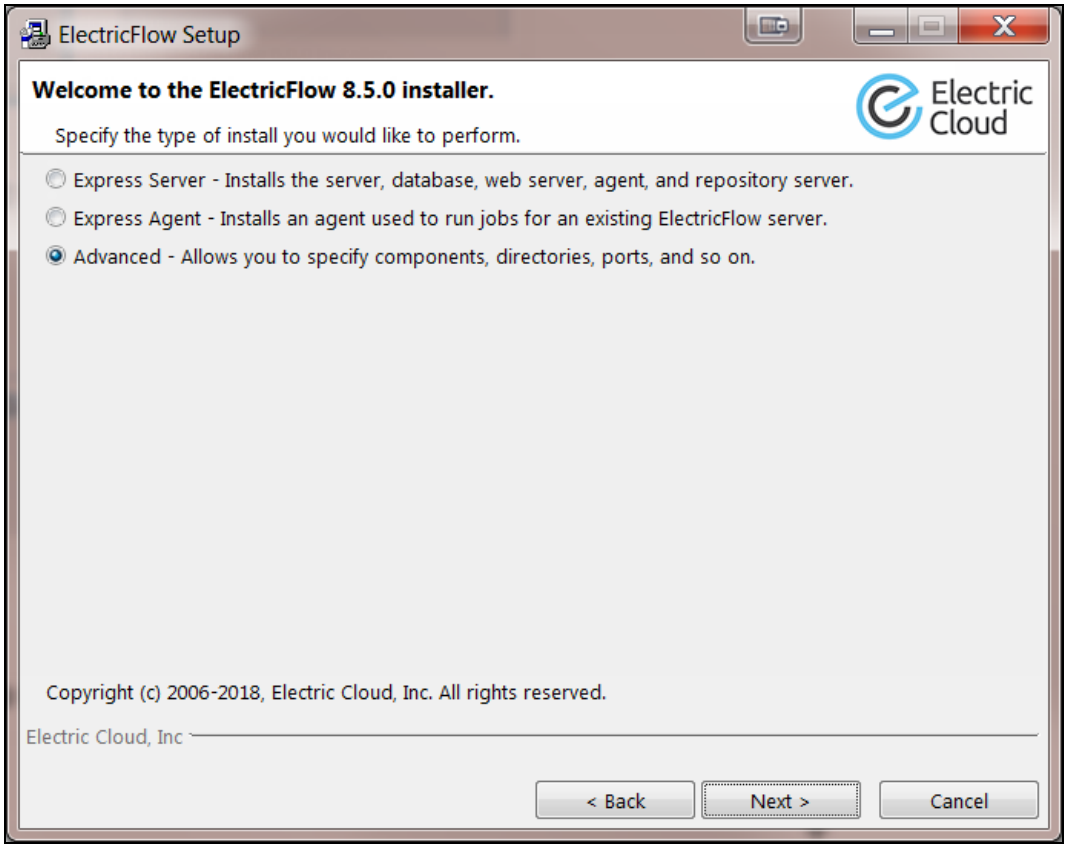
     `./ElectricFlow-<version> --nonRoot`

     For this installation type, the following warning appears:



3. For non-root/non-`sudo` installations, click **Yes** to dismiss the warning.

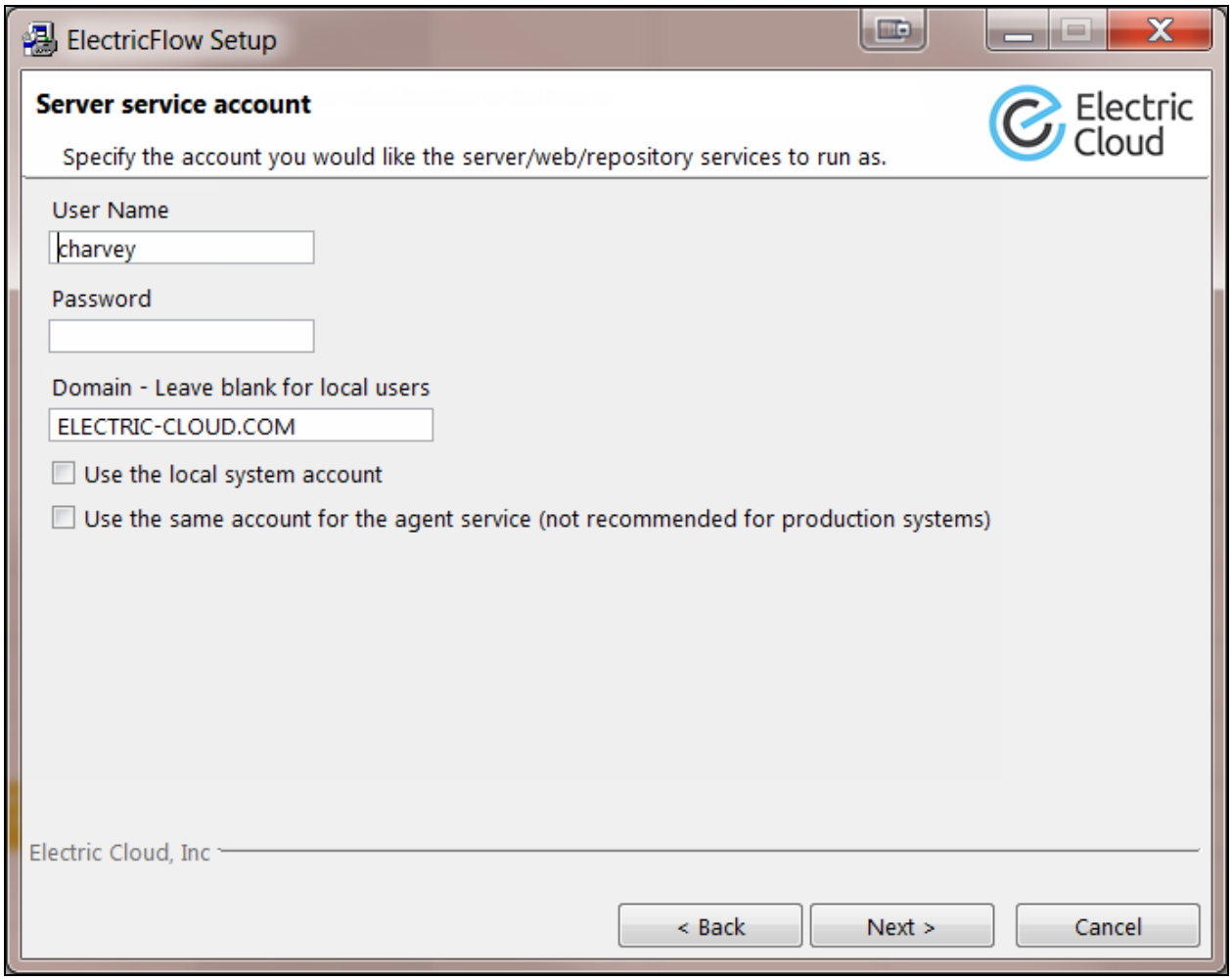
   The **Welcome to the ElectricFlow Installer** screen appears:



> **Note:** The screen examples in this procedure are from a Windows system. Different options will appear in some windows on a Linux system.

4. Select the **Express Server** installation option, and then click **Next** to continue.

The **Server service account**screen appears:



5. Select the appropriate step for your platform and complete the information for the server service account.
   - Windows:
     - **User Name**—Enter the name of the user who will run the ElectricFlow server, web server, and repository server services.
     - **Password**—Enter the password of the user who will run the ElectricFlow server, web server, and repository server services.
     - **Domain**—Enter the domain name information for the user. For example, electric-cloud.com. Leave this field blank if this is a local user.

- **Use the local system account**—Select this check box if you want the ElectricFlow server, repository server, and web server services to run as the local Windows system account.

  > **Note:**
  > The Windows local system account cannot access network resources such as shared file systems used for plugins or workspaces. Therefore, do not use this option for a clustered server deployment, which requires a shared file system for plugins. This option is typically used only for installing agents on numerous machines, which would otherwise require that you create a new account on each of those machines.

- **Use the same account for the agent service**—Select this check box if you want the agent on the ElectricFlow server machine to run as the same account.

  For security reasons in production environments, you should use a separate account for the agent service because the server account has permission to read the key file (`/opt/electriccloud/electriccommander/conf/passkey` in Linux or `C:\ProgramData\Electric Cloud\ElectricCommander\conf\passkey` in Windows). The key file is used to decrypt passwords stored in ElectricFlow. Using a different account for the agent service ensures that a process running on the agent cannot gain access to the key file.

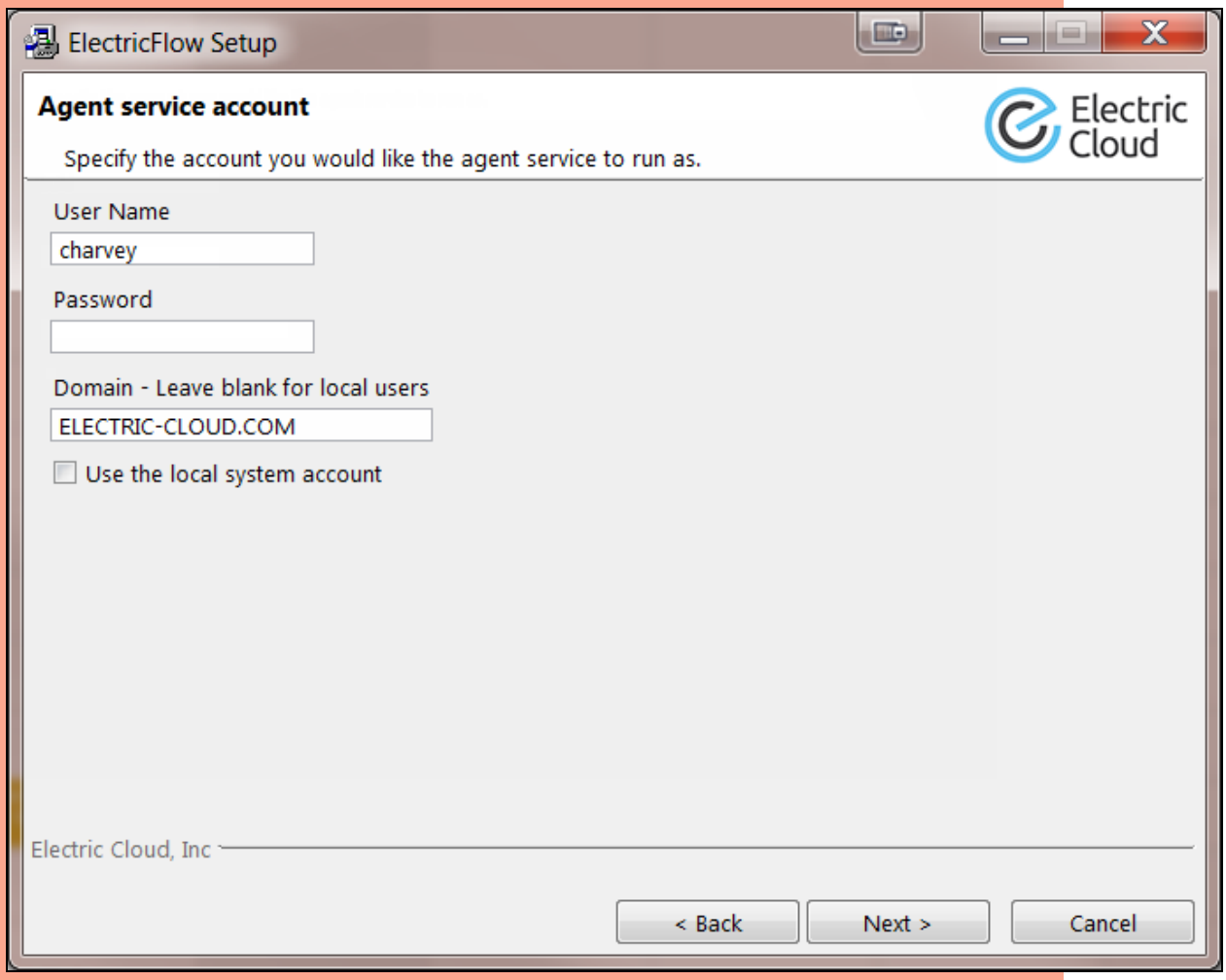
- Linux:
  - **User Name**—Enter the name of the user who owns the ElectricFlow server, repository server, and web server processes.
  - **Group Name**—Enter the name of the group who owns the ElectricFlow server, repository server, and web server processes.
  - **Use the same account for the agent service**—Select this check box if you want the same user and group to own the agent process on the ElectricFlow server machine.

    For security reasons in production environments, you should use a separate user and group for the agent service because the server service has permission to read the key file (`/opt/electriccloud/electriccommander/conf/passkey` in Linux or `C:\ProgramData\Electric Cloud\ElectricCommander\conf\passkey` in Windows). The key file is used to decrypt passwords stored in ElectricFlow. Using a different user and group for the agent service ensures that a process running on the agent cannot gain access to the key file.

5. Click **Next** to continue.

   The **Agent Service Account** screen appears.

Important: If you selected the **Use the same account for the agent service** check box on the previous screen, you will not see this screen.



6. Select the appropriate step for your platform and complete the information for the agent service account.

   - Windows:

     - **User Name**—Enter the name of the user who will run the ElectricFlow agent service.

       The user that the agent runs as must have permission to write to the `$INSTALL_DIRECTORY/log` directory.

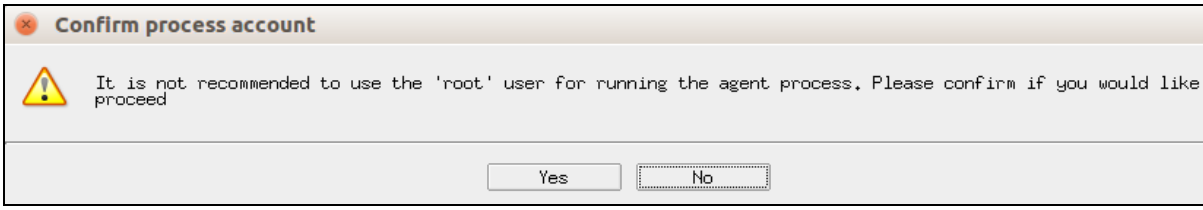
     - **Password**—Enter the password of the user who will run the ElectricFlow agent service.
     - **Domain**—Enter the domain name information for the user. For example, electric-cloud.com. Leave this field blank if this is a local user.

- **Use the local system account**—Select this check box if you want the ElectricFlow agent service to run as the local Windows system account.

> **Note:** The local system account does not have access to network shares.

- Linux:
  - **User Name**—Use this field to enter the name of the user who owns the ElectricFlow agent process.

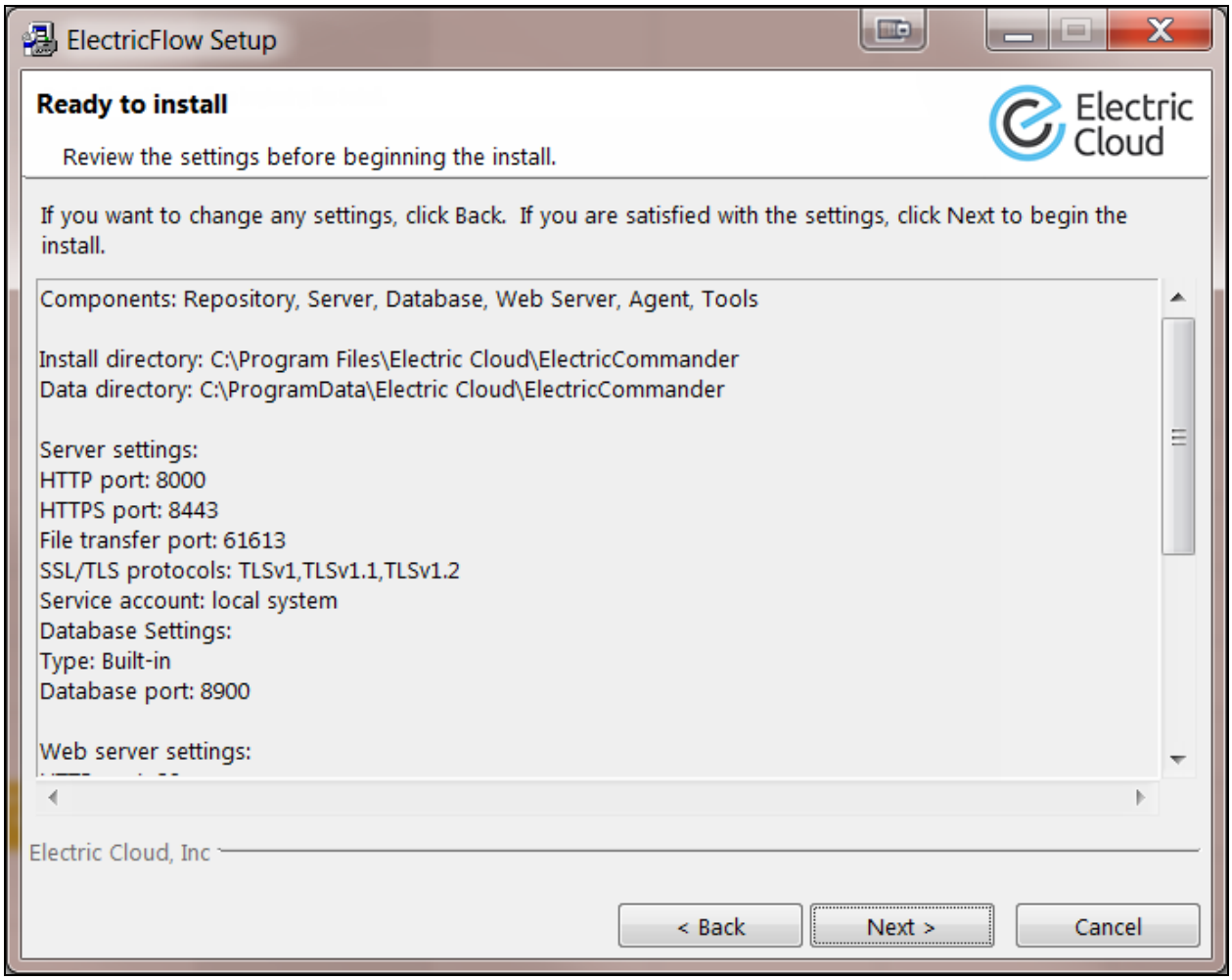
    The user/group that the agent runs as must have permission to write to the `$INSTALL_DIRECTORY/log` directory. If you specify `root`, click **Yes** when the following confirmation appears:



  - **Group Name**—Use this field to enter the name of the group that owns the ElectricFlow agent process.

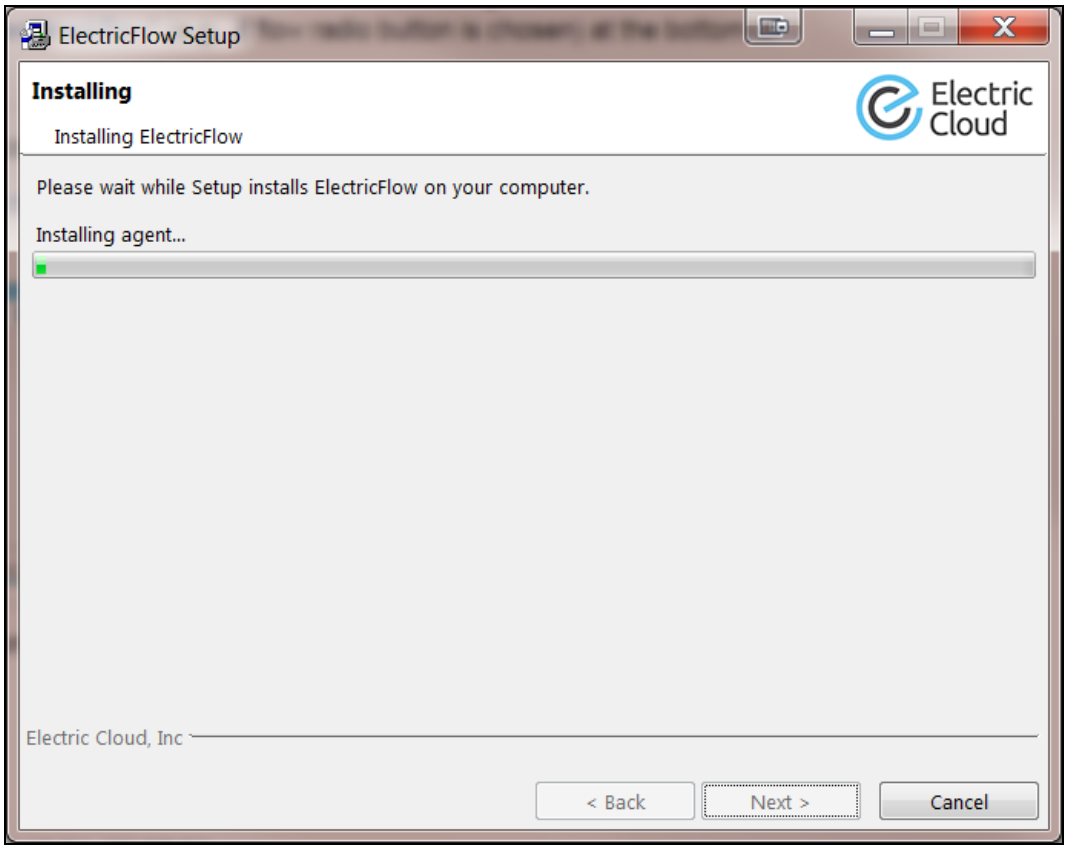
7. Click **Next** to continue.

The **Ready to Install** screen appears:

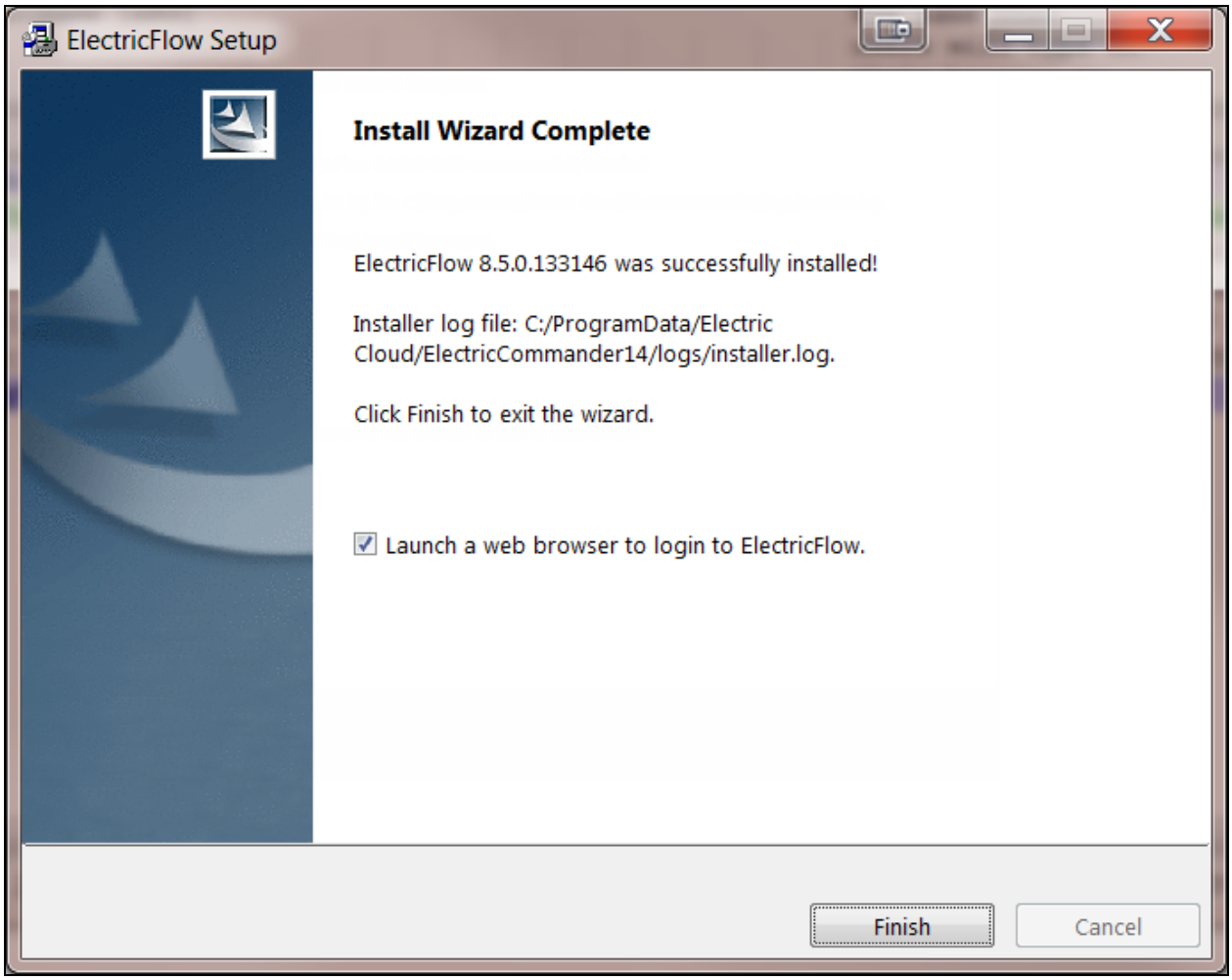


8. Review the default settings and your service account selections. Use the **Back** button to change your service account selections if necessary.
9. Click **Next** to continue.

   The installer displays a status bar to show the progress of the installation, which can take fifteen minutes:



When the install process is complete, the **Install Wizard Complete** screen appears:



Note: The ElectricFlow server will automatically start when the installation is complete.

10. Select the **Launch a web browser to login to ElectricFlow** check box if you want ElectricFlow to open the login screen now.
11. Click **Finish** to close the wizard.
12. For non-root/non-`sudo` Linux installations, configure autostart for the ElectricFlow services.

    For instructions, see Configuring Services Autostart for Non-Root/Non-sudo Linux Installations on page 5-11.

Running an Advanced Graphical User Interface Installation

The advanced graphical user interface installation lets you install individual ElectricFlow components such as an ElectricFlow server, built-in database, web server, repository server, or ElectricFlow tools on specific machines. You can also change the default installation settings to accommodate your environment. Review Before You Install ElectricFlow on page 3-7 before performing this procedure.

1. (Linux only) Enter the following command to make the installer file executable:

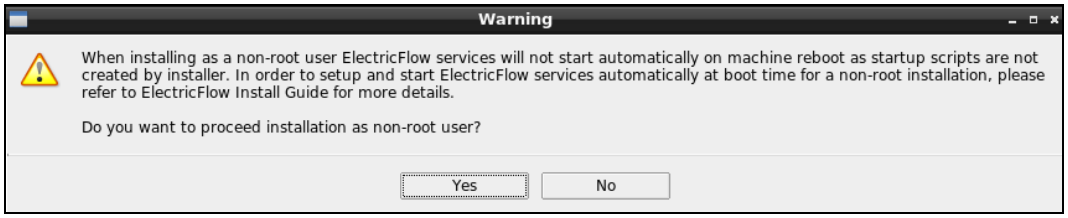
   ```
   chmod +x ./ElectricFlow-<version>
   ```

2. Do one of the following to start the installation:

   - For Linux with root or `sudo` privileges or for Windows installations, double-click the installer file.
   - For Linux non-root/non-`sudo` installations, enter:

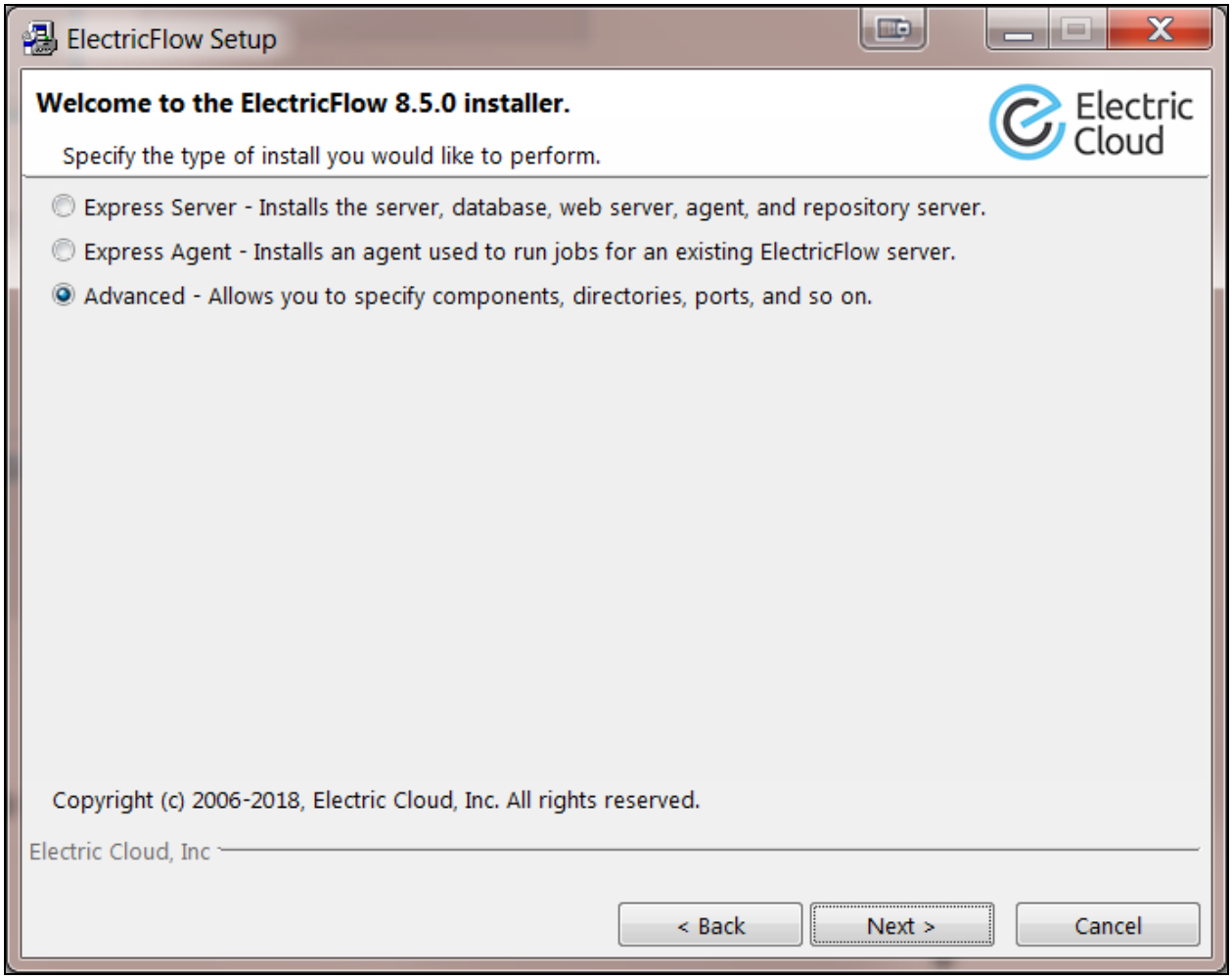
     ```
     ./ElectricFlow-<version> --nonRoot
     ```

     For this installation type, the following warning appears:



3. For non-root/non-`sudo` installations, click **Yes** to dismiss the warning.

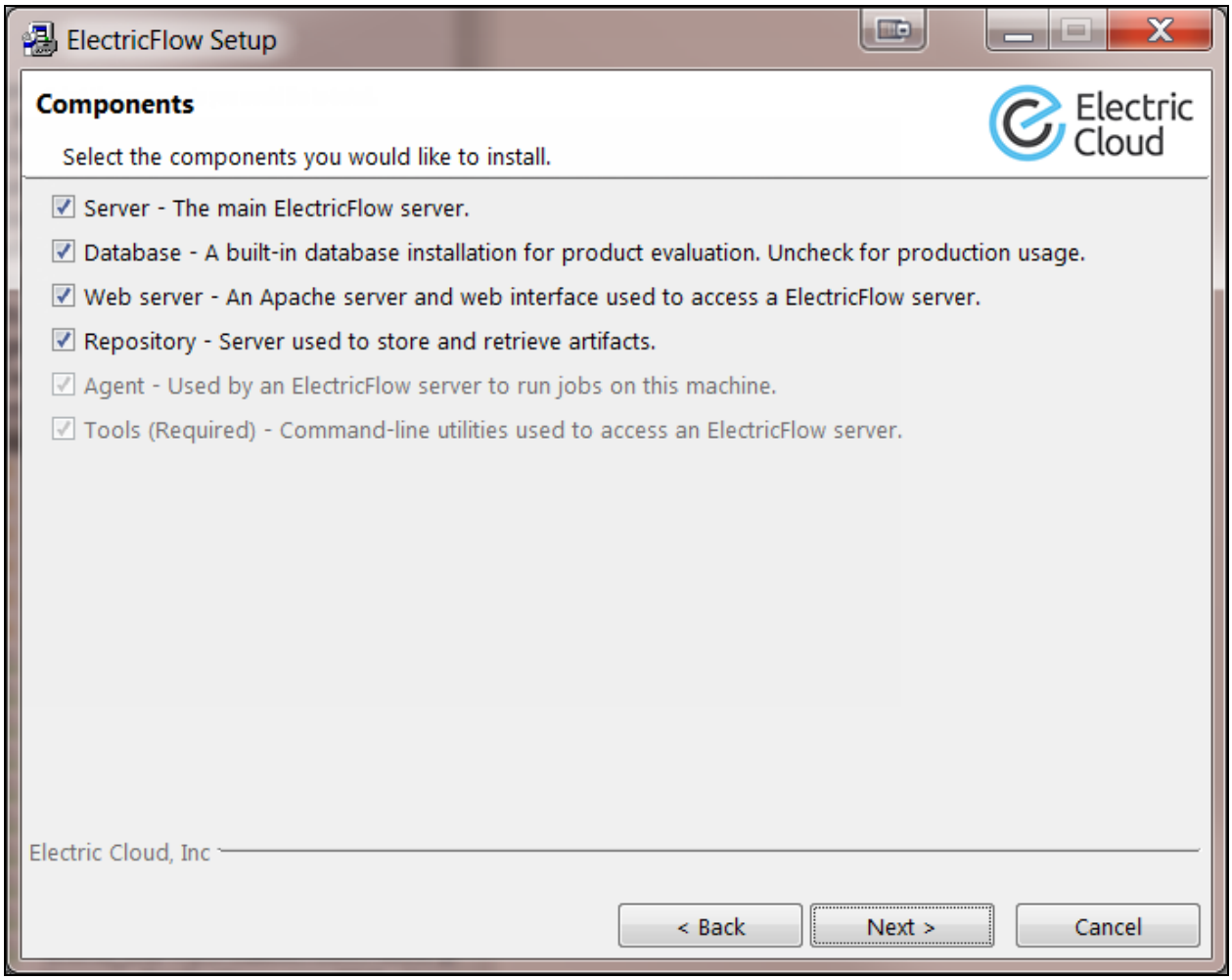
   The Welcome to the ElectricFlow Installer screen appears:



> **Note:** The screen examples in this procedure are from a Windows system. Different options will appear in some windows on a Linux system.

4. Select the **Advanced** installation option, and then click **Next** to continue.

   The Components screen appears. All options are selected by default:



5. Clear the check boxes for servers that you do *not* want to install. For details, see Architecture on page 1-4.

   Available options are:

   - **Server**—Installs an ElectricFlow server.

     > **Note:** If you uncheck this check box, the **Remote ElectricFlow Server** screen appears later (shown below).

   - **Database**—Installs the built-in database. This is not recommended for production systems. Also, the built-in database is not supported in a clustered ElectricFlow configuration. Clear this check box if you plan to use an external database. If you plan to use MySQL, see Installing the MySQL JDBC Driver on page 3-181.

- **Web server**—Select this check box if you want to install an Apache web server. If you select this option, an agent is also required on this machine and is therefore automatically installed. For details about why local agents are required on web server machines, see Local Agent Installation Requirement for Web Server Machines on page 3-11.

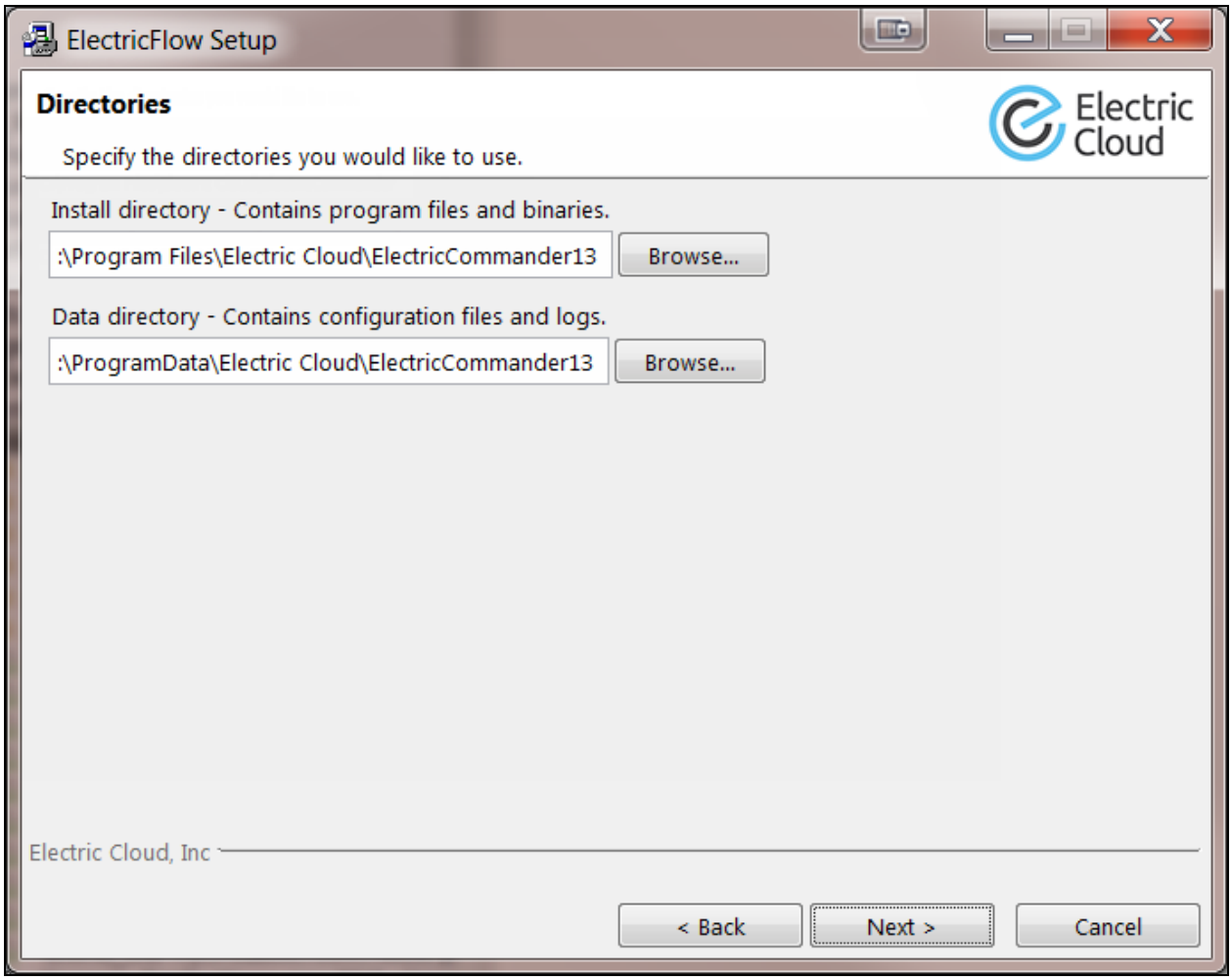
  > **Note:** You should not use these local agents to run jobs.

- **Repository**—Installs an ElectricFlow repository server. If you select this option, an agent is also installed.
- **Agent**—Installs ElectricFlow agent software.
- **Tools**—Installs ElectricFlow tools. To install only the ElectricFlow tools, clear all the check boxes. This option does not automatically install an ElectricFlow agent, unlike the other options.

  > **Note:** Any combination of the following installation screens will appear depending on which servers you install.

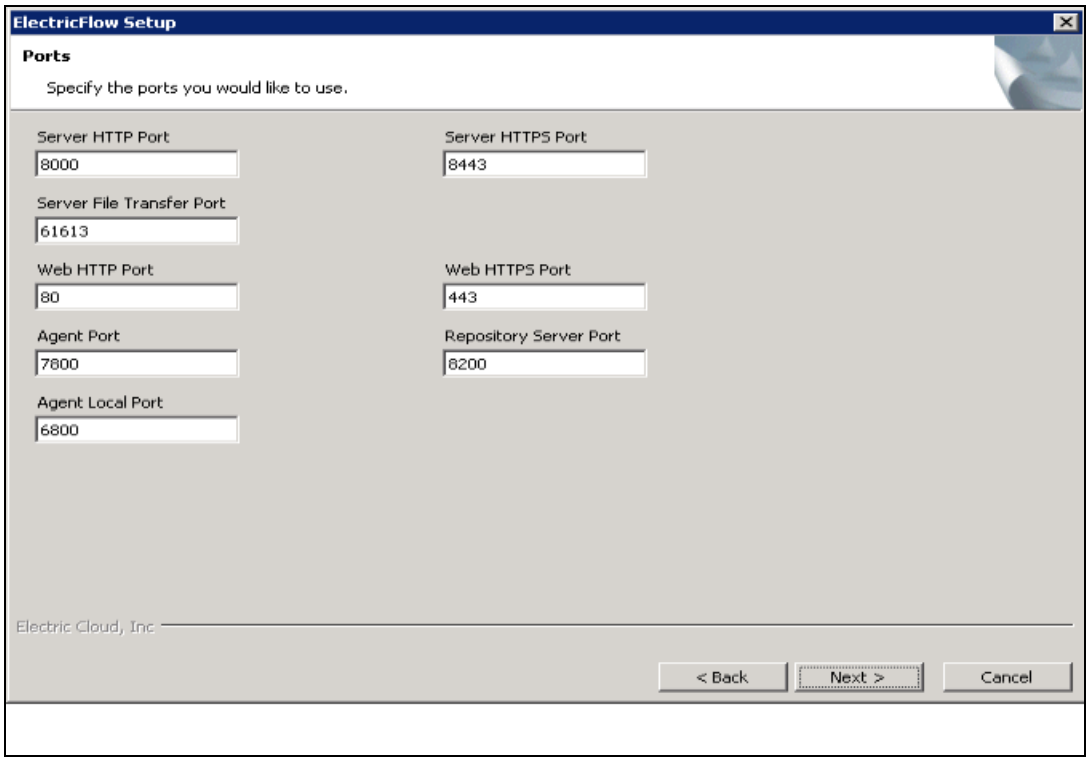
6. Click **Next**.

   The **Directories** screen appears. ElectricFlow uses the default directories to install files and components:



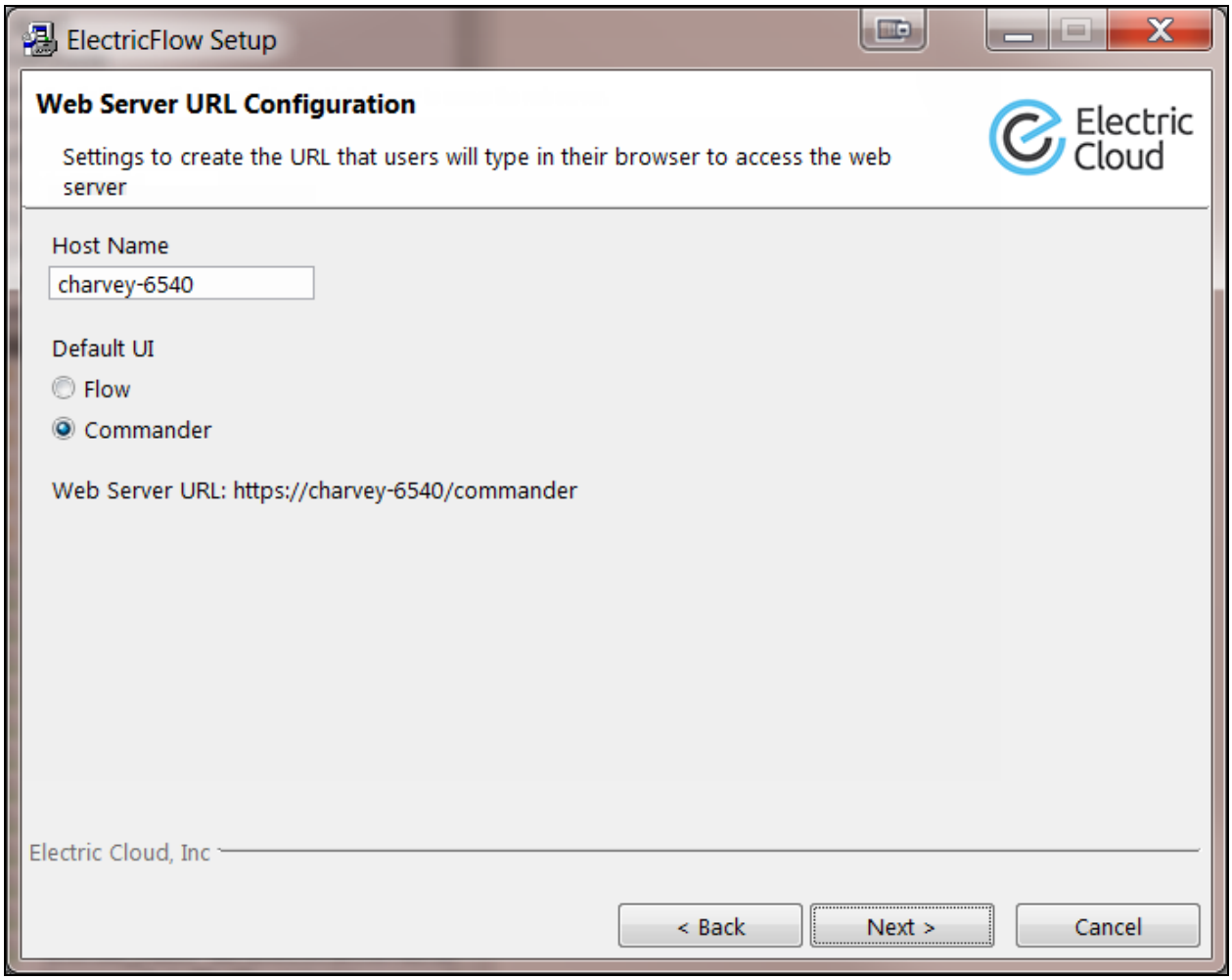
7. Click **Next** to continue, or click **Browse** to specify different directory locations.

   The **Ports** screen with the default ElectricFlow port values appears if you are installing an ElectricFlow, web, or repository server:



8. Complete the information for the **Ports** screen, and click **Next** to continue. You can enter alternate port numbers if you need to specify different port values.

   The **Web Server URL Configuration** screen appears if you are installing a web server:

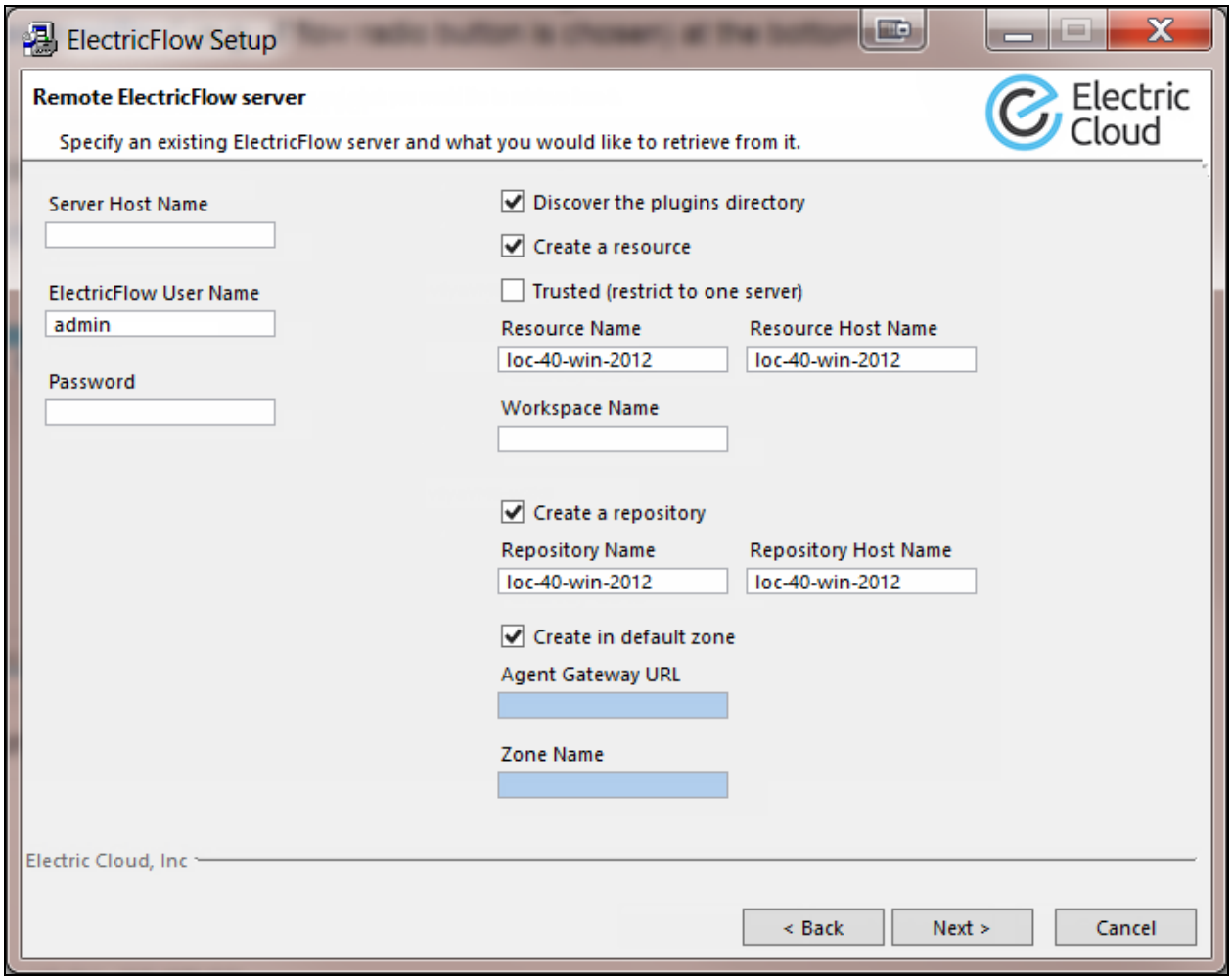


9. Complete the information for the **Web Server URL Configuration** screen, and click **Next** to continue.

   - **Host Name**—Name that users must enter in their browser to access the ElectricFlow web server.
   - **Default UI**—Determines whether the Deploy UI or the Automation Platform UI appears when users browse to https://<*ElectricFlow_server*> without appending /flow or /commander respectively to the end of the URL. For example, you can configure ElectricFlow so that it opens the Deploy UI whether you browse to https://ecdevopsserver1 or https://ecdevopsserver1/flow.

     You can reconfigure this behavior post-installation by using the ecconfigure --`webDefaultUI` option. For details, see the "ecconfigure" section in the "Automation Platform" chapter of the *ElectricFlow User Guide* at http://docs.electric-cloud.com/eflow_doc/FlowIndex.html.

   If you unchecked the **Server** check box above, the **Remote ElectricFlow Server** screen appears.



10. Complete the following information on the **Remote ElectricFlow Server** screen:

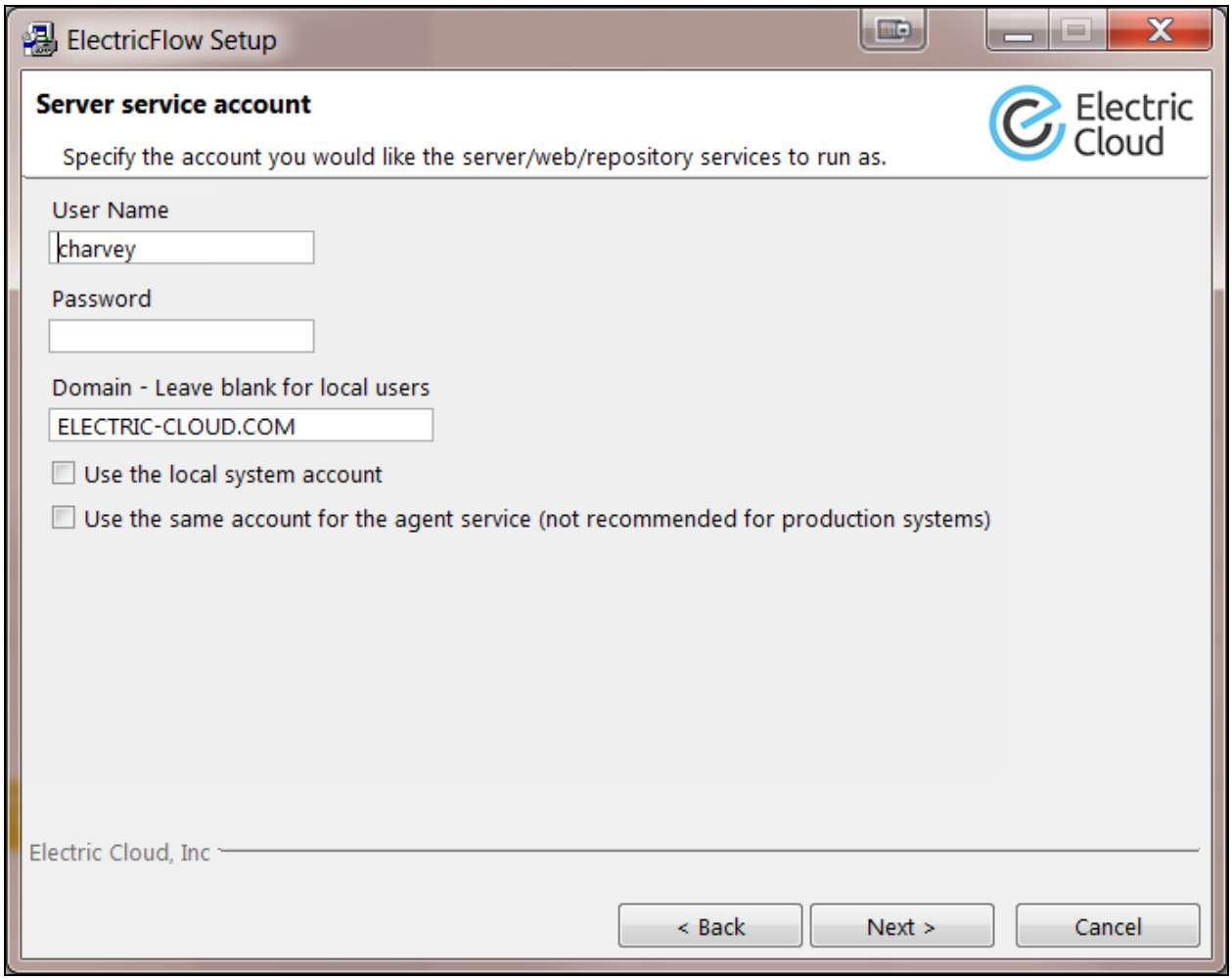
    - **Server Host Name**—Use this field to enter the name of the ElectricFlow server that will communicate with this web server. If the remote server is using a non-default HTTPS port, you must specify the Server Host Name as `<host>:<port>`. If you do not specify a port, HTTPS port 8443 is assumed (the same as the ElectricFlow server default port).
    - **ElectricFlow User Name**—Use this field to enter the name of an ElectricFlow user on the ElectricFlow server who has sufficient privileges to create a resource. This field defaults to the ElectricFlow-supplied `admin` user.
    - **Password**—Use this field to enter the password for the ElectricFlow user. The default password for the `admin` user is `changeme`.
    - **Discover the plugins directory**—Select this check box if you want the web server machine to have access to the plugins directory. You should allow access to the plugins directory so agents have access to collections of features, third-party integrations, or third-party tools.

> **Note:** The plugins directory on the ElectricFlow server must be "shared" before the web server machine can use "discover" to find the directory. For more information, see Universal Access to the Plugins Directory on page 5-21

- **Create a resource**—Select this check box if you want to create a resource on the remote ElectricFlow server for the web server you are installing.
- **Trusted**—Select this check box to restrict this web server to one ElectricFlow server. The web server will not respond to incoming communication from any other ElectricFlow server. This is useful when you want to create a secure production environment, but generally not needed for test or development systems.
- **Resource name**—Use this field to enter the name of the name of the resource to use.
- **Workspace Name**—Use this field to enter the name of the workspace you would like to use for the web server.
- **Create a repository**—Create an artifact repository on this machine.
- **Repository name**—Name of the artifact repository to create.
- **Create in default zone**—Select this check box if you want to create the agent in the default zone.
- **Agent Gateway URL**—Use this field to enter the URL of the gateway used to communicate with the ElectricFlow server. This field is available for use when the Create in default zone check box is cleared.
- **Zone Name**—Use this field to enter the name of the zone used during remote agent and/or remote repository creation. This field is available for use when the Create in default zone check box is cleared.

11. Click **Next** to continue.

    The **Server Service Account** screen appears if you are installing an ElectricFlow, web, or repository server:



12. Complete the information on the **Server Service Account** screen, and click **Next** to continue.
    - Windows:
      - **User Name**—Use this field to enter the name of the user who will run the ElectricFlow server, web server, and repository server services.
      - **Password**—Use this field to enter the password of the user who will run the ElectricFlow server, web server, and repository server services.
      - **Domain**—Use this field to enter the domain name information for the user. For example, electric-cloud.com. Leave this field blank if this is a local user.

- **Use the local system account**—Select this check box if you want the ElectricFlow server, repository server, and web server services to run as the Windows local system account.

  > **Note:**
  > The Windows local system account cannot access network resources such as shared file systems used for plugins or workspaces. Therefore, do not use this option for a clustered server deployment, which requires a shared file system for plugins. This option is typically used only for installing agents on numerous machines, which would otherwise require that you create a new account on each of those machines.

- **Use the same account for the agent service**—Select this check box if you want the agent on the ElectricFlow server machine to run as the same account.

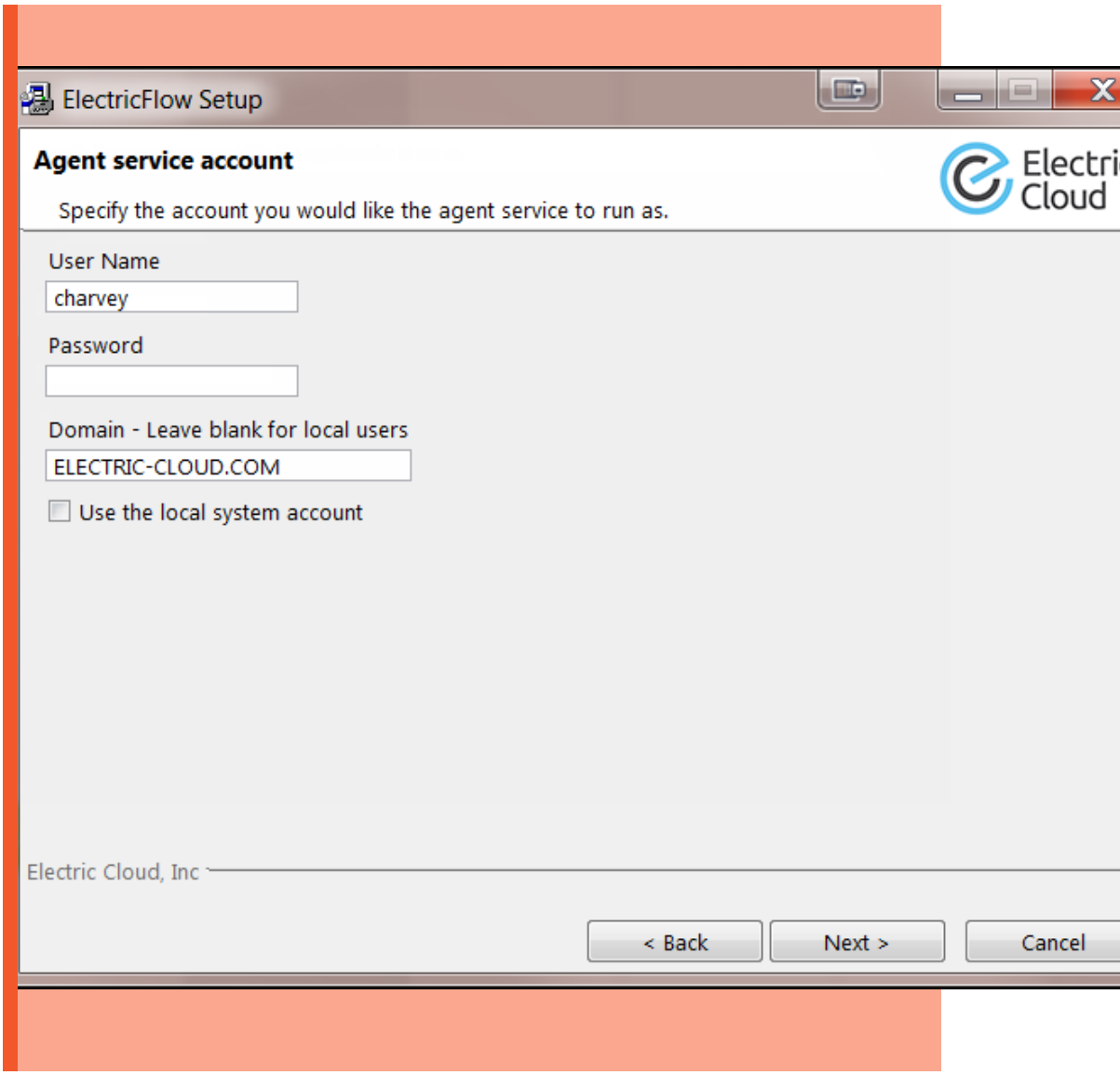
  For security reasons in production environments, you might want to use a separate account for the agent service because the server account has permission to read the key file (`/opt/electriccloud/electriccommander/conf/passkey` in Linux or `C:\ProgramData\Electric Cloud\ElectricCommander\conf\passkey` in Windows). The key file is used to decrypt passwords stored in ElectricFlow. Using a different account for the agent service ensures that a process running on the agent cannot gain access to the key file.

- Linux:
  - **User Name**—Use this field to enter the name of the user who owns the ElectricFlow server, repository server, and web server processes.
  - **Group Name**—Use this field to enter the name of the group who owns the ElectricFlow server, repository server, and web server processes.
  - **Use the same account for the agent service**—Select this check box if you want the same user and group to own the agent process on the ElectricFlow server machine.

    For security reasons in production environments, you might want to use a separate user and group for the agent service because the server service has permission to read the key file (`/opt/electriccloud/electriccommander/conf/passkey` in Linux or `C:\ProgramData\Electric Cloud\ElectricCommander\conf\passkey` in Windows). The key file is used to decrypt passwords stored in ElectricFlow. Using a different user and group for the agent service ensures that a process running on the agent cannot gain access to the key file.

    The Agent Service Account screen appears if you are installing an agent. An agent is automatically installed on the machine to run jobs if you are installing a web or repository server.

    > **Important:** If you selected the **Use the same account for the agent service** check box on the previous screen, you will not see the fields to supply your agent service account information.


ElectricFlow Setup
Agent service account
Specify the account you would like the agent service to run as.
Electri
Cloud
User Name
charvey
Password
Domain - Leave blank for local users
ELECTRIC-CLOUD.COM
Use the local system account
Electric Cloud, Inc
< Back
Next >
Cancel

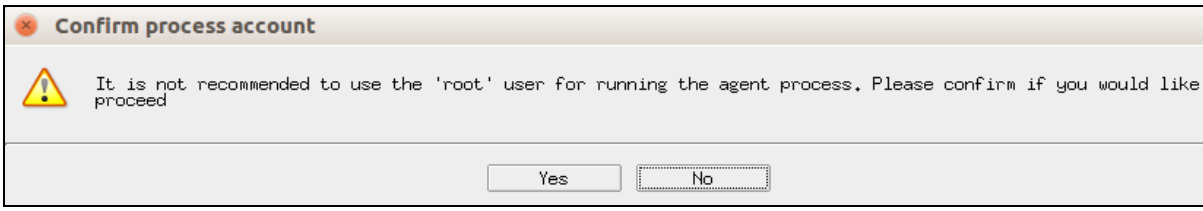
13. Complete the information on the **Agent Service Account** screen, and click **Next** to continue.

- Windows:
  - **User Name**—Use this field to enter the name of the user who will run the ElectricFlow agent service.

    The user that the agent runs as must have permission to write to the `$INSTALL_DIRECTORY/log` directory.
  - **Password**—Use this field to enter the password of the user who will run the ElectricFlow agent service.
  - **Domain**—Use this field to enter the domain name information for the user. For example, electric-cloud.com. Leave this field blank if this is a local user.
  - **Use the local system account**—Select this check box if you want the ElectricFlow agent service to run as the local Windows system account.

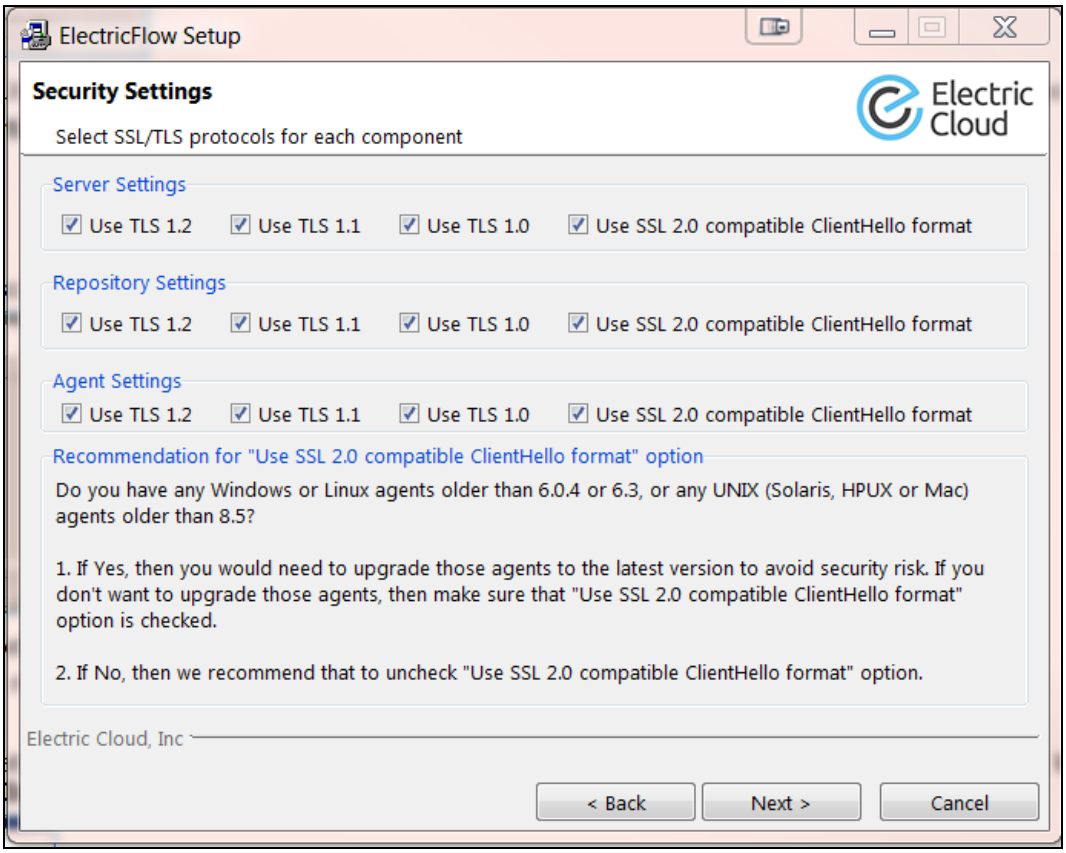
    Note: The local system account does not have access to network shares.
- Linux:
  - **User Name**—Use this field to enter the name of the user who owns the ElectricFlow agent process.

    The user/group that the agent runs as must have permission to write to the `$INSTALL_DIRECTORY/log` directory. If you specify `root`, click **Yes** when the following confirmation appears:

  - **Group Name**—Use this field to enter the name of the group that owns the ElectricFlow agent process.

    After you click **Next**, the **Security Settings** screen appears.



This screen specifies the list of SSL/TLS protocols that will be allowed for ElectricFlow server, repository server, and agent connections using HTTPS. The possible values are any combination of **TLSv1**, **TLSv1.1**, **TLSv1.2**, and **SSLv2Hello**. You must select at least one protocol for each connection.

The default security configurations are as follows:

- First-time ElectricFlow installations: TLSv1, TLSv1.1, and TLSv1.2 are enabled
- Existing ElectricFlow installations: TLSv1, TLSv1.1, TLSv1.2, and SSLv2Hello are enabled

The default for upgrades from version 8.5 and newer versions is to inherit the settings from the existing installation being upgraded.

To avoid the following warning in the Automation Platform web UI, we recommend removing the `SSL 2.0 Client Hello` or `SSLv2Hello` protocol from your security configurations for all components:

We recommend removing "SSL 2.0 Client Hello format from server configuration and upgrade older agents as indicated on the Cloud / Resources Page to avoid security risk.

To safely remove this protocol, enter the following command on the ElectricFlow server:

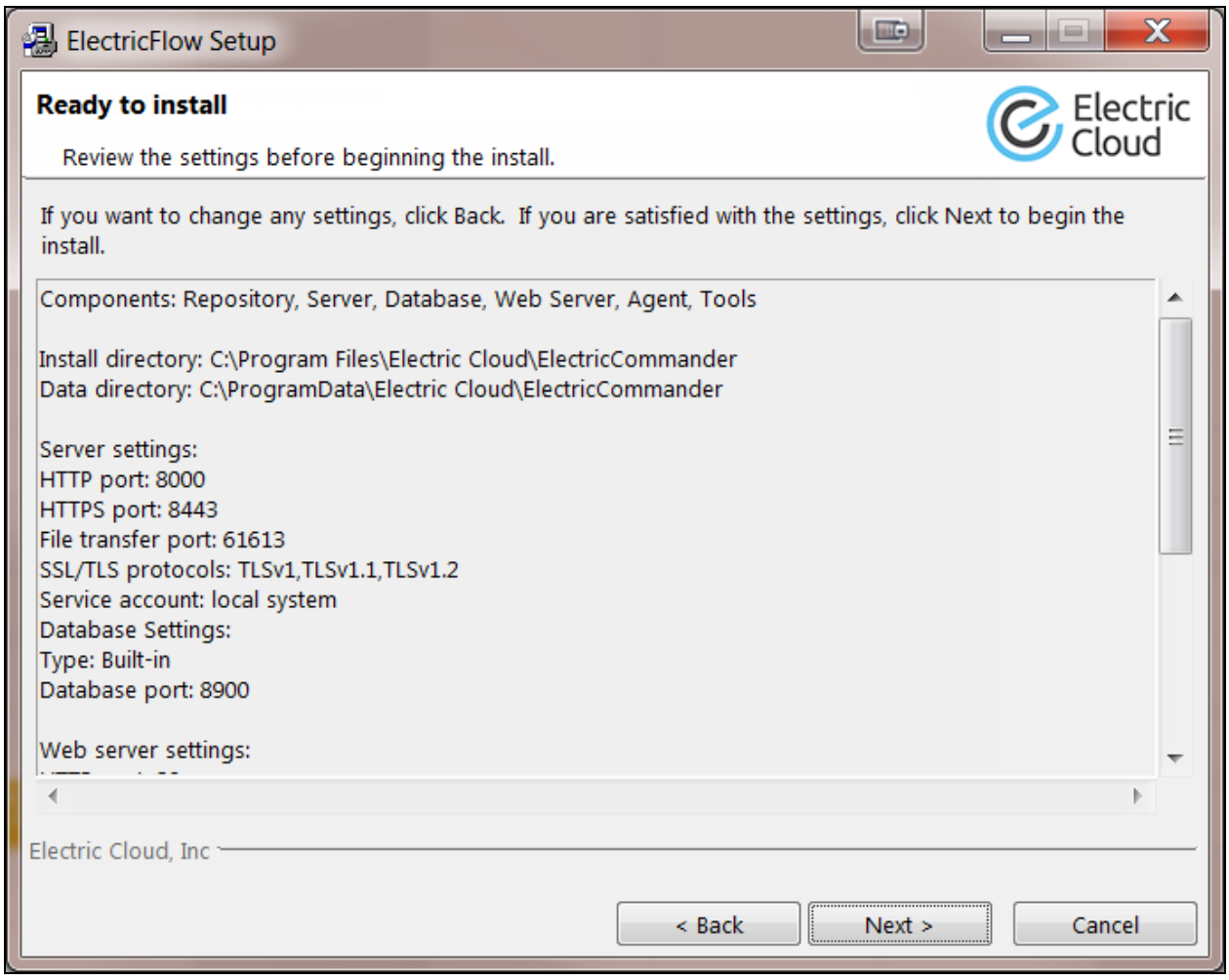
```
$ ecconfigure --serverTLSEnabledProtocol=TLSv1,TLSv1.1,TLSv1.2
```

When you do this, you would also need to upgrade older agents to the latest version to avoid security risks. You would need to upgrade agents if you are using the following agent versions:

- Windows, Linux: 6.0.3 or older; 6.2 or older
- Sun Solaris, HP UX, Mac OS: 8.4 or older

14. Complete the information in the **Security Settings** screen, and click **Next**.

    The **Ready to Install** screen appears.

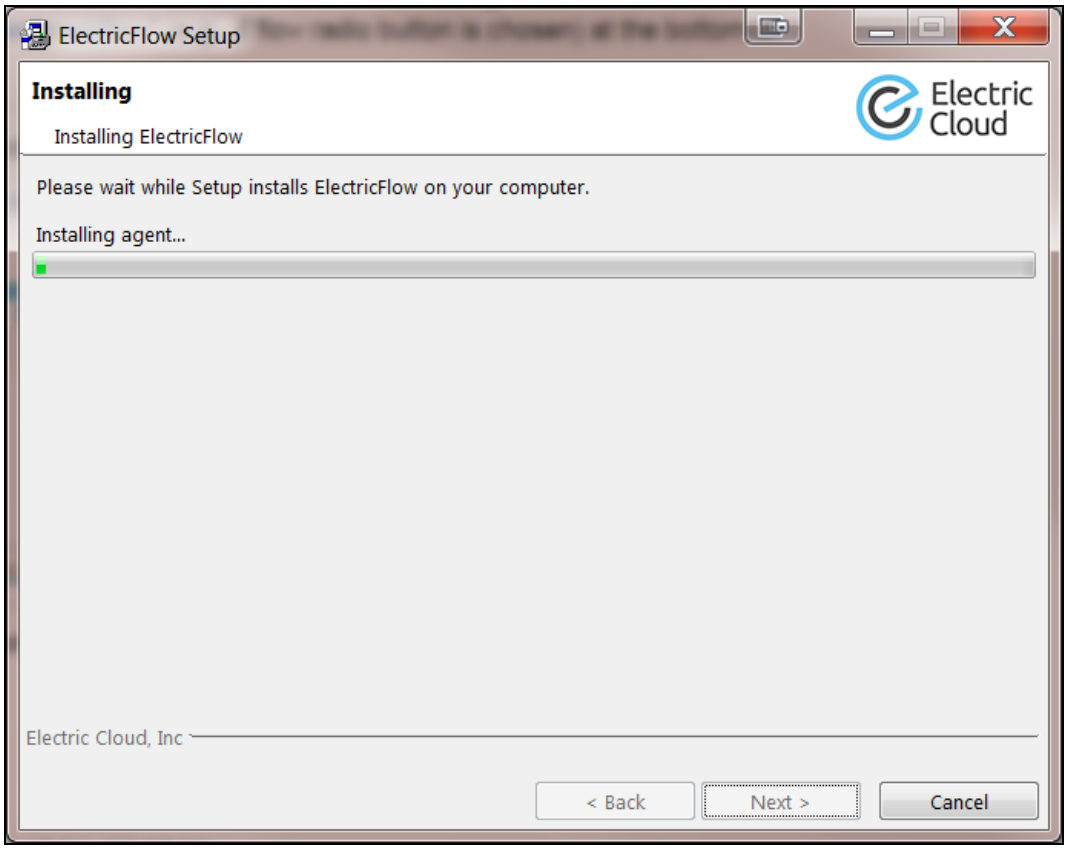


15. Review your installation settings.

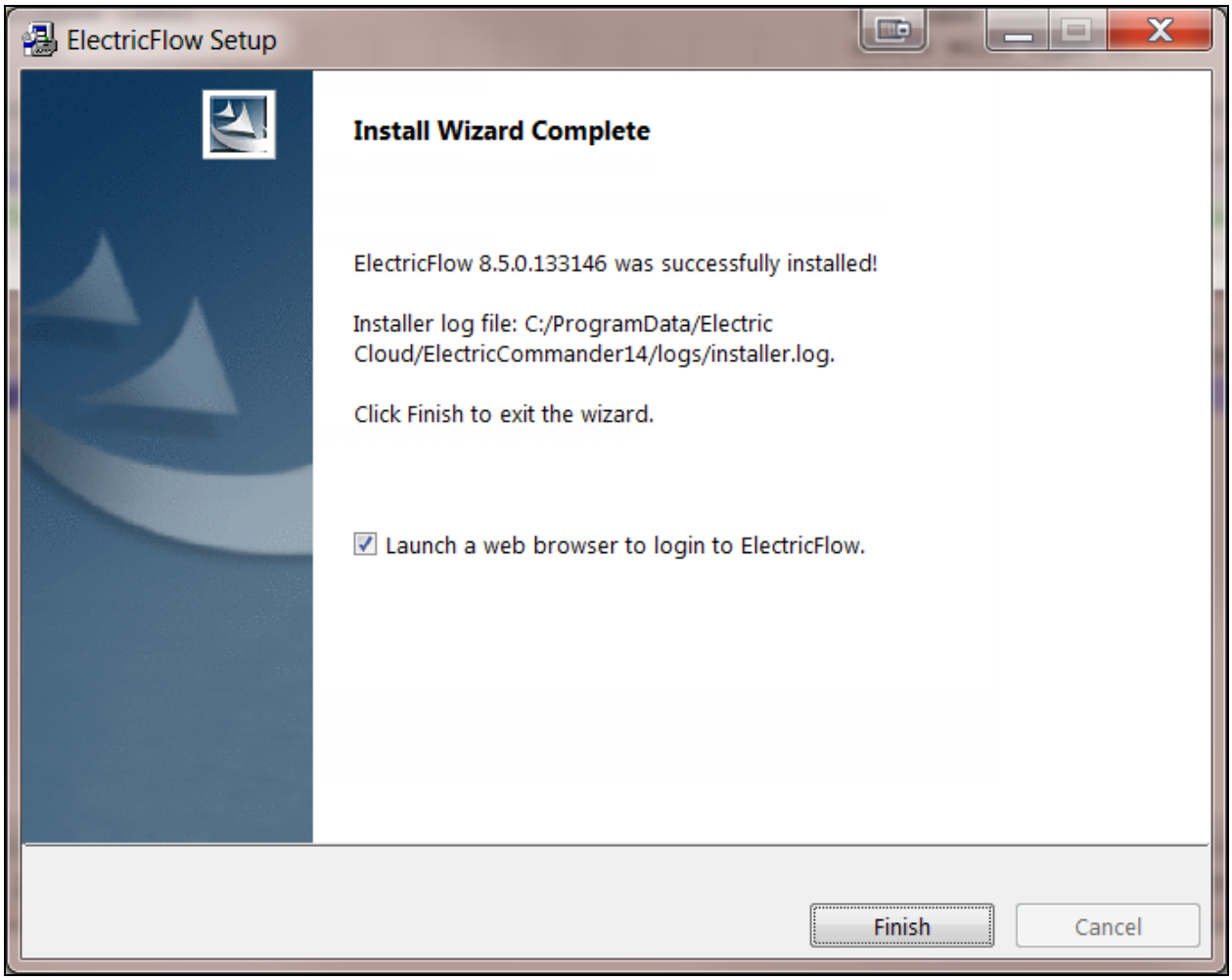
    Use the **Back** button to modify any information if necessary.

16. Click **Next** to continue.

    The installer displays a status bar to show the progress of the installation, which can take up to fifteen minutes:



When the install process is complete, the **Install Wizard Complete** screen appears:



Note: The ElectricFlow server will automatically start when the installation is complete.

17. Select the **Launch a web browser to login to ElectricFlow** check box if you want the ElectricFlow login screen to open.
18. Click **Finish** to close the wizard.
19. For non-root/non-`sudo` Linux installations, configure autostart for the ElectricFlow services.

    For instructions, see Configuring Services Autostart for Non-Root/Non-sudo Linux Installations on page 5-11.

Running an Express Agent Graphical User Interface Installation

The ElectricFlow agent software must be installed on each agent machine you intend to use with ElectricFlow. Review Before You Install ElectricFlow on page 3-7 before performing this procedure.

Note: You install ElectricFlow agent software on Windows or Linux with this installation method. For Solaris, HP-UX, macOS, AIX, or other supported UNIX-only agent machines, see Non-Server Platform Installation Method for UNIX Agents on page 14-30.

1. (Linux only) Enter the following command to make the installer file executable:

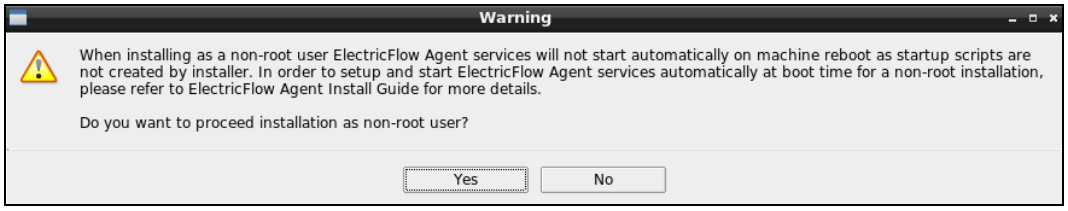
   ```
   chmod +x ./ElectricFlow-<version>
   ```

2. Do one of the following to start the installation:

   - For Linux with root or `sudo` privileges or for Windows installations, double-click the installer file.
   - For non-root/non-`sudo` installations, enter:

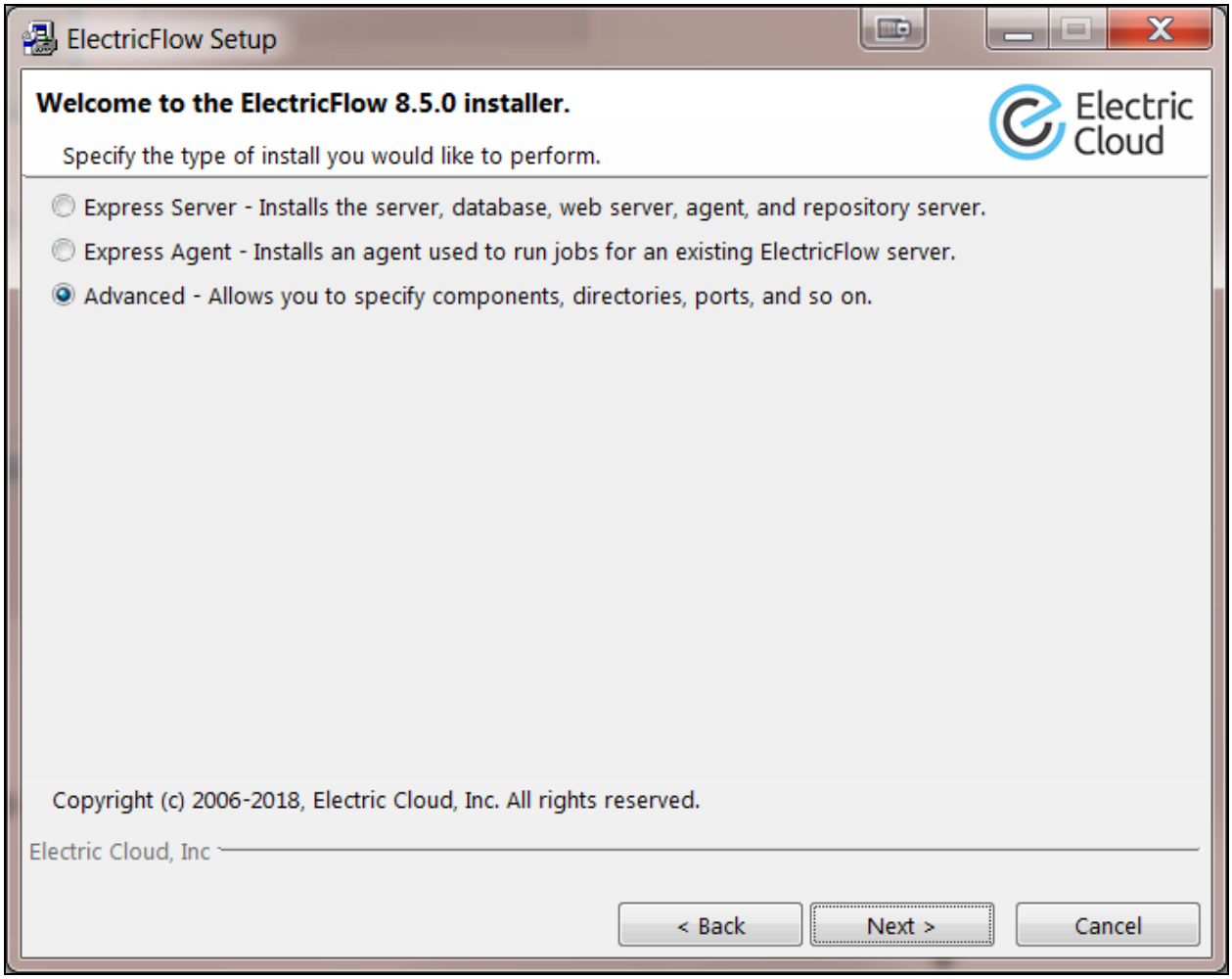
     ```
     ./<full_installer_file> --nonRoot
     ```

     For this installation type, the following warning appears:



3. For non-root/non-`sudo` installations, click **Yes** to dismiss the warning.

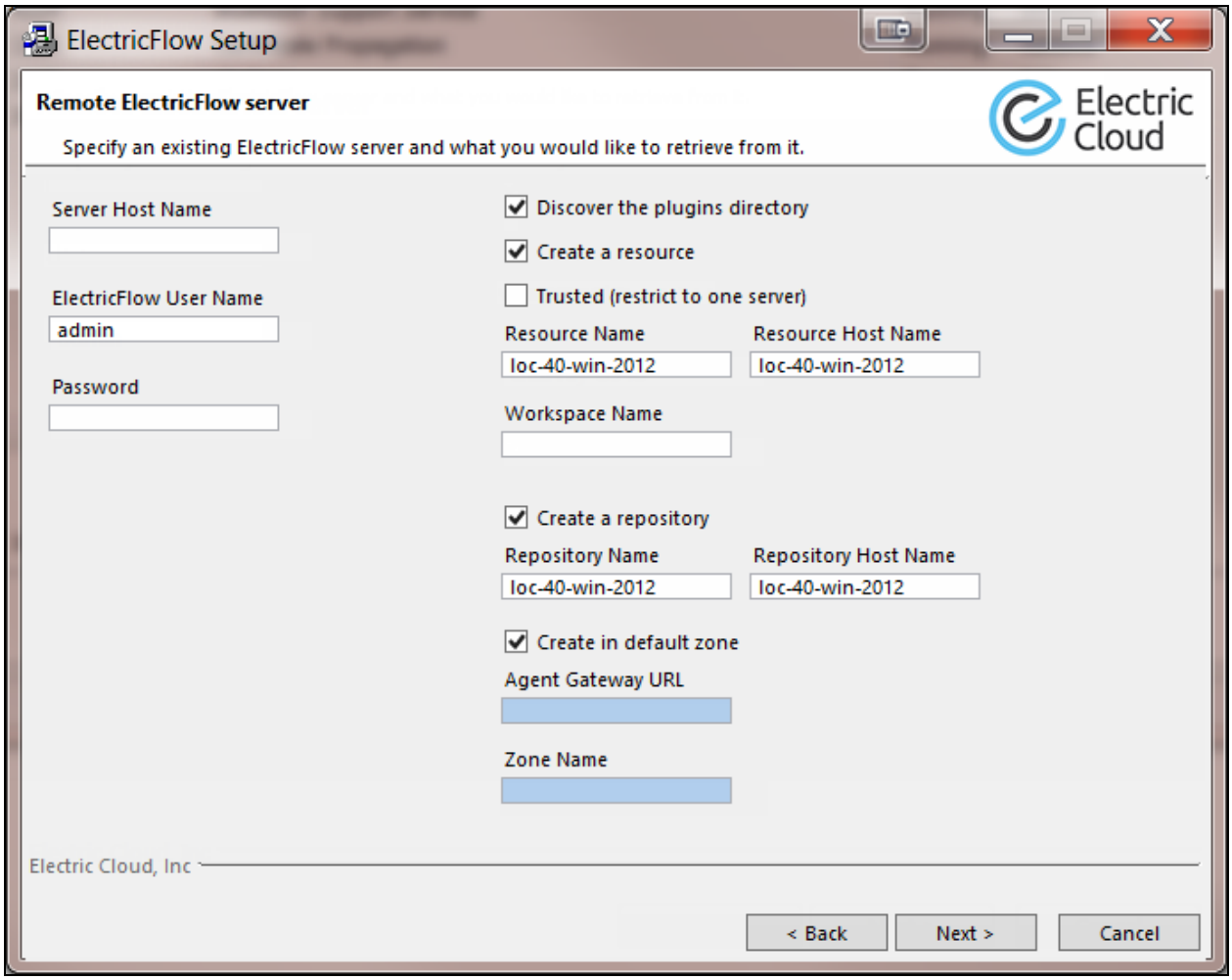
   The Welcome to the ElectricFlow Installer screen appears.



> **Note:** The screen examples in this procedure are from a Windows system. Different options will appear in some windows on a Linux system.

4. Select the **Express Agent** installation option, and then click **Next** to continue.

   The **Remote ElectricFlow server** screen appears.



5. Complete the following information on the **Remote ElectricFlow server** screen.

   - **Server Host Name**—Use this field to enter the name of the ElectricFlow server that will communicate with this agent. If the remote server is using a non-default HTTPS port, you must specify the Server Host Name as *<host>*:*<port>*. If you do not specify a port, HTTPS port 8443 is assumed (the same as the ElectricFlow server default port).
   - **ElectricFlow User Name**—Use this field to enter the name of an ElectricFlow user on the ElectricFlow server who has sufficient privileges to create a resource. This field defaults to the ElectricFlow-supplied `admin` user.
   - **Password**—Use this field to enter the password for the ElectricFlow user. The default password for the `admin` user is `changeme`.

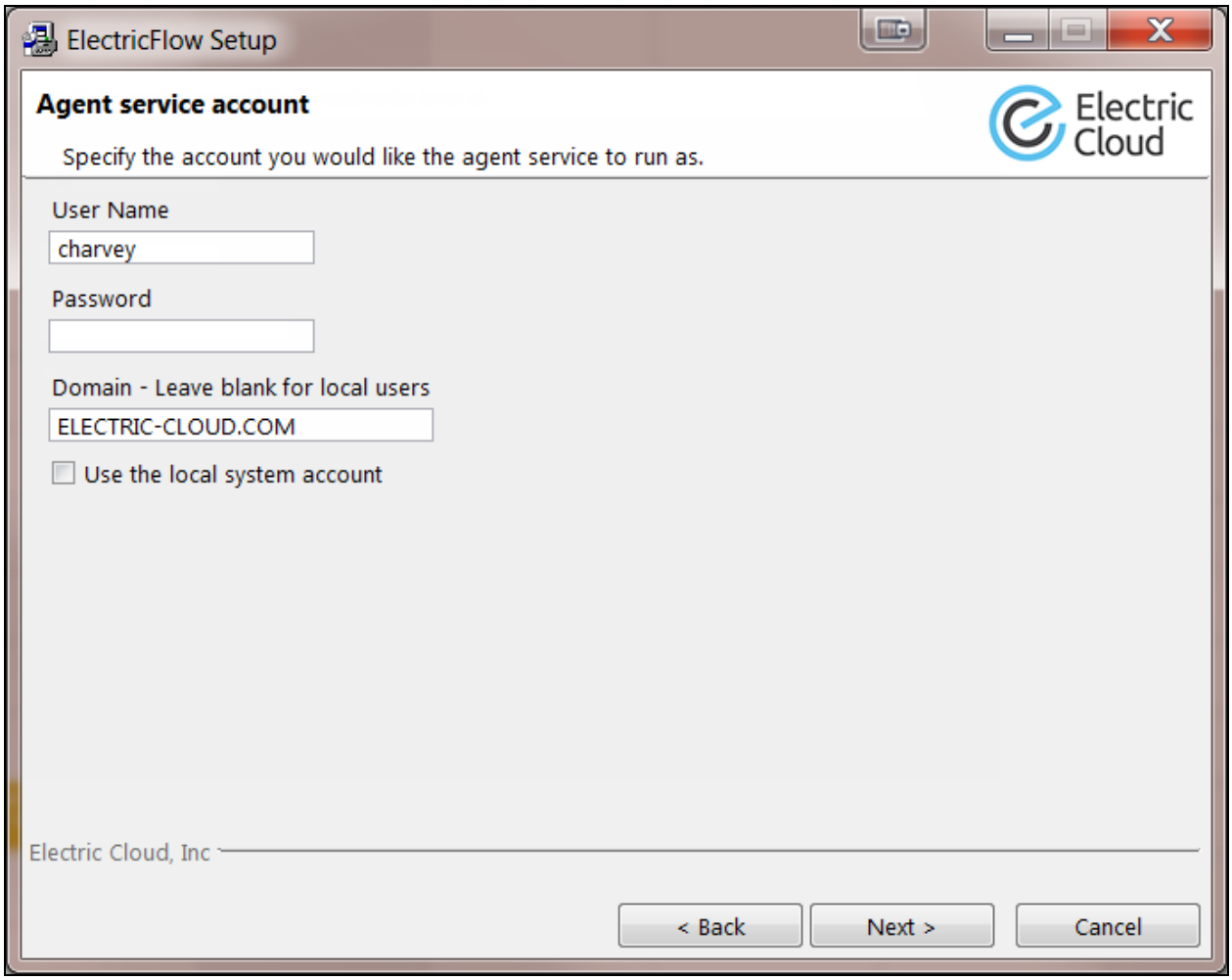
- **Discover the plugins directory**—Select this check box if you want the agent machine to have access to the plugins directory. You should allow access to the plugins directory so agents have access to collections of features, third-party integrations, or third-party tools.

  > **Note:** The plugins directory on the ElectricFlow server must be "shared" before the agent machine can use "discover" to find the directory. For more information, see Universal Access to the Plugins Directory on page 5-21.

- **Create a resource**—Select this check box if you want to create a resource on the remote ElectricFlow server for the agent you are installing.
- **Trusted**—Select this check box to restrict this agent to one ElectricFlow server. The agent will not respond to incoming communication from any other ElectricFlow server. This is useful when you want to create a secure production environment, but generally not needed for test or development systems.
- **Resource Name**—Use this field to enter the name of the resource you would like to use for the agent. This field is available for use when the Create a resource check box is selected.
- **Create in default zone**—Select this check box if you want to create the agent in the default zone.
- **Agent Gateway URL**—Use this field to enter the URL of the gateway used to communicate with the ElectricFlow server. This field is available for use when the Create in default zone check box is cleared.
- **Zone Name**—Use this field to enter the name of the zone used during remote agent and or remote repository creation. This field is available for use when the Create in default zone check box is cleared.

6. Click **Next** to continue.

   The **Agent service account** screen appears.



7. Select the appropriate steps for your platform and complete the following information on the screen.

   - If you have a Windows system:
     - **User Name**—Use this field to enter the name of the user who will run the ElectricFlow agent service.

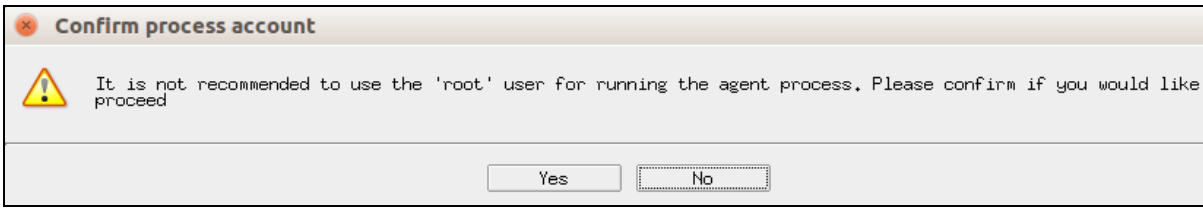
       The user that the agent runs as must have permission to write to the `$INSTALL_DIRECTORY/log` directory.
     - **Password**— Use this field to enter the password of the user who will run the ElectricFlow agent service.
     - **Domain**—Use this field to enter the domain name information for the user. For example, electric-cloud.com. Leave this field blank if this is a local user.

- **Use the local system account**—Select this check box if you want the ElectricFlow agent service to run as the Windows local system account.

  > **Note:**
  > The Windows local system account cannot access network resources such as shared file systems used for plugins or workspaces. Therefore, do not use this option for a clustered server deployment, which requires a shared file system for plugins. This option is typically used only for installing agents on numerous machines, which would otherwise require that you create a new account on each of those machines.

- If you have a Linux system:
  - **User Name**—Use this field to enter the name of the user who owns the ElectricFlow agent process.

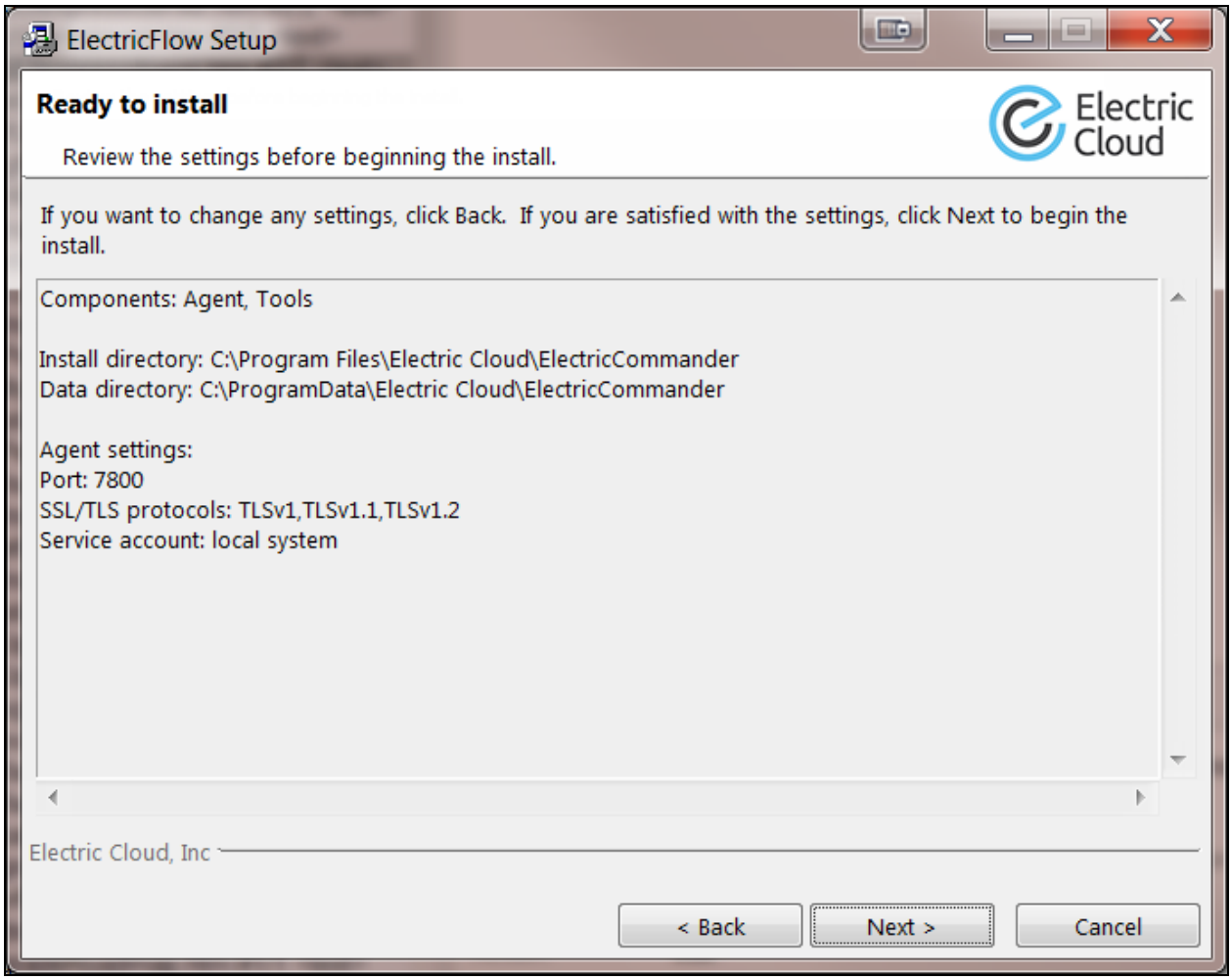
    The user/group that the agent runs as must have permission to write to the `$INSTALL_DIRECTORY/log` directory. If you specify `root`, click **Yes** when the following confirmation appears:

  - **Group Name**—Use this field to enter the name of the group who owns the ElectricFlow agent process.

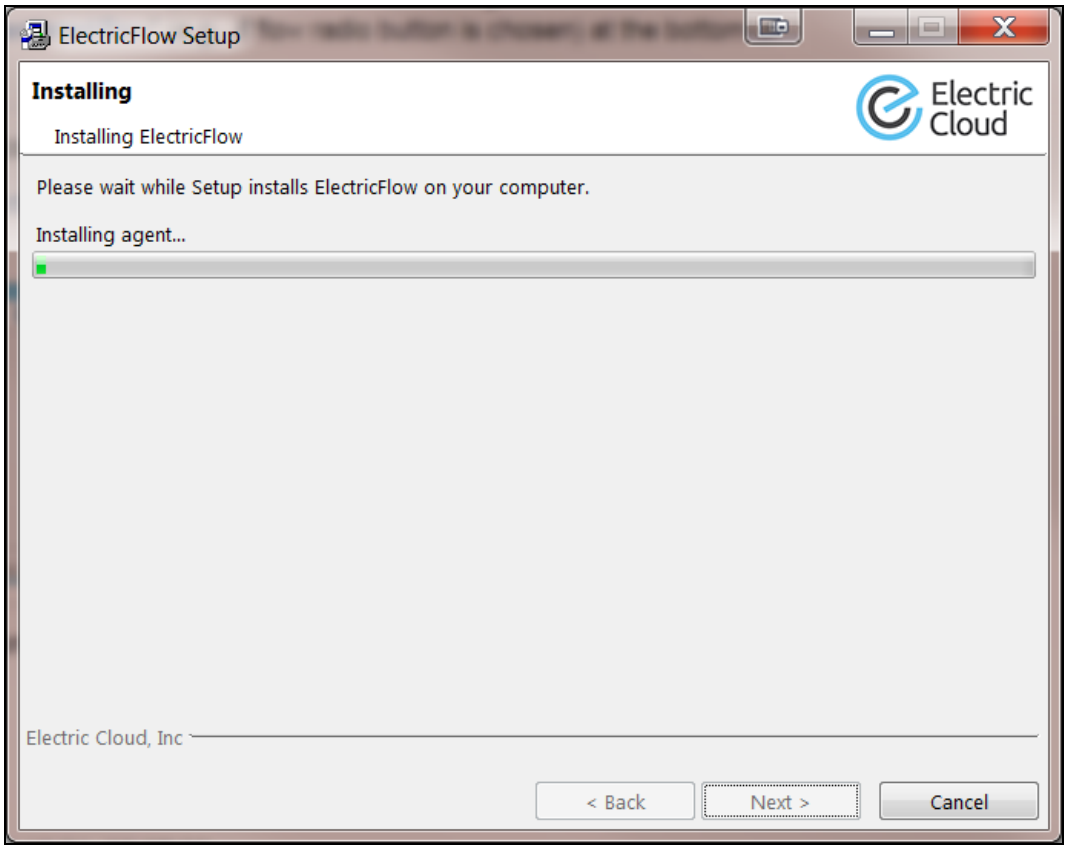
8. Click **Next** to continue.

   The **Ready to Install** appears.

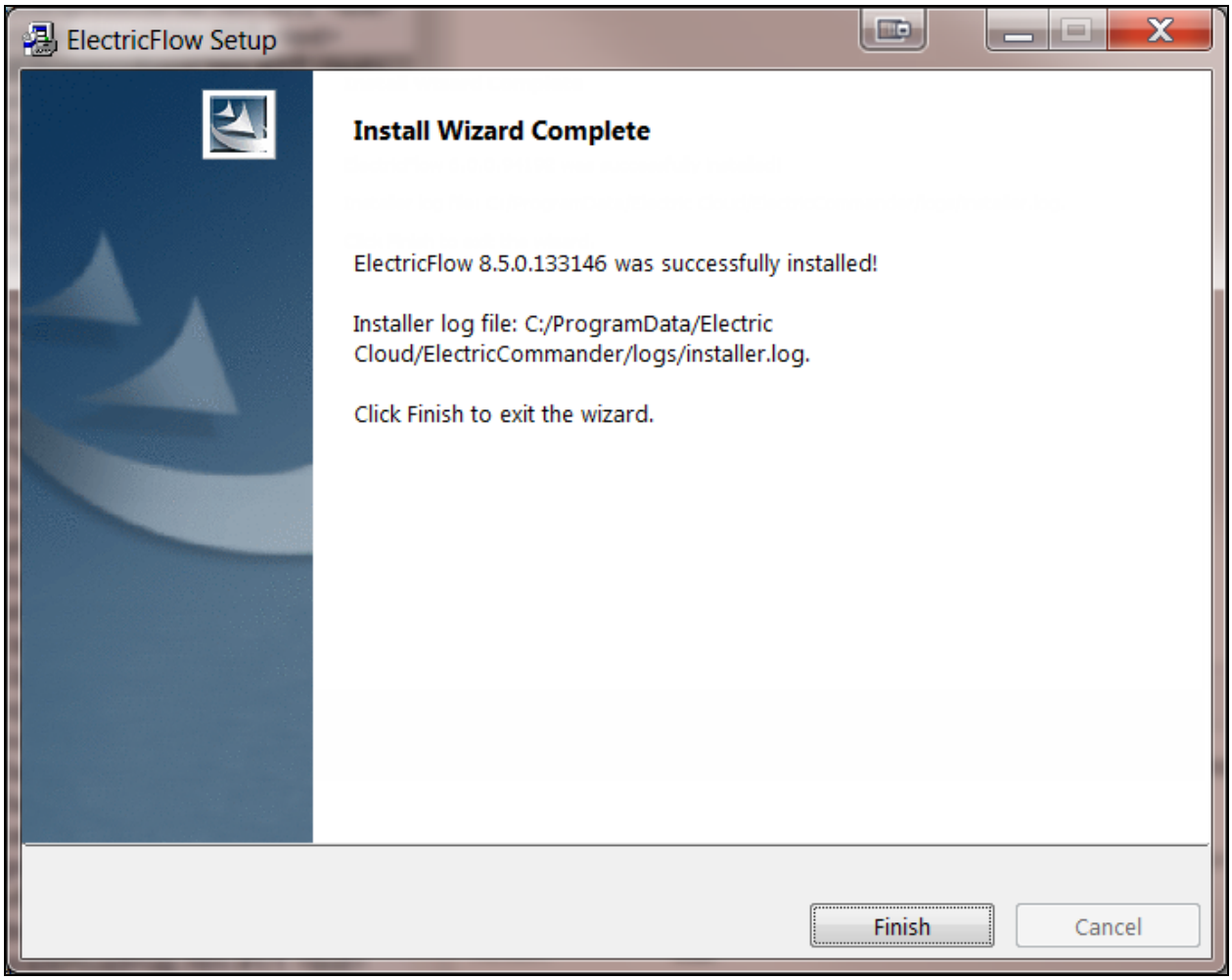


9. Review this screen to verify your selections. Use the **Back** button to change any of your settings if necessary.
10. Click **Next** to continue.

    ElectricFlow installs the agent and tools components. This process can take a few minutes:



When the install process is complete, the **Install Wizard Complete** screen appears:



11. Click **Finish** to complete the installation.

Running an Express Agent Graphical User Interface Installation (Agent-Only Installer)

The agent software must be installed on each machine you intend to use with ElectricFlow. An agent is an ElectricFlow component that runs on a machine resource. The agent executes ElectricFlow job steps, monitors step progress, and records job completion information.

Certain ElectricFlow installers allow you to perform installations as a non-root user or a user without `sudo` privileges. To determine whether a particular installer has an option to run in this mode, see Availability of Installers with a Non-Root/Non-sudo or Non-Administrator Mode on page 3-2.

Review Before You Install ElectricFlow on page 3-7 before performing these steps.

> **Note:** You install ElectricFlow agent software on Windows or Linux with this installation method. For Solaris, HP-UX, macOS, AIX, or other supported UNIX agent-only machines, see Non-Server Platform Installation Method for UNIX Agents on page 14-30.

1. Download the appropriate agent-only installer file.

   For details, see ElectricFlow Installer Files on page 3-1.

2. (Linux only) Enter the following command to make the installer file executable:

   ```
   chmod +x <agent_installer_file>
   ```

   For example, enter:

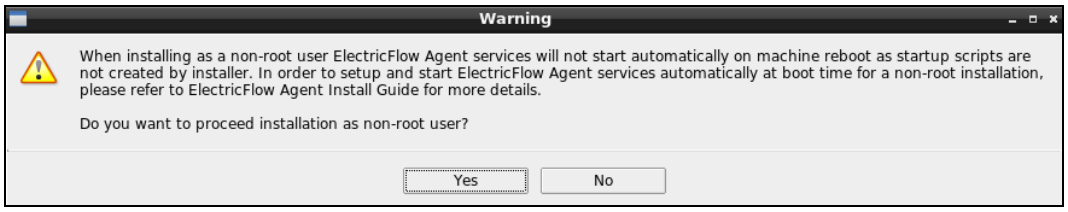
   ```
   chmod +x ElectricFlowAgent-x64-8.4.0.129860-new-with-64bit-perl
   ```

3. Do one of the following to start the installation:

   - For Linux with root or `sudo` privileges or for Windows installations, double-click the installer file.
   - For non-root/non-`sudo` installations, enter:

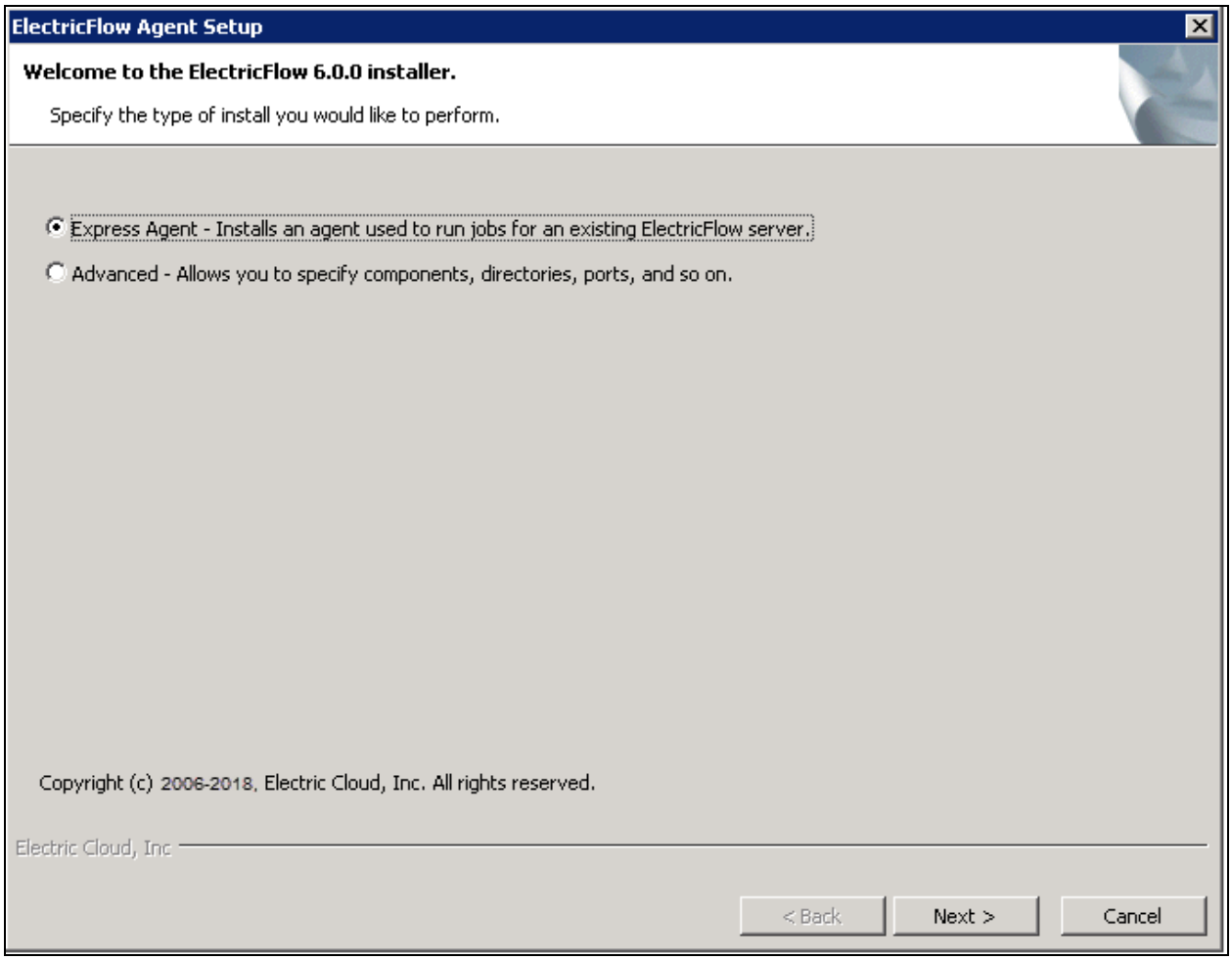
     ```
     ./<agent_installer_file> --nonRoot
     ```

     For this installation type, the following warning appears:



4. For non-root/non-`sudo` installations, click **Yes** to dismiss the warning.

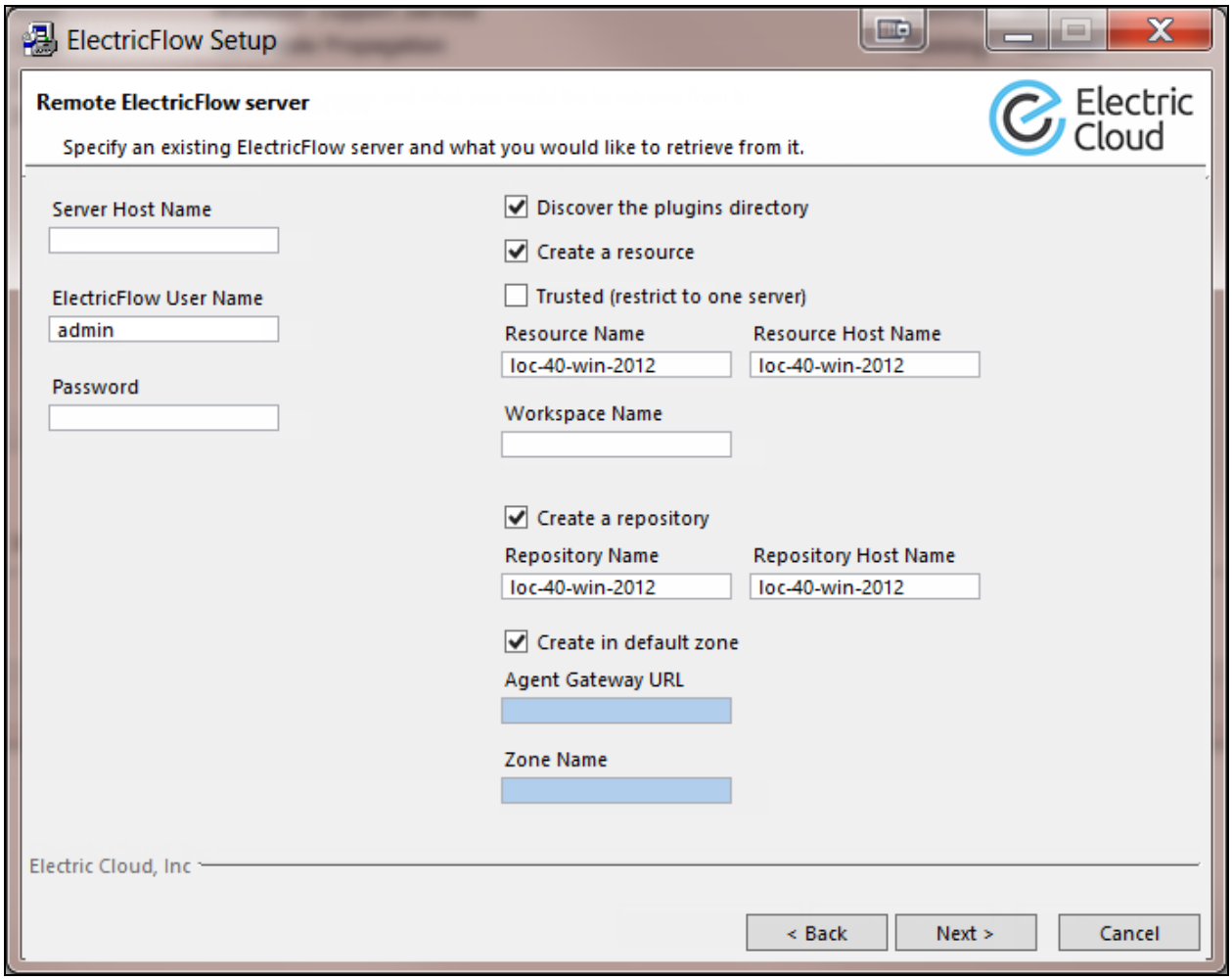
   The **Welcome to the ElectricFlow Installer** screen appears:



> **Note:** Different options might appear depending on the operating system.

5. Select the **Express Agent** installation option, and then click **Next** to continue.

   The Remote ElectricFlow Server screen appears:



6. Complete the following information on the **Remote ElectricFlow Server** screen:

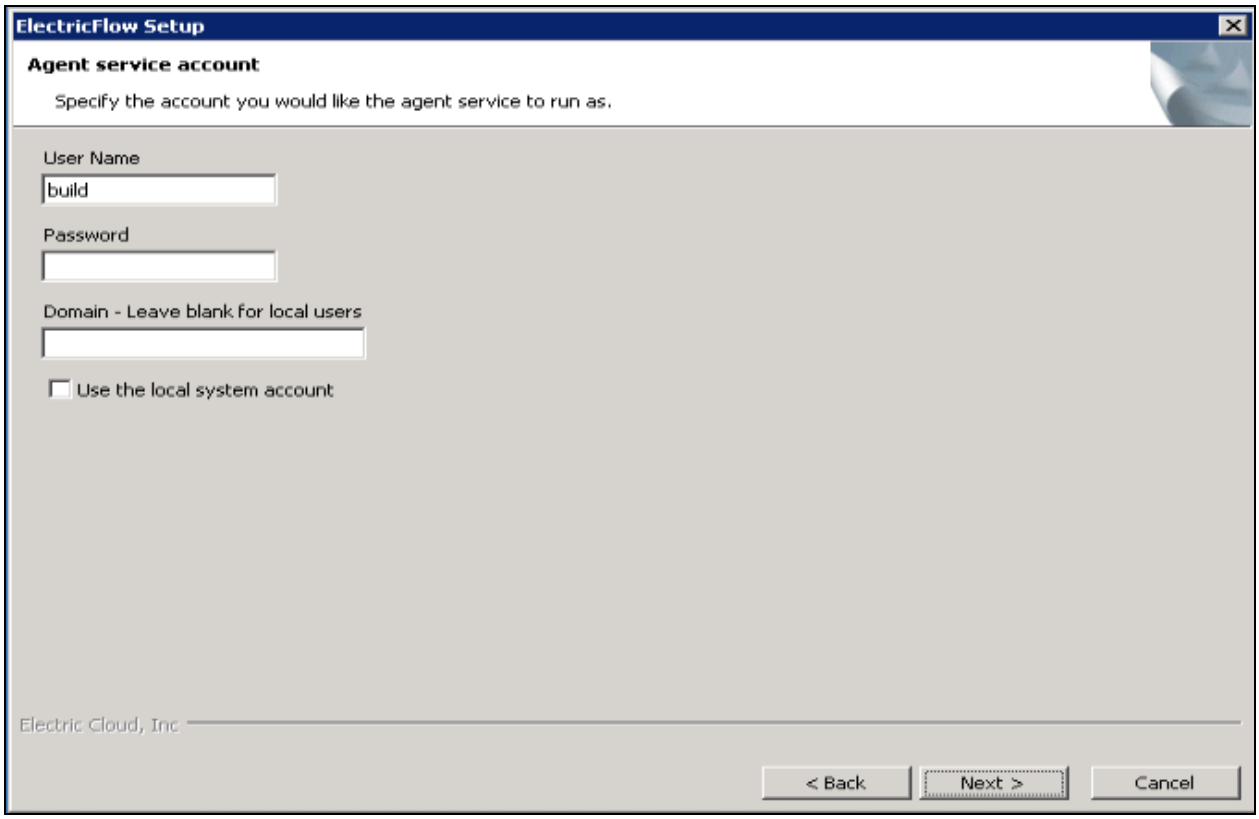
   - **Server Host Name**—Use this field to enter the name of the ElectricFlow server that will communicate with this agent. If the remote server is using a non-default HTTPS "port, you must specify the Server Host Name as *<host>*:*<port>*. If you do not specify a port, HTTPS port 8443 is assumed (the same as the ElectricFlow server default port).
   - **ElectricFlow User Name**—Use this field to enter the name of an ElectricFlow user on the ElectricFlow server who has sufficient privileges to create a resource. This field defaults to the ElectricFlow-supplied `admin` user.
   - **Password**—Use this field to enter the password for the ElectricFlow user. The default password for the `admin` user is `changeme`.
   - **Discover the plugins directory**—Select this check box if you want the agent machine to have access to the plugins directory. You should allow access to the plugins directory so agents have access to collections of features, third-party integrations, or third-party tools.

> **Note:** The plugins directory on the ElectricFlow server must be "shared" before the agent machine can use "discover" to find the directory. For more information, see Universal Access to the Plugins Directory on page 5-21

- **Create a resource**—Select this check box if you want to create a resource on the remote ElectricFlow server for the agent you are installing.
- **Trusted**—Select this check box to restrict this agent to one ElectricFlow server. The agent will not respond to incoming communication from any other ElectricFlow server. This is useful when you want to create a secure production environment, but generally not needed for test or development systems.
- **Resource Name**—Use this field to enter the name of the resource you would like to use for the agent. This field is available for use when the Create a resource check box is selected.
- **Create in default zone**—Select this check box if you want to create the agent in the default zone.
- **Agent Gateway URL**—Use this field to enter the URL of the gateway used to communicate with the ElectricFlow server. This field is available for use when the Create in default zone check box is cleared.
- **Zone Name**—Use this field to enter the name of the zone used during remote agent or remote repository creation. This field is available for use when the Create in default zone check box is cleared.

7. Click **Next** to continue.

   The Agent Service Account screen appears:

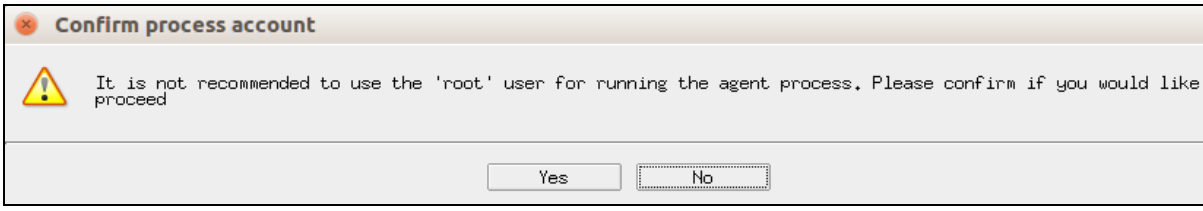


8. Select the appropriate steps for your platform and complete the following information on the screen.

   - On Linux root or `sudo` installations:

     - **User Name**—Use this field to enter the name of the user who owns the ElectricFlow agent process.

       The user/group that the agent runs as must have permission to write to the `$INSTALL_DIRECTORY/log` directory. If you specify `root`, click **Yes** when the following confirmation appears:



     - **Group Name**—Use this field to enter the name of the group who owns the ElectricFlow agent process.

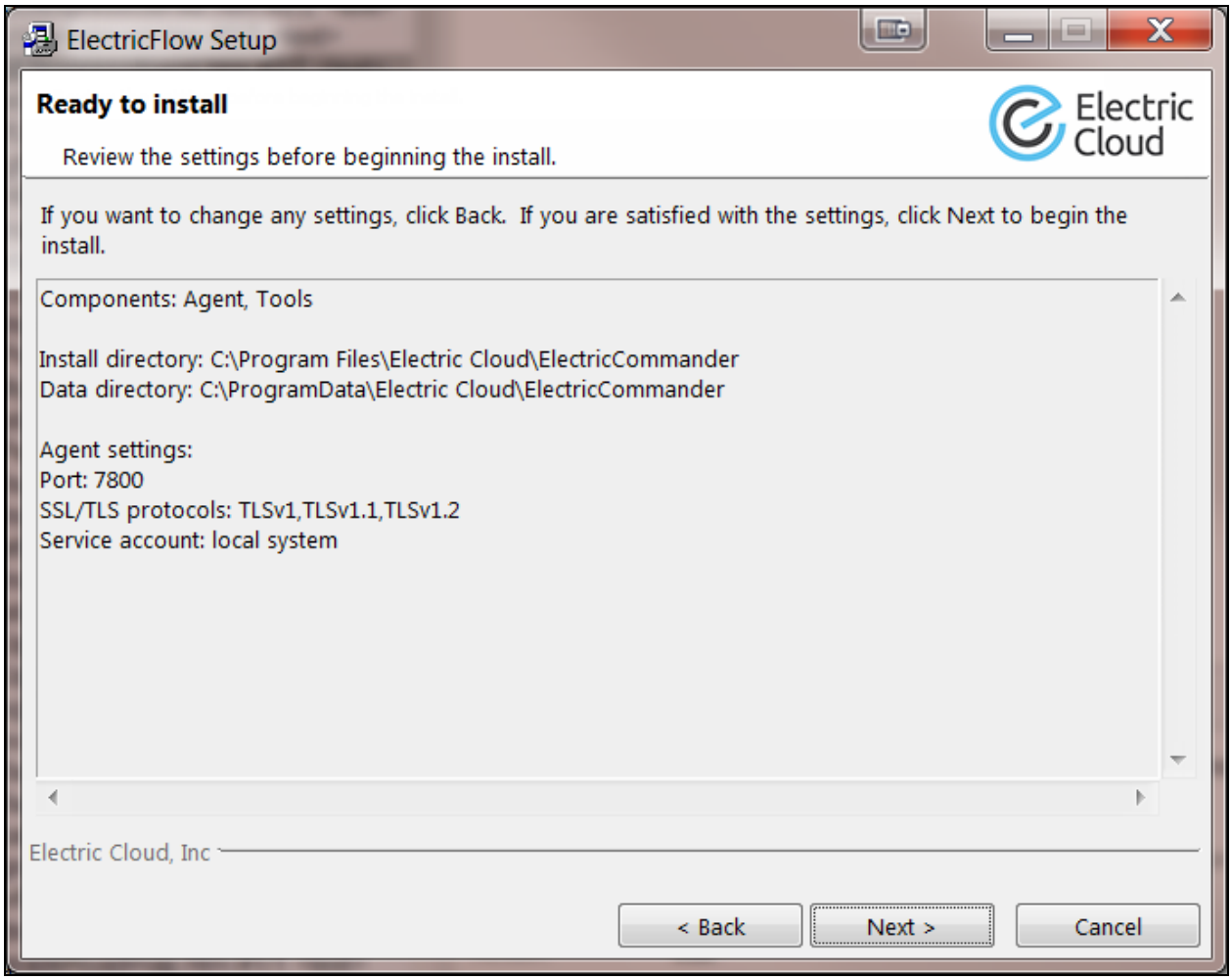
- On Windows:
  - **User Name**—Use this field to enter the name of the user who will run the ElectricFlow agent service.

    The user that the agent runs as must have permission to write to the `$INSTALL_DIRECTORY/log` directory.
  - **Password**— Use this field to enter the password of the user who will run the ElectricFlow agent service.
  - **Domain**—Use this field to enter the domain name information for the user. For example, electric-cloud.com. Leave this field blank if this is a local user.
  - **Use the local system account**—Select this check box if you want the ElectricFlow agent service to run as the Windows local system account.

    > **Note:**
    > The Windows local system account cannot access network resources such as shared file systems used for plugins or workspaces. Therefore, do not use this option for a clustered server deployment, which requires a shared file system for plugins. This option is typically used only for installing agents on numerous machines, which would otherwise require that you create a new account on each of those machines.

9. Click **Next** to continue.

   The **Ready to Install** screen appears:

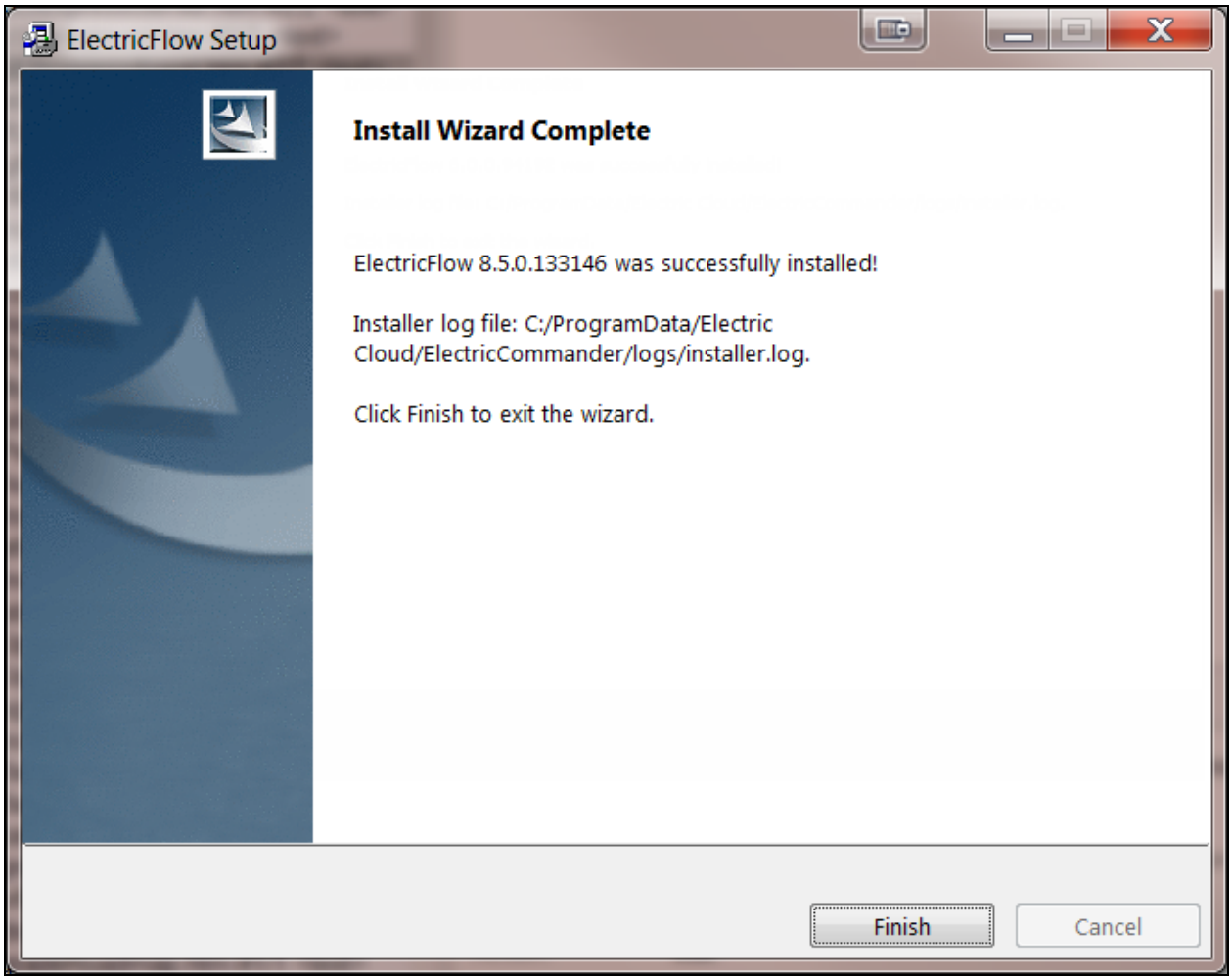


10. Review your selections.

    Use the **Back** button to change settings if necessary.

11. Click **Next** to continue.

    ElectricFlow installs the agent and tools components. This process can take a few minutes. **The Installation Wizard Complete** screen appears:



12. Click **Finish** to complete the installation.
13. For non-root/non-`sudo` Linux installations, configure autostart for the ElectricFlow agent service.

    For instructions, see Configuring Services Autostart for Non-Root/Non-sudo Linux Installations on page 5-11.

Running an Advanced Agent Graphical User Interface Installation (Agent-Only Installer)

The agent software must be installed on each machine you intend to use with ElectricFlow. An agent is an ElectricFlow component that runs on a machine resource. The agent executes ElectricFlow job steps, monitors step progress, and records job completion information.

Certain ElectricFlow installers allow you to perform installations as a non-root user or a user without `sudo` privileges. To determine whether a particular installer has an option to run in this mode, see Availability of Installers with a Non-Root/Non-sudo or Non-Administrator Mode on page 3-2.

Review Before You Install ElectricFlow on page 3-7 before performing this procedure.

Note: You install ElectricFlow agent software on Windows or Linux with this installation method. For Solaris, HP-UX, macOS, AIX, or other supported UNIX agent-only machines, see Non-Server Platform Installation Method for UNIX Agents on page 14-30.

1. Download the appropriate agent-only installer file.

   For details, see ElectricFlow Installer Files on page 3-1.

2. (Linux only) Enter the following command to make the installer file executable:

   ```
   chmod +x <agent_installer_file>
   ```

   For example, enter:

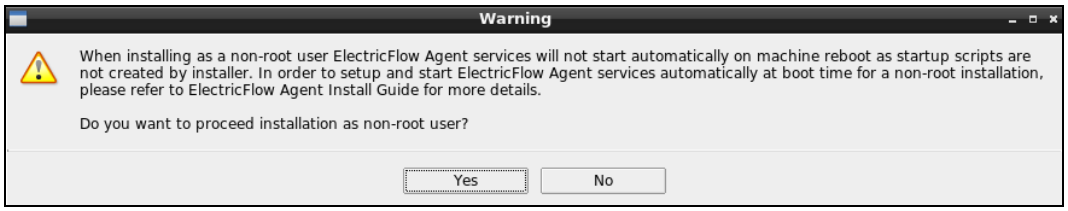
   ```
   chmod +x ElectricFlowAgent-x64-8.4.0.129860-new-with-64bit-perl
   ```

3. Do one of the following to start the installation:

   - For Linux with root or `sudo` privileges or for Windows installations, double-click the installer file.
   - For non-root/non-`sudo` installations, enter:

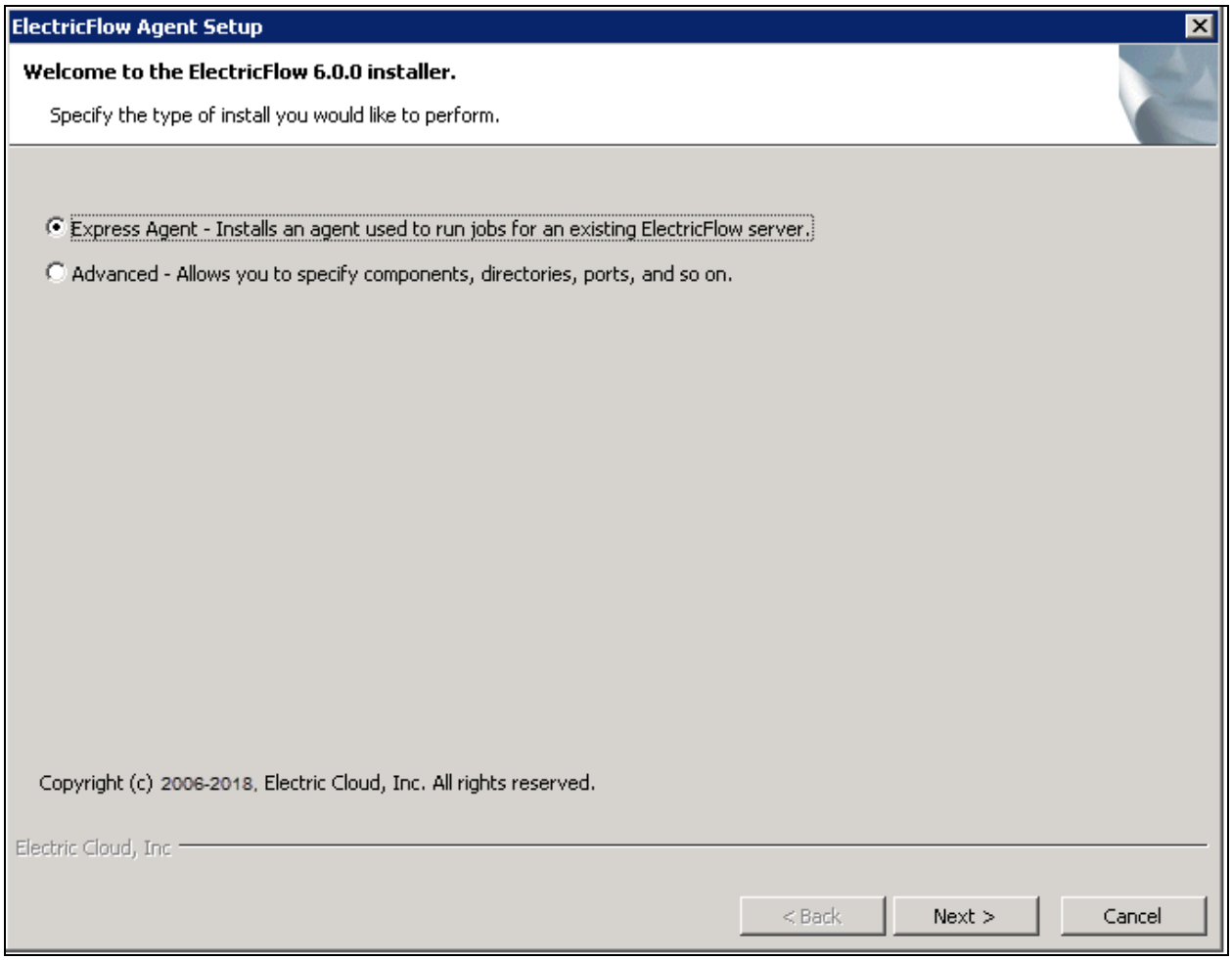
     ```
     ./<agent_installer_file> --nonRoot
     ```

     For this installation type, the following warning appears:

4. For non-root/non-`sudo` installations, click **Yes** to dismiss the warning.

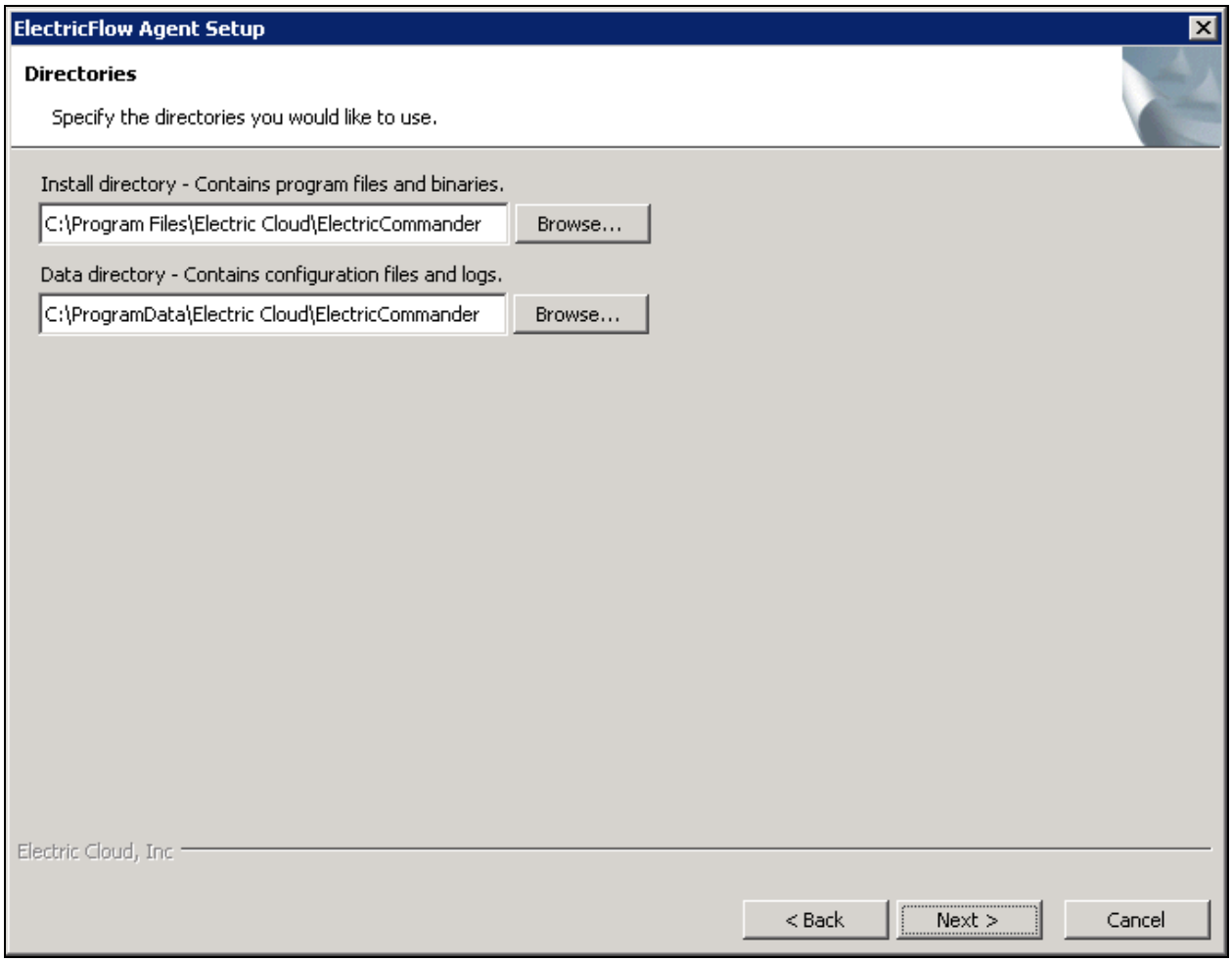
   The **Welcome to the ElectricFlow Installer** screen appears:



Note: Different options might appear depending on the operating system.

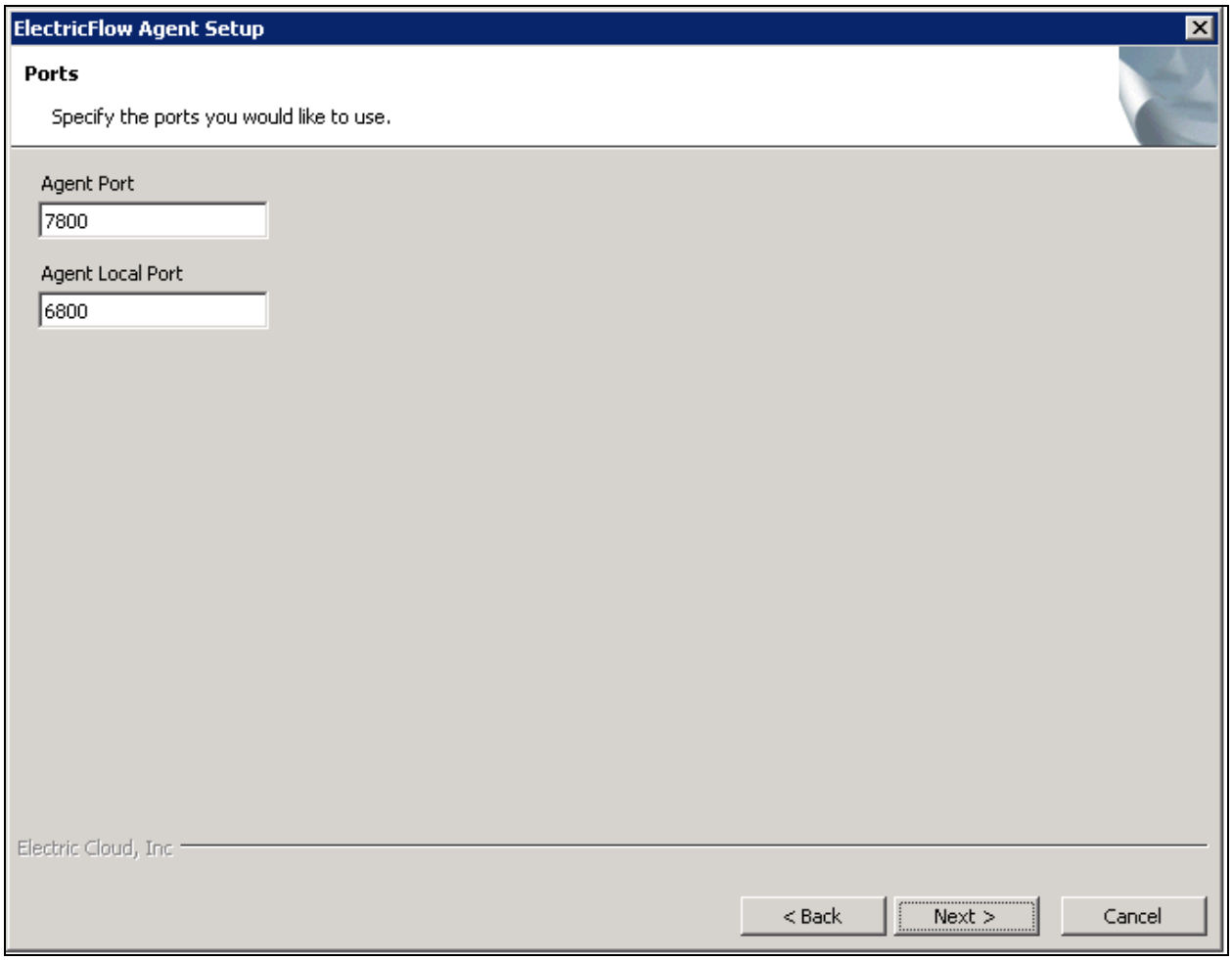
5. Select the **Advanced Agent**installation option, and then click **Next** to continue.

   The **Directories** screen appears:



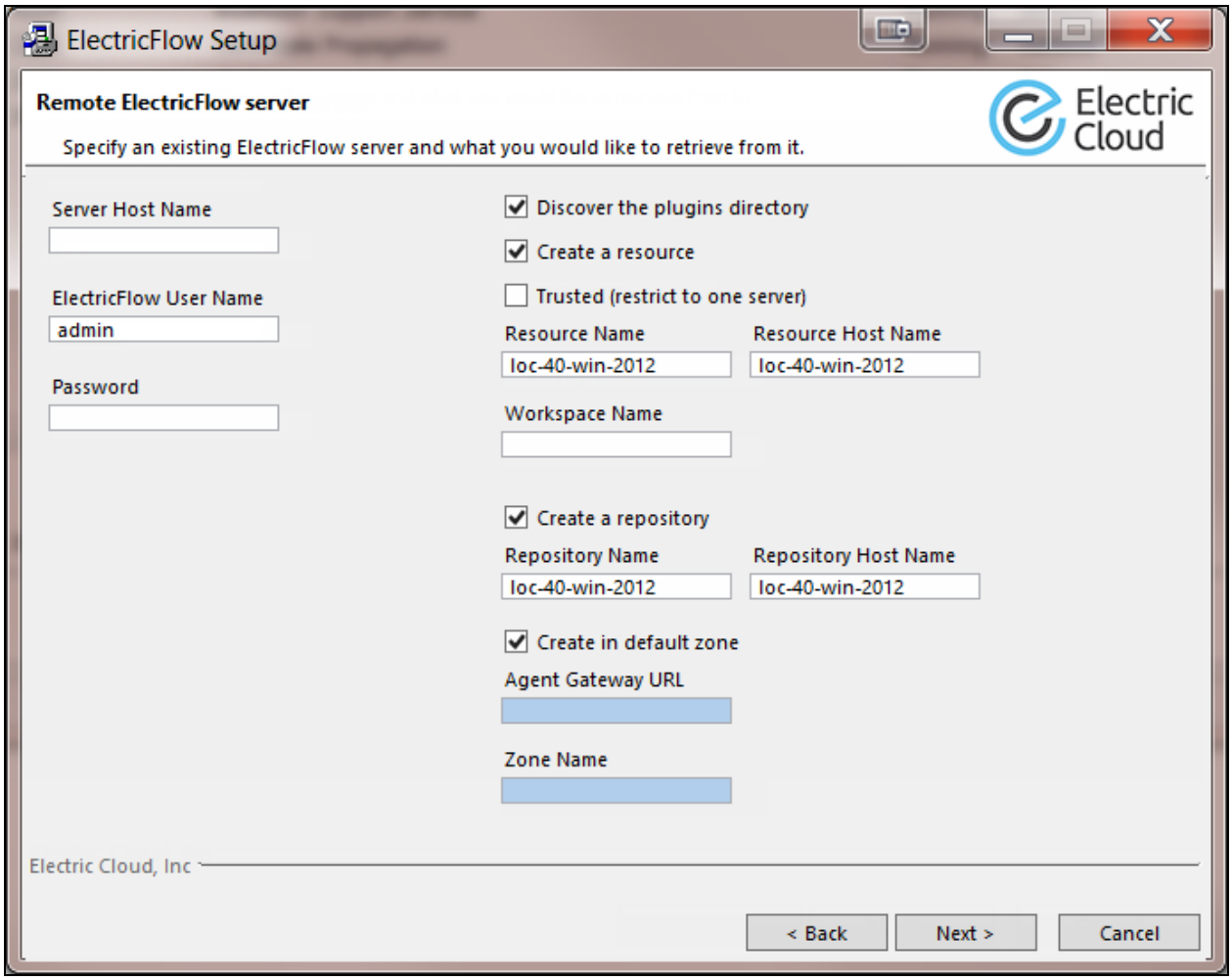
6. Complete the following information on the **Directories** screen:
   - **Install directory**—Use this field to enter a new installation directory path for program files and binaries.
   - **Data directory**—Use this field to enter a new installation directory path for configuration files and logs.
7. Click **Next** to continue.

   The **Ports** screen appears:



8. Complete the following information on the **Ports** screen:
   - **Agent port**—Use this field to specify a different port to eliminate any conflicts with your existing system configuration.
   - **Agent local port**—Use this field to specify a different port to be used by the agent for HTTP communication on the localhost network interface.
9. Click **Next** to continue.

   The Remote ElectricFlow Server screen appears:



10. Complete the following information on the Remote ElectricFlow Server screen:

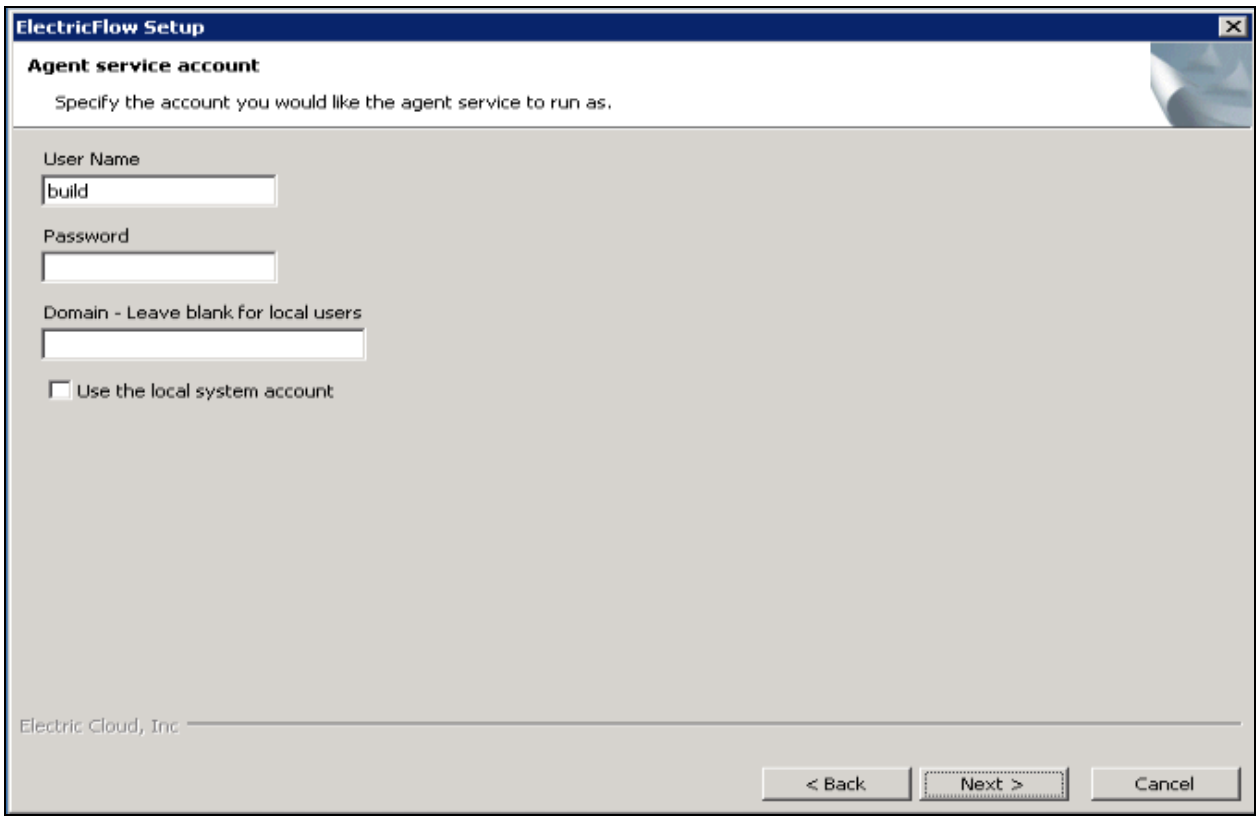
    - **Server Host Name**—Use this field to enter the name of the ElectricFlow server that will communicate with this agent. If the remote server is using a non-default HTTPS port, you must specify the Server Host Name as `<host>:<port>`. If you do not specify a port, HTTPS port 8443 is assumed (the same as the ElectricFlow server default port).
    - **ElectricFlow User Name**—Use this field to enter the name of an ElectricFlow user on the ElectricFlow server who has sufficient privileges to create a resource. This field defaults to the ElectricFlow-supplied `admin` user.
    - **Password**—Use this field to enter the password for the ElectricFlow user. The default password for the `admin` user is `changeme`.
    - **Discover the plugins directory**—Select this check box if you want the agent machine to have access to the plugins directory. You should allow access to the plugins directory so agents have access to collections of features, third-party integrations, or third-party tools.

> **Note:** The plugins directory on the ElectricFlow server must be "shared" before the agent machine can use "discover" to find the directory. For more information, see Universal Access to the Plugins Directory on page 5-21

- **Create a resource**—Select this check box if you want to create a resource on the remote ElectricFlow server for the agent you are installing.
- **Trusted**—Select this check box to restrict this agent to one ElectricFlow server. The agent will not respond to incoming communication from any other ElectricFlow server. This is useful when you want to create a secure production environment, but generally not needed for test or development systems.
- **Resource Name**—Use this field to enter the name of the resource you would like to use for the agent. This field is available for use when the Create a resource check box is selected.
- **Create in default zone**—Select this check box if you want to create the agent in the default zone.
- **Agent Gateway URL**—Use this field to enter the URL of the gateway used to communicate with the ElectricFlow server. This field is available for use when the Create in default zone check box is cleared.
- **Zone Name**—Use this field to enter the name of the zone used during remote agent and or remote repository creation. This field is available for use when the Create in default zone check box is cleared.

11. Click **Next** to continue.

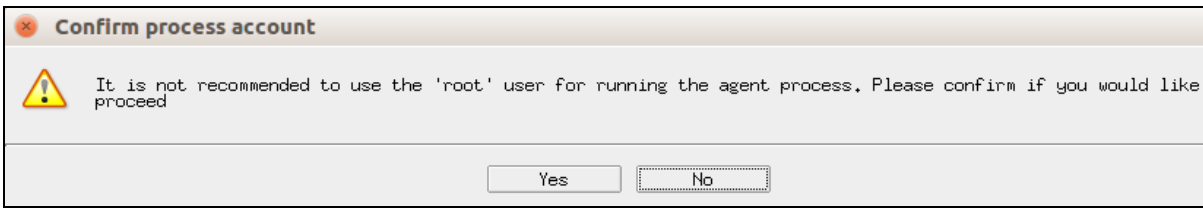
    The **Agent Service Account** screen appears:



12. Select the appropriate steps for your platform and complete the following information on the screen:

    - On Linux root or `sudo` installations:
      - **User Name**—Use this field to enter the name of the user who owns the ElectricFlow agent process.

        The user/group that the agent runs as must have permission to write to the `$INSTALL_DIRECTORY/log` directory. If you specify `root`, click **Yes** when the following confirmation appears:



- **Group Name**—Use this field to enter the name of the group who owns the ElectricFlow agent process.

- Windows systems:
  - **User Name**—Use this field to enter the name of the user who will run the ElectricFlow agent service.

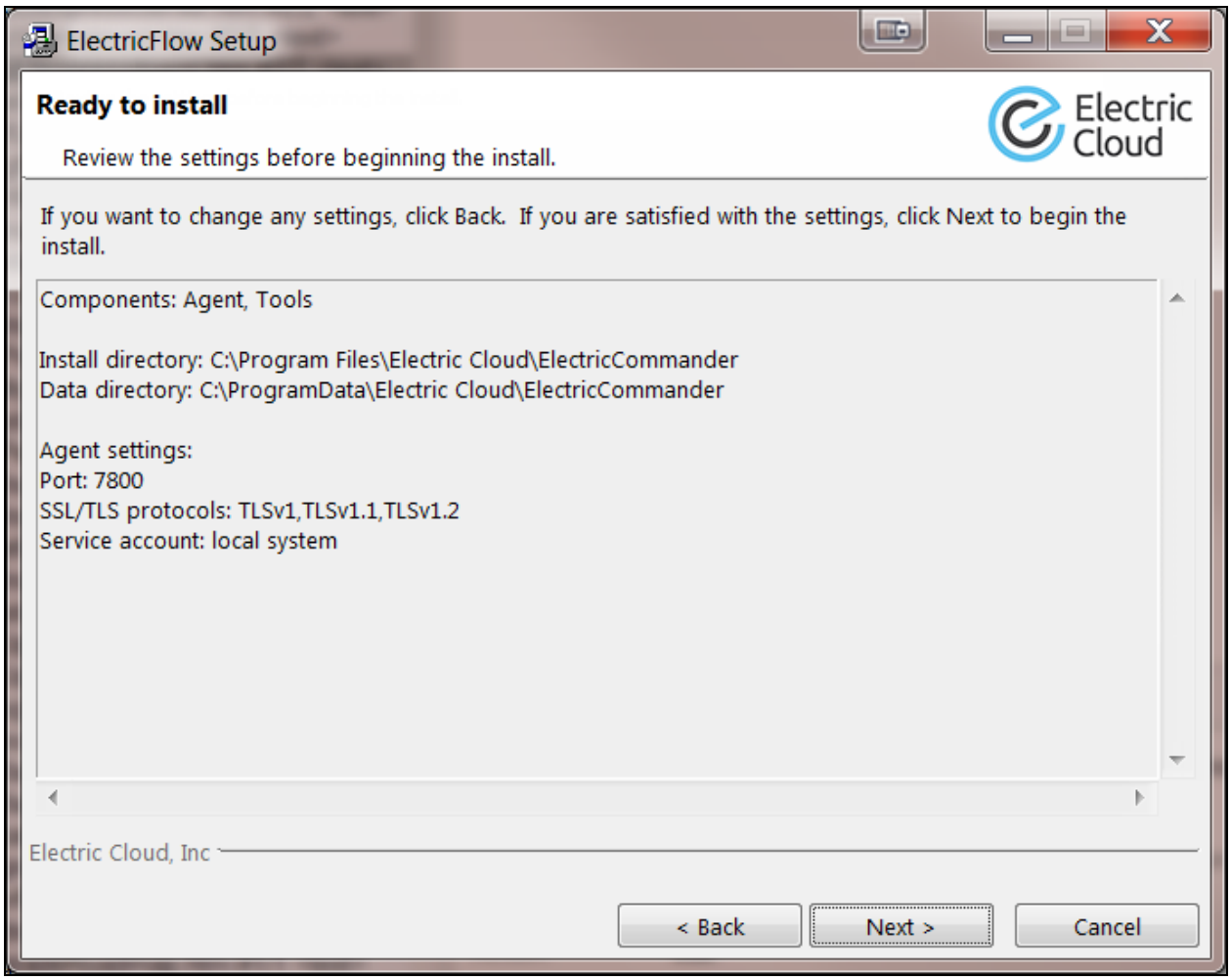
    The user that the agent runs as must have permission to write to the `$INSTALL_DIRECTORY/log` directory.
  - **Password**— Use this field to enter the password of the user who will run the ElectricFlow agent service.
  - **Domain**—Use this field to enter the domain name information for the user. For example, electric-cloud.com. Leave this field blank if this is a local user.
  - **Use the local system account**—Select this check box if you want the ElectricFlow agent service to run as the Windows local system account.

    > **Note:**
    > The Windows local system account cannot access network resources such as shared file systems used for plugins or workspaces. Therefore, do not use this option for a clustered server deployment, which requires a shared file system for plugins. This option is typically used only for installing agents on numerous machines, which would otherwise require that you create a new account on each of those machines.

13. Select the appropriate steps for your platform and complete the information on the screen.

14. Click **Next** to continue.

    The **Ready to Install Screen** appears:

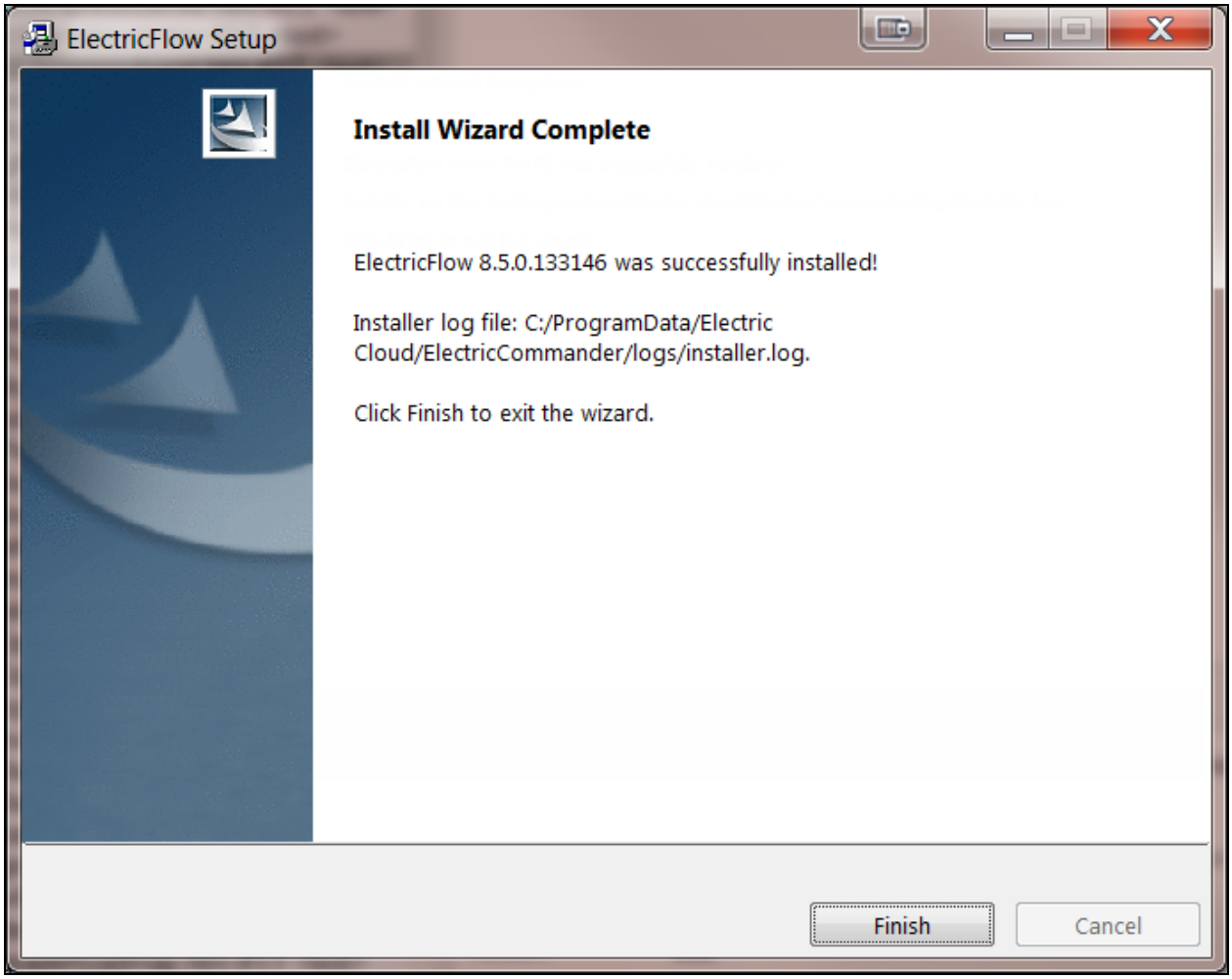


15. Verify your selections.

    Use the **Back** button to change settings if needed.

16. Click **Next** to continue.

    ElectricFlow installs the agent and tools components. This process can take a few minutes. **The Installation Wizard Complete** screen appears:



17. Click **Finish** to complete the installation.
18. For non-root/non-`sudo` Linux installations, configure autostart for the ElectricFlow agent service.

    For instructions, see Configuring Services Autostart for Non-Root/Non-sudo Linux Installations on page 5-11.

Running a DevOps Insight Server Graphical User Interface Installation

The graphical user interface installation method is supported by Windows platforms and Linux platforms running the X Window System. The following procedure includes instructions for adding a system to a DevOps Insight cluster during installation.

For details about the overall steps for installing DevOps Insight on a group of servers to create a DevOps Insight server cluster, see Creating a DevOps Insight Server Cluster on page 4-36.

Installing the DevOps Insight Server on a System with Other ElectricFlow Components

For a production environment, Electric Cloud recommends that you install the DevOps Insight server on a system other than systems running other ElectricFlow components (such as the ElectricFlow server, web server, repository server, or agent). If you must install it on the same system (such as for testing or other non-production or trial-basis situations) see Running the DevOps Insight Server on a System with Other ElectricFlow Components on page 3-8 for details.

Installing the DevOps Insight Server

1. (Linux only) Enter the following command to make the installer file executable:

   ```
   chmod +x ElectricFlowDevOpsInsightServer-x64-<version>
   ```

2. Do one of the following to start the installation:

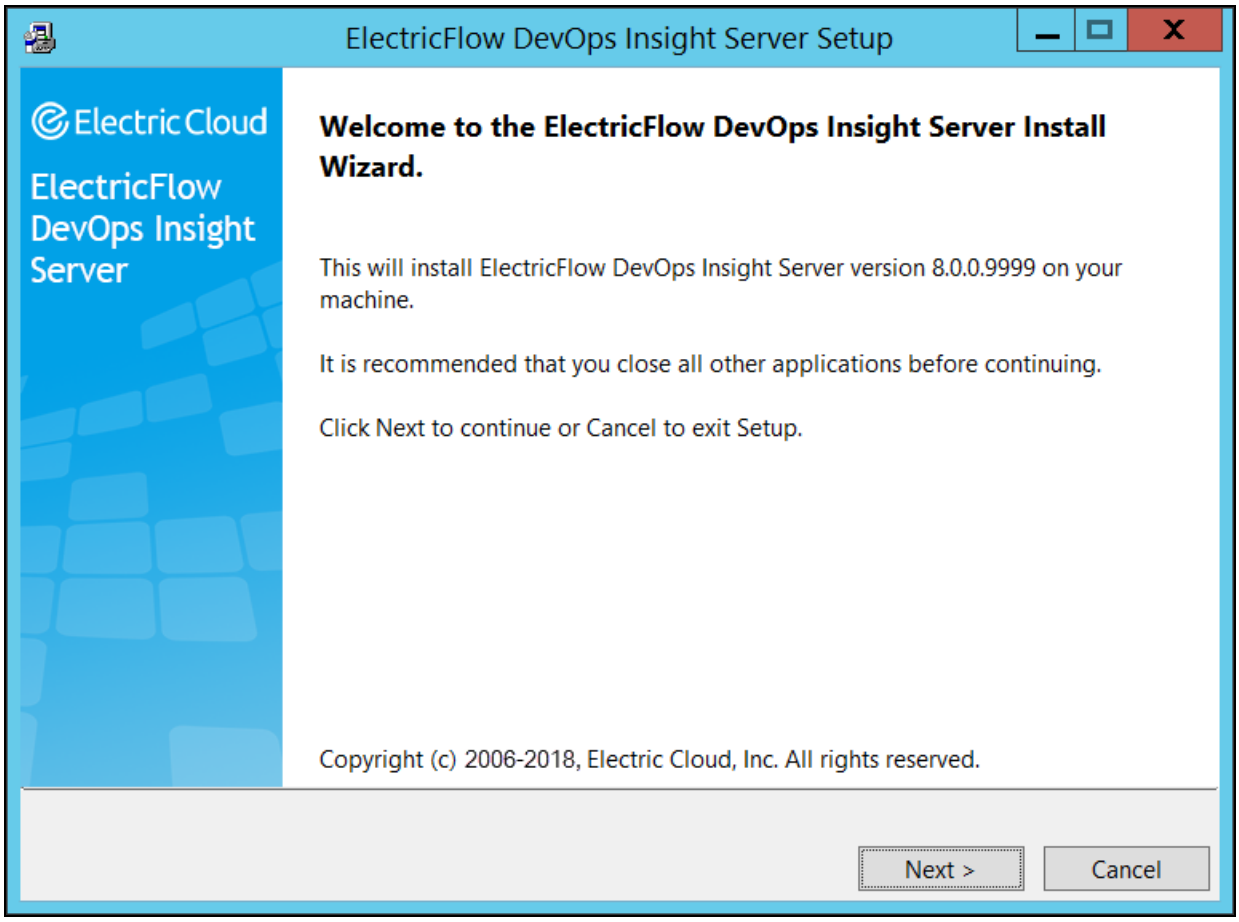
   - For Linux with root or `sudo` privileges or for Windows installations, double-click the installer file.
   - For Linux non-root/non-`sudo` installations, enter:

     ```
     ./ElectricFlowDevOpsInsightServer-x64-<version> --nonRoot
     ```

     For this installation type, the following warning appears:

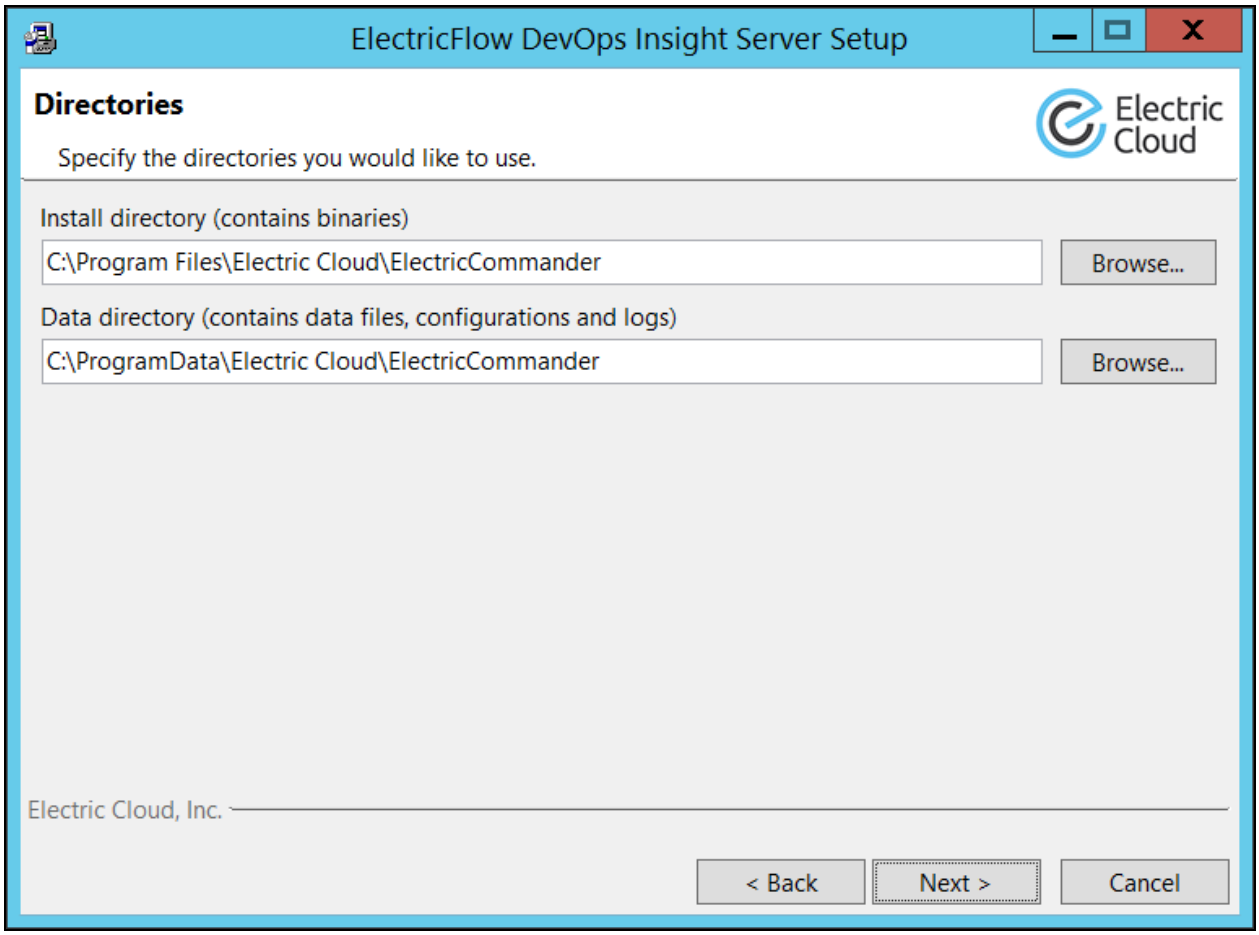
3. For non-root/non-`sudo` installations, click **Yes** to dismiss the warning.

   The **Welcome to the ElectricFlow DevOps Insight Server Install Wizard** screen appears:



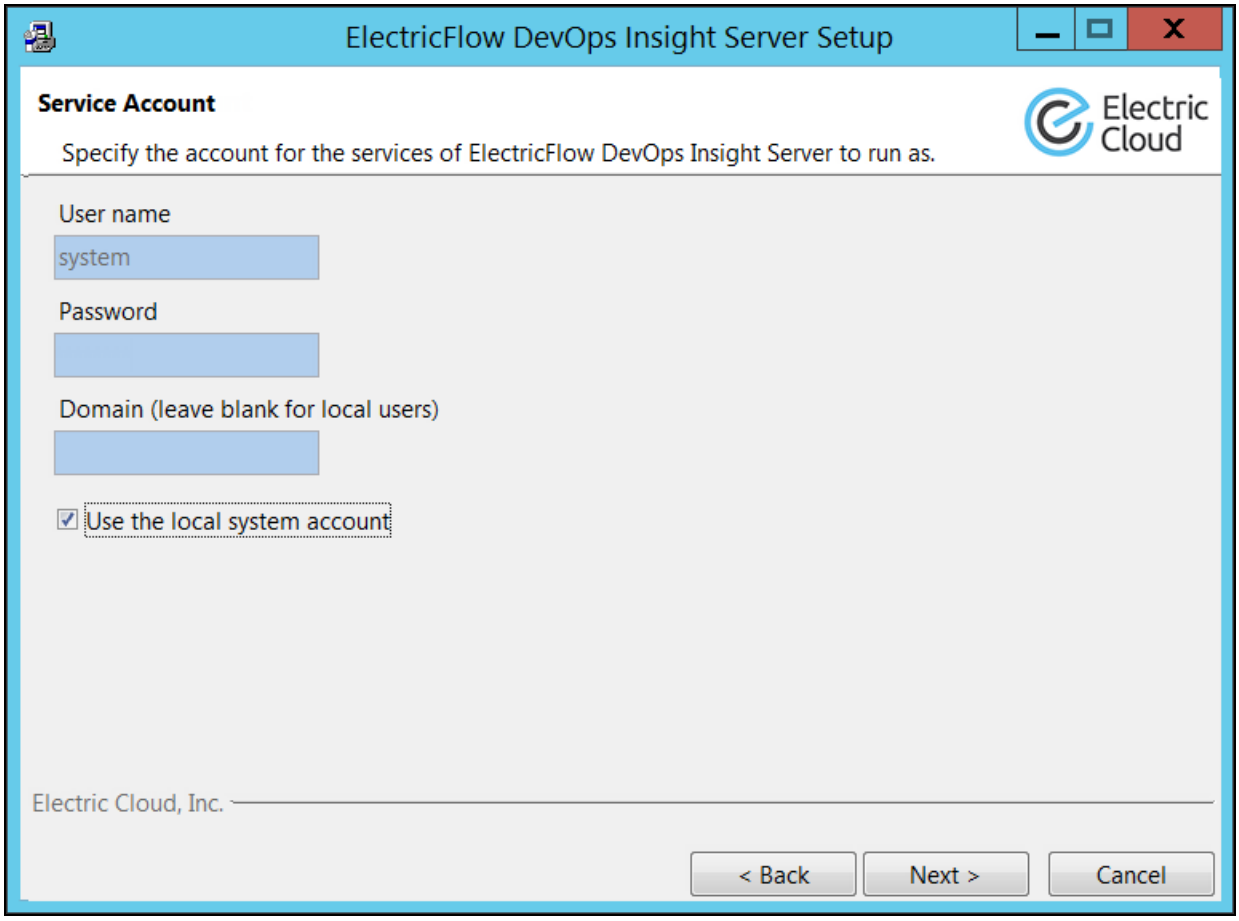
4. Click **Next** to continue.

   The **Directories** screen appears. The installer uses the default directories to install files and components:



5. Click **Next** to continue, or click **Browse** to specify different directory locations.

   The **Service Account** screen appears:



6. If you have a Windows system, complete the information on the screen as follows:

   - **User Name**—Name of the user who will run the ElectricFlow DevOps Insight server services.
   - **Password**—Password of the user who will run the ElectricFlow DevOps Insight server services.
   - **Domain**—Domain name information for the user. For example, electric-cloud.com. Leave this field blank if this is a local user.
   - **Use the local system account**—Determines if the ElectricFlow DevOps Insight server services will run as the local Windows system account.

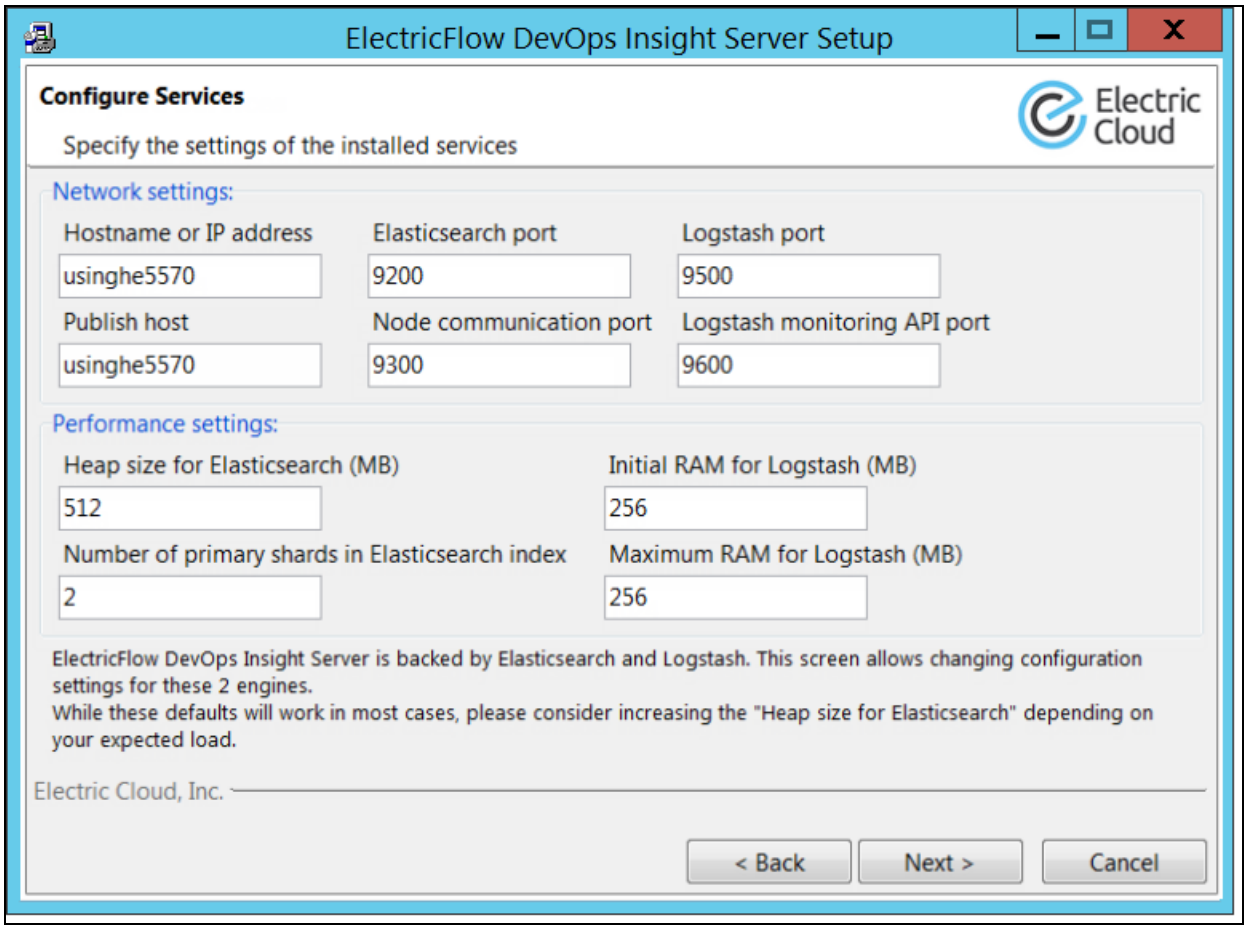
7. If you have a Linux system, complete the information on the screen as follows:

   User Name—Name of the user who owns the ElectricFlow DevOps Insight server processes.

   Group Name—Name of the group who owns the ElectricFlow DevOps Insight server processes.

8. Click **Next** to continue.

   The **Configure Services** screen appears:



Hostname or IP address—Name of the host that will be used to access the installed ElectricFlow DevOps Insight server.

Publish host—The network address that the Elasticsearch node advertises to other nodes in the cluster, so that those nodes can connect to it.

Elasticsearch port—Port number to be used to access Elasticsearch.

The DevOps Insight server uses the Elasticsearch search engine and the Logstash data-collection and log-parsing engine to gather data from the ElectricFlow server for use in the Deployments, Releases, and Release Command Center dashboards.

Node communication port—Port number used for internal communication between nodes within the Elasticsearch cluster.

Logstash port—Port number to be used to store information in Logstash.

Logstash monitoring API port—Port number used by the Logstash monitoring APIs that provide runtime metrics about Logstash.

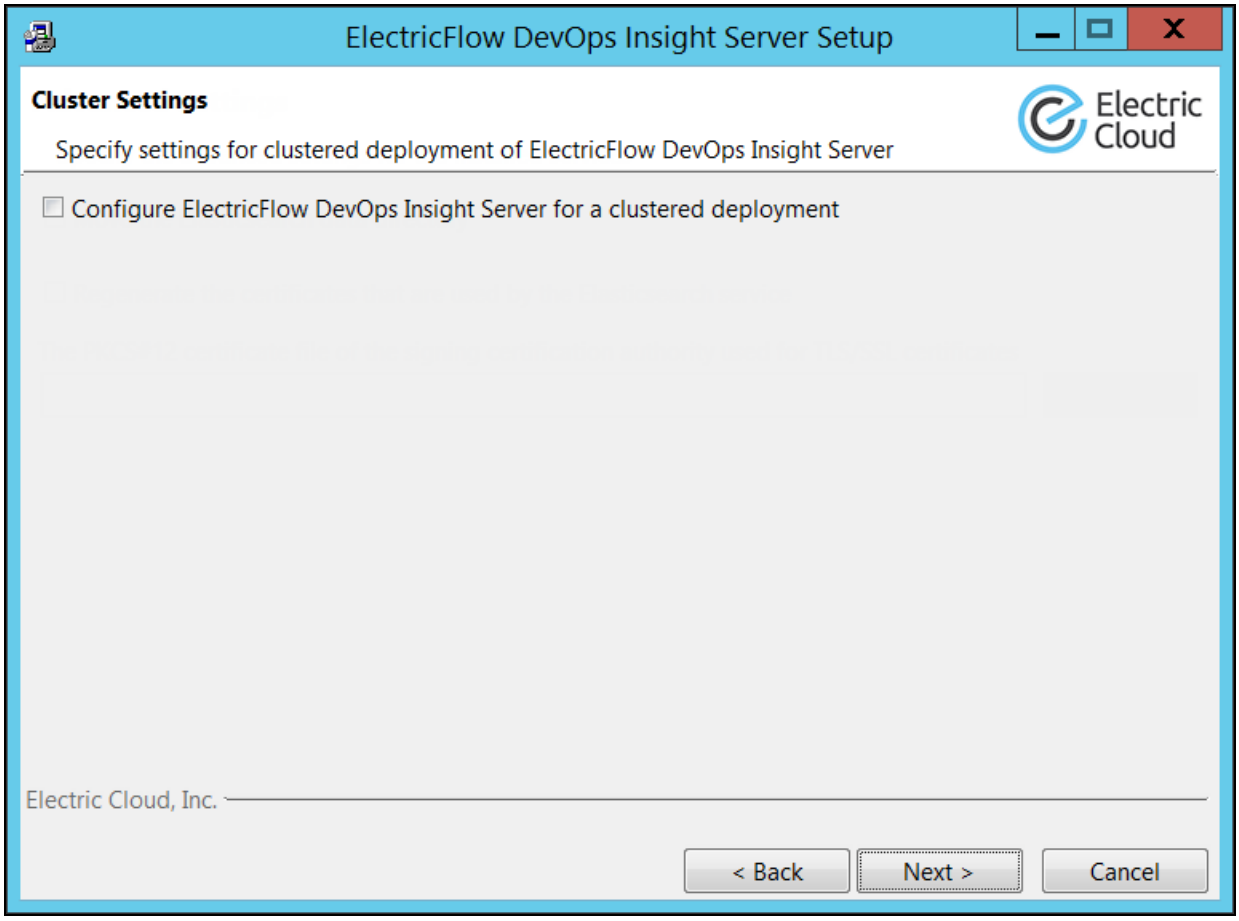
Heap size for Elasticsearch (MB)—Heap size for Elasticsearch in megabytes.

Number of primary shards in Elasticsearch index—Number of primary shards in the Elasticsearch index.

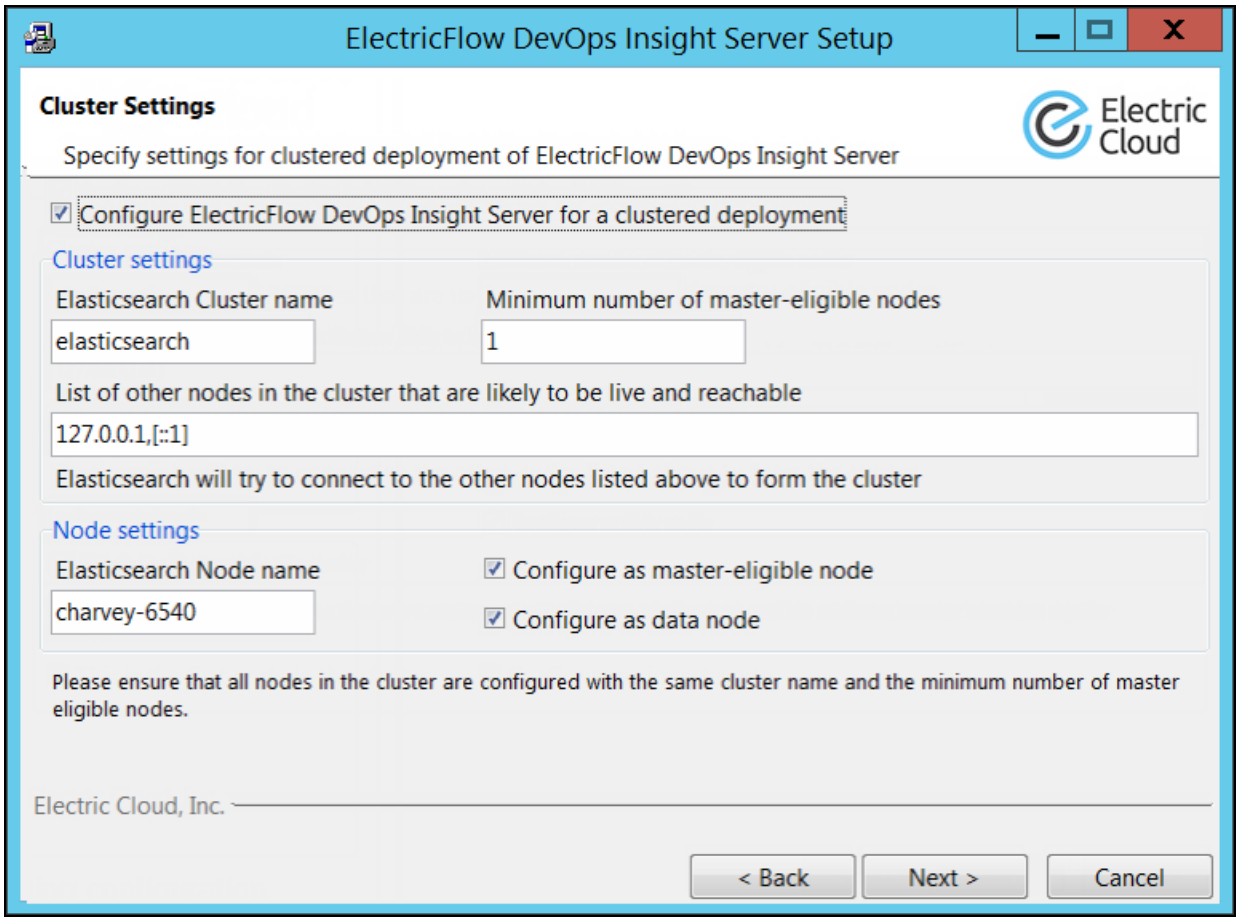
Initial RAM for Logstash (MB)—Initial heap size for Logstash in megabytes.

Maximum RAM for Logstash (MB)—Maximum heap size for Logstash in megabytes.

9. Complete the information on the **Configure Services** screen, and click **Next** to continue. The **Cluster Settings** screen appears:



Configure ElectricFlow DevOps Insight Server for a clustered deployment—Check this field if you want to add this system to a DevOps Insight server cluster. If you do so, additional fields appear to let you enter the details about this node and the cluster:

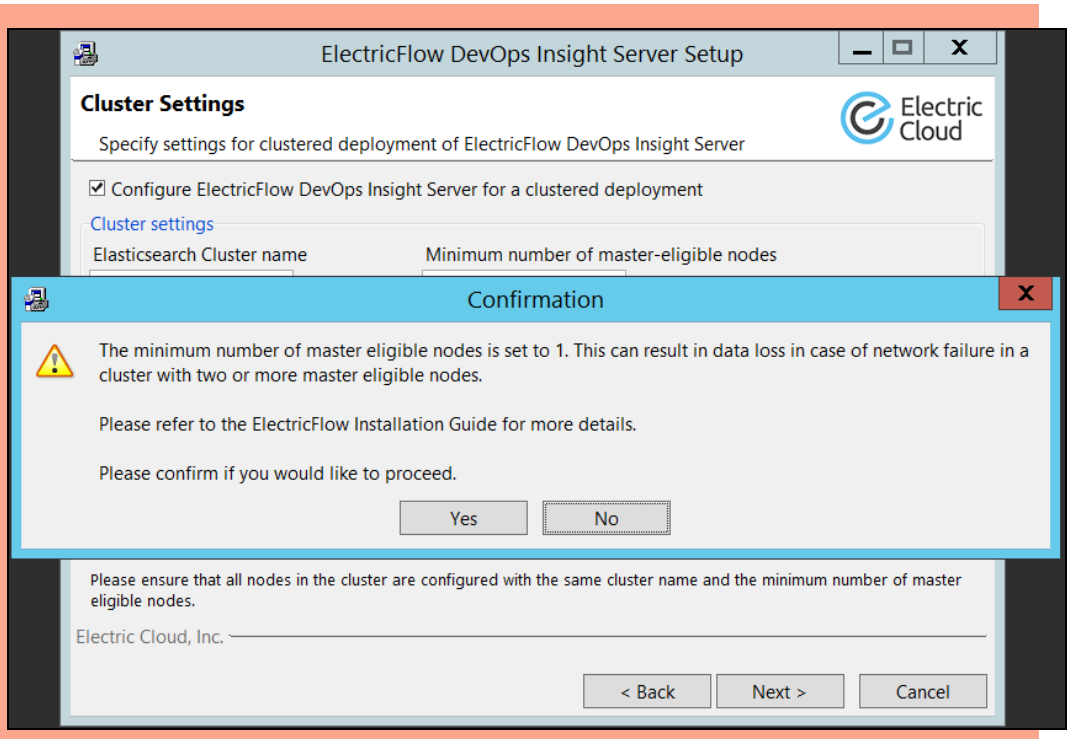


Elasticsearch Cluster name—Name of the cluster. The cluster name must be unique across all Elasticsearch clusters in the network.

Minimum number of master-eligible nodes—Minimum number of master-eligible nodes that must be visible in order to form a cluster. For details about how to determine how many master-eligible nodes you need for your cluster, see 1. Planning the Total Number of Master-Eligible Nodes on page 4-37. The master node will be elected from the list of master-eligible nodes.

For details about master-eligible nodes, see the Node module in the *Elasticsearch Reference*. For details about master elections, see the Zen Discovery module in the *Elasticsearch Reference*.

> **Important:**
> If you specify `1`, you are asked to confirm this number in the following warning:



To prevent data loss in case of network failure, the minimum number of master-eligible nodes that must be visible in the cluster must be set to a quorum of master-eligible nodes:

(Number of master-eligible nodes in the cluster / 2) + 1

For example, in a cluster with three master-eligible nodes, minimum number of master-eligible nodes should be set to 2.

The minimum number of master-eligible nodes should be set to 1 only if you intend to run a single-node cluster. For a multi-node cluster, the minimum number of master-eligible nodes must be set to a quorum as described above.

List of other nodes in the cluster that are likely to be live and reachable—Additional nodes that are running DevOps Insight and can become part of the cluster. These can be any nodes (whether they are master-eligible or not). You can enter any combination of IP addresses or host names.

This is mandatory for additional nodes and optional for the first node. You should specify in this list all available master nodes.

Elasticsearch Node name—Name of this node in the cluster. This serves as a unique identifier and therefore must be a unique name in the cluster.

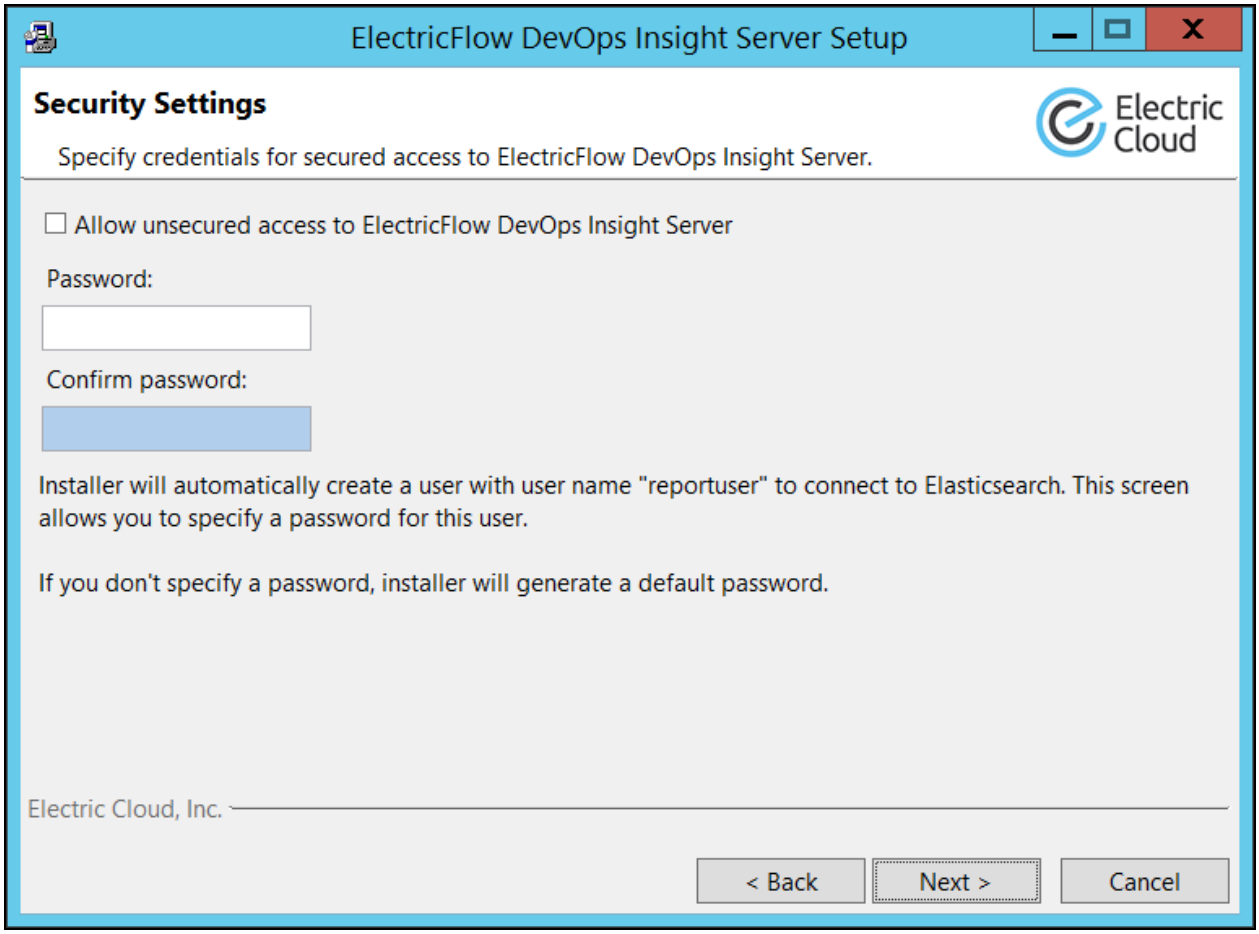
This is the first node in the cluster—Check this checkbox if this is the first node that you are adding to the cluster.

Configure as master-eligible node—Makes this node eligible to be elected as a master node. Master-eligible nodes participate in updating the cluster state as well as elections of the master node. A master-eligible node can also be a data node. The first node that you add to a cluster is always a master-eligible node (and also a data node).

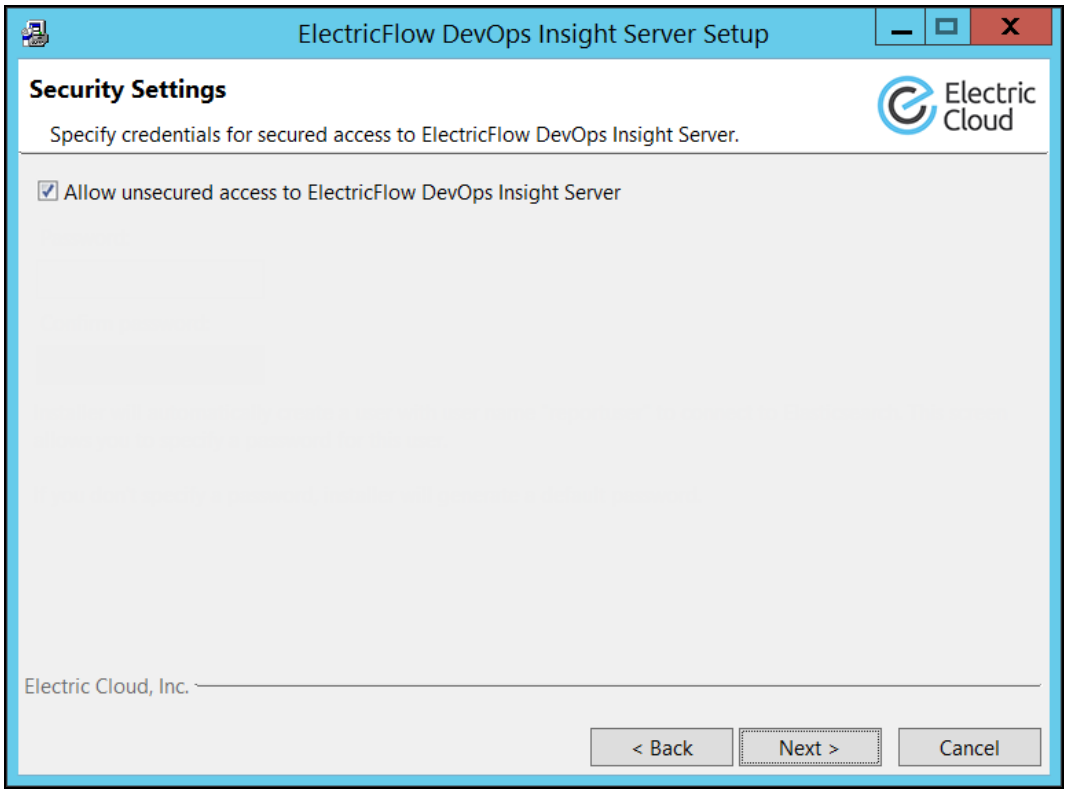
Configure as data node—Determines whether this node will be a data node. A data node stores data that is indexed into Elasticsearch and performs data-related operations such as CRUD, search, and aggregations. A data node can also be a master-eligible node. The first node that you add to a cluster is always a data node (and also a master-eligible node).

10. Complete the information on the **Cluster Settings** screen.

11. Click **Next** to continue.
    The **Security Settings** screen appears:



Allow unsecured access to ElectricFlow DevOps Insight Server—Check this field if you do *not* want to use a secure protocol and authentication when accessing the DevOps Insight server:



Otherwise, the **Password** and **Confirm password** fields let you enter the server password:

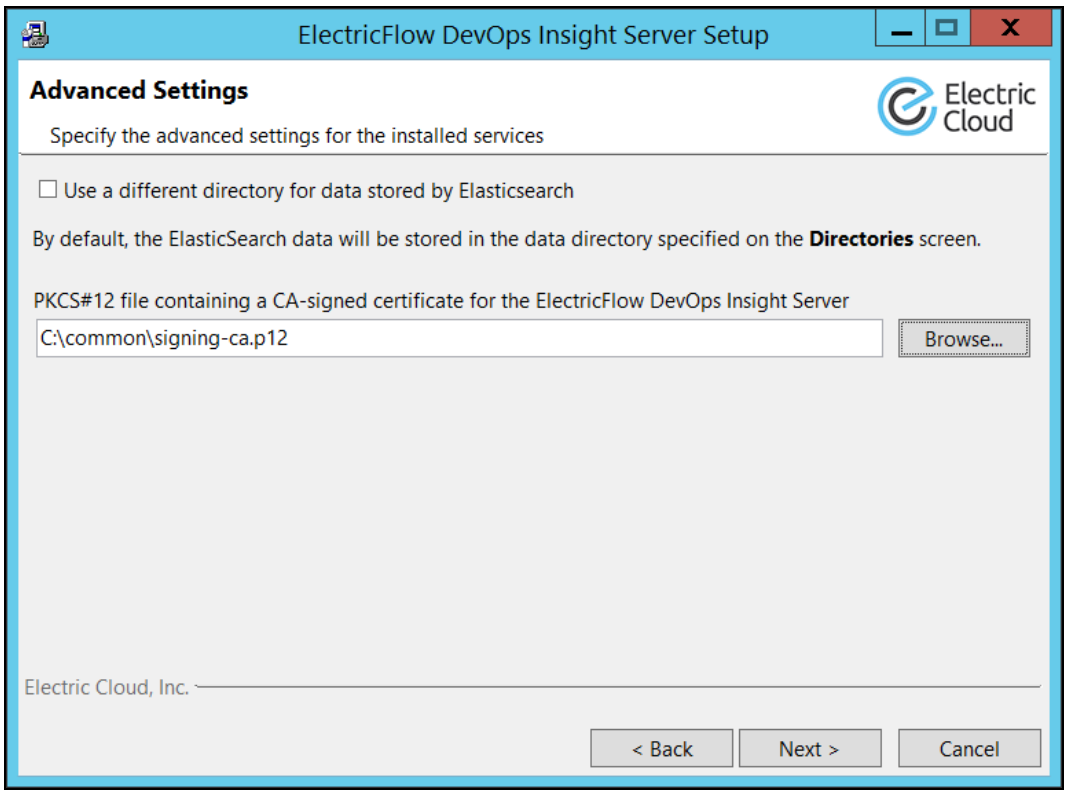
Password—Password to be used to access the server. The installer will automatically create a user with user name `reportuser` and the password that you specified. If you do not specify a password, the installer will generate a default password. (Electric Cloud recommends that you change this password.)

Confirm password—Confirm the password. Enter the same password in this field as in the previous field.

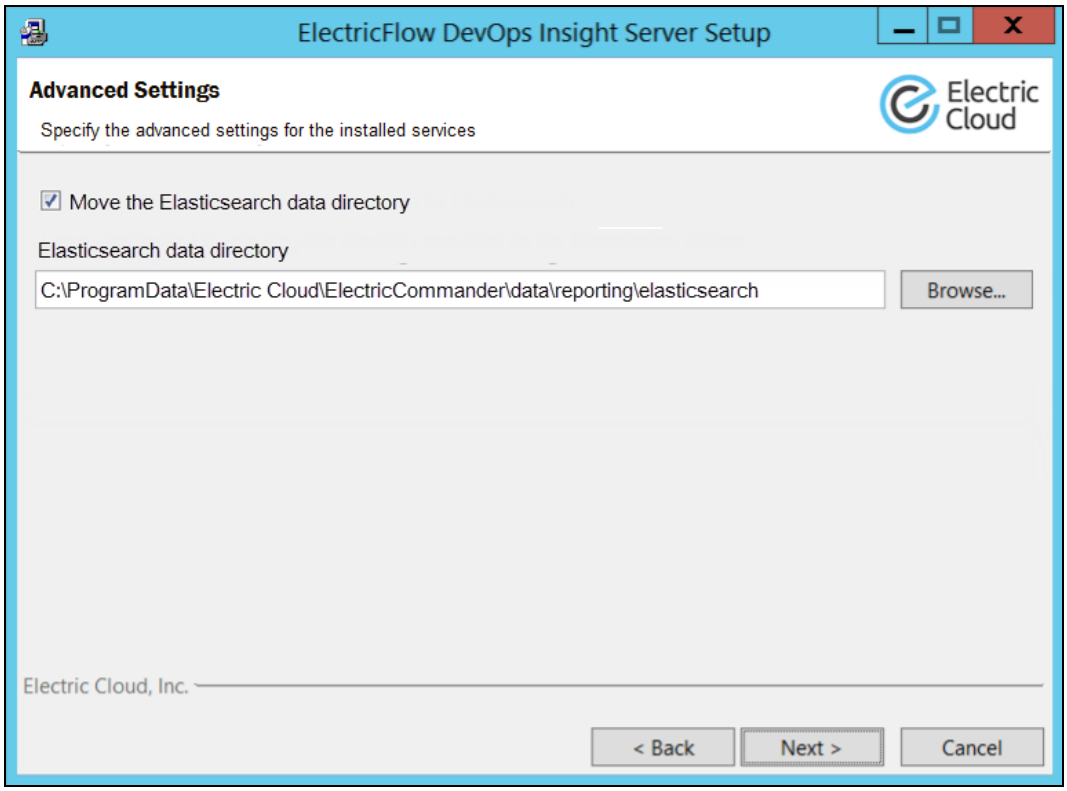
> **Important:** Unsecured access is not recommended for use in a production environment.

12. Complete the information on the **Security Settings** screen, and click **Next** to continue.

    The **Advanced Settings** screen appears:



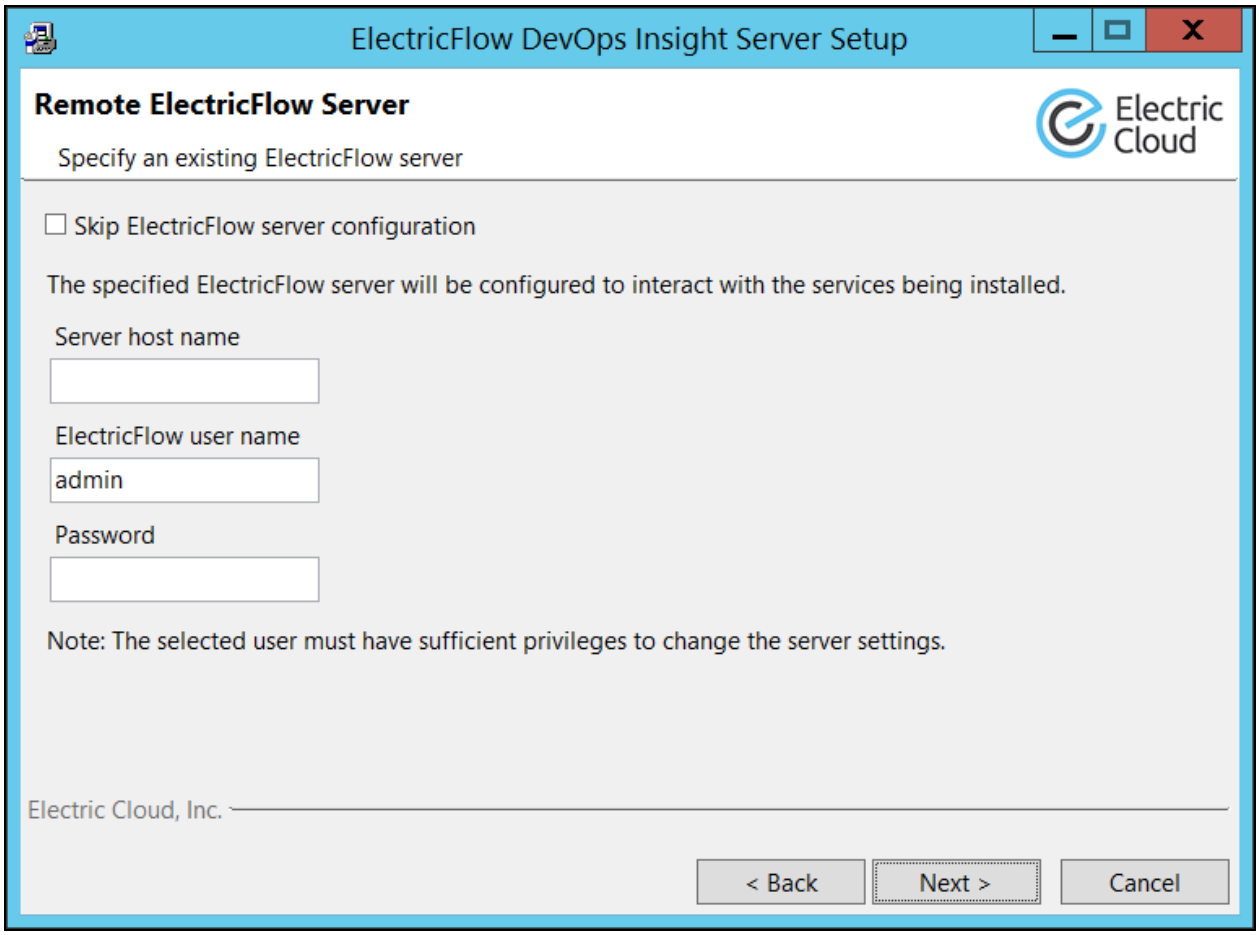
Use a different directory for data stored by Elasticsearch—Check this checkbox if you want to use a non-default directory for Elasticsearch index data. If you do so, the **Elasticsearch data directory** field appears to let you enter that directory:



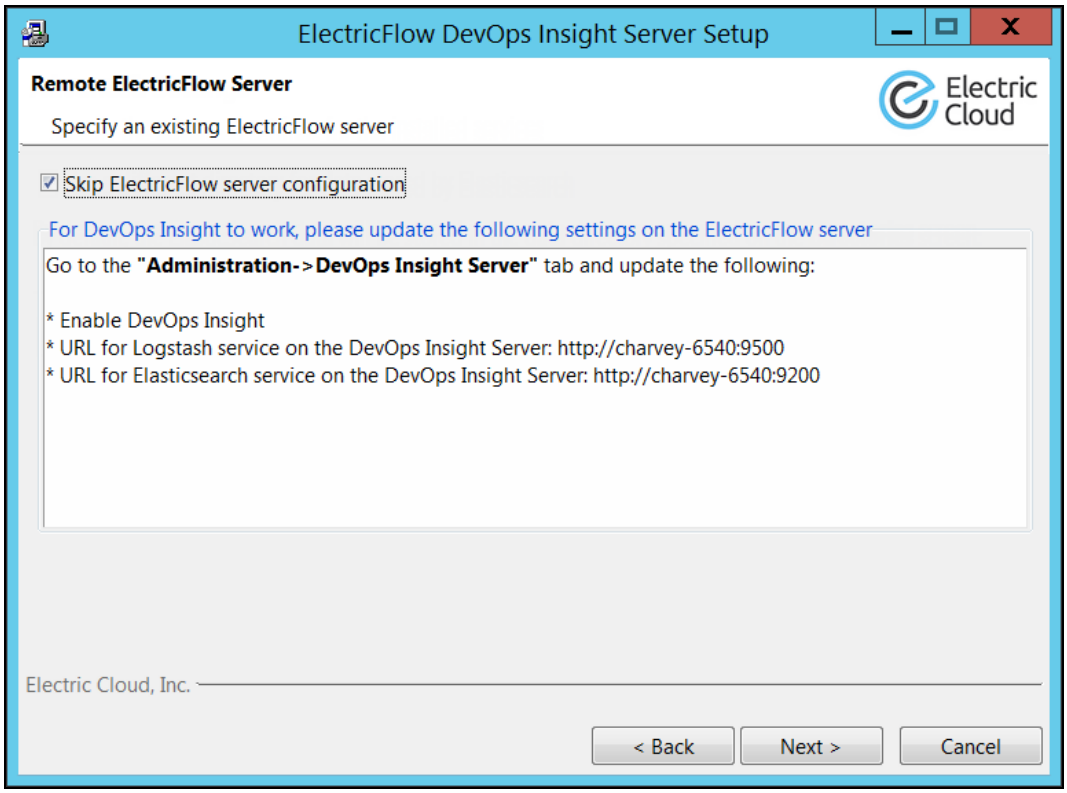
Then specify the certificate file containing a CA-signed certificate retrieved from the first node at the **PKCS#12 file containing a CA-signed certificate for the ElectricFlow DevOps Insight Server** prompt.

> **Note:** You can leave this entry blank for a new installation in non-clustered mode or for the first node in clustered mode. In this case, the installer will generate a new self-signed certificate and will use it to sign other TLS certificates.

13. Complete the information on the **Advanced Settings** screen, and click **Next** to continue.
14. The **Remote ElectricFlow Server** screen appears:



Skip ElectricFlow server configuration—Determines whether to skip the automatic configuration of the remote ElectricFlow server with the services being installed. If you choose this option, the screen changes as follows:



Otherwise, fill in the fields in the screen as follows:

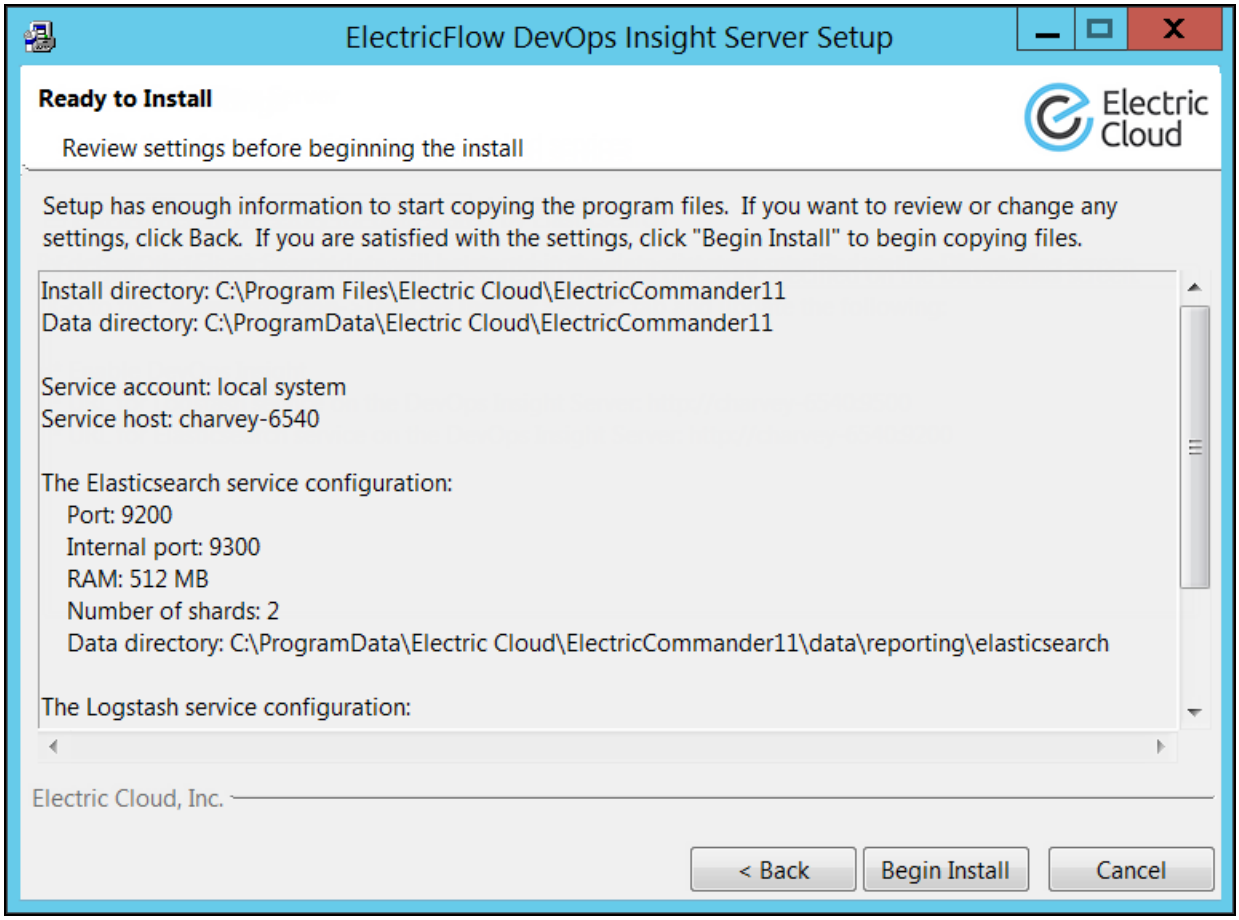
Server host name—Name of the ElectricFlow server that will communicate with this DevOps Insight server. If the remote server is using a non-default HTTPS port, you must enter `<host>:<port>`.

ElectricFlow User Name—Name of an ElectricFlow user on the ElectricFlow server who has sufficient privileges to edit server settings. This field defaults to the ElectricFlow-supplied `admin` user.

Password—Password for the ElectricFlow user. The default password for the `admin` user is `changeme`.

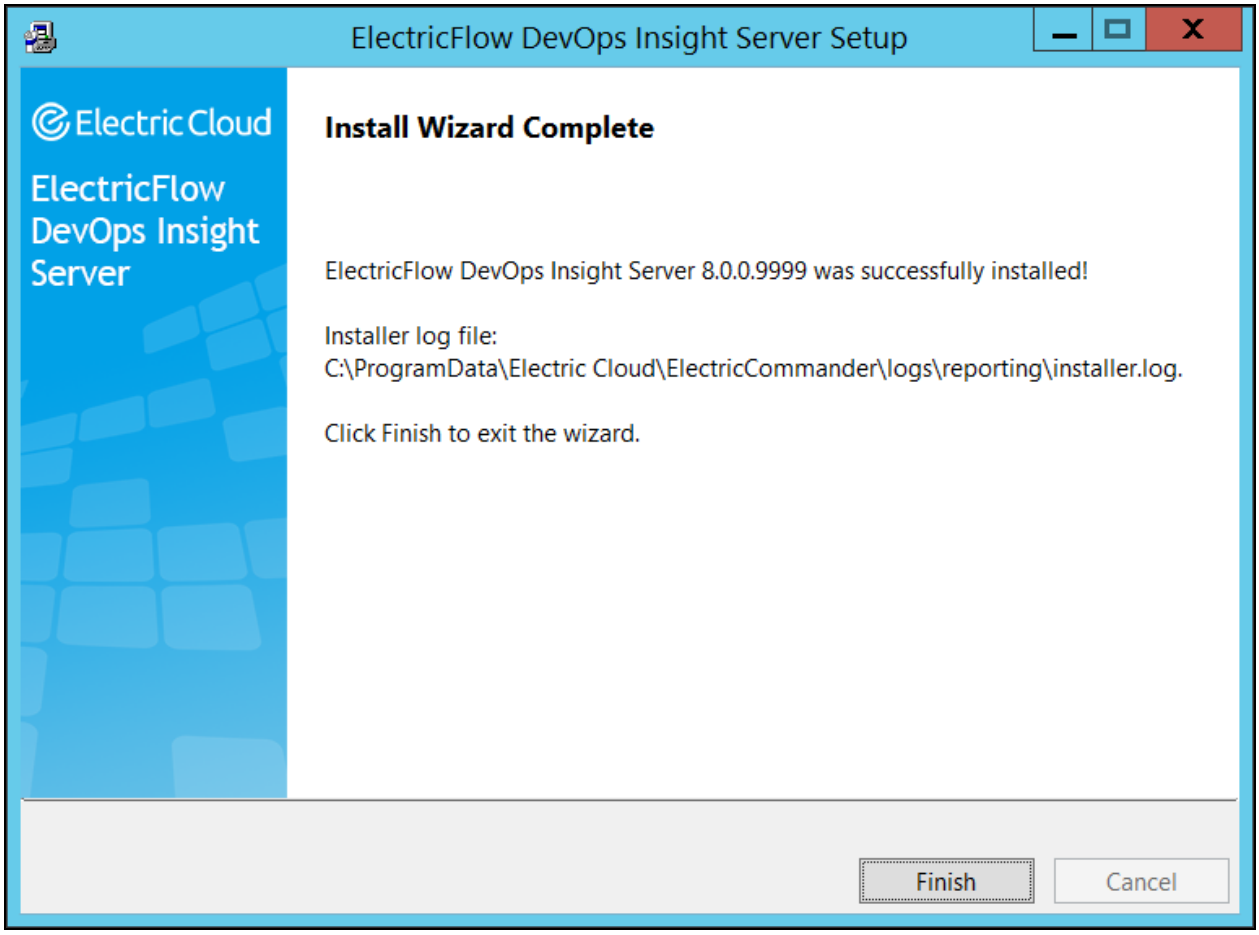
15. Complete the information on the **Remote ElectricFlow Server** screen, and click **Next** to continue.

The **Ready to Install** screen appears:



16. Review this screen to verify your selections. Use the **Back** button to change any of your settings if needed.
17. Click **Begin Install**.

    The installer displays a status bar to show the progress of the installation, which can take a few minutes. When the installation is complete, the **Install Wizard Complete** screen appears:



18. **Click Finish** to close the wizard.

Configuring DevOps Insight Server Services Autostart for Non-Root/Non-sudo Linux Installations

For non-root/non-`sudo` Linux installations, you must configure autostart for the DevOps Insight services. For instructions, see Configuring Services Autostart for Non-Root/Non-sudo Linux Installations on page 5-11.

Configuring the DevOps Insight Server on the ElectricFlow Server

If you chose to skip the option to configure the remote ElectricFlow server during the installation or upgrade of the DevOps Insight server, you must do so afterward to ensure connectivity and authentication between the DevOps Insight server and the ElectricFlow server. To do this, you use the **Administration** > **DevOps Insight Server** tab in the Automation Platform. For details, see the "DevOps Insight Server Configuration" section in the *ElectricFlow User Guide* at http://docs.electric-cloud.com/eflow_doc/FlowIndex.html.

Checking the DevOps Insight Server Configuration on the ElectricFlow Server

You can confirm the correct DevOps Insight Server settings by entering the following ectool command on the ElectricFlow server:

```
ectool getDevOpsInsightServerConfiguration
```

Following is sample output:

```
<response requestId="1" nodeId="192.168.5.138">
    <devOpsInsightServerConfiguration>
      <devOpsInsightServerConfigurationId>12642169-71c4-11e7-8a08-
0050568f29b0</devOpsInsightServerConfigurationId>
      <createTime>2017-07-26T05:34:19.404Z</createTime>
      <elasticSearchUrl>https://192.168.5.54:9200</elasticSearchUrl>
      <enabled>1</enabled>
      <lastModifiedBy>admin</lastModifiedBy>
      <logStashUrl>https://192.168.5.54:9500</logStashUrl>
      <modifyTime>2017-07-26T05:40:13.458Z</modifyTime>
      <owner>admin</owner>
      <userName>reportuser</userName>
    </devOpsInsightServerConfiguration>
  </response>
```

For details about the `getDevOpsInsightServerConfiguration` options, enter

```
ectool getDevOpsInsightServerConfiguration --help
```

Testing Connectivity and Authentication Between the DevOps Insight Server and the ElectricFlow Server

After you enable connectivity and authentication between the DevOps Insight server and the ElectricFlow server, you can perform a test by using one of the following methods:

- Check the **Test Connection** checkbox in the **Administration** > **DevOps Insight Server** subtab of the Administration Platform web UI on the ElectricFlow server and click **OK**.

- Enter the following ectool command on the ElectricFlow server:

  ```
  ectool setDevOpsInsightServerConfiguration --testConnection 1
  ```

  For details about the `setDevOpsInsightServerConfiguration` options, enter

  ```
  ectool setDevOpsInsightServerConfiguration --help
  ```

  For example, the following response appears if the user name or password is incorrect:

  ```
  ectool error [InvalidCredentials]: HTTP/1.1 401 Unauthorized: Access to
    'https://192.168.5.54:9500' is denied due to invalid credentials.
  ```

  Also, for example, the following response appears if you specify an invalid `elasticSearchUrl` or `logstashUrl`:

  ```
  ectool error [InvalidUrl]: The url 'https://192.168.5.54:9500' is invalid
  ```

  The following example shows the response when a valid `elasticSearchUrl` is used:

  ```
  /opt/electriccloud/electriccommander/bin$ ./ectool
  setDevOpsInsightServerConfiguration
    --elasticSearchUrl https://192.168.5.54:9200 --testConnection 1
  ```

Interactive Command-Line Installation Methods

The interactive command-line installation methods are supported only for Linux-only installations on a local Linux volume. Electric Cloud does not support installing the ElectricFlow server on a network volume.

> **Note:** You install ElectricFlow agent software on Linux with this installation method. For Solaris, HP-UX, macOS, AIX, or other supported UNIX agent machines, see Non-Server Platform Installation Method for UNIX Agents on page 14-30.

Running an Express Server Command-Line Installation

This option installs the ElectricFlow server, built-in database, web server, and repository server on one machine. The default ElectricFlow server settings are used. A local agent (required for running jobs), and ElectricFlow tools are also installed.

This option is available via a "full" installer file (see ElectricFlow Installer Files on page 3-1). This option is best for quickly installing the ElectricFlow software for evaluation purposes.

Review Before You Install ElectricFlow on page 3-7 before performing this procedure.

> **Note:** The built-in database is not supported in a clustered ElectricFlow configuration.

1. Enter the following command to make the installer file executable:

   ```
   chmod +x ./ElectricFlow-<version>
   ```

2. Do one of the following to start the installation:

   - For installations with root or `sudo` privileges, enter:

     `./ElectricFlow-x64-<version>`

   - For installations with root or `sudo` privileges and the X Window System, override the installer GUI by entering:

     `./ElectricFlow-x64-<version> --mode console`

   - For non-root/non-`sudo` installations, enter:

     `./ElectricFlow-x64-<version> --mode console --nonRoot`

     (After this command, enter `Y` at the following message:

     `Do you want to proceed installation as non-root user? [n/Y]`)

   The following prompt appears:

   ```
   Copyright (c) 2006-2018, Electric Cloud, Inc. All rights reserved.

   This will install ElectricFlow on your computer. Continue? [n/Y]
   ```

3. Continue the installation by entering `y`.

   The following prompt appears:

   ```
   Specify the type of setup you would like to perform:
   expressServer, expressAgent, or advanced. [expressServer]
   ```

4. Enter: `expressServer`.

   The following prompt appears:

   ```
   Specify the install directory (for program files and binaries).
   [/opt/electriccloud/electriccommander]
   ```

5. Press `Enter` to accept the default installation directory or enter a new directory.

   The following prompt appears:

   ```
   Specify the user the server, web, and/or repository will run as. []
   ```

6. Enter a user name.

   This is the user who owns the ElectricFlow server, repository server, and web server processes. For example, you might enter `build`.

   The following prompt appears:

   ```
   Specify the group the server, web, and/or repository will run as. []
   ```

7. Enter a group name.

   This is the group who owns the ElectricFlow server, repository server, and web server processes. For example, you might enter `build`.

   The following prompt appears:

```
Use the same service account for the agent (not recommended for production
systems)? [y/N]
```

For security reasons in production environments, you should use a separate user and group for the agent service. This is because the server service has permission to read the key file (`/opt/electriccloud/electriccommander/conf/passkey` in Linux or `C:\ProgramData\Electric Cloud\ElectricCommander\conf\passkey` in Windows). The key file is used to decrypt passwords stored in ElectricFlow. Using a different user and group for the agent service ensures that a process running on the agent cannot access the key file.

8. Choose one of the following options:

   - Enter `y` to use the same user and group for the agent service. This is not recommended for production systems.
   - Enter `n` to use a separate user and group for the agent service.

     The following prompt appears:

     ```
     Specify the user the agent will run as. []
     ```

     1. Enter a user name.

        This is the user who owns the ElectricFlow agent process. For example, you might enter `build`.

        The user/group that the agent runs as must have permission to write to the `$INSTALL_DIRECTORY/log` directory. If you specify `root`, enter `y` when the following confirmation appears:

        ```
        It is not recommended to use the 'root' user for running the agent
        process. Please confirm if you would like to proceed [y/N]
        ```

        The following prompt appears:

        ```
        Specify the group the agent will run as. []
        ```

     2. Enter a group name.

        This is the group that owns the ElectricFlow agent process. For example, you might enter `build`.

9. For non-root/non-`sudo` Linux installations, configure autostart for the ElectricFlow services.

   For instructions, see Configuring Services Autostart for Non-Root/Non-sudo Linux Installations on page 5-11.

   ElectricFlow is installed on the machine. When the installation completes successfully, a message that contains the line `ElectricFlow <version> was successfully installed!` appears.

Running an Advanced Command-Line Installation

The advanced command-line installation lets you install individual ElectricFlow components such as an ElectricFlow server, built-in database, web server, repository server, or ElectricFlow tools on specific machines. You can also change the default installation settings to accommodate your environment. Review Before You Install ElectricFlow on page 3-7 before performing this procedure.

1. Enter the following command to make the installer file executable:

   ```
   chmod +x ./ElectricFlow-<version>
   ```

2. Do one of the following to start the installation.

   - For installations with root or `sudo` privileges, enter:

     ```
     ./ElectricFlow-x64-<version>
     ```

   - For installations with root or `sudo` privileges and the X Window System, override the installer GUI by entering:

     ```
     ./ElectricFlow-x64-<version> --mode console
     ```

   - For non-root/non-`sudo` installations, enter:

     ```
     ./ElectricFlow-x64-<version> --mode console --nonRoot
     ```

     (After this command, enter `Y` at the following message:

     ```
     Do you want to proceed installation as non-root user? [n/Y]
     ```
)

   The following prompt appears:

   ```
   Copyright (c) 2006-2018, Electric Cloud, Inc. All rights reserved.

   This will install ElectricFlow on your computer. Continue? [n/Y]
   ```

3. Enter `y` to continue the installation.

   The following prompt appears:

   ```
   Specify the type of setup you would like to perform: expressServer,
   expressAgent, or advanced. [expressServer]
   ```

4. Enter `advanced`.

   The following prompt appears:

   ```
   Specify the install directory (for program files and binaries).
   [/opt/electriccloud/electriccommander]
   ```

5. Press `Enter` to accept the default or enter another directory.

   The following prompt appears:

   ```
   Install an ElectricFlow server? [n/Y]
   ```

6. Select the servers that you want to install on the current machine.

   For more information, see Architecture on page 1-4.

   > **Note:** If you want to install only the ElectricFlow tools, enter `n` for every server option. The ElectricFlow tools are installed automatically even if you choose not to install any server software.

   1. Enter `y` to install an ElectricFlow server.

      The following prompt appears:

```
Install a built-in database? [n/Y]
```

2. Choose one of the following options:

   - Enter `y` to install a built-in database.

     This is not recommended for production systems. Also, the built-in database is not supported in a clustered ElectricFlow configuration.

   - Enter `n` if you plan to use an external database.

     For more information, see External Database Configuration on page 5-2. If you plan to use MySQL, see Installing the MySQL JDBC Driver on page 3-181.

     The following prompt appears:

     ```
     Install an Apache web server? [n/Y]
     ```

3. Enter `y` to install an Apache web server.

   The following prompt appears:

   ```
   Install an ElectricFlow repository server? [n/Y]
   ```

4. Enter `y` to install an ElectricFlow repository server.

   The following prompt appears:

   ```
   Specify the install directory (for program files and binaries).
   [/opt/electriccloud/electriccommander]
   ```

> **Note:** Any combination of the following installation screens will appear depending on which servers you install.

7. Press `Enter` to accept the default installation directory, or enter a new installation directory path for program files and binaries.

   The following prompt appears:

   ```
   Specify the data directory (for configuration files and logs).
   [/opt/electriccloud/electriccommander]
   ```

8. Press `Enter` to accept the default installation directory, or enter a new installation directory path for configuration files and logs.

   The software displays prompts for server port values. The prompts that appear will vary depending on the server software you previously selected to install.

9. Press `Enter` to accept the default port values, or enter alternate port numbers if you need to specify a different port value.

   The following prompt only appears if you are installing an Apache web server. If you are not installing a web server, you will see a prompt to enter a user name.

   ```
   Specify the host name that users will type in their browser to access the web
   server. [hostName]
   ```

10. Enter a web server host name if you are installing an Apache web server.

    This is the host name users need to type into their browser to access the ElectricFlow web server.

    The following prompt appears:

    ```
    Specify the user the server, web, and/or repository will run as. []
    ```

11. Enter a user name if you are installing an ElectricFlow, web, or repository server.

    This is the user who owns the ElectricFlow server, repository server, and web server processes. For example, you might enter `build`.

    The following prompt appears:

    ```
    Specify the group the server, web, and/or web repository will run as. []
    ```

12. Enter a group name if you are installing an ElectricFlow, web, or repository server.

    This is the group who owns the ElectricFlow server, repository server, and web server processes. For example, you might enter `build`.

    The following message only appears if an agent is installed on this machine. An agent is required on this machine whenever you install a web or repository server and is therefore automatically installed. For details about why local agents are required on web server machines, see Before You Install ElectricFlow on page 3-7.

    > **Note:** You should not use these local agents to run jobs.

    ```
    Use the same service account for the agent (not recommended for production
    systems)? [y/N]
    ```

    > **Note:** For security reasons in production environments, you might want to use a separate user and group for the agent service because the server service has permission to read the key file (`/opt/electriccloud/electriccommander/conf/passkey` in Linux or `C:\ProgramData\Electric Cloud\ElectricCommander\conf\passkey` in Windows). The key file is used to decrypt passwords stored in ElectricFlow. Using a different user and group for the agent service ensures that a process running on the agent cannot gain access to the key file.

13. Choose one of the following options if an agent is automatically installed with the server:

    - Enter `y` to use the same user and group for the agent service. This is not recommended for production systems.
    - Enter `n` to use a separate user and group for the agent service.

      The following prompt appears:

      ```
      Specify the user the agent will run as. []
      ```

      - Enter a user name. This is the user who owns the ElectricFlow agent process.

        The user/group that the agent runs as must have permission to write to the `$INSTALL_DIRECTORY/log` directory. If you specify `root`, enter `y` when the following confirmation appears:

```
It is not recommended to use the 'root' user for running the agent
process. Please confirm if you would like to proceed [y/N]
```

The following prompt appears:

```
Specify the group the agent will run as. []
```

2. Enter a group name. This is the group that owns the ElectricFlow agent process.

Several prompts appear that ask you to specify the TLS or SSL protocols for the agent, ElectricFlow server, and repository server components:

```
Specify SSL/TLS protocols for Agent component [TLSv1,TLSv1.1,TLSv1.2]

Specify SSL/TLS protocols for Server component [TLSv1,TLSv1.1,TLSv1.2]

Specify SSL/TLS protocols for Repository component [TLSv1,TLSv1.1,TLSv1.2]
```

14. Answer the security protocol prompts as follows.

For each component, you can specify a comma-separated list of any combination of `TLSv1`, `TLSv1.1`, `TLSv1.2`, and `SSLv2Hello`.

The default security configurations are as follows:

- First-time ElectricFlow installations: TLSv1, TLSv1.1, and TLSv1.2 are enabled
- Existing ElectricFlow installations: TLSv1, TLSv1.1, TLSv1.2, and SSLv2Hello are enabled

The default security configuration for upgrades from version 8.5 and newer versions is inherited from the existing installation being upgraded.

To avoid the following warning in the Automation Platform web UI, we recommend removing the `SSL 2.0 Client Hello` or `SSLv2Hello` protocol from your security configurations for all components:

We recommend removing "SSL 2.0 Client Hello format from server configuration and upgrade older agents as indicated on the Cloud / Resources Page to avoid security risk.

To safely remove this protocol, enter the following command on the ElectricFlow server:

```
$ ecconfigure --serverTLSEnabledProtocol=TLSv1,TLSv1.1,TLSv1.2
```

When you do this, you would also need to upgrade older agents to the latest version to avoid security risks. You would need to upgrade agents if you are using the following agent versions:

- Windows, Linux: 6.0.3 or older; 6.2 or older
- Sun Solaris, HP UX, Mac OS: 8.4 or older

The following message appears:

```
Installing ElectricFlow...
```

ElectricFlow is installed on the machine. When the installation completes successfully, a message that contains the line `ElectricFlow <version> was successfully installed!` appears.

14. For non-root/non-`sudo` Linux installations, configure autostart for the ElectricFlow services.

    For instructions, see Configuring Services Autostart for Non-Root/Non-sudo Linux Installations on page 5-11.

Running an Express Agent Command-Line Installation

The agent software must be installed on each machine you intend to use with ElectricFlow. An agent is an ElectricFlow component that runs on a machine resource. The agent executes ElectricFlow job steps, monitors step progress, and records job completion information.

Review Before You Install ElectricFlow on page 3-7 before performing this procedure.

1. Enter the following command to make the installer file executable:

   ```
   chmod +x ./ElectricFlow-<version>
   ```

2. Choose one of the following commands to begin the upgrade:

   - If you have a Linux platform, enter `./ElectricFlow-<version>`.
   - For installations with root or `sudo` privileges and the X Window System, override the installer GUI by entering:

     ```
     ./<agent_installer_file> --mode console
     ```

   The following prompt appears:

   ```
   Copyright (c) 2006-2018, Electric Cloud, Inc. All rights reserved.

   This will install ElectricFlow on your computer. Continue? [n/Y]
   ```

3. Continue the installation by entering `y`.

   The following prompt appears:

   ```
   Specify the type of setup you would like to perform: expressServer,
   expressAgent, or advanced. [expressServer]
   ```

4. Enter: `expressAgent`.

   The following prompt appears:

   ```
   Discover the plugins directory from a remote ElectricFlow server? [n/Y]
   ```

5. Enter `y` if you want the agent machine to have access to the plugins directory. You should allow access to the plugins directory so agents have access to collections of features, third-party integrations, or third-party tools.

   > **Important:** The plugins directory on the ElectricFlow server must be "shared" before the agent machine can use "discover" to find the directory. For more information, see Universal Access to the Plugins Directory on page 5-21.

   The following prompt appears:

   ```
   Create a resource for the installed agent on a remote ElectricFlow server? [n/Y]
   ```

6. Enter `y` to automatically create a resource object for the agent on a remote ElectricFlow server. This option is recommended to save time configuring new ElectricFlow resources for **pre-existing** ElectricFlow servers.

The following prompt appears:

```
Register as trusted agent (required for gateway)? [y/N]
```

> **Note:** Making an agent trusted restricts the agent to one ElectricFlow server. The agent will not respond to incoming communication from any other ElectricFlow server. This is useful when you want to create a secure production environment, but generally not needed for test or development systems.

> **Important:**
> You can run gateways without trusted agents. However, you should use gateways with trusted agents to prevent security issues in the firewall between zones connected by a gateway.
>
> There are exceptions to using gateways without trusted agents:
>
> - The firewall between two zones is not required in your environment or is needed only to protect the ElectricFlow server.
> - There is a specific reason to use gateways without trusted agents, such as a requirement to prevent unauthorized users from accessing your network. All incoming traffic from the internet is routed to a data center through a load balancer, and the load balancer routes the traffic to the appropriate machine in your network.

7. Choose one of the following options:

   - If a gateway used to communicate with the ElectricFlow server, you must select `y`. This option allows you to create a trusted network connection between the agent and server under the same certificate authority. This will allow the agent and the ElectricFlow server to communicate across the network.
   - If there is no gateway between the agent and ElectricFlow server, enter `n`.

   > **Important:** If you deviated from the recommended agent options, you will see variations in the installation options that appear on your system.

   The following prompt appears:

   ```
   Create repository and/or agent in the default zone? [n/Y]
   ```

8. Enter `y` to create the agent in the default zone.

   The following prompt appears:

   ```
   Specify the hostName:port of a remote ElectricFlow server the agent, repository
   server and/or web server being installed can link to. The port is only required
   if it is not the default. [] <hostName:port>
   ```

9. Enter the Server Host Name of the ElectricFlow server that will communicate with this agent. You must specify the Server Host Name as *<hostName>:>port>* if the remote server is using a non-default HTTPS port. If you do not specify a port, HTTPS port 8443 is assumed (the same as the ElectricFlow server default port).

   The following prompt appears:

```
Specify the user name with which to login to <hostName>:<port>. [admin]
```

10. Enter the user name of a user on the ElectricFlow server who has sufficient privileges to create a resource. The default is the ElectricFlow-supplied `admin` user.

    The following prompt appears:

    ```
    Specify the password for "<electricflow_user>" on <hostName>:<port>. []
    ```

11. Enter the password for the ElectricFlow user. The default password for the `admin` user is `changeme`.

    The following prompt appears:

    ```
    Specify the name of the resource to create on <<hostName>:<port>. [<resource_
    name>]
    ```

12. Enter the following information if the agent must be registered as a trusted agent. These options only appear if you entered `y` for `Register as trusted agent (required for gateway)? [y/N]`.

    1. Enter a resource name to use on the ElectricFlow server.

       The following prompt appears:

       ```
       Specify the agent gateway URL in the form of 'ipOrHostname:port' []
       ```

    2. Enter an agent gateway URL. This is the URL of the gateway used to communicate with the ElectricFlow server.

       The following prompt appears:

       ```
       Specify the zone name for the agent and/or repository []
       ```

    3. Enter the Zone Name. This is the zone used during remote agent and or remote repository creation.

       The following prompt appears:

       ```
       Specify the user the agent will run as. []
       ```

    4. Enter a user name. This is the user who owns the ElectricFlow agent process. For example, you might enter `build`.

13. The user/group that the agent runs as must have permission to write to the `$INSTALL_DIRECTORY/log` directory. If you specify `root`, enter `y` when the following confirmation appears:

    ```
    It is not recommended to use the 'root' user for running the agent process.
    Please confirm if you would like to proceed [y/N]
    ```

    The following prompt appears:

    ```
    Specify the group the agent will run as. []
    ```

14. Enter a Group Name. This is the group that owns the ElectricFlow agent process. For example, you might enter `build`.

    ElectricFlow is installed on the machine. When the installation completes successfully, a prompt that contains the line "`ElectricFlow <version> was successfully installed!`" appears.

Running an Express Agent Command-Line Installation (Agent-Only Installer)

The agent software must be installed on each machine you intend to use with ElectricFlow. An agent is an ElectricFlow component that runs on a machine resource. The agent executes ElectricFlow job steps, monitors step progress, and records job completion information.

Certain ElectricFlow installers allow you to perform installations as a non-root user or a user without `sudo` privileges. To determine whether a particular installer has an option to run in this mode, see Availability of Installers with a Non-Root/Non-sudo or Non-Administrator Mode on page 3-2.

Review Before You Install ElectricFlow on page 3-7 before performing this procedure.

1. Download the appropriate agent-only installer file.

   For details, see ElectricFlow Installer Files on page 3-1.

2. Enter the following command to make the installer file executable:

   ```
   chmod +x <agent_installer_file>
   ```

   For example, enter:

   ```
   chmod +x ElectricFlowAgent-x64-8.4.0.129860-new-with-64bit-perl
   ```

3. Choose one of the following commands to begin the installation:

   - For installations with root or `sudo` privileges, enter:

     ```
     ./<agent_installer_file>
     ```

   - For installations with root or `sudo` privileges and the X Window System, override the installer GUI by entering:

     ```
     ./<agent_installer_file> --mode console
     ```

   - For non-root/non-`sudo` installations, enter:

     ```
     ./<agent_installer_file> --mode console --nonRoot
     ```

4. After the confirmation prompt, continue the installation by entering `y`.

   The following prompt appears:

   ```
   Specify the type of setup you would like to perform: expressAgent or advanced.
   [expressAgent]
   ```

5. Press `Enter` to accept `expressAgent`.

   The following prompt appears:

   ```
   Discover the plugins directory from a remote ElectricFlow server? [n/Y]
   ```

6. Enter `y` if you want the agent machine to have access to the plugins directory. You should allow access to the plugins directory so agents have access to collections of features, third-party integrations, or third-party tools.

   Important: The plugins directory on the ElectricFlow server must be "shared" before the agent machine can use "discover" to find the directory. For more information, see Universal Access to the Plugins Directory on page 5-21.

The following prompt appears:

```
Create a resource for the installed agent on a remote ElectricFlow server? [n/Y]
```

7. Enter `y` to automatically create a resource object for the agent on a remote ElectricFlow server. This option is recommended to save time configuring new ElectricFlow resources for existing ElectricFlow servers.

   The following prompt appears:

   ```
   Register as trusted agent? [y/N]
   ```

   Making an agent trusted restricts the agent to one ElectricFlow server. The agent will not respond to incoming communication from any other ElectricFlow server. This is useful when you want to create a secure production environment, but generally not needed for test or development systems.

   Important:
   You can run gateways without trusted agents. However, you should use gateways with trusted agents to prevent security issues in the firewall between zones connected by a gateway.

   There are exceptions to using gateways without trusted agents:

   - The firewall between two zones is not required in your environment or is needed only to protect the ElectricFlow server.
   - There is a specific reason to use gateways without trusted agents, such as a requirement to prevent unauthorized users from accessing your network. All incoming traffic from the internet is routed to a data center through a load balancer, and the load balancer routes the traffic to the appropriate machine in your network.

8. Choose one of the following options:

   - If a gateway is used to communicate with the ElectricFlow server, you must select `y`. This option allows you to create a trusted network connection between the agent and server under the same certificate authority. This will allow the agent and the ElectricFlow server to communicate across the network.
   - If there is no gateway between the agent and ElectricFlow server, enter `n`.

     Note: If you deviated from the recommended agent options, you will see variations in the installation options that appear on your system.

     For root or `sudo` installations, The following prompt appears:

     ```
     Specify the user the agent will run as. []
     ```

9. (Root or `sudo` installations) Enter a user name. This is the user who owns the ElectricFlow agent process. For example, you might enter `build`.

   The user/group that the agent runs as must have permission to write to the `$INSTALL_DIRECTORY/log` directory. If you specify `root`, enter `y` when the following confirmation appears:

   ```
   It is not recommended to use the 'root' user for running the agent process.
   Please confirm if you would like to proceed [y/N]
   ```

   The following prompt appears:

```
Specify the group the agent will run as. []
```

10. (Root or `sudo` installations) Enter a Group Name. This is the group that owns the ElectricFlow agent process. For example, you might enter `build`.

    ElectricFlow is installed on the machine. When the installation completes successfully, a prompt that contains the line "`ElectricFlow <version> was successfully installed!`" appears.

11. For non-root/non-`sudo` Linux installations, configure autostart for the ElectricFlow agent service.

    For instructions, see Configuring Services Autostart for Non-Root/Non-sudo Linux Installations on page 5-11.

Running an Advanced Agent Command-Line Installation (Agent-Only Installer)

The agent software must be installed on each machine you intend to use with ElectricFlow. An agent is an ElectricFlow component that runs on a machine resource. The agent executes ElectricFlow job steps, monitors step progress, and records job completion information.

Certain ElectricFlow installers allow you to perform installations as a non-root user or a user without `sudo` privileges. To determine whether a particular installer has an option to run in this mode, see Availability of Installers with a Non-Root/Non-sudo or Non-Administrator Mode on page 3-2.

Review Before You Install ElectricFlow on page 3-7 before performing this procedure.

1. Download the appropriate agent-only installer file.

   For details, see ElectricFlow Installer Files on page 3-1.

2. Enter the following command to make the installer file executable:

   ```
   chmod +x <agent_installer_file>
   ```

   For example, enter:

   ```
   chmod +x ElectricFlowAgent-x64-8.4.0.129860-new-with-64bit-perl
   ```

3. Choose one of the following commands to begin the upgrade:

   - For installations with root or `sudo` privileges, enter:

     ```
     ./<agent_installer_file>
     ```

   - For installations with root or `sudo` privileges and the X Window System, override the installer GUI by entering:

     ```
     ./<agent_installer_file> --mode console
     ```

   - For non-root/non-`sudo` installations, enter:

     ```
     ./<agent_installer_file> --mode console --nonRoot
     ```

4. After the confirmation prompt, continue the installation by entering `y`.

   The following prompt appears:

   ```
   Specify the type of setup you would like to perform: expressAgent or advanced.
   [expressAgent]
   ```

5. Enter `advanced`.

   The following prompt appears:

   ```
   Specify the install directory (for program files and binaries).
   [/opt/electriccloud/electriccommander]
   ```

6. Enter a new installation directory path for program files and binaries.

   The following prompt appears:

   ```
   Specify the data directory (for configuration files and logs).
   [/opt/electriccloud/electriccommander]
   ```

7. Enter a new installation directory path for configuration files and logs.

   The following prompt appears:

   ```
   Specify the agent port. [7800]
   ```

8. Enter a different port to eliminate any conflicts with your existing system configuration.

   The following prompt appears:

   ```
   Specify the agent local port. [6800]
   ```

9. Enter a different port to be used by the agent for HTTP communication on the localhost network interface.

   The following prompt appears:

   ```
   Discover the plugins directory from a remote ElectricFlow server? [n/Y]
   ```

10. Enter `y` if you want the agent machine to have access to the plugins directory. You should allow access to the plugins directory so agents have access to collections of features, third-party integrations, or third-party tools.

    > **Note:** The plugins directory on the ElectricFlow server must be "shared" before the agent machine can use "discover" to find the directory. For more information, see Universal Access to the Plugins Directory on page 5-21.

    The following prompt appears:

    ```
    Create a resource for the installed agent on a remote ElectricFlow server? [n/Y]
    ```

11. Enter `y` to automatically create a resource object for the agent on a remote ElectricFlow server. This option is recommended to save time configuring new ElectricFlow resources for existing ElectricFlow servers.

    The following prompt appears:

    ```
    Register as trusted agent? [y/N]
    ```

    Making an agent trusted restricts the agent to one ElectricFlow server. The agent will not respond to incoming communication from any other ElectricFlow server. This is useful when you want to create a secure production environment, but generally not needed for test or development environments.

    > **Important:**
    > You can run gateways without trusted agents. However, you should use gateways with

trusted agents to prevent security issues in the firewall between zones connected by a gateway.

There are exceptions to using gateways without trusted agents:

- The firewall between two zones is not required in your environment or is needed only to protect the ElectricFlow server.
- There is a specific reason to use gateways without trusted agents, such as a requirement to prevent unauthorized users from accessing your network. All incoming traffic from the internet is routed to a data center through a load balancer, and the load balancer routes the traffic to the appropriate machine in your network.

12. Choose one of the following options:

    - If a gateway is used to communicate with the ElectricFlow server, you must select `y`. This option allows you to create a trusted network connection between the agent and server under the same certificate authority. This will allow the agent and the ElectricFlow server to communicate across the network.
    - If there is no gateway between the agent and ElectricFlow server, enter `n`.

    Important: If you deviated from the recommended agent options, you will see variations in the installation options that appear on your system.

    For root or `sudo` installations, The following prompt appears:

    ```
    Specify the user the agent will run as. []
    ```

13. (Root or `sudo` installations) Enter a user name. This is the user who owns the ElectricFlow agent process. For example, you might enter `build`.

    The user/group that the agent runs as must have permission to write to the `$INSTALL_DIRECTORY/log` directory. If you specify `root`, enter `y` when the following confirmation appears:

    ```
    It is not recommended to use the 'root' user for running the agent process.
    Please confirm if you would like to proceed [y/N]
    ```

    The following prompt appears:

    ```
    Specify the group the agent will run as. []
    ```

14. (Root or `sudo` installations) Enter a Group Name. This is the group that owns the ElectricFlow agent process. For example, you might enter `build`.

    ElectricFlow is installed on the machine. When the installation completes successfully, a prompt that contains the line "`ElectricFlow <version> was successfully installed!`" appears.

15. For non-root/non-`sudo` Linux installations, configure autostart for the ElectricFlow agent service.

    For instructions, see Configuring Services Autostart for Non-Root/Non-sudo Linux Installations on page 5-11.

Running an Express Agent Command-Line Installation (Agent-Only Installer) When the Server Uses Registered and Concurrent Licenses

Use this procedure when the ElectricFlow server uses a mix of registered and concurrent licenses.

The agent software must be installed on each machine you intend to use with ElectricFlow. An agent is an ElectricFlow component that runs on a machine resource. The agent executes ElectricFlow job steps, monitors step progress, and records job completion information.

Certain ElectricFlow installers allow you to perform installations as a non-root user or a user without `sudo` privileges. To determine whether a particular installer has an option to run in this mode, see Availability of Installers with a Non-Root/Non-sudo or Non-Administrator Mode on page 3-2.

Review Before You Install ElectricFlow on page 3-7 before performing this procedure.

1. Download the appropriate agent-only installer file.

   For details, see ElectricFlow Installer Files on page 3-1.

2. Enter the following command to make the installer file executable:

   ```
   chmod +x <agent_installer_file>
   ```

   For example, enter:

   ```
   chmod +x ElectricFlowAgent-x64-8.4.0.129860-new-with-64bit-perl
   ```

3. Choose one of the following commands to begin the upgrade:

   - For installations with root or `sudo` privileges, enter:

     ```
     ./<agent_installer_file>
     ```

   - For installations with root or `sudo` privileges and the X Window System, override the installer GUI by entering:

     ```
     ./<agent_installer_file> --mode console
     ```

   - For non-root/non-`sudo` installations, enter:

     ```
     ./<agent_installer_file> --mode console --nonRoot
     ```

4. After the confirmation prompt, enter `y` to continue the installation.

   The following prompt appears:

   ```
   Specify the type of setup you would like to perform: expressAgent or advanced.
   [expressAgent]
   ```

5. Press `Enter` to accept `expressAgent`.

   The following prompt appears:

   ```
   Discover the plugins directory from a remote ElectricFlow server? [n/Y]
   ```

6. Enter `y` if you want the agent machine to have access to the plugins directory. You should allow access to the plugins directory so agents have access to collections of features, third-party integrations, or third-party tools.

> **Note:** The plugins directory on the ElectricFlow server must be "shared" before the agent machine can use "discover" to find the directory. For more information, see Universal Access to the Plugins Directory on page 5-21.

The following prompt appears:

```
Create a resource for the installed agent on a remote ElectricFlow server? [n/Y]
```

7. Enter `y` to automatically create a resource object for the agent on a remote ElectricFlow server. This option is recommended to save time configuring new ElectricFlow resources for *pre-existing* ElectricFlow servers.

   The following prompt appears:

   ```
   Register as trusted agent? [y/N]
   ```

   Making an agent trusted restricts the agent to one ElectricFlow server. The agent will not respond to incoming communication from any other ElectricFlow server. This is useful when you want to create a secure production environment, but generally not needed for test or development environments.

   > **Important:**
   > You can run gateways without trusted agents. However, you should use gateways with trusted agents to prevent security issues in the firewall between zones connected by a gateway.
   >
   > There are exceptions to using gateways without trusted agents:
   >
   > - The firewall between two zones is not required in your environment or is needed only to protect the ElectricFlow server.
   > - There is a specific reason to use gateways without trusted agents, such as a requirement to prevent unauthorized users from accessing your network. All incoming traffic from the internet is routed to a data center through a load balancer, and the load balancer routes the traffic to the appropriate machine in your network.

8. Enter `n` if you are installing the ElectricFlow Community Edition.

   The following prompt appears:

   ```
   Create repository and/or agent in the default zone? [y/n]
   ```

9. Enter `y`.

   The following prompt appears:

   ```
   Specify the host:port of a remote ElectricFlow server that the agent,
   repository server and/or web server being installed can link to. The
   port is only required if it is not the default. []
   ```

10. Enter the `<host:port>`.

    The following prompt appears:

    ```
    Specify the user name with which to login to "<host:port>". [admin]
    ```

11. Enter `admin`.

    The following prompt appears:

    ```
    Specify the password for "admin" on "<host:port>". []
    ```

12. Enter a password.

    The following prompt appears:

    ```
    Specify the name of the resource to create on "<host:port>". []
    ```

13. Enter a resource name.

    The following prompt appears:

    ```
    Specify resource type for remote server: Registered or Concurrent. []
    ```

14. Enter `Registered`.

    For root or `sudo` installations, The following prompt appears:

    ```
    Specify the user the agent will run as. []
    ```

15. (Root or `sudo` installations) Enter a user name.

    This is the user who owns the ElectricFlow agent process. For example, you can enter `deploy`.

    The user/group that the agent runs as must have permission to write to the `$INSTALL_DIRECTORY/log` directory. If you specify `root`, enter `y` when the following confirmation appears:

    ```
    It is not recommended to use the 'root' user for running the agent process.
    Please confirm if you would like to proceed [y/N]
    ```

    The following prompt appears:

    ```
    Specify the group the agent will run as. []
    ```

16. (Root or `sudo` installations) Enter a group name.

    This is the group that owns the ElectricFlow agent process. For example, you can enter `deploy`.

    ElectricFlow is installed on the machine.

    When the installation completes successfully, a prompt that contains the line "`ElectricFlow <version> was successfully installed!`" appears.

17. For non-root/non-`sudo` Linux installations, configure autostart for the ElectricFlow agent service.

    For instructions, see Configuring Services Autostart for Non-Root/Non-sudo Linux Installations on page 5-11.

Running a DevOps Insight Server Interactive Command-Line Installation

The command-line user interface installation method is supported only by Linux platforms. In this mode, additional command line parameters that are listed in Windows or Linux DevOps Insight Server Silent Unattended Installation Example on page 3-130 can be used. The following procedure includes instructions for adding a system to a DevOps Insight cluster during installation.

For details about the overall steps for installing DevOps Insight on a group of servers to create a DevOps Insight server cluster, see Creating a DevOps Insight Server Cluster on page 4-36.

Installing the DevOps Insight Server on a System with Other ElectricFlow Components

For a production environment, Electric Cloud recommends that you install the DevOps Insight server on a system other than systems running other ElectricFlow components (such as the ElectricFlow server, web server, repository server, or agent). If you must install it on the same system (such as for testing or other non-production or trial-basis situations) see Running the DevOps Insight Server on a System with Other ElectricFlow Components on page 3-8 for details.

Installing the DevOps Insight Server

1. Enter the following command to make the installer file executable:

   ```
   chmod +x ./ElectricFlowDevOpsInsightServer-x64-<version>
   ```

2. Choose one of the following commands to begin the installation:

   - For installations with root or `sudo` privileges, enter:

     ```
     ./ElectricFlowDevOpsInsightServer-x64-<version>
     ```

   - For installations with root or `sudo` privileges and the X Window System, override the installer GUI by entering:

     ```
     ./ElectricFlowDevOpsInsightServer-x64-<version> --mode console
     ```

   - For non-root/non-`sudo` installations, enter:

     ```
     ./ElectricFlowDevOpsInsightServer-x64-<version> --mode console --nonRoot
     ```

     (After this command, enter `Y` at the following message:

     ```
     Do you want to proceed installation as non-root user? [n/Y]
     ```
)

   The following prompt appears:

   ```
   Logging to "/tmp/ijtmp_00CB8424-9E21-C4E5-3357-0E5B11BADFA6/installer-
   EFlowReportServ.log"
   Installing temporary...
   Copyright (c) 2006-2018, Electric Cloud, Inc. All rights reserved.

   This will install ElectricFlow DevOps Insight Server on your computer.
   Continue? [n/Y]
   ```

3. Continue the installation by entering `y`.

   The following prompt appears:

   ```
   Specify the install directory (for binaries)
   [/opt/electriccloud/electriccommander]
   ```

4. Press `Enter` to accept the default installation directory, or enter a new installation directory path for program files and binaries.

   The following prompt appears:

   ```
   Specify the data directory (for data files, configurations and logs)
   [/opt/electriccloud/electriccommander]
   ```

5. Press `Enter` to accept the default installation directory, or enter a new installation directory path for data files, configurations, and logs.

   The following prompt appears:

   ```
   Specify the user the services will run as []
   ```

6. Enter the name of the user who owns the ElectricFlow DevOps Insight server processes.

   The following prompt appears:

   ```
   Specify the group the services will run as [<primary group>]
   ```

7. Enter the name of the group who owns the ElectricFlow DevOps Insight server processes, or accept the default primary group of the chosen user by pressing `Enter`.

   The following prompt appears:

   ```
   Choose the port which will be used by Elasticsearch [9200]
   ```

   The DevOps Insight server uses the Elasticsearch search engine and the Logstash data-collection engine to gather data from the ElectricFlow server for use in the DevOps Insight dashboards.

8. If you want to specify a non-default port number, enter that number, or accept the default port number by pressing `Enter`.

   The following prompt appears:

   ```
   Choose the port which will be used by the Elasticsearch service for
   communication between nodes within the Elasticsearch cluster [9300]
   ```

   This port is used for internal communication between nodes within the Elasticsearch cluster.

9. If you want to specify a non-default port number, enter that number, or accept the default port number by pressing `Enter`.

   The following prompt appears:

   ```
   Choose the port which will be used by Logstash [9500]
   ```

10. If you want to specify a non-default port number, enter that number, or accept the default port number by pressing `Enter`.

    The following prompt appears:

    ```
    Choose the port which will be used by the Logstash service for the Logstash
    monitoring APIs [9600]
    ```

    This port is used by the Logstash monitoring APIs that provide runtime metrics about Logstash.

11. If you want to specify a non-default port number, enter that number, or accept the default port number by pressing `Enter`.

    The following prompt appears:

    ```
    Do you want to specify additional Elasticsearch cluster mode settings? [y/N] y
    ```

12. (Optional) Enter `y` if you want to add this system to a DevOps Insight server cluster. Otherwise, enter `n`.

    If you enter `y`, the following prompt appears:

    ```
    Specify the name of the Elasticsearch cluster [elasticsearch]
    ```

    > **Note:** The following prompts related to the cluster are skipped if you declined to configure it automatically.

13. Enter the name of the cluster.

    The following prompt appears:

    ```
    Specify comma-delimited list of other nodes in the Elasticsearch cluster that
    are likely to be live and reachable [127.0.0.1,[::1]]
    ```

14. Enter any additional nodes that are running DevOps Insight and can become part of the cluster.

    These can be nodes (whether they are master-eligible or not). You can enter any combination of IP addresses or host names.

    The following prompt appears:

    ```
    Specify minimum number of master-eligible nodes that must be visible in order
    to form an Elasticsearch cluster [1]
    ```

15. Enter the minimum number of master-eligible nodes that must be visible in order to form a cluster.

    For details about how to determine how many master-eligible nodes you need for your cluster, see Creating a DevOps Insight Server Cluster on page 4-36. The master node will be elected from the list of master-eligible nodes.

    For details about master-eligible nodes, see the Node module in the *Elasticsearch Reference*. For details about master elections, see the Zen Discovery module in the *Elasticsearch Reference*.

    Important:
    If you specify `1`, you are asked to confirm this number in the following warning:

    ```
    The minimum number of master eligible nodes is set to 1. This can result
    in data loss in case of network failure in a cluster with two or more
    master eligible nodes.

    Please refer to the ElectricFlow Installation Guide for more details.

    Please confirm if you would like to proceed. [N/y] n
    ```

    To prevent data loss in case of network failure, the minimum number of master-eligible nodes that must be visible in the cluster must be set to a quorum of master-eligible nodes:

    (Number of master-eligible nodes in the cluster / 2) + 1

    For example, in a cluster with three master-eligible nodes, the minimum number of master-eligible nodes should be set to 2.

    The minimum number of master-eligible nodes should be set to 1 only if you intend to run a single-node cluster. For a multi-node cluster, the minimum number of master-eligible nodes must be set to a quorum as described above.

    The following prompt appears:

    ```
    Specify the name of this node in the Elasticsearch cluster [loc-10-lin-ub1604-
    64]
    ```

16. Enter the name of this node in the cluster.

    This serves as a unique identifier and therefore must be a unique name in the cluster.

    The following prompt appears:

    ```
    Is this node the first node to be installed in the Elasticsearch cluster? [n/Y]
    y
    ```

17. If this is the first node that you are adding to the cluster, enter `y`.

    The following prompts appear:

    ```
    The first node will be automatically configured as eligible to be elected as
    the master node.

    The first node will be automatically configured to hold data and perform data
    related operations.

    Installer will automatically create a user with user name "reportuser" to
    connect to Elasticsearch.

    Specify a password for this user []
    ```

18. Enter the password that will be used to access the server. The installer will automatically create a user named `reportuser` with the password that you provide. If you do not specify a password, the installer generates a default password `changeme`.

    The installer asks you to confirm the password that you entered. Enter the same password as before.

    The following prompt appears:

    ```
    Do you want to provide the certificate file containing a CA-signed certificate
    for the ElectricFlow DevOps Insight Server, any intermediate CA certificates
    and a private key? [y/N]
    ```

19. If you want to use your own certificate file, enter `y` and then enter the file path at the `Specify the PKCS#12 certificate file []` prompt, or enter `n`.

    > **Note:** You can enter `n` for a new installation in non-clustered mode or for the first node in clustered mode. In this case, the installer will generate a new self-signed certificate and will use it to sign other TLS certificates.

    The following prompt appears:

    ```
    Specify the directory for data stored by Elasticsearch if the Elasticsearch
    data should be stored in a different location than the DevOps Insight Server
    data directory.
    ```

20. If you want specify a non-default directory to contain the Elasticsearch index data, enter that directory path, or accept the default directory by pressing `Enter`.

    This option is useful if the system lacks enough disk space for the expected growth of data on the volume containing the default data directory.

    The following prompt appears:

    ```
    Do you want to specify the remote ElectricFlow server which will be configured
    to interact with the services being installed? [n/Y]
    ```

21. Enter `y` if you want to automatically configure the remote ElectricFlow server to interact with the services being installed.

    > **Note:** The following prompts related to the configuration of the remote ElectricFlow server are skipped if you declined to configure it automatically.

    The following prompt appears:

    ```
    Specify the host[:port] of the remote ElectricFlow server []
    ```

22. Enter the name of the ElectricFlow server that will communicate with this DevOps Insight server. If the remote server is using a non-default HTTPS port, you must specify the host name as *<host>*:*<port>*. If you do not specify a port, HTTPS port 8443 is assumed (the same as the ElectricFlow server default port).

    The following prompt appears:

    ```
    Specify the user name with which to login to "<remote host>" [admin]
    ```

23. Enter the name of an ElectricFlow user on the ElectricFlow server who has sufficient privileges to edit server settings. This field defaults to the ElectricFlow-supplied `admin` user.

    The following prompt appears:

    ```
    Specify the password for "<remote user>" on "<remote host>" []
    ```

24. Enter the password for the ElectricFlow user. The default password for the `admin` user is `changeme`.

    The following prompt appears:

    ```
    The ElectricFlow DevOps Insight Server will be configured on ElectricFlow
    server version <version> on <remote host>
    ```

    ElectricFlow is installed on the machine. When the installation completes successfully, a message that contains the line `Installation complete` appears.

Configuring DevOps Insight Server Services Autostart for Non-Root/Non-sudo Linux Installations

For non-root/non-`sudo` Linux installations, you must configure autostart for the DevOps Insight services. For instructions, see Configuring Services Autostart for Non-Root/Non-sudo Linux Installations on page 5-11.

Configuring the DevOps Insight Server on the ElectricFlow Server

If you chose to skip the option to configure the remote ElectricFlow server during the installation or upgrade of the DevOps Insight server, you must do so afterward to ensure connectivity and authentication between the DevOps Insight server and the ElectricFlow server. To do this, you use the **Administration** > **DevOps Insight Server** tab in the Automation Platform. For details, see the "DevOps Insight Server Configuration" section in the *ElectricFlow User Guide* at http://docs.electric-cloud.com/eflow_doc/FlowIndex.html.

Checking the DevOps Insight Server Configuration on the ElectricFlow Server

You can confirm the correct DevOps Insight Server settings by entering the following ectool command on the ElectricFlow server:

```
ectool getDevOpsInsightServerConfiguration
```

Following is sample output:

```
<response requestId="1" nodeId="192.168.5.138">
    <devOpsInsightServerConfiguration>
      <devOpsInsightServerConfigurationId>12642169-71c4-11e7-8a08-
0050568f29b0</devOpsInsightServerConfigurationId>
      <createTime>2017-07-26T05:34:19.404Z</createTime>
      <elasticSearchUrl>https://192.168.5.54:9200</elasticSearchUrl>
      <enabled>1</enabled>
      <lastModifiedBy>admin</lastModifiedBy>
      <logStashUrl>https://192.168.5.54:9500</logStashUrl>
      <modifyTime>2017-07-26T05:40:13.458Z</modifyTime>
      <owner>admin</owner>
      <userName>reportuser</userName>
    </devOpsInsightServerConfiguration>
  </response>
```

For details about the `getDevOpsInsightServerConfiguration` options, enter

```
ectool getDevOpsInsightServerConfiguration --help
```

Testing Connectivity and Authentication Between the DevOps Insight Server and the ElectricFlow Server

After you enable connectivity and authentication between the DevOps Insight server and the ElectricFlow server, you can perform a test by using one of the following methods:

- Check the **Test Connection** checkbox in the **Administration** > **DevOps Insight Server** subtab of the Administration Platform web UI on the ElectricFlow server and click **OK**.

- Enter the following ectool command on the ElectricFlow server:

  ```
  ectool setDevOpsInsightServerConfiguration --testConnection 1
  ```

  For details about the `setDevOpsInsightServerConfiguration` options, enter

  ```
  ectool setDevOpsInsightServerConfiguration --help
  ```

  For example, the following response appears if the user name or password is incorrect:

  ```
  ectool error [InvalidCredentials]: HTTP/1.1 401 Unauthorized: Access to
    'https://192.168.5.54:9500' is denied due to invalid credentials.
  ```

  Also, for example, the following response appears if you specify an invalid `elasticSearchUrl` or `logstashUrl`:

  ```
  ectool error [InvalidUrl]: The url 'https://192.168.5.54:9500' is invalid
  ```

  The following example shows the response when a valid `elasticSearchUrl` is used:

  ```
  /opt/electriccloud/electriccommander/bin$ ./ectool
  setDevOpsInsightServerConfiguration
    --elasticSearchUrl https://192.168.5.54:9200 --testConnection 1
  ```

Silent Unattended Installation Method

You can run the ElectricFlow installers and the DevOps Insight installer in unattended (silent) mode with no user interface on either Windows or Linux. Use the arguments in the following list to construct the commands that you need for a specific installation. For example: a server, agent, or web server.

Running a Silent Install

You can silently install the full version of ElectricFlow, agents only, or the DevOps Insight server. *<arguments>* represents optional silent install arguments. For a list of available arguments, see Silent Install Arguments on page 3-113.

Linux

```
./ElectricFlow-<version> --mode silent <--arguments>
```

Linux Agent-Only

"Pseudo" 64-bit : `./ElectricFlowAgent-x64-<version> --mode silent <arguments>`

"Pure" 64-bit : `./ElectricFlowAgent-x64-<version>-new-with-64bit-perl --mode silent <arguments>`

Windows

```
ElectricFlow-<version>.exe --mode silent <--arguments>
```

Windows Agent-Only

32-bit: `ElectricFlowAgent-x86-<version>.exe --mode silent <arguments>`

64-bit: `ElectricFlowAgent-x64-<version>.exe --mode silent <arguments>`

Linux DevOps Insight Server

```
sudo ./ElectricFlowDevOpsInsightServer-<version> --mode silent --unixServerUser
[<user_name>] --unixServerGroup [<group_name>] <--arguments>
```

or

```
sudo ./ElectricFlowDevOpsInsightServer-x64-<version> --mode silent --unixServerUser
[<user_name>] --unixServerGroup [<group_name>] <--arguments>
```

Windows DevOps Insight Server

```
ElectricFlowDevOpsInsightServer-<version>.exe --mode silent --windowsServerUser
[<user_name>] --windowsServerPassword [<password>] --windowsServerDomain [<domain_
name>] <--arguments>
```

or

```
ElectricFlowDevOpsInsightServer-<version>.exe --mode silent --windowsServerLocalSystem
<--arguments>
```

or

```
ElectricFlowDevOpsInsightServer-x64-<version>.exe --mode silent --windowsServerUser
[<user_name>] --windowsServerPassword [<password>] --windowsServerDomain [<domain_
name>] <--arguments>
```

or

```
ElectricFlowDevOpsInsightServer-x64-<version>.exe --mode silent --
windowsServerLocalSystem <--arguments>
```

Silent Install Arguments

The following argument table is an excerpt from the installer help text. You can view the full installer help by entering the `<installer_file> --help` command.

> **Note:** Only limited validity checking is performed on these values during an unattended installation, which means typing errors or other mistakes might cause unexpected issues.

> **Note:** The `response-file` and `save-response-file` arguments are not supported as of versions newer than 6.0.4 and 6.4.

Argument	Description
`--agentAllowRootUser`	(Linux platforms) Lets you specify `root` as the user to own the ElectricFlow agent process when you use the `--unixAgentUser` argument. If you do not use this argument and specify `root` as the agent user, the installation will exit, and the installer log will contain the following error prompt: `It is not recommended to use the 'root' user for running the agent process. Please use the --agentAllowRootUser flag to override this error and proceed with the 'root' user.`
`--agentArtifactCache [<directory>]`	Directory containing cached artifact versions.
`--agentGatewayURL [<URL>]`	URL of the gateway used to communicate with the ElectricFlow server.
`--agentIdleConnectionTimeout [<milliseconds>]`	Idle connection timeout for the ElectricFlow agent (in milliseconds). The default is 0 (the connection does not time out).
`--agentInitMemory [<percent>]`	Initial Java heap size as a percentage of the total system memory for the ElectricFlow agent. This setting has no default and is overridden if you have set `agentInitMemoryMB` to a non-default value.
`--agentInitMemoryMB [<megabytes>]`	Initial Java heap size for the ElectricFlow agent (in MB). The default is 16.
`--agentMaxConnections [<number>]`	Maximum number of network connections for the ElectricFlow agent. The default is 200.
`--agentMaxConnectionsPerRoute [<number>]`	Maximum number of network connections per route for the ElectricFlow agent. The default is 20.
`--agentMaxMemory [<percent>]`	Maximum Java heap size as a percentage of the total system memory. This setting has no default and is overridden if you have set `agentMaxMemoryMB` to a non-default value.
`--agentMaxMemoryMB [<megabytes>]`	Maximum Java heap size for the ElectricFlow agent (in MB). The default is 64.
`--agentLocalPort [<port>]`	Port used by the ElectricFlow agent for HTTP communication on the localhost network interface.
`--agentOutboundConnectTimeout <milliseconds>`	Timeout for the ElectricFlow agent establishing outbound connections (in milliseconds). The default is 30000.
`--agentPort [<port>]`	Port used by the installed ElectricFlow agent for HTTPS communication on any network interface.

Argument	Description
`--agentTLSEnabledProtocol <protocols>`	Comma-delimited list of SSL/TLS protocols that will be allowed for agent connections using HTTPS. The possible values are any combination of `TLSv1`, `TLSv1.1`, `TLSv1.2`, and `SSLv2Hello`. The default security configurations are as follows: • First-time ElectricFlow installations: TLSv1, TLSv1.1, and TLSv1.2 are enabled • Existing ElectricFlow installations: TLSv1, TLSv1.1, TLSv1.2, and SSLv2Hello are enabled To avoid the following warning in the Automation Platform web UI, we recommend removing the `SSL 2.0 Client Hello` or `SSLv2Hello` protocol from your security configurations for all components: **We recommend removing "SSL 2.0 Client Hello format from server configuration and upgrade older agents as indicated on the Cloud / Resources Page to avoid security risk.** To safely remove this protocol, enter the following command on the ElectricFlow server: `$ ecconfigure --serverTLSEnabledProtocol=TLSv1,TLSv1.1,TLSv1.2` When you do this, you would also need to upgrade older agents to the latest version to avoid security risks. You would need to upgrade agents if you are using the following agent versions: • Windows, Linux: 6.0.3 or older; 6.2 or older • Sun Solaris, HP UX, Mac OS: 8.4 or older
`--agentWindowsServiceStartType [<start_type>]`	(Windows only) Start type of the ElectricFlow agent service. Available values are `auto_start` and `delayed_auto_start`.
`--databaseMemoryBufferSize [<size>]`	Size of the database memory buffer. The value can be suffixed with a unit (K, M, or G). The default unit is bytes. The default size is 256 MB.
`--databasePassword [<password>`	]Password used to access the database. The default password is `changeme`.
`--databasePort [<port>]`	Port used by the built-in (default) database. The default port is 8900.

Argument	Description
`--dataDirectory [<directory>]`	Directory used to store configuration files, logs, and database artifacts.
`--force32Bit`	Force a 32-bit install, even if the machine is 64-bit.
`--help`	Display this information.
`--installAgent`	Install the ElectricFlow agent. This option is not required when using the agent-only installers. It is ignored when used.
`--installDatabase`	Install a local built-in database to use with the main ElectricFlow server. This database works only for standard licenses (shipped with ElectricFlow by default) and evaluation licenses. This database is not recommended for production systems.
`--installDirectory [<directory>]`	Directory used to store program files and binaries.
`--installRepository`	Install an ElectricFlow artifact repository server.
`--installServer`	Install the main ElectricFlow server.
`--installWeb`	Install the local web server and ElectricFlow web interface.
`--mode [<installer_mode>]`	Mode in which the installer will run. Available values: `console`, `silent`, or `standard`.
`--nonRoot`	(Linux full installations and Linux agent-only installations) Install as a non-root user or a user without `sudo` privileges. You cannot use this argument when logged in as the root user.
`--remoteServer [<host>:<port>]`	`host:port` for the remote ElectricFlow server. The port is optional and can be omitted if the server is using the default HTTP port.
`--remoteServerCreateRepository`	Create a repository object on the remote ElectricFlow server. You can specify the host name of the repository to create on the remote ElectricFlow server by also using the `--remoteServerRepositoryHostName` option.
`--remoteServerCreateResource`	Create a resource for the installed agent on the remote ElectricFlow server. You can specify the host name of the resource to create on the remote ElectricFlow server by also using the `--remoteServerResourceHostName` option.

Argument	Description
`--remoteServerDiscoverPlugins`	Set the plugins directory for the installed agent and/or web server to the shared plugins directory defined on the remote ElectricFlow server.
`--remoteServerPassword [<password>]`	Password to use when logging in to the remote ElectricFlow server.
`--remoteServerRepository [<repository_name>]`	Name of the repository to create on the remote ElectricFlow server.
`--remoteServerRepositoryHostName [<repository_host_name>]`	Host name of the repository to create on the remote ElectricFlow server. The `--remoteServerRepositoryHostName` option requires that you also specify the `--remoteServerCreateRepository` option.
`--remoteServerResource [<resource_name>]`	Name of the resource to create on the remote ElectricFlow server.
`--remoteServerResourceHostName [<server_resource_name>]`	Host name of the resource to create on the remote ElectricFlow server. The `--remoteServerResourceHostName` option requires that you also specify the `--remoteServerCreateResource` option.
`--remoteServerResourceType [<type>]`	Type of resource to create on the remote ElectricFlow server. This argument is available only when the ElectricFlow server is using a mixed-mode license (concurrent resources and registered hosts). Valid options for this argument are `concurrent` or `registered`.
`--remoteServerUser [<user_name>]`	User name to use when logging in to the remote ElectricFlow server.
`--repositoryPort [<port>]`	Port used by the ElectricFlow artifact repository server (the default is 8200).

Argument	Description
`--repositoryTLSEnabledProtocol <protocols>`	Comma-delimited list of SSL/TLS protocols that will be allowed for ElectricFlow repository server connections using HTTPS. The possible values are any combination of `TLSv1`, `TLSv1.1`, `TLSv1.2`, and `SSLv2Hello`. The default security configurations are as follows: • First-time ElectricFlow installations: TLSv1, TLSv1.1, and TLSv1.2 are enabled • Existing ElectricFlow installations: TLSv1, TLSv1.1, TLSv1.2, and SSLv2Hello are enabled To avoid the following warning in the Automation Platform web UI, we recommend removing the `SSL 2.0 Client Hello` or `SSLv2Hello` protocol from your security configurations for all components: **We recommend removing "SSL 2.0 Client Hello format from server configuration and upgrade older agents as indicated on the Cloud / Resources Page to avoid security risk.** To safely remove this protocol, enter the following command on the ElectricFlow server: `$ ecconfigure --serverTLSEnabledProtocol=TLSv1,TLSv1.1,TLSv1.2` When you do this, you would also need to upgrade older agents to the latest version to avoid security risks. You would need to upgrade agents if you are using the following agent versions: • Windows, Linux: 6.0.3 or older; 6.2 or older • Sun Solaris, HP UX, Mac OS: 8.4 or older
`--serverFileTransferPort [<port>]`	File transfer port used by the installed ElectricFlow server.
`--serverHttpPort [<port>]`	HTTP port used by the installed ElectricFlow server.
`--serverHttpsPort [<port>]`	HTTPS port used by the installed ElectricFlow server.
`--serverServicePrincipalName=name`	The Kerberos Service Principal Name that will be used to authorize users. This command changes the `#wrapper.java.additional.950=-Dsun.security.krb5.principal=*` line in the *DATA_DIR*`/conf/wrapper.conf` file.

Argument	Description
`--serverTLSEnabledProtocol <protocols>`	Comma-delimited list of SSL/TLS protocols that will be allowed for ElectricFlow server connections using HTTPS. The possible values are any combination of `TLSv1`, `TLSv1.1`, `TLSv1.2`, and `SSLv2Hello`. The default security configurations are as follows: • First-time ElectricFlow installations: TLSv1, TLSv1.1, and TLSv1.2 are enabled • Existing ElectricFlow installations: TLSv1, TLSv1.1, TLSv1.2, and SSLv2Hello are enabled To avoid the following warning in the Automation Platform web UI, we recommend removing the `SSL 2.0 Client Hello` or `SSLv2Hello` protocol from your security configurations for all components: **We recommend removing "SSL 2.0 Client Hello format from server configuration and upgrade older agents as indicated on the Cloud / Resources Page to avoid security risk.** To safely remove this protocol, enter the following command on the ElectricFlow server: `$ ecconfigure --serverTLSEnabledProtocol=TLSv1,TLSv1.1,TLSv1.2` When you do this, you would also need to upgrade older agents to the latest version to avoid security risks. You would need to upgrade agents if you are using the following agent versions: • Windows, Linux: 6.0.3 or older; 6.2 or older • Sun Solaris, HP UX, Mac OS: 8.4 or older

Argument	Description
`--skipCheckUserHomeDirectory`	Disables checking for the existence of a valid home directory. This argument overrides the default checks (whether the `HOME` environment variable is defined and points to a writeable directory). Certain ElectricFlow installers allow you to perform installations as a non-root user or a user without `sudo` privileges. To determine whether a particular installer has an option to run in this mode, see Availability of Installers with a Non-Root/Non-sudo or Non-Administrator Mode on page 3-2. The installer writes installation data to the home directory of the user who invoked the installer. The installer will read this data during subsequent upgrades or uninstallations. This argument ensures that the installer finishes successfully for non-root or non-`sudo` installations in which the logged-in user lacks a home directory. **Important:** Do not use this argument for a non-root or non-`sudo` installation if you anticipate a same-system future upgrade (or uninstallation). Instead, you must ensure that you have a home directory before invoking the installer.
`--temp [<directory>]`	Directory used to store temporary files used by the installer.
`--trustedAgent`	Restricts the agent to one ElectricFlow server. The agent will not respond to incoming communication from any other ElectricFlow server. This is useful when you want to create a secure production environment, but generally not needed for test or development systems.
`--unixAgentGroup [<group_name>]`	Group the installed ElectricFlow agent runs as.
`--unixAgentUser [<user_name>]`	User the installed ElectricFlow agent runs as. The user and group that the agent runs as must have permission to write to the `$INSTALL_DIRECTORY/log` directory. If you specify root, you must also use the `--agentAllowRootUser` argument (described above).
`--unixServerGroup [<group_name>]`	Group the installed ElectricFlow, web, or repository server runs as.
`--unixServerUser [<user_name>]`	User the installed ElectricFlow, web, or repository server runs as.

Argument	Description
`--useSameServiceAccount`	Use the same account for server and agent services. Not recommended for production systems.
`--version`	Display installer version information.
`--webDefaultUI=<flow|commander>`	The type of the Commander server user interface that will be used by default. This setting determines whether the Deploy UI or the Automation Platform UI appears when users browse to https://*<ElectricFlow_server>* without appending /flow or /commander respectively to the end of the URL. You can reconfigure this behavior post-installation by using the ecconfigure `--webDefaultUI` option. For details, see the "ecconfigure" section in the "Automation Platform" chapter of the *ElectricFlow User Guide* at http://docs.electric-cloud.com/eflow_doc/FlowIndex.html.
`--webEnableKerberosConstrainedDelegation=<true|false>`	Enable (`true`) or disable (`false`) support for constrained delegation authorization when using the Kerberos SSO protocol. This parameter manages the `KrbConstrainedDelegation` setting in *`DATA_DIR`*`/apache/conf/extra/auth-kerberos.conf`.
`--webEnableKrb5Trace=<1|0>`	Enable (1) or disable (0) additional Kerberos protocol logging for the web server. This parameter manages the `webEnableKrb5Trace` setting in *`DATA_DIR`*`/apache/conf/extra/auth-kerberos.conf`.
`--webEnableSsoKerberos =<true|false>`	Enable (`true`) or disable (`false`) authentication using the Kerberos SSO protocol. This command changes the `$config["ssoEnabledKerberos"]` variable in the *`INSTALL_DIR`*`/apache/htdocs/commander/config.php` file.
`--webHostName [<name>]`	Name users need to type in their browser to access the web server.
`--webHttpPort [<port>]`	HTTP port used by the installed web server.
`--webHttpsPort [<port>]`	HTTPS port used by the installed web server.
`--webServicePrincipalName=[<name>]`	The Kerberos Service Principal Name that will be used to authorize users. This command changes the `KrbServiceName HTTP` setting in the *`DATA_DIR`*`/apache/conf/extra/auth-kerberos.conf` file.
`--windowsAgentDomain [<domain_name>]`	Domain of the account the installed ElectricFlow agent runs as.
`--windowsAgentLocalSystem`	Run the ElectricFlow agent as the local system account.

Argument	Description
`--windowsAgentPassword [<password>]`	Password of the account the installed ElectricFlow agent runs as.
`--windowsAgentUser [<user_name>]`	User name of the account the installed ElectricFlow agent runs as. User that the agent runs as must have permission to write to the `$INSTALL_DIRECTORY/log` directory.
`--windowsServerDomain [<domain_name>]`	Domain of the account the installed ElectricFlow, web, or repository server runs as.
`--windowsServerLocalSystem`	Run the ElectricFlow, web, or repository server as the local system account. **Note:** The Windows local system account cannot access network resources such as shared file systems used for plugins or workspaces. Therefore, do not use this option for a clustered server deployment, which requires a shared file system for plugins. This option is typically used only for installing agents on numerous machines, which would otherwise require that you create a new account on each of those machines.
`--windowsServerPassword [<password>]`	Password of the account the installed ElectricFlow, web, or repository server runs as.
`--windowsServerUser [<user_name>]`	User name of the account the installed ElectricFlow, web, or repository server runs as.
`--windowsSkipAdminCheck`	Do not check that the user running the installer is a direct member of group Administrators.
`--zoneName [<zone_name>]`	Zone name used during remote agent and or remote repository creation.

Linux Silent Installation Examples

The following examples are command strings to use for unattended (silent) installations. In many instances, the command text with the associated options wraps to the following lines.

Important: You must enter the installation command and all options on a single line.

Complete ElectricFlow Installation

This installation installs the ElectricFlow server, including the web, repository, and database servers, one agent, and ElectricFlow tools. Review Before You Install ElectricFlow on page 3-7 before performing this procedure.

Enter:

```
chmod +x ./ElectricFlow-<version>
```

```
./ElectricFlow-<version>
--mode silent
--installServer
--installAgent
--installDatabase
--installWeb
--installRepository
--unixServerUser <server_user>
--unixServerGroup <server_group>
--unixAgentUser <agent_user>
--unixAgentGroup <agent_group>
```

Where:

- *<server_user>* is the user who owns the ElectricFlow server, repository server, and web server processes.
- *<server_group>* is the group who owns the ElectricFlow server, repository server, and web server processes.
- *<agent_user>* is the user who owns the ElectricFlow agent process.
- *<agent_group>* is the group that owns the ElectricFlow agent process.

Agent Installation

This installation example installs an ElectricFlow agent, a repository server, and ElectricFlow tools. Review Before You Install ElectricFlow on page 3-7 before performing this procedure.

Enter:

```
chmod +x ./ElectricFlow-<version>
```

```
./ElectricFlow-<version>
--mode silent
--installRepository
--installAgent
--unixAgentUser <agent_user>
--unixAgentGroup <agent_group>
--unixServerUser <server_user>
--unixServerGroup <server_group>
--remoteServer <existing_ElectricFlow_server>
```

Where:

- *<server_user>* is the user who owns the ElectricFlow server, repository server, and web server processes.
- *<server_group>* is the group who owns the ElectricFlow server, repository server, and web server processes.

- <*agent_user*> is the user who owns the ElectricFlow agent process.
- <*agent_user*> is the group that owns the ElectricFlow agent process.

ElectricFlow Agent Installation (Full Installer)

The ElectricFlow agent software must be installed on each agent machine you intend to use with ElectricFlow. This installation also installs Tools. Review Before You Install ElectricFlow on page 3-7 before performing this procedure.

Enter:

```
chmod +x ./ElectricFlow-<version>
```

```
./ElectricFlow-<version>
--mode silent --installAgent
--unixAgentUser <agent user>
--unixAgentGroup <agent group>
```

Where:

- <*agent_user*> is the user who owns the ElectricFlow agent process.
- <*agent_group*> is the group that owns the ElectricFlow agent process.

ElectricFlow Agent Installation (Agent-Only Installers)

The ElectricFlow agent software must be installed on each agent machine you intend to use with ElectricFlow. Review Before You Install ElectricFlow on page 3-7 before performing this procedure. This example uses the "pseudo" 64-bit agent-only installer file.

Note that this example creates a resource for the installed agent on the remote ElectricFlow server. The `--remoteServerResourceHostName` option requires that you also specify the `--remoteServerCreateResource` option.

- Enter:

  ```
  ./ElectricFlowAgent-x64-<version>
  --mode silent
  --unixAgentUser <agent user>
  --unixAgentGroup <agent group>
  --remoteServerCreateResource
  --remoteServerResourceHostName <server_resource_name>
  ```

  Where:

  - <*agent user*> is the user who owns the ElectricFlow agent process.
  - <*agent group*> is the group that owns the ElectricFlow agent process.
  - <*server_resource_name*> is the host name of the resource to create on the remote ElectricFlow server.

Remote Web Server Installation

A remote web server configuration helps prevent network latency. If you have multiple sites, ElectricFlow can be configured in numerous ways to help you work more efficiently. For details about the architecture for this configuration as well as a discussion of the benefits of using a central web server and web servers at each remote site, see Remote Web Server Configuration on page 1-6.

An agent is required on the machine when you install a web server. For details about why local agents are required on web server machines, see Local Agent Installation Requirement for Web Server Machines on page 3-11.

> **Note:** You should not use these local agents to run jobs.

This installation example installs an ElectricFlow web server, an agent, and ElectricFlow tools. Review Before You Install ElectricFlow on page 3-7 before performing this procedure.

- Enter:

```
chmod +x ./ElectricFlow-<version>
```

```
./ElectricFlow-<version>
--mode silent --installWeb
--installAgent
--unixAgentUser <agent user>
--unixAgentGroup <agent group>
--remoteServer <your existing ElectricFlow server>
--unixServerUser <server user>
--unixServerGroup <server group>
```

  Where:

  - *<server user>* is the user who owns the ElectricFlow server, repository server, and web server processes.
  - *<server group>* is the group who owns the ElectricFlow server, repository server, and web server processes.
  - *<agent user>* is the user who owns the ElectricFlow agent process.
  - *<agent group>* is the group that owns the ElectricFlow agent process.

Repository Server Installation

This installation example installs an ElectricFlow repository server, an agent, and ElectricFlow tools. Review Before You Install ElectricFlow on page 3-7 before performing this procedure.

Enter:

```
chmod +x ./ElectricFlow-<version>
```

```
./ElectricFlow-<version>
--mode silent
--installRepository
--installAgent
--unixAgentUser <agent_user>
--unixAgentGroup <agent_group>
--unixServerUser <server_user>
--unixServerGroup <server_group>
--remoteServer <existing_ElectricFlow_server>
```

Where:

- *<server_user>* is the user who owns the ElectricFlow server, repository server, and web server processes.

- <*server_group*> is the group who owns the ElectricFlow server, repository server, and web server processes.
- <*agent_user*> is the user who owns the ElectricFlow agent process.
- <*agent_user*> is the group that owns the ElectricFlow agent process.

Tools Only Installation

This installation example installs only the ElectricFlow tools. Review Before You Install ElectricFlow on page 3-7 before performing this procedure.

- Enter:

```
chmod +x ./ElectricFlow-<version>

./ElectricFlow-<version> --mode silent
```

Windows Silent Installation Examples

The following examples are command strings to use for unattended (silent) installations. In many instances, the command text with the associated options wraps to the following lines.

> **Important:** You must enter the installation command and all options on a single line.

Complete ElectricFlow Installation

This installation installs the ElectricFlow server, including the web, repository, and database servers, one agent, and ElectricFlow tools. Review Before You Install ElectricFlow on page 3-7 before performing this procedure.

- Enter:

```
ElectricFlow-<version>.exe
--mode silent
--installServer
--installAgent
--installDatabase
--installWeb
--installRepository
--windowsServerUser <server user>
--windowsServerDomain <domain>
--windowsServerPassword <password>
--windowsAgentUser <agent user>
--windowsAgentDomain <domain>
--windowsAgentPassword <password>
```

Where:

- <*server_user*> is the user who owns the ElectricFlow server, repository server, and web server processes.
- <*agent_user*> is the user who owns the ElectricFlow agent process.

Agent Installation

This installation example installs an ElectricFlow agent, a repository server, and ElectricFlow tools. Review Before You Install ElectricFlow on page 3-7 before performing this procedure.

- Enter:

```
ElectricFlow-<version>.exe
--mode silent
--installRepository
--installAgent
--windowsAgentUser <agent_user>
--windowsAgentDomain <domain>
--windowsAgentPassword <password>
--windowsServerUser <server_user>
--windowsServerDomain <domain>
--windowsServerPassword <password>
--remoteServer <existing_ElectricFlow_server>
```

  Where:

  - <server_user> is the user who owns the ElectricFlow server, repository server, and web server processes.
  - <agent_user> is the user who owns the ElectricFlow agent process.

ElectricFlow Agent Installation (Full Installer)

The ElectricFlow agent software must be installed on each agent machine you intend to use with ElectricFlow. This installation also installs Tools. Review Before You Install ElectricFlow on page 3-7 before performing this procedure.

- Enter:

```
ElectricFlow-<version>.exe --mode silent --installAgent --windowsAgentUser
<agent_user> --windowsAgentDomain <domain> --windowsAgentPassword <password>
```

  Where <agent_user> is the user who owns the ElectricFlow agent process.

ElectricFlow Agent Installation (Agent-Only Installer)

The ElectricFlow agent software must be installed on each agent machine you intend to use with ElectricFlow. Review Before You Install ElectricFlow on page 3-7 before performing this procedure. This example uses the "pseudo" 64-bit agent-only installer file.

Note that this example creates a resource for the installed agent on the remote ElectricFlow server. The `--remoteServerResourceHostName` option requires that you also specify the `--remoteServerCreateResource` option.

- Enter:

```
ElectricFlowAgent-x64-<version>.exe --mode silent --windowsAgentUser <agent
user> --windowsAgentDomain <domain> --windowsAgentPassword <password> --
remoteServerCreateResource --remoteServerResourceHostName <server_resource_name>
```

  Where:

- <agent_user> is the user who owns the ElectricFlow agent process.
- <server_resource_name> is the host name of the resource to create on the remote ElectricFlow server.

Remote Web Server Installation

A remote web server configuration helps prevent network latency. If you have multiple sites, ElectricFlow can be configured in numerous ways to help you work more efficiently. For details about the architecture for this configuration as well as a discussion of the benefits of using a central web server and web servers at each remote site, see Remote Web Server Configuration on page 1-6.

An agent is required on the machine when you install a web server. For details about why local agents are required on web server machines, see Local Agent Installation Requirement for Web Server Machines on page 3-11.

> **Note:** You should not use these local agents to run jobs.

This installation example installs an ElectricFlow web server, an agent, and ElectricFlow tools. Review Before You Install ElectricFlow on page 3-7 before performing this procedure.

- Enter:

```
ElectricFlow-<version>.exe
--mode silent
--installAgent
--installWeb
--windowsAgentUser <agent_user>
--windowsAgentDomain <domain>
--windowsAgentPassword <password>
--remoteServer <existing_ElectricFlow_server>
--windowsServerUser <server_user>
--windowsServerDomain <domain>
--windowsServerPassword <password>
```

  Where:

  - `<server user>` is the user who owns the ElectricFlow server, repository server, and web server processes.
  - `<agent user>` is the user who owns the ElectricFlow agent process.

Repository Server Installation

This installation example installs an ElectricFlow repository server, an agent, and ElectricFlow tools. Review Before You Install ElectricFlow on page 3-7 before performing this procedure.

- Enter:

```
ElectricFlow-<version>.exe
--mode silent
--installRepository
--installAgent
--windowsAgentUser <agent_user>
--windowsAgentDomain <domain>
--windowsAgentPassword <password>
--windowsServerUser <server_user>
--windowsServerDomain <domain>
--windowsServerPassword <password>
--remoteServer <existing_ElectricFlow_server>
```

  Where:

- <*server_user*> is the user who owns the ElectricFlow server, repository server, and web server processes.
- <*agent_user*> is the user who owns the ElectricFlow agent process.

Tools Only Installation

This installation example installs only the ElectricFlow tools. Review Before You Install ElectricFlow on page 3-7 before performing this procedure.

- Enter:

```
ElectricFlow-<version>.exe --mode silent
```

Linux Repository Server Silent Installation

This installation example installs an ElectricFlow repository server, an agent, and ElectricFlow tools. Review Before You Install ElectricFlow on page 3-7 before performing this procedure.

Enter:

```
chmod +x ./ElectricFlow-<version>
```

```
./ElectricFlow-<version>
--mode silent
--installRepository
--installAgent
--unixAgentUser <agent_user>
--unixAgentGroup <agent_group>
--unixServerUser <server_user>
--unixServerGroup <server_group>
--remoteServer <existing_ElectricFlow_server>
```

Where:

- <*server_user*> is the user who owns the ElectricFlow server, repository server, and web server processes.
- <*server_group*> is the group who owns the ElectricFlow server, repository server, and web server processes.
- <*agent_user*> is the user who owns the ElectricFlow agent process.
- <*agent_user*> is the group that owns the ElectricFlow agent process.

Windows Repository Server Silent Installation Example

This installation example installs an ElectricFlow repository server, an agent, and ElectricFlow tools. Review Before You Install ElectricFlow on page 3-7 before performing this procedure.

- Enter:

```
ElectricFlow-<version>.exe
--mode silent
--installRepository
--installAgent
--windowsAgentUser <agent_user>
--windowsAgentDomain <domain>
```

```
--windowsAgentPassword <password>
--windowsServerUser <server_user>
--windowsServerDomain <domain>
--windowsServerPassword <password>
--remoteServer <existing_ElectricFlow_server>
```

Where:

- *<server_user>* is the user who owns the ElectricFlow server, repository server, and web server processes.
- *<agent_user>* is the user who owns the ElectricFlow agent process.

Windows or Linux DevOps Insight Server Silent Unattended Installation Example

You can run the ElectricFlow DevOps Insight server installer in unattended (silent) mode with no user interface on either Windows or Linux. This section includes information for adding a system to a DevOps Insight cluster during installation.

For details about the overall steps for installing DevOps Insight on a group of servers to create a DevOps Insight server cluster, see Creating a DevOps Insight Server Cluster on page 4-36.

Installing the DevOps Insight Server on a System with Other ElectricFlow Components

For a production environment, Electric Cloud recommends that you install the DevOps Insight server on a system other than systems running other ElectricFlow components (such as the ElectricFlow server, web server, repository server, or agent). If you must install it on the same system (such as for testing or other non-production or trial-basis situations) see Running the DevOps Insight Server on a System with Other ElectricFlow Components on page 3-8 for details.

Silent Installation Arguments (Single Server and Cluster Mode)

The following argument table is an excerpt from the installer help text. You can view the full installer help by entering `ElectricFlowDevOpsInsightServer-x64-<version> --help`.

Argument	Description
`--dataDirectory <arguments>`	Directory used to store binaries Default value on Linux: `/opt/electriccloud/electriccommander`. Default value on Windows: `%ProgramFiles%\Electric Cloud\ElectricCommander` (usually: `C:\Program Files\Electric Cloud\ElectricCommander`)

Argument	Description
`--disableSSL`	Do not configure the Elasticsearch service to use SSL connections and authentication **Note:** Using this option is not recommended in production environments. The DevOps Insight server uses the Elasticsearch search engine and the Logstash data-collection and log-parsing engine to gather data from the ElectricFlow server for use in the Deployments, Releases, and Release Command Center dashboards.
`--elasticsearchDataDirectory <directory>`	Path to the directory for data stored by Elasticsearch • New installations—The DevOps Insight server data directory will be used by default. • Upgrades—The existing directory will continue to be used by default.
`--elasticsearchIndexNumberOfShards <arguments>`	Number of primary shards that an index should have. Default value: 2
`--elasticsearchInternalPort <arguments>`	Port number used for internal communication between nodes within the Elasticsearch cluster. Default value: 9300
`--elasticsearchMemoryMB <arguments>`	Heap size in MB for the Elasticsearch service. Default value: 2048
`--elasticsearchPort <arguments>`	Port number to be used by the Elasticsearch service. Default value: 9200
`--elasticsearchRegenerateCertificates`	During the update, regenerate the certificates that are used by the Elasticsearch service
`--elasticsearchUserPassword <arguments>`	Password to use for regular access to the DevOps Insight server services. The installer automatically creates a user with the user name `reportuser` to connect to Elasticsearch. This parameter lets you specify a password for this user. Electric Cloud recommends that you change the default password **Note:** If you do specify a password, the installer will generate a default password `changeme`.
`--help`	Display the information in this table

Argument	Description
`--hostName <arguments>`	Host name or IP address to be used by the remote ElectricFlow server to communicate with the DevOps Insight server. The default value is the current host name of the machine.
`--installDirectory <arguments>`	Directory used to store binaries Default value on Linux: `/opt/electriccloud/electriccommander` Default value on Windows: `%ProgramFiles%\Electric Cloud\ElectricCommander` (usually: `C:\Program Files\Electric Cloud\ElectricCommander`)
`--logstashInitMemoryMB <arguments>`	Initial java heap size in MB for the Logstash service. Default value: 256 The DevOps Insight server uses the Elasticsearch search engine and the Logstash data-collection and log-parsing engine to gather data from the ElectricFlow server for use in the Deployments, Releases, and Release Command Center dashboards.
`--logstashInternalPort <arguments>`	Port number used by the Logstash monitoring APIs that provide runtime metrics about Logstash. Default value: 9600
`--logstashMaxMemoryMB <arguments>`	Maximum Java heap size in MB for the Logstash service. Default value: 1024
`--logstashPort <arguments>`	Internal port number to be used by the Logstash service. Default value: 9500
`--mode <arguments>`	Set the installer mode Available values on Linux: `console`, `silent` or `standard` Available values on Windows: `silent` or `standard`
`--nonRoot`	(Linux installations) Install as a non-root user or a user without `sudo` privileges. You cannot use this argument when logged in as the root user.
`--remoteServer <arguments>`	*<host>*`[:`*<port>*`]` of the remote ElectricFlow server
`--remoteServerPassword <arguments>`	Password to use when logging in to the remote ElectricFlow server
`--remoteServerUser <arguments>`	User name to use when logging in to the remote ElectricFlow server
`--temp <arguments>`	Set the temporary directory used by this program

Argument	Description
`--unixServerGroup <arguments>`	(Linux only) Group name that the ElectricFlow DevOps Insight server services run as **Note:** This is required for silent installation on Linux.
`--unixServerUser <arguments>`	(Linux only) User name that the ElectricFlow DevOps Insight server services run as **Note:** This is required for silent installation on Linux.
`--version`	Display installer version information
`--windowsServerDomain <arguments>`	(Windows only) Domain of the account the ElectricFlow DevOps Insight server services will run as on Windows **Note:** Do not use this parameter if a local account used.
`--windowsServerLocalSystem`	(Windows only) Run the ElectricFlow DevOps Insight server services as the local system account
`--windowsServerPassword <arguments>`	(Windows only) Password of the account that the ElectricFlow DevOps Insight server services will run as **Note:** This is required for silent installation on Windows if the `--windowsServerLocalSystem` option is not specified.
`--windowsServerUser <arguments>`	User name of the account the ElectricFlow DevOps Insight server services will run as on Windows **Note:** This is required for silent installation on Windows if the `--windowsServerLocalSystem` option is not specified.

Additional Silent Installation Arguments for DevOps Insight Server Cluster Mode

The following argument table is an excerpt from the installer help text. You can view the full installer help by entering `ElectricFlowDevOpsInsightServer-x64-<version> --help`.

Argument	Description
`--elasticsearchClusterDiscoveryHosts <list>`	Comma-delimited list of other nodes in the Elasticsearch cluster that are likely to be live and reachable. The default is `127.0.0.1,[::1]`.
`--elasticsearchClusterMinimumMasterNodes <number>`	Minimum number of master-eligible nodes that must be visible in order to form an Elasticsearch cluster. The default is `1`.
`--elasticsearchClusterName <name>`	The name of the Elasticsearch cluster. The default is `elasticsearch`.
`--elasticsearchNodeAdditional`	This node is not the first node to be installed in the Elasticsearch cluster.
`--elasticsearchNodeData <boolean>`	This node holds data and performs data -related operations such as CRUD, search, and aggregations. The default is `true`.
`-elasticsearchNodeIngest <boolean>`	This node is able to apply an ingest pipeline to a document in order to transform and enrich the document before indexing. The default is `true`.
`--elasticsearchNodeMaster <boolean>`	This node is eligible to be elected as the master node, which controls the Elasticsearch cluster. The default is `true`.
`--elasticsearchNodeName <name>`	The name of this node in the Elasticsearch cluster. The default is the actual hostname.
`--elasticsearchPublishHost <host>`	The single interface that the Elasticsearch node advertises to other nodes in the cluster, so that those nodes can connect to it. The default is the value set by the `--hostName` argument.
`--elasticsearchCACertificateFile <file>`	The PKCS#12 file containing a CA-signed certificate for the ElectricFlow DevOps Insight Server, any intermediate CA certificates and a private key. **Note:** You can omit this option for a new installation in non-clustered mode or for the first node in clustered mode. In this case, the installer will generate a new self-signed certificate and will use it to sign other TLS certificates.

Installing the DevOps Insight Server

Enter one of the following commands from a command line.

- Linux: `sudo ./ElectricFlowDevOpsInsightServer-x64-<version> --mode silent <arguments>`
- Windows: `ElectricFlowDevOpsInsightServer-x64-<version>.exe --mode silent <arguments>`

where:

- *<version>* is your ElectricFlow DevOps Insight server version number.
- *<arguments>* represents any additional silent install arguments.

For a successful installation in this mode, you must specify the following:

- Linux: Use the `--unixServerUser` and `--unixServerGroup` options to specify the user name and group that the ElectricFlow DevOps Insight server service runs as.
- Windows: Use the `--windowsServerUser`, `--windowsServerPassword`, and `--WindowsServerDomain` options to specify the user name, password, and domain (if the account is not local) of the account the ElectricFlow DevOps Insight server service runs as or the `--windowsServerLocalSystem` option to use the local system account.

Configuring DevOps Insight Server Services Autostart for Non-Root/Non-sudo Linux Installations

For non-root/non-`sudo` Linux installations, you must configure autostart for the DevOps Insight services. For instructions, see Configuring Services Autostart for Non-Root/Non-sudo Linux Installations on page 5-11.

Configuring the DevOps Insight Server on the ElectricFlow Server

If you chose to skip the option to configure the remote ElectricFlow server during the installation or upgrade of the DevOps Insight server, you must do so afterward to ensure connectivity and authentication between the DevOps Insight server and the ElectricFlow server. To do this, you use the **Administration** > **DevOps Insight Server** tab in the Automation Platform. For details, see the "DevOps Insight Server Configuration" section in the *ElectricFlow User Guide* at http://docs.electric-cloud.com/eflow_doc/FlowIndex.html.

Checking the DevOps Insight Server Configuration on the ElectricFlow Server

You can confirm the correct DevOps Insight Server settings by entering the following ectool command on the ElectricFlow server:

```
ectool getDevOpsInsightServerConfiguration
```

Following is sample output:

```
<response requestId="1" nodeId="192.168.5.138">
    <devOpsInsightServerConfiguration>
      <devOpsInsightServerConfigurationId>12642169-71c4-11e7-8a08-
0050568f29b0</devOpsInsightServerConfigurationId>
      <createTime>2017-07-26T05:34:19.404Z</createTime>
      <elasticSearchUrl>https://192.168.5.54:9200</elasticSearchUrl>
      <enabled>1</enabled>
      <lastModifiedBy>admin</lastModifiedBy>
      <logStashUrl>https://192.168.5.54:9500</logStashUrl>
      <modifyTime>2017-07-26T05:40:13.458Z</modifyTime>
      <owner>admin</owner>
```

```
      <userName>reportuser</userName>
    </devOpsInsightServerConfiguration>
  </response>
```

For details about the `getDevOpsInsightServerConfiguration` options, enter

```
ectool getDevOpsInsightServerConfiguration --help
```

Testing Connectivity and Authentication Between the DevOps Insight Server and the ElectricFlow Server

After you enable connectivity and authentication between the DevOps Insight server and the ElectricFlow server, you can perform a test by using one of the following methods:

- Check the **Test Connection** checkbox in the **Administration** > **DevOps Insight Server** subtab of the Administration Platform web UI on the ElectricFlow server and click **OK**.
- Enter the following ectool command on the ElectricFlow server:

  ```
  ectool setDevOpsInsightServerConfiguration --testConnection 1
  ```

  For details about the `setDevOpsInsightServerConfiguration` options, enter

  ```
  ectool setDevOpsInsightServerConfiguration --help
  ```

  For example, the following response appears if the user name or password is incorrect:

  ```
  ectool error [InvalidCredentials]: HTTP/1.1 401 Unauthorized: Access to
    'https://192.168.5.54:9500' is denied due to invalid credentials.
  ```

  Also, for example, the following response appears if you specify an invalid `elasticSearchUrl` or `logstashUrl`:

  ```
  ectool error [InvalidUrl]: The url 'https://192.168.5.54:9500' is invalid
  ```

  The following example shows the response when a valid `elasticSearchUrl` is used:

  ```
  /opt/electriccloud/electriccommander/bin$ ./ectool
  setDevOpsInsightServerConfiguration
    --elasticSearchUrl https://192.168.5.54:9200 --testConnection 1
  ```

Non-Server Platform Installation Method for UNIX Agents

To install agents and tools on UNIX machines that are not supported ElectricFlow server platforms, you must use a UNIX installer file instead of the `./ElectricFlow-<version>` installer file (which works only for server installation). This file is named `commander_<OStype>.bin` and is available on the Electric Cloud FTP site. For more information about supported agent platforms, see Supported Agent Platforms on page 2-2.

Interactive Command-Line Installation Method for UNIX or macOS Agents

The agent software must be installed on each machine you intend to use with ElectricFlow. An agent is an ElectricFlow component that runs on a machine resource. The agent executes ElectricFlow job steps, monitors step progress, and records job completion information.

This section describes how to install agents and tools on UNIX (not Linux or Windows) machines. These include Solaris, HP-UX, macOS, and AIX machines. Agent upgrades are not supported on these platforms.

You can install agents using any of the following accounts:

- root
- Any account with sudo privileges
- (UNIX or macOS only) Any non-root account without sudo privileges

Installing Agents Using root or an Account with sudo Privileges

To install agents and tools on UNIX or macOS machines using root or an account with sudo privileges:

1. Obtain the UNIX or macOS installer file for your agent platform as described in Non-Server Platform Installation Method for UNIX Agents on page 14-30.
2. Log in as root.
3. Enter `chmod +x ./commander_<OStype>.bin` to ensure that the installer is executable.

   where *<OStype>* is the agent platform. For example:

   ```
   chmod +x ./commander_powerpc_AIX71.bin
   ```

4. Run `./commander_<OStype>.bin`.

   The following prompts appear:

   ```
   Checking installer integrity, please wait...
   ElectricFlow 7.2.0.116649 for AIX Installer
   Copyright 2006-2018 Electric Cloud, Inc. All rights reserved.


   Press CTRL-C to exit at any time.


   Press Enter to accept default settings.


   log file: /tmp/commander_install_20170321_115947.log

   This suite installer can install several different product options.

   Note: The default is to install everything.

   Which products would you like to install (agent, tools):
   ```

5. Enter `agent` or press `Enter`.

   (You can also install the tools only by entering `Tools`.) The agent and tools will be installed. The following prompts appear:

   ```
   Installing agent and tools.


   Where would you like the software to be installed?

   NOTE: The destination should NOT be an nfs filesystem.
   ```

```
Enter destination directory (default is /opt):
```

6. Enter the destination directory path.

   The following prompt appears:

   ```
   Enter an existing user to own installed agent files and run agent processes:
   ```

7. Enter the name of the user to own the ElectricFlow agent files and run the agent processes.

   The following prompt appears:

   ```
   Enter an existing user group to own installed agent files and run agent
   processes.
   Or hit Enter to choose the primary group (default is '<primary group>'):
   ```

8. Enter the group name of the user to own the ElectricFlow agent files and run the agent processes or press `Enter` to use the user's primary group.

   The following prompt appears:

   ```
   Enter the agent port (default is 7800):
   ```

9. Accept the default port or specify a different port if needed to eliminate conflicts with your existing system configuration, and then press `Enter`.

   The installer extracts and installs the software. When the installation is complete, the following prompt appears:

   ```
   OK: Installation successful!
   ```

Installing Agents Using a Non-root Account or an Account Without sudo Privileges

In this type of installation, the installer starts the agent service and runs it as the user that performed the installation.

> **Important:** Running the installer without root or sudo privileges is not recommended. When run without root or sudo privileges, the installer cannot install the files that provide automatic start for the agent services, so you must configure automatic restart manually.

To install agents and tools on UNIX or macOS machines using a non-root account without sudo privileges:

1. Log in as the user to own the installed agent files and run the agent processes.
2. Obtain the UNIX or macOS installer file for your agent platform as described in Non-Server Platform Installation Method for UNIX Agents on page 14-30.
3. Run `chmod +x ./commander_<OStype>.bin` to ensure that the installer is executable.

   <OStype> is the agent platform. For example:

   ```
   chmod +x ./commander_powerpc_AIX71.bin
   ```

4. Enter `./commander_<OStype>.bin --nonRoot` to start the installation.

   The following prompts appear:

```
Checking installer integrity, please wait...
ElectricFlow 7.2.0.116649 for AIX Installer
Copyright 2006-2018 Electric Cloud, Inc. All rights reserved.


Press CTRL-C to exit at any time.


Press Enter to accept default settings.


log file: /tmp/commander_install_20170321_115947.log
This suite installer can install several different product options.
Note: The default is to install everything.
Which products would you like to install (agent, tools):
```

   Note:
   Failure to include the `--nonRoot` argument causes the following error:

```
This installer must be invoked in a root context.


ERROR: Install failed. Exiting installer.
```

5. Enter `agent` or press `Enter`.

   (You can also install the tools only by entering `Tools`.) The agent and tools will be installed. The following prompts appear:

```
Installing agent and tools.


Where would you like the software to be installed?
NOTE: The destination should NOT be an nfs filesystem.


Enter destination directory (default is /opt):
```

6. Enter the destination directory path.

   Note:
   If you lack sufficient privileges on the destination directory, the following error appears, and you must obtain sufficient privileges before continuing:

```
Could not create "/bin/electriccloud/electriccommander".
```

   If the directory that you entered already exists, the following prompts appear:

```
Directory "/opt/electriccloud/electriccommander" already exists.

Do you want to delete and overwrite it (Y/n)?
```

7. If the directory already exists, enter `Y` to overwrite it.

   The following prompts appear:

   ```
   Non-root install mode. Current user 'build' will be used as owner for installed
   agent files and run agent processes.

   Enter an existing user group to own installed agent files and run agent
   processes.
   Or hit Enter to choose the primary group (default is '<primary group>'):
   ```

8. Enter the group name of the user to own the ElectricFlow agent files and run the agent processes or press `Enter` to use the user's primary group.

   The group that the agent runs as must have permission to write to the `$INSTALL_DIRECTORY/log` directory.

   Note:
   If you are not a member of the group, the following prompt appears, and you must enter a different group:

   ```
   The combination of agent user 'build' and agent group 'foo' is invalid.
   Please try again.

   Enter an existing user group to own installed agent files and run agent
   processes.
   Or hit Enter to choose the primary group (default is '<primary group>'):
   ```

   After you successfully enter the group name, the following prompt appears:

   ```
   Enter the agent port (default is 7800):
   ```

9. Accept the default port or specify a different port if needed to eliminate conflicts with your existing system configuration, and then press `Enter`.

   The installer extracts and installs the software. Then the following prompts appear. Note that the directory to contain the agent services varies by platform:

```
Please wait while the services are configured and started...

Services are started automatically during configuration.

To manually start services use following command(s):
/opt/electriccloud/electriccommander/startup/ecmdrAgent start

To start services at system startup,
copy files at /opt/electriccloud/electriccommander/startup
to the init.d directory '/etc/rc.d/init.d'
and make corresponding links in /etc/rcX.d directories.
```

   When the installation is complete, the following prompt appears:

```
OK: Installation successful!
```

Unattended (Silent) Installation Method for UNIX or macOS Agents

The agent software must be installed on each machine you intend to use with ElectricFlow. An agent is an ElectricFlow component that runs on a machine resource. The agent executes ElectricFlow job steps, monitors step progress, and records job completion information.

This section describes how to install agents and tools silently on UNIX (not Linux or Windows) machines. These include Solaris, HP-UX, macOS, and AIX machines. Agent upgrades are not supported on these platforms.

You can install agents using any of the following accounts:

- root
- Any account with sudo privileges
- (UNIX or macOS only) Any non-root account without sudo privileges

Silent Installation Command Arguments

The following table lists the available arguments.

Argument	Description
`-q`	Runs the installer in silent mode. The default installation options are used unless you override them on the command line or in an installation configuration file.
`--nonRoot`	(UNIX or macOS only) Runs the installer using a non-root account without sudo privileges. The agent service will run as the user that performed the installation.

Argument	Description
`-f`	Removes and replaces any existing files in the destination directory. This argument completely removes the directory but does *not* uninstall the previous version. For details about upgrades, see Roadmap for Upgrading ElectricFlow on page 6-1.
`--config`	Specifies a file containing installation parameters and values.

Running a Silent Installation

Important: Running the installer without root or sudo privileges is not recommended. When run without root or sudo privileges, the installer cannot install the files that provide automatic start for the agent services, so you must configure automatic restart manually.

To run a silent UNIX or macOS agent installation:

1. Obtain the UNIX or macOS installer file for your agent platform as described in Non-Server Platform Installation Method for UNIX Agents on page 14-30.
2. If you are *not* installing as a non-root user without sudo privileges, log in as root or as a user with sudo privileges.
3. Run `chmod +x ./commander_<OStype>.bin` to ensure that the installer is executable.
4. Run `commander_<OStype>.bin -q <arguments>`.

   where <*OStype*> is the agent platform. For example:

   ```
   commander_powerpc_AIX71.bin -q -f --config myconfig
   ```

   For installation using a non-root account without sudo privileges, you must include the `--nonRoot` argument. Failure to do so causes the following error:

   ```
   This installer must be invoked in a root context.

   ERROR: Install failed. Exiting installer.
   ```

Example Parameters in an Installation Configuration File

Following is an example of parameters in a configuration file for silent installation of agents using root or an account with sudo privileges:

```
EC_INSTALL_TYPE=agent
DESTINATION_DIR="/opt"
AGENT_USER_TO_RUN_AS="bill jones"
AGENT_GROUP_TO_RUN_AS=engineering
EC_AGENT_PORT=7800
EC_AGENT_LOCAL_PORT=6800
```

Following is an example of parameters in a configuration file for silent installation of tools using root or an account with sudo privileges:

```
EC_INSTALL_TYPE=tools
DESTINATION_DIR="/opt"
USER_TO_RUN_AS=sally
GROUP_TO_RUN_AS=engineering
```

Installing or Upgrading Remote Agents

You can install or upgrade remote hosts that are running Linux (x86 or x64), Windows (x86 or x64), macOS, Solaris, Solaris x86, or HP-UX. The installation or upgrade processes take advantage of the underlying ElectricFlow Centralized Agent Management (CAM) feature, which significantly simplifies distribution of new agents and reduces the administrative cost of upgrading ElectricFlow to newer versions.

Prerequisites

General Prerequisites

- You must have an artifact repository server installed.
- At least one version of each agent installer must be published to the artifact repository for any required OS.
- The user that you specify in the **Authentication Options** dialog box must have administrator privileges on the target machines.
- The agent service user of the driving agent must have administrator rights on the target machine.

Linux and UNIX Prerequisites

- The target machines must be running the SSH daemon.
- For SSH, the user account on the target machine must have passwordless sudo configured for running the installer with root privileges.

  For example, in the `/etc/sudoers` file, you must add "`<username> ALL=(ALL) NOPASSWD:ALL`".
- For each target machine that you want to upgrade, ensure that `PasswordAuthentication=yes` in the `/etc/ssh/sshd_config` file. For example:

  ```
  # To disable tunneled clear text passwords, change to no here!
  PasswordAuthentication yes
  ```

  On SUSE platforms, `PasswordAuthentication=no` is the default.

Windows Prerequisites

Windows Driving Resources

- They must be in the same Windows domain as the remote windows hosts where you will install the agents.
- They must be in the same zone as the zone where you will install the agents.
- Powershell 3.0 or newer must be installed on older versions of Windows. To check the Powershell version:

  1. Open Powershell on the host by selecting **Start** >**All Programs** > **Accessories** > **Windows PowerShell** > **Windows PowerShell**.

  2. Enter the `Get-Host | Select-Object Version` command.
- Windows Remote Management (WinRM) must be installed and configured.
  - WinRM is installed automatically with all currently-supported versions of Windows.

- The WinRM service starts automatically on Windows Server. On Windows Vista, the service must be started manually.
- Before using a domain user to install agents on remote hosts, you must enable multi-hop support for WinRM on the driving resource system and on each remote host. For details, see the *Enabling Multi-Hop Support for Windows Remote Management Before Installing or Upgrading Remote Agents* Knowledge Base article.
- You can enable the WS-Management protocol on the local system and set up the default configuration for remote management by using the `Winrm quickconfig` command.

- Maximum Powershell memory must be set to at least 1024 MB. To set the memory setting to 1024 MB:

  1. Open a Windows command prompt as Administrator.
  2. Enter the `winrm set winrm/config/winrs @{MaxMemoryPerShellMB="1024"}` command.

Remote Windows Hosts

- Maximum Powershell memory (MaxMemoryPerShellMB) must be set to at least 1024 MB.
- WinRM must be installed and configured.
  - WinRM is installed automatically with all currently-supported versions of Windows.
  - The WinRM service starts automatically on Windows Server. On Windows Vista, the service must be started manually.

You can enable the WS-Management protocol on the local computer and set up the default configuration for remote management by using the `winrm quickconfig` command.

Permissions for Installing or Upgrading Remote Agents

The following table describes the user permissions required to install or upgrade remote agents. For information about modifying permissions, see the "Access Control" section in the "Automation Platform" chapter of the *ElectricFlow User Guide* at http://docs.electric-cloud.com/eflow_doc/FlowIndex.html.

Allowed action	Object	Required privilege ("allow")
Install a resource in a zone	The "Resources" system object	Read
		Modify
		Execute
	Zone (such as `US1`)	Read
		Modify

Allowed action	Object	Required privilege ("allow")
Upgrade a resource	Resource (such as `ResourceA`)	Read
		Modify
		Execute

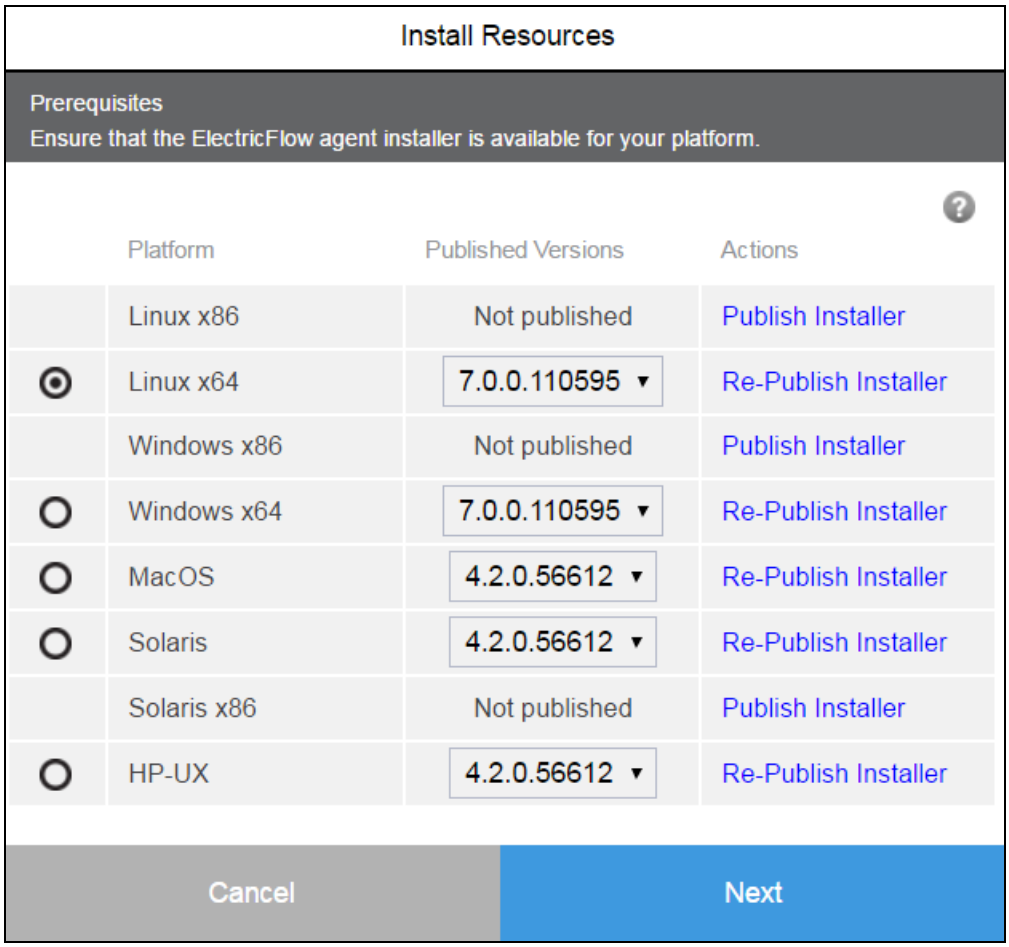
Installing Remote Agents Using the Web Interface

This section describes how to use the ElectricFlow web UI to install remote agents. This process uses the underlying ElectricFlow Centralized Agent Management (CAM) functionality.

In the **Cloud** > **Resources** page of the Automation Platform, click the (add) menu and then click **Install Resource(s)** to install resources on host machines to use as agent hosts for your ElectricFlow resources. If the **Install Resource(s)** menu option is not visible, you must log out and then log in as a user with the required permissions. For details about required permissions, see Permissions for Installing or Upgrading Remote Agents on page 3-144

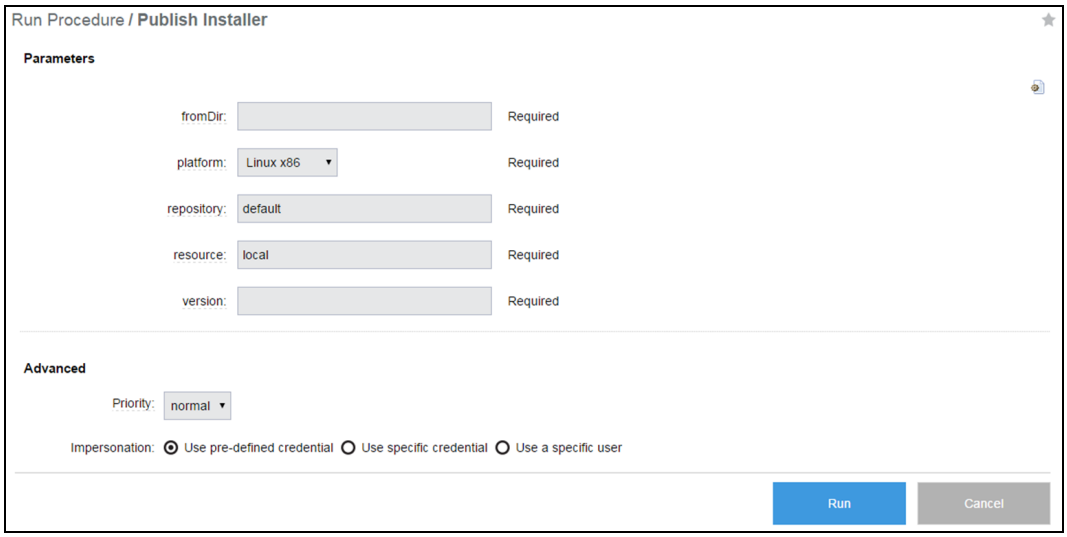
> **Note:** On Solaris platforms, the installation directory path must have fewer than 70 characters.

The **Install Resource(s)** dialog box appears:



If an ElectricFlow agent installer is already published and available for your desired platform, then you can select it to continue with the installation. Otherwise, click the appropriate **Publish Installer** or **Re-Publish Installer** link for the appropriate platform, then click **Next**.

The **Run Procedure / Publish Installer** page opens:



Enter the following information:

Parameters

- **fromDir**—Directory on the resource used for publishing. For example, enter `/var/tmp` for Linux or `C:/<temp>` for Windows.
- **platform**—Host platform type.
- **repository**—Repository name. Use **default** to use the repository installed during the ElectricFlow installation or enter another name.
- **resource**—Name of the resource used for publishing.
- **version**—Build version for the ElectricFlow installation. For example, enter `5.0.0.56390`. You can find the build version from the file name of the ElectricFlow installer.

(Optional) Advanced

Set advanced options if needed.

Click **Run**.

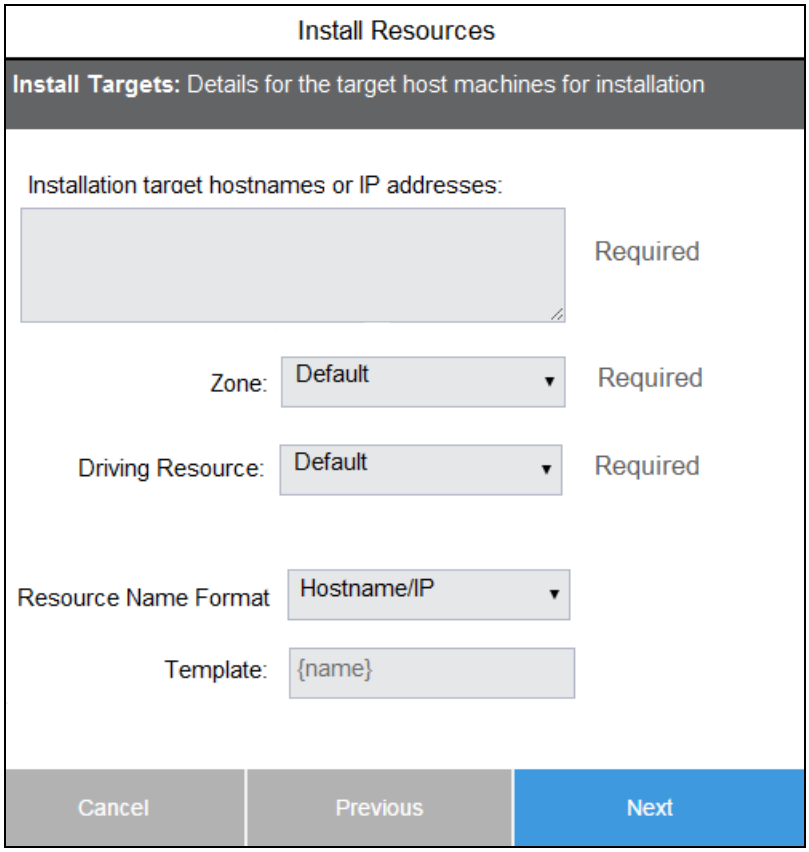
When the Publish Installer procedure runs, you can see the job status on the **Job Details** page.

If you want to see the published installers, click the **Artifacts** tab and then click the `com.electriccloud:installer` artifact. The **Artifact Details** page lists the published installers.

> **Note:** You can republish a new agent installer version. To do so, return to the **Cloud** > **Resources** page and choose + (add) > **Install Resource(s)** and click the radio button for the installer.

If the **Install Resource(s)** menu option is not visible, you must log out and then log in as a user with the required permissions. For details about required permissions, see Permissions for Installing or Upgrading Remote Agents on page 3-144

The **Install Targets** dialog box appears:



Enter the information for the target host machines as follows:

- (Required) **Installation target hostnames or IP addresses**—One or more host names or IP addresses separated by any combination of spaces, commas, semicolons, or newlines. Host names cannot contain spaces.
- (Required) **Zone**—Zone where you are installing agents. If only one zone exists, this option is grayed out. For multiple zones, only the zones for which you have permissions are displayed.

  You can install agents to only one zone at a time. A functioning gateway must exist before an agent can be installed into the non-default zone, so installing the first agent into a zone must be done manually.
- (Required) **Driving Resource**—Resource to perform all actions for installing a host on behalf of the server. This is the agent that installs the agents on the remote hosts.

  The driving resource must belong to the same platform and zone as the hosts being installed or upgraded. Also, on Windows systems containing the driving resource, Windows PowerShell 3.0 or newer must be installed on older Windows versions.

- **Resource Name Format**—Naming scheme for newly-created resources to correspond to the hosts on which to perform the installation.
  - **Hostname/IP**—Use fully-qualified domain names (FQDNs), which will be derived from the host names or IP addresses of the hosts.
  - **Short name**—Use host names that do not include domain names. For example, the `host11` part of `host11.bigco.com`.

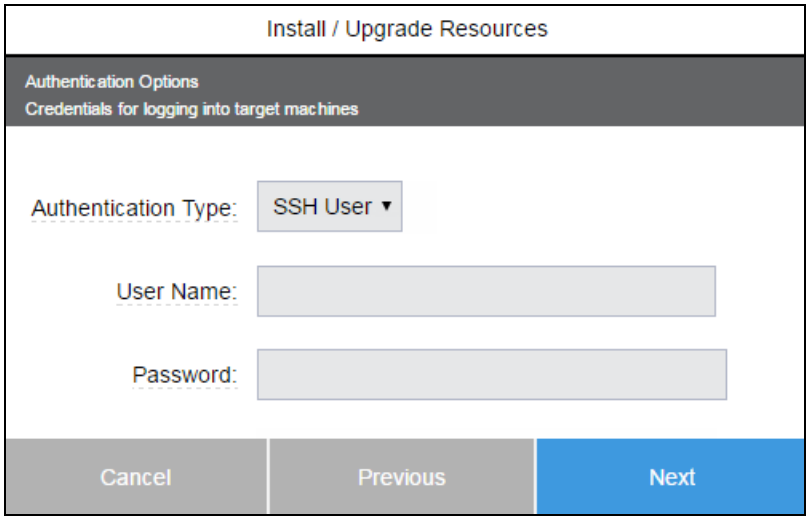
    If you provided IP addresses for the target hosts for installation, resources are named using the short names of the FQDNs, which are derived from those IP addresses. If you provided short host names for installation, resources are named using the short names.

    If you provided IP addresses for the target hosts for installation and the FDQNs cannot be determined, the installation will fail for those resources (but will continue with the rest of the installation). Also, because there might be duplicate resource names across domains and subdomains if you use short names, if a resource by that name already exists, the installation will fail for the resource (but will continue with the rest of the installation).
  - **Custom**—Custom template for resource naming. The value in this field undergoes property expansion in a global context and is scanned for the {name}, {counter}, and {shortname} special tokens.

Click **Next**.

One of two **Authentication Settings** dialog boxes appears depending on the platform.

Linux- or UNIX-Based Platforms

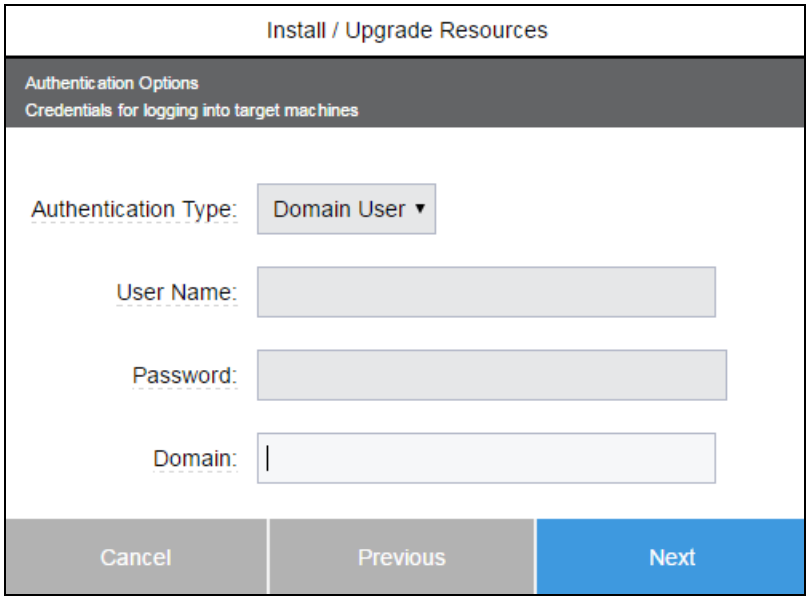


From the **Authentication Type** menu, choose **SSH User** or **SSH Key** and enter authentication information required to connect to the remote machines:

- **SSH User**:
  - (Required) **User Name**—User name. This user must have must have administrator privileges on the target machines.
  - (Required) **Password**—Password for the user name.

- **SSH Key**:
  - (Required) **User Name**—User name. This user must have must have administrator privileges on the target machines.
  - (Required) **Public Key Path**—Path to the SSH public key file.
  - (Required) **Private Key Path**—Path to the SSH private key file.
  - (Optional) **Passphrase**—Passphrase for unlocking the private key file.

Windows Platforms



> **Note:** Before using a domain user to install agents on remote hosts, you must enable multi-hop support for WinRM on the driving resource system and on each remote host. For more information, see the *Enabling Multi-Hop Support for Windows Remote Management Before Installing or Upgrading Remote Agents* KB article.

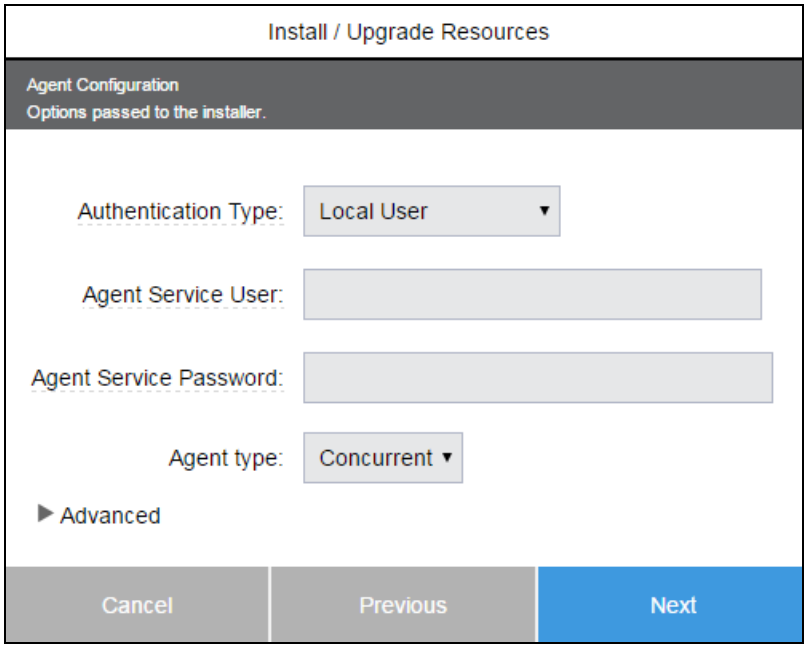
Enter the authentication information required to connect to the remote machines:

- (Required) **User Name**—User name. This user must have must have administrator privileges on the target machines.
- (Required) **Password**—Password for the user name.
- (Required) **Domain**—User's Windows domain.

Click **Next**.

One of two **Agent Configuration** dialog boxes appears depending on the platform.

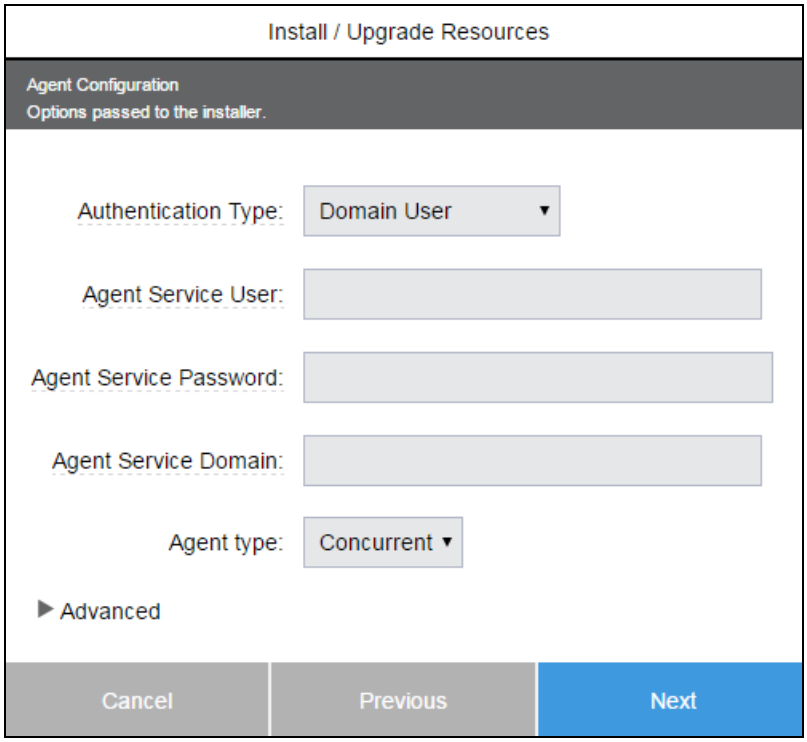
Linux- or UNIX-Based Platforms



Provide the agent settings to be passed to the ElectricFlow agent installer. The agent settings appear in two sections: Information for the user account to be used to run the agent and the agent or resource configurations such as agent port:

- (Required) **Agent Service User**—User that the ElectricFlow agent runs as.
- (Required) **Agent Service Group**—Group that the ElectricFlow agent runs as.

Windows Platforms



Provide the agent settings to be passed to the ElectricFlow agent installer. The agent settings appear in two sections: Information for the user account to be used to run the agent and the agent or resource configurations such as agent port:

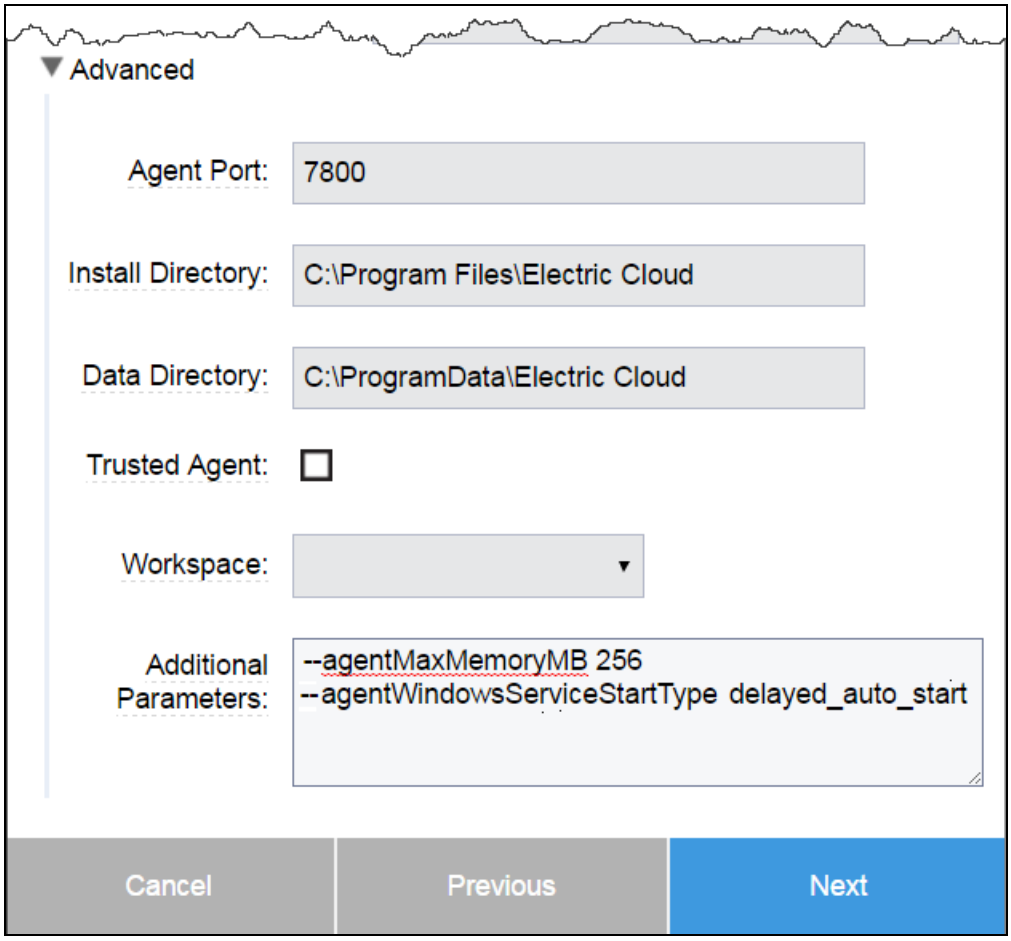
- **Local User**:
  - (Required) **User Name**—User name.
  - (Required) **Password**—Password for the user name.
- **Domain User**:
  - (Required) **User Name**—User name.
  - (Required) **Password**—Password for the user name.
  - (Required) **Domain**—User's Windows domain.
- **Built-in User Account**:
  - (Required) **Service Account**—**LocalSystem** is the only option available in this release.

All Platforms

- **Agent type**—Agent type. (This menu appears only if a mixed-mode license is installed on the server for both registered and concurrent hosts.)
  - If the license on the server is a concurrent resource license, the host type defaults to **Concurrent**, and this menu does not appear.
  - If the license on the server is a registered host license, the host type defaults to **Registered**, and this menu does not appear.

- If the license on the server is a mixed-mode license (concurrent resources and registered hosts), you must choose the host type when adding a resource.

(Optional) Click **Advanced** to view additional options:



Complete the advanced options as follows:

- **Agent Port**—Port for communication between the local agent and the server. The default is 7800.
- **Install Directory**—Path to the install directory. Spaces are allowed. Do not use quotation marks.
- **Data Directory**—Path to the directory where your modified ElectricFlow files are stored (configuration and log files). Spaces are allowed. Do not use quotation marks.
- **Trusted Agent**—Check this check box if all agent machines being installed or upgraded are trusted. For more information, see the "Switching a Non-Trusted Agent to Trusted" section in the "Automation Platform" chapter of the *ElectricFlow User Guide* at http://docs.electric-cloud.com/eflow_doc/FlowIndex.html.
- **Workspace**—Name of the workspace to set as the default workspace for the new resource(s).

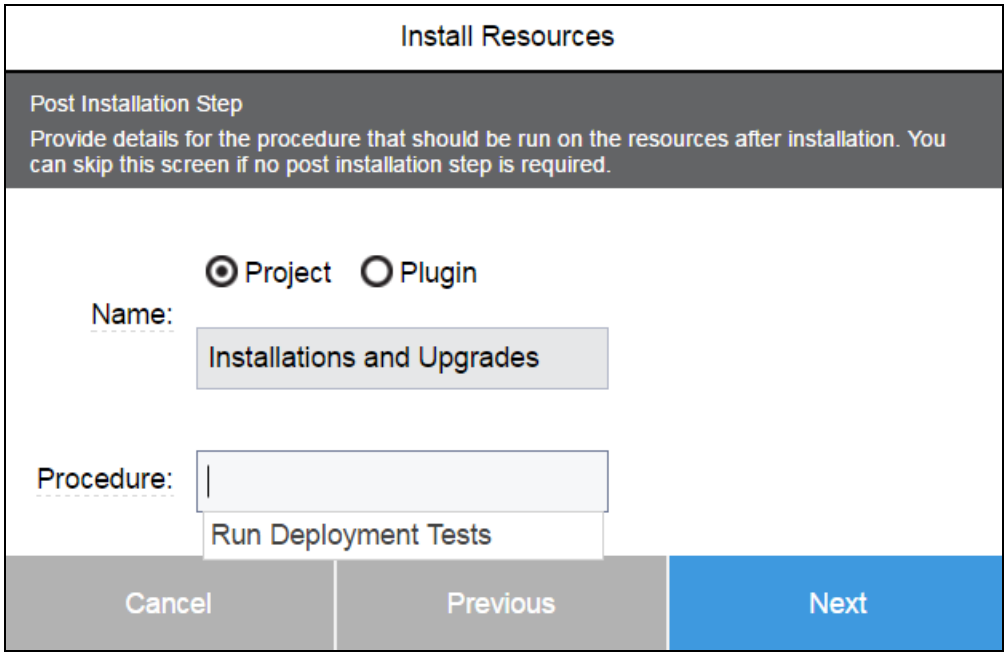
- **Additional Parameters**—Parameters that are not available in the UI. For example, `--agentMaxMemoryMB 256`. All arguments for a parameter must be on the same line.

Click **Next**.

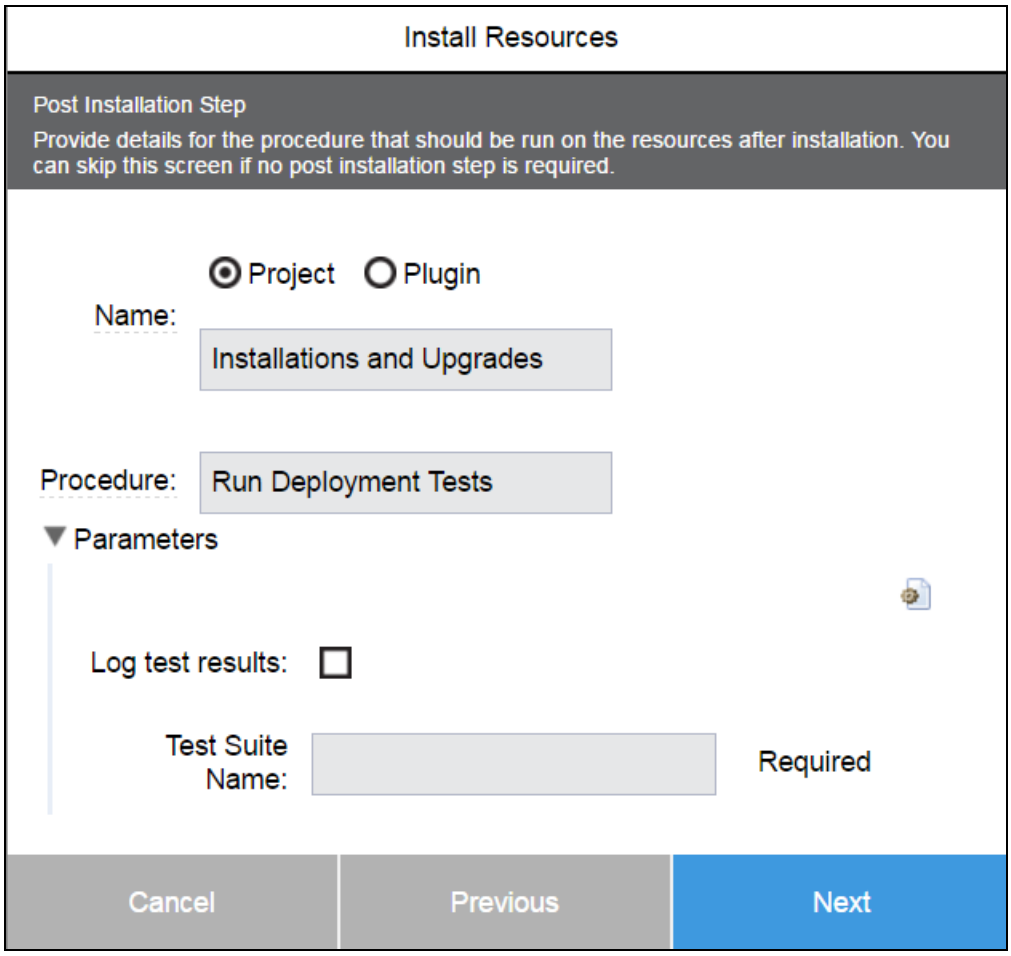
The **Post Installation Step** dialog box appears. (You can skip the dialog box by clicking **Next** at this point.)

You can choose **Project** or **Plugin**. The following examples use **Project**.

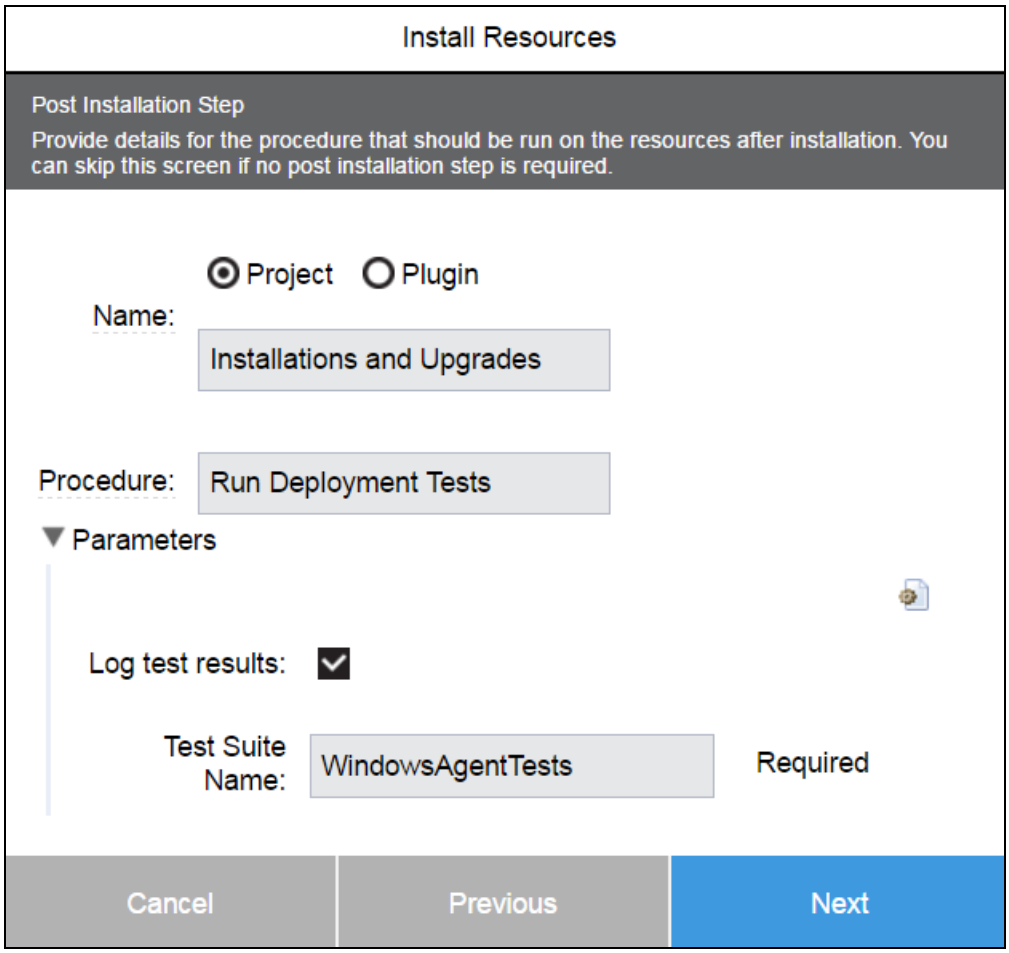
Enter a project name into the **Name** dialog box. For example:



Enter a procedure name. If the procedure has parameters, the **Parameters** menu appears, which includes the parameter fields for the selected procedure. For example:



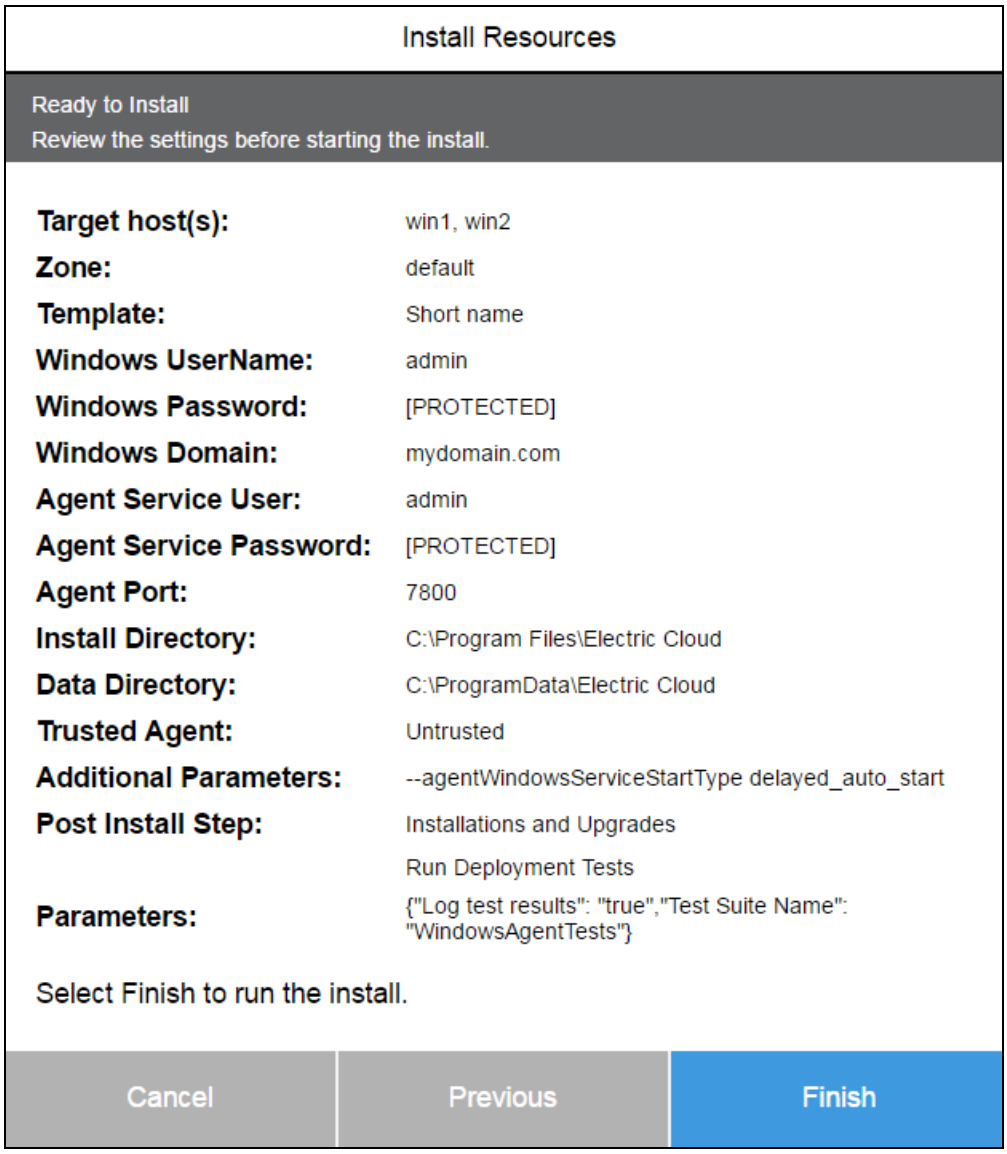
Enter the parameters for the selected procedure. For example:



Note: Credential parameters are not supported in post-installation steps.

Click **Next**.

The **Ready to Install** dialog box appears, which summarizes your settings. For example:



For more than five hosts, the number of hosts appears (instead of a space-separated list of host names).

Review your settings in the **Ready to Install** dialog box, then click **Finish** to start the installation or upgrade. The **Job Details** page appears.

When the installation or upgrade finishes, you can return to the **Resources** page to see the hosts that were just installed. To verify a resource version, click a resource name (in table view) to open the **Resource Details** panel for that resource.

Installing Remote Agents Using the API

This section describes how to use the ElectricFlow API to install remote agents. This process uses the underlying ElectricFlow Centralized Agent Management (CAM) functionality.

You can automate remote installations by using scripts that call the following ElectricFlow APIs. These APIs provide the same remote installation capability as the web interface.

Parameters for Running the Procedure to Publish an Installer to the Artifact Repository

The `Publish Installer` procedure is used to publish an ElectricFlow installer to the artifact repository. The parameters for the `Publish Installer` procedure appear below. These parameters have equivalent options in the web interface.

Parameter	Description
`fromDir`	Directory where the installer is located. Only the installer itself will be published.
`platform`	Platform of the installer. The possible values are Linux_x86, Linux_x64, Windows_x86, Windows_x64, Darwin, SunOS, SunOSx86, or HP-UX.
`repository`	ElectricFlow repository server where the installer will be published.
`resource`	Resource used by the publishing procedure.
`version`	Version of the installer. Use the format *<major>.<minor>.<patch>.<build>*. Because the build number changes with every build of the corresponding plugin, you should check this number often and update it if needed.

Parameters for Running the Procedure to Perform the Installations

A top-level wrapper procedure named `Centralized Agent Management` is used for installer publishing. This procedure calls the `Install Agent` procedure to publish installers to the repository.

The parameters for the `Centralized Agent Management` procedure appear below. Most of these parameters have equivalent options in the web interface.

Parameter	Windows Support	Other Platform Support	Description
Installation or Upgrade Parameters Based on Platform and Target Host			
`isUpgrade`	Yes	Yes	Specifies whether you are installing agents or upgrading existing agents. Use `0` if installing agents and `1` if upgrading agents.
`resources`	Yes	Yes	(Upgrades only) Space-delimited list of resources to upgrade. The resources should belong to the same platform (such as Windows or Linux) and the same zone.

Parameter	Windows Support	Other Platform Support	Description
`hostnames`	Yes	Yes	(Installations only) List of hostnames or IP addresses on which to install the agent. You can delimit the host names or IP addresses by using any combination of spaces, commas, semicolons, or newlines. Host names cannot contain spaces.The hosts should belong to the same platform and the same network.
`zoneName`	Yes	Yes	ElectricFlow zone in which the agents will be installed or upgraded.
`drivingResource`	Yes	Yes	Existing resource within the selected zone that will be used to perform all proxy commands for installing or upgrading the target hosts. The driving resource must belong to the same platform and the same zone as the hosts.
`resourceNameTemplate`	Yes	Yes	(Installations only) String templates. Used for naming the resources in ElectricFlow for the corresponding installation target hosts. The special tokens `{name}`, `{counter}`, and `{shortname}` can be used as place holders for the host name and for incrementing counters in the template value. For example, `{name}-Win-{counter}`.
`version`	Yes	Yes	Version of the agent previously published to the repository server to use. For example, 7.0.0.110954.
`platform`	Yes	Yes	Platform of the resources being upgraded or installed. The possible values are Linux_x86, Linux_x64, Windows_x86, Windows_x64, Darwin, SunOS, SunOSx86, or HP-UX.
Connection Parameters for Communicating With the Target Hosts			

Parameter	Windows Support	Other Platform Support	Description
connectionType	Yes	Yes	Type of authentication to use when connecting to the target machines. Use WINDOWS when installing or upgrading resources on Windows. Otherwise, use either SSH_PASSWORD or SSH_KEY depending on whether the SSH user name and password will be used or an SSH key will be used.
connectionCredential	Yes	Yes	Connection credentials based on the type of authentication used for connecting to the target hosts. Use the user account name and password if connectionType is WINDOWS or SSH_USER. Use the user name and SSH key passphrase if connectionType is SSH_KEY.
connectionDomain	Yes	-	(Windows platforms only) Name of the windows domain for the specified user account.
connectionPublicKey	-	Yes	Path to the SSH public key file on the driving resource.
connectionPrivateKey	-	Yes	Path to the SSH private key file on the driving resource.
Agent Configuration Parameters for Installing Agents on Target Hosts (There are additional parameters for agent configuration that are not listed here. To specify additional agent configuration parameters, use the **Additional Parameters** field as described above.)			
agentPort	Yes	Yes	(Installations only) Port that will be used by the agents installed on the target hosts. The default is 7800.
installDirectory	Yes	Yes	(Installations only) Path to the installation directory on the target hosts.
dataDirectory	Yes	Yes	(Installations only) Path to the data directory on the target hosts.
agentWorkspaceName	Yes	Yes	(Installations only) Name of the workspace that will be used by the agents installed on the target hosts.

Parameter	Windows Support	Other Platform Support	Description
`agentCredential`	Yes	Yes	(Installations only) Password for the user account that the installed agent service should run as. (This is not applicable if you are using `LocalSystem` as the agent user on Windows.)
`agentDomain`	Yes	-	(Installations only) Windows domain name for the agent user, if the user account belongs to a domain.
`trustedAgent`	Yes	Yes	(Installations only) Specifies whether the agent will be installed as a trusted agent. A trusted agent is "certificate verified," which means that the agent verifies the server or upstream agent's certificate. If this is set to `1`, the agent will be installed as a trusted agent. For more information, see the “Switching a Non-Trusted Agent to Trusted” section in the “Automation Platform” chapter of the *ElectricFlow User Guide* at http://docs.electric-cloud.com/eflow_doc/FlowIndex.html.
`agentGroup`	-	Yes	(Installations only) Group of the user account that the installed agent service should run as.
`hostType`	Yes	Yes	(Installations only) Host type when the service is licensed for both concurrent and registered agent hosts. Specify `CONCURRENT` to install concurrent host agents. Specify `REGISTERED` to install registered host agents.

Parameter	Windows Support	Other Platform Support	Description
additionalParameters	Yes	Yes	Specifies additional agent configuration parameters. **Note:** Do not use this parameter to specify agent configuration parameters (listed above) in conjunction with their equivalent arguments in the standard agent installer. Doing so might cause unpredictable behavior. For example, do not use `ectool runProcedure EC-AgentManagement-1.2.0.111083 --procedureName "Centralized Agent Management" --actualParameter "additionalParameters=--agentPort 7800"`. Instead, use `ectool runProcedure EC-AgentManagement-1.2.0.111083 --procedureName "Centralized Agent Management" --actualParameter "agentPort=7800"`. For a complete list of agent installer arguments, see Silent Install Arguments on page 3-113.
Post-Installation Parameters			
postStepProjectName	Yes	Yes	Name of the project that contains the procedure to run as a post-installation step on each of the just-installed or upgraded resources.
postStepProcedureName	Yes	Yes	Name of the procedure to run as a post-installation step on each of the just-installed or upgraded resources.

Parameter	Windows Support	Other Platform Support	Description
postStepParameters	Yes	Yes	Parameters in JSON format. You must escape special characters that are not supported by JSON.

Examples for Publishing an ElectricFlow Installer to the Artifact Repository

As mentioned above, `Publish Installer` is a procedure used to publish an ElectricFlow installer to the artifact repository. Note that the ec-perl and DSL scripts below contain commands for using the promoted plugin version without hard-coding the version.

ec-perl Example

```
use strict;
use ElectricCommander;
my $ec = new ElectricCommander();
my $xpath      = $ec->getPlugin("EC-AgentManagement");
my $pluginProject = $xpath->findvalue('//projectName')->value;
my $jobId = $ec->runProcedure($pluginProject, {
       procedureName => "Publish Installer",
       actualParameter => [
               {
               actualParameterName => 'fromDir',
               value => '/home/build'
               },
               {
               actualParameterName => 'platform',
               value => 'Linux_x64'
               },
               {
               actualParameterName => 'repository',
               value => 'default'
               },
               {
               actualParameterName => 'resource',
               value => 'local'
               },
               {
               actualParameterName => 'version',
               value => '7.0.0.110576'
               }
       ]
})->findvalue('//jobId')->value();
print "jobId is: $jobId";
```

ectool Example

```
ectool runProcedure EC-AgentManagement-1.2.0.111083 --procedureName "Publish
Installer"
--actualParameter "fromDir=/home/build"
--actualParameter "platform=Linux_x86"
--actualParameter "repository=default"
```

```
--actualParameter "resource=local"
--actualParameter "version=7.0.0.110576"
```

DSL Example

```
ectool evalDsl "
def pluginProject = getPlugin(pluginName: 'EC-AgentManagement').projectName;
runProcedure(pluginProject, procedureName: 'Publish Installer',
    actualParameter: [
        fromDir:    '/home/build',
        platform:   'Linux_x86',
        repository: 'default',
        resource:   'local',
        version:    '7.0.0.110576'
    ]
)"
```

Examples for Installing Remote Agents

As mentioned above, `Centralized Agent Management` is a top-level wrapper procedure that calls the `Install Agent` procedure for installing or upgrading individual agents on the agent hosts. Note that the DSL scripts below contain commands for using the promoted plugin version without hard-coding the version.

ectool Example (Linux)

```
ectool runProcedure EC-AgentManagement-1.2.0.111083 --procedureName "Centralized Agent
Management"
--actualParameter "isUpgrade=0"
--actualParameter "platform=Linux_x64"
--actualParameter "version=7.0.0.110954"
--actualParameter "zoneName=default"
--actualParameter "hostnames=192.168.4.208 192.168.4.210"
--actualParameter "drivingResource=local"
--actualParameter "resourceNameTemplate=prefix-{counter}"

--actualParameter "agentCredential=agentCredential"
--actualParameter "agentGroup=remote"
--actualParameter "agentPort=7800"

--actualParameter "trustedAgent=0"
--actualParameter "hostType=CONCURRENT"

--actualParameter "connectionType=SSH_PASSWORD"
--actualParameter "connectionCredential=connectionCredential"

--credential connectionCredential=remote
--credential agentCredential=remote
```

ectool Example (Linux with Post-Installation Procedure)

```
ectool runProcedure EC-AgentManagement-1.3.0.0 --procedureName "Centralized Agent
Management" \
--actualParameter "isUpgrade=0" \
--actualParameter "platform=Linux_x64" \
--actualParameter "version=7.0.0.111324" \
--actualParameter "zoneName=default" \
--actualParameter "hostnames=10.200.1.109" \
--actualParameter "drivingResource=local" \
```

```
--actualParameter "resourceNameTemplate=prefix-{counter}" \
--actualParameter "agentCredential=agentCredential" \
--actualParameter "agentGroup=vagrant" \
--actualParameter "agentPort=7800" \
--actualParameter "trustedAgent=0" \
--actualParameter "hostType=CONCURRENT" \
--actualParameter "connectionType=SSH_PASSWORD" \
--actualParameter "connectionCredential=connectionCredential" \
--actualParameter "postStepProcedureName=postinstall" \
--actualParameter "postStepProjectName=postinstall" \
--actualParameter "postStepParameters={\"another_resource\": \"local\",\"checkbox\":
\"true\",\"credential\": \"\",\"dropdown\": \"option3\",\"project\": \"Default\",
\"radio\": \"option3\",\"required1\": \"required1\",\"saved_filter\":
\"/projects/EC-Examples/ec_savedSearches/Test Outcome Filter\",\"textarea\":
\"Lorem ipsum amet\ntempor incididunt \nquis nostrud\nconsequat.\"}" \
--credential connectionCredential=vagrant \
--credential agentCredential=vagrant
```

ectool Example (Windows)

```
ectool runProcedure EC-AgentManagement-1.2.0.111083 --procedureName "Centralized Agent
Management"
--actualParameter "isUpgrade=0"
--actualParameter "platform=Windows_x64"
--actualParameter "version=7.0.0.110954"
--actualParameter "zoneName=default"
--actualParameter "hostnames=10.1.216.227"
--actualParameter "drivingResource=3EC-IT-3614"
--actualParameter "resourceNameTemplate={name}"

--actualParameter "connectionDomain=electric-cloud"
--actualParameter "agentCredential=agentCredential"
--actualParameter "agentPort=7800"

--actualParameter "trustedAgent=0"
--actualParameter "hostType=CONCURRENT"

--actualParameter "connectionType=WINDOWS"
--actualParameter "connectionCredential=connectionCredential"

--credential connectionCredential=remote
--credential agentCredential=remote
```

DSL Example (Linux)

In the following example, `agentGroup` is the group for the Linux user to connect to the remote machines.

```
ectool evalDsl "
def pluginProject = getPlugin(pluginName: 'EC-AgentManagement').projectName;
runProcedure(pluginProject, procedureName: 'Publish Installer',
    actualParameter: [
        isUpgrade: '0',
        platform: 'Linux_x64',
        version: '7.0.0.110954',
        zoneName: 'default',hostnames: '192.168.4.210',
        drivingResource: 'local',
        resourceNameTemplate: 'prefix-{counter}',
```

```
        agentCredential: 'agentCredential',
        agentGroup: 'remote',
        agentPort: '7800',

        trustedAgent: '0',
        hostType: 'CONCURRENT',

        connectionType: 'SSH_PASSWORD',
        connectionCredential: 'connectionCredential'

    ],
    credential: [
        [
            credentialName: 'connectionCredential',
            userName: 'remote',
            password: ''Rem0te2''
        ],
        [
            credentialName: 'agentCredential',
            userName: 'remote',
            password: ''Rem0te3''
        ]
    ]
)"
```

DSL Example (Linux with Post-Installation Procedure)

```
ectool evalDsl "
def pluginProject = getPlugin(pluginName: 'EC-AgentManagement').projectName;
runProcedure(pluginProject, procedureName: 'Centralized Agent Management',
    actualParameter: [
        isUpgrade: '0',
        platform: 'Linux_x64',
        version: '7.0.0.111324',
        zoneName: 'custom',
        hostnames: '10.200.1.109',
        drivingResource: 'local',
        resourceNameTemplate: '{shortname}-{name}-{counter} ',

        agentCredential: 'agentCredential',
        agentGroup: 'vagrant',
        agentPort: '7800',

        trustedAgent: '0',
        hostType: 'CONCURRENT',

        connectionType: 'SSH_PASSWORD',
        connectionCredential: 'connectionCredential',

        postStepProcedureName: 'postinstall',
        postStepProjectName: 'postinstall',
        postStepParameters: '{"another_resource": "local","checkbox":
"true","credential": "","dropdown": "option3","project": "Default",
"radio": "option3","required1": "required1","required2": "req2",
"saved_filter": "/projects/EC-Examples/ec_savedSearches/Test Outcome
Filter","textarea": "Lorem ipsum amet\ntempor incididunt \nquis
```

```
nostrud\ nconsequat."}'
    ],
    credential: [
        [
            credentialName: 'connectionCredential',
            userName: 'remote',
            password: 'remote4'
        ],
        [
            credentialName: 'agentCredential',
            userName: 'remote',
            password: 'remote5'
        ]
    ]
)"
```

DSL Example (Windows)

In the following example, `connectionDomain` is the Windows domain for the Windows user to connect to the remote machines.

```
ectool evalDsl "
def pluginProject = getPlugin(pluginName: 'EC-AgentManagement').projectName;
runProcedure(pluginProject, procedureName: 'Publish Installer',
    actualParameter: [
        isUpgrade: '0',
        platform: 'Windows_x64',
        version: '7.0.0.110954',
        zoneName: 'default',
        hostnames: '10.1.216.235',
        drivingResource: '10.1.216.227',
        resourceNameTemplate: '{name}',

        connectionDomain: 'electric-cloud',
        agentCredential: 'agentCredential',
        agentPort: '7800',

        trustedAgent: '0',
        hostType: 'CONCURRENT',

        connectionType: 'WINDOWS',
        connectionCredential: 'connectionCredential'
    ],
    credential: [
        [
            credentialName: 'connectionCredential',
            userName: 'remote',
            password: ''Rem0te6''
        ],
        [
            credentialName: 'agentCredential',
            userName: 'remote',
            password: ''Rem0te7''
        ]
    ]
)"
```

Upgrading Remote Agents Using the Web Interface

This section describes how to use the ElectricFlow web interface to upgrade remote agents. This process uses the underlying ElectricFlow Centralized Agent Management (CAM) functionality.

> **Note:** Agents installed by root or using sudo can be upgraded only by root or using sudo.

In the **Cloud** > **Resources** page of the Automation Platform, check the check boxes for resources that you want to upgrade on agent hosts. If the selected resources have different platforms or zones, an error prompt appears:

> **Resources belonging to different platforms and zones cannot be upgraded at the same time. Please select resources belonging to the same platform and the same zone**.

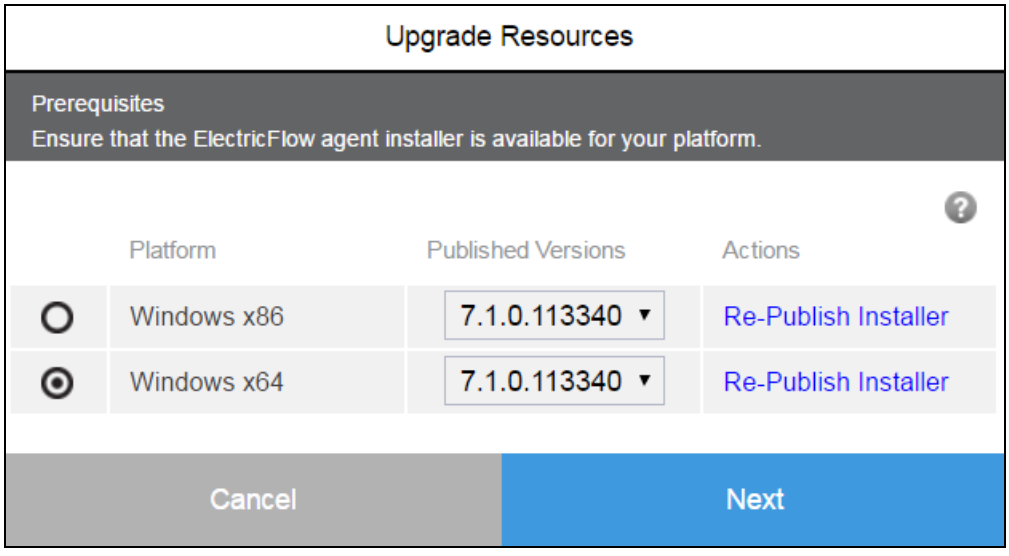
If any selected resource is a proxy resource, an error prompt appears:

> **Proxy resources are not supported for upgrade. Select the resources acting as proxy agents directly for upgrade**.

Then click the (add) menu and then click **Upgrade Resource(s)**. If the **Upgrade Resource (s)** menu option is not visible, you must log out and then log in as a user with the required permissions. For details about required permissions, see Permissions for Installing or Upgrading Remote Agents on page 3-144

> **Note:** On Solaris platforms, the installation directory path must have fewer than 70 characters.

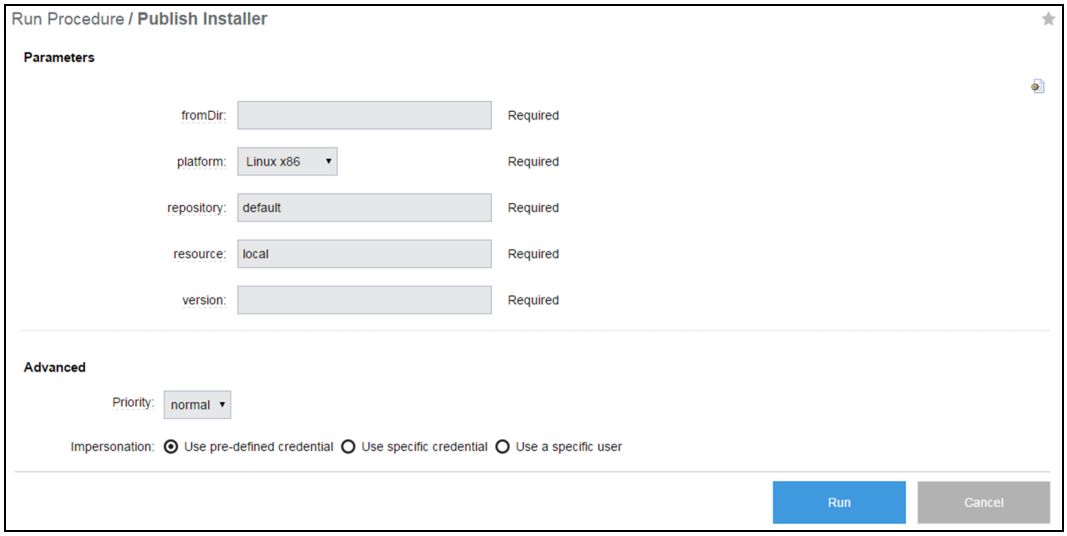
The **Upgrade Resources** dialog box appears. For example:



Either one or two platforms for the selected resource are listed. Two platforms are listed if the platform (such as Linux x86 or Linux x64) could not be determined.

Choose the platform and published version, click the appropriate **Re-Publish Installer** link, then click **Next**.

The **Run Procedure / Publish Installer** page opens:



Enter the following information:

Parameters

- (Required) **fromDir**—Directory on the resource used for publishing. For example, enter `/var/tmp` for Linux or `C:/<temp>` for Windows.
- (Required) **platform**—Host platform type.
- (Required) **repository**—Repository name. Use **default** to use the repository installed during the ElectricFlow installation or enter another name.
- (Required) **resource**—Name of the resource used for publishing.
- (Required) **version**—Build version for the ElectricFlow installation. For example, enter `5.0.0.56390`. You can find the build version from the file name of the ElectricFlow installer.

(Optional) Advanced

Set advanced options if needed.

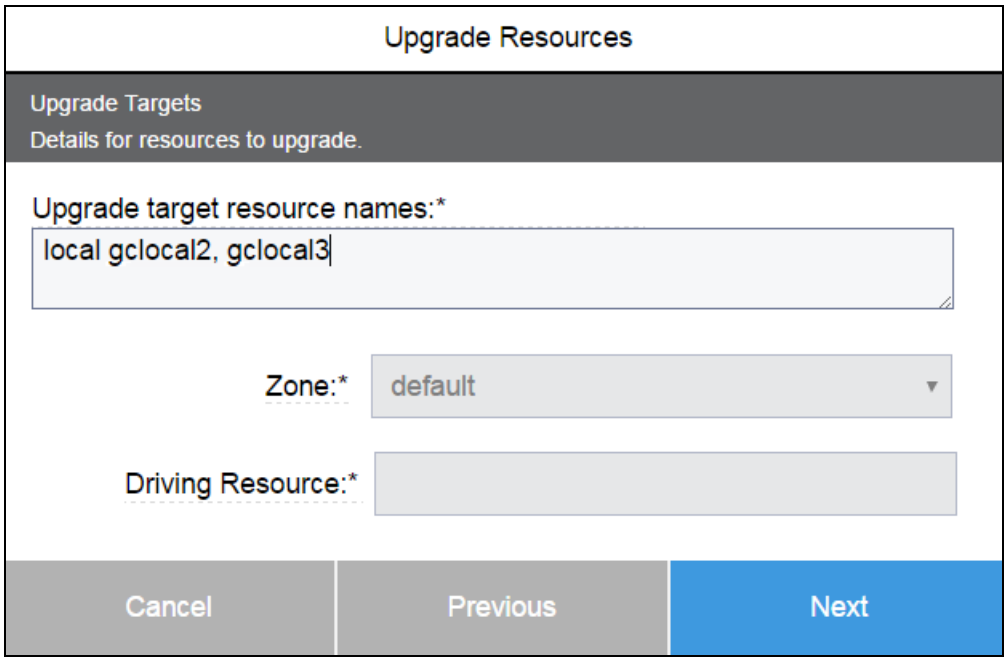
When the Publish Installer procedure runs, you can see the job status on the **Job Details** page.

If you want to see the published installers, click the **Artifacts** tab and then click the `com.electriccloud:installer` artifact. The **Artifact Details** page lists the published installers.

Click **Run**.

The **Upgrade Targets** dialog box appears.

Enter a list of resource names to upgrade into the **Upgrade target resource names** field. Names must be separated by any combination of spaces, commas, semicolons, or newlines. The resources must already exist, must be in the same zone, and must be of the same platform. For example:

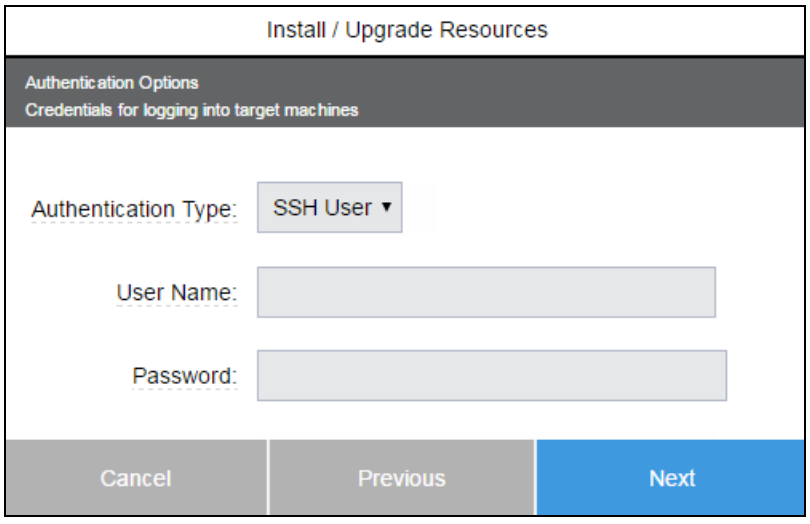


The **Zone** field displays the zone of the selected resources for upgrade. **Driving Resource** is the resource to perform all actions for installing or upgrading a host on behalf of the server. This is the agent that upgrades the agents on the remote hosts.

Click **Next**.

One of two **Authentication Settings** dialog boxes appears depending on the platform.

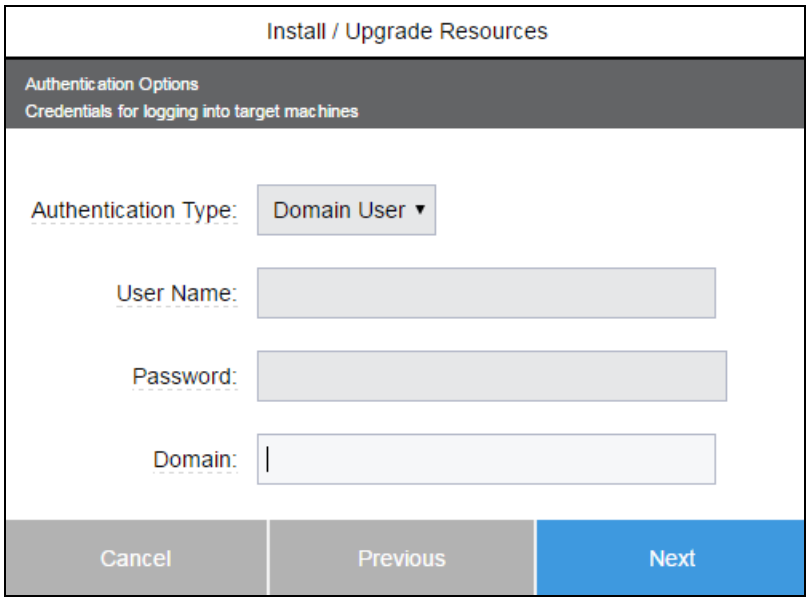
Linux- or UNIX-Based Platforms



From the **Authentication Type** menu, choose **SSH User** or **SSH Key** and enter authentication information required to connect to the remote machines:

- **SSH User**:
  - (Required) **User Name**—User name. This user must have must have administrator privileges on the target machines.
  - (Required) **Password**—Password for the user name.
- **SSH Key**:
  - (Required) **User Name**—User name. This user must have must have administrator privileges on the target machines.
  - (Required) **Public Key Path**—Path to the SSH public key file.
  - (Required) **Private Key Path**—Path to the SSH private key file.
  - (Optional) **Passphrase**—Passphrase for unlocking the private key file.

Windows Platforms



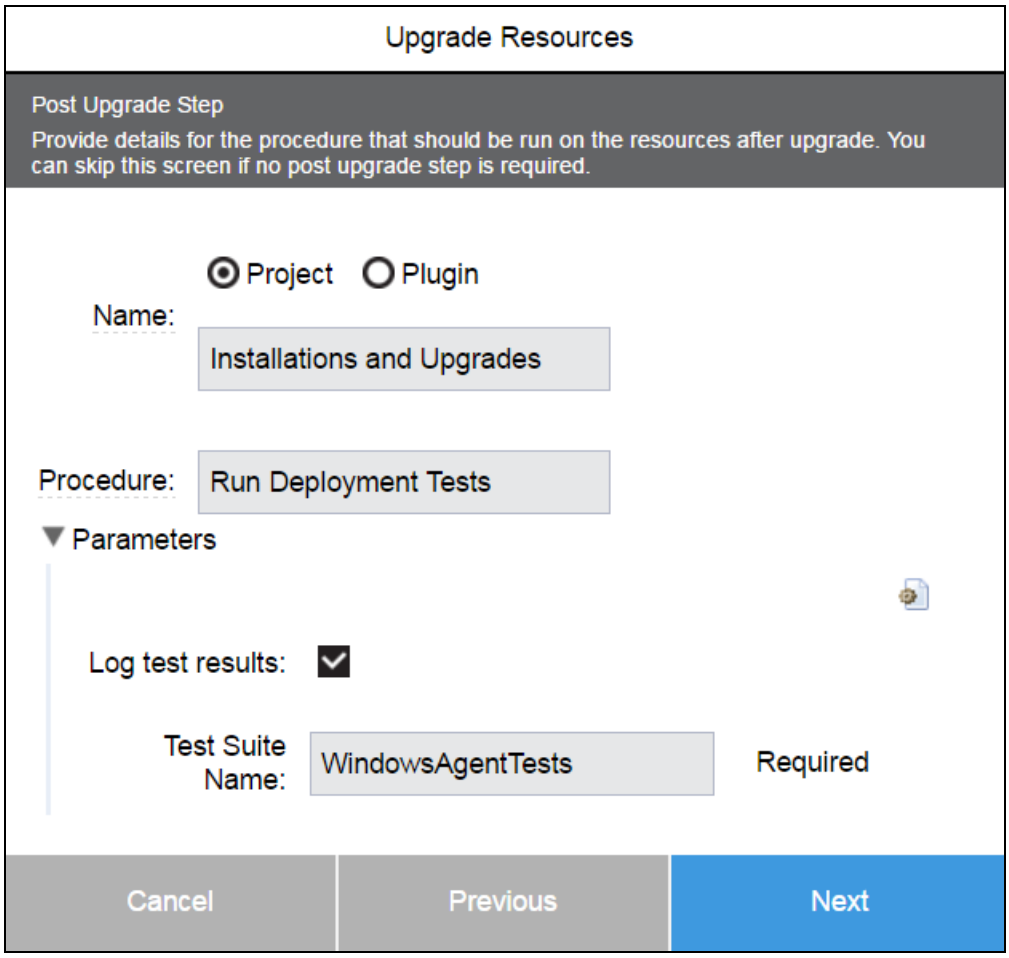
Enter the authentication information required to connect to the remote machines:

- (Required) **User Name**—User name. This user must have must have administrator privileges on the target machines.
- (Required) **Password**—Password for the user name.
- (Required) **Domain**—User's Windows domain.

Click **Next**.

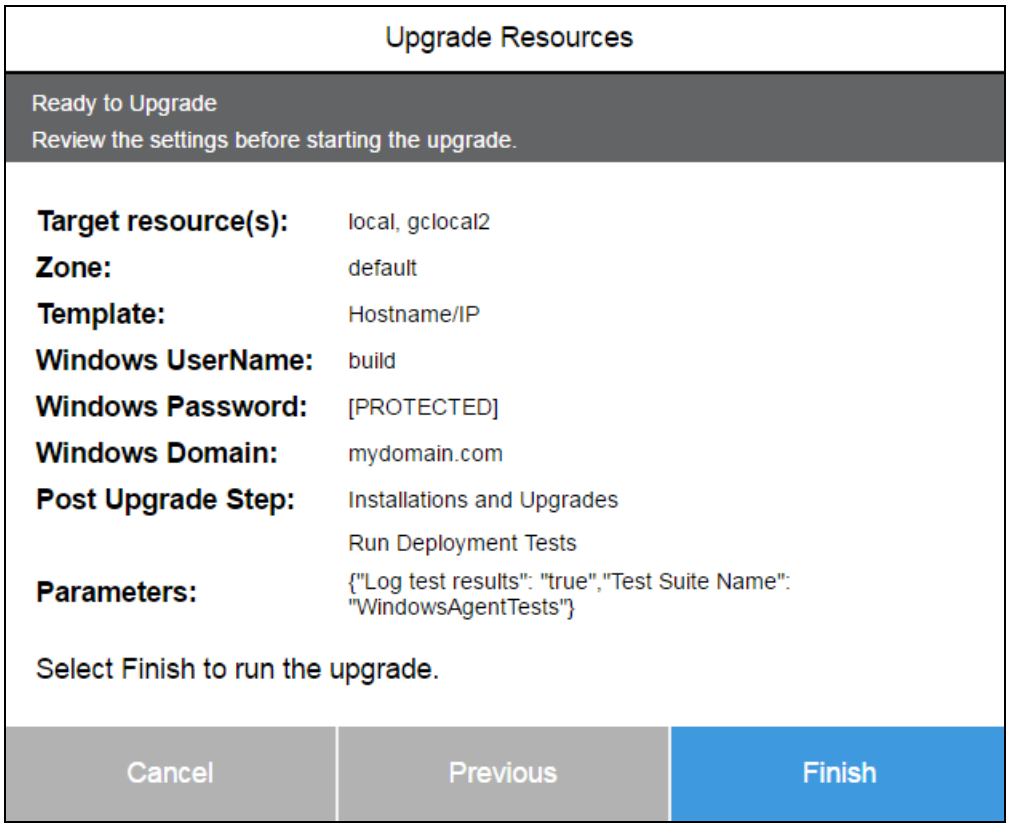
The **Post Upgrade Step** dialog box appears. (You can skip the dialog box by clicking **Next** at this point.)

You can choose **Project** or **Plugin**. The following example uses **Project**. Enter a project name, procedure name, and if the selected procedure has parameters, enter its parameters. For example:



Note: Credential parameters are not supported in post-installation steps.

The **Ready to Upgrade** dialog box appears, which summarizes your settings. For example:



For more than five hosts, the number of hosts appears (instead of a space-separated list of host names).

Review your settings in the **Ready to Upgrade** dialog box and click **Finish** to start the installation or upgrade. The **Job Details** page appears.

When the installation or upgrade finishes, you can return to the **Resources** page to see the hosts that were just upgraded. To verify a resource version, click a resource name (in table view) to open the **Resource Details** panel for that resource.

Upgrading Remote Agents Using the API

This section describes how to use the ElectricFlow API to upgrade remote agents. This process uses the underlying ElectricFlow Centralized Agent Management (CAM) functionality.

> **Note:** Agents installed by root or using sudo can be upgraded only by root or using sudo.

You can automate remote upgrades for multiple resources by using scripts that call ElectricFlow APIs to run those upgrades. These APIs provide the same remote upgrade capability as the web interface.

Parameters for Running the Procedure to Perform the Installations

See Parameters for Running the Procedure to Perform the Installations on page 3-158 above.

Examples for Upgrading Remote Agents

As mentioned above, `Centralized Agent Management` is a top-level wrapper procedure that calls the `Install Agent` procedure for installing or upgrading individual agents on the agent hosts. Note that the DSL scripts below contain commands for using the promoted plugin version without hard-coding the version.

ectool Example (Linux)

```
ectool runProcedure EC-AgentManagement-1.2.0.111083 --procedureName "Centralized Agent
Management"
--actualParameter "isUpgrade=1"
--actualParameter "platform=Linux_x64"
--actualParameter "version=7.0.0.110954"
--actualParameter "zoneName=default"
--actualParameter "resources=4.208 4.210"
--actualParameter "drivingResource=local"

--actualParameter "connectionType=SSH_PASSWORD"
--actualParameter "connectionCredential=connectionCredential"
--actualParameter "agentCredential=agentCredential"

--credential connectionCredential=remote
--credential agentCredential=remote
```

ectool Example (Windows)

```
ectool runProcedure EC-AgentManagement-1.2.0.111083 --procedureName "Centralized Agent
Management"
--actualParameter "isUpgrade=1"
--actualParameter "platform=Windows_x64"
--actualParameter "version=7.0.0.110954"
--actualParameter "zoneName=default"
--actualParameter "resources=10.1.216.227"
--actualParameter "drivingResource=3EC-IT-3614"

--actualParameter "connectionDomain=electric-cloud"

--actualParameter "connectionType=WINDOWS"
--actualParameter "connectionCredential=connectionCredential"
--actualParameter "agentCredential=agentCredential"


--credential connectionCredential=remote
--credential agentCredential=remote
```

DSL Example (Linux)

```
ectool evalDsl "
def pluginProject = getPlugin(pluginName: 'EC-AgentManagement').projectName;
runProcedure(pluginProject, procedureName: 'Publish Installer',
    actualParameter: [
        isUpgrade: '1',
        platform: 'Linux_x64',
        version: '7.0.0.110954',
        zoneName: 'default',
        resources: '4.208',
        drivingResource: 'local',
```

```
        connectionType: 'SSH_PASSWORD',
        connectionCredential: 'connectionCredential',
        agentCredential: 'agentCredential'
    ],
    credential: [
        [
            credentialName: 'connectionCredential' ,
            userName: 'remote',
            password: ''Rem0te4''
        ],
        [
            credentialName: 'agentCredential' ,
        ]
    ]
)"
```

DSL Example (Windows)

```
ectool evalDsl "
def pluginProject = getPlugin(pluginName: 'EC-AgentManagement').projectName;
runProcedure(pluginProject, procedureName: 'Publish Installer',
    actualParameter: [
        isUpgrade: '1',
        platform: 'Windows_x64',
        version: '7.0.0.110954',
        zoneName: 'default',
        resources: '10.1.216.235',
        drivingResource: '10.1.216.227',

        connectionDomain: 'electric-cloud',

        connectionType: 'WINDOWS',
        connectionCredential: 'connectionCredential',
        agentCredential: 'agentCredential'

    ],
    credential: [
        [
            credentialName: 'connectionCredential',
            userName: 'remote',
            password: ''Rem0te5''
        ],
        [
            credentialName: 'agentCredential',
        ]
    ]
)"
```

Moving the Artifact Repository in Linux

In this scenario, the `/opt/electriccloud/electriccommander/conf/repository/wrapper.conf` file has these settings:

- `set.default.INSTALL_DIRECTORY=/opt/electriccloud/electriccommander`
- `set.default.DATA_DIRECTORY=/opt/electriccloud/electriccommander`

If the Linux server where the current artifact repository is stored is full and you want to move it to a new device with more disk space, map the artifact repository to a new network location.

In this procedure, the `set.default.DATA_DIRECTORY` value will change to the `NEW_DATA_DIRECTORY` value. The `REPOSITORY_BACKING_STORE` value in `/opt/electriccloud/electriccommander/conf/repository/server.properties` will remain the same, relative to the `DATA_DIRECTORY`, which will be `NEW_DATA_DIRECTORY`.

In the example, the `DATA_DIRECTORY` will be changed to `/vagrant_data`, and the REPOSITORY_BACKING_STORE value will be relative to this directory.

The `REPOSITORY_BACKING_STORE` value is always relative to the `DATA_DIRECTORY` and cannot be set to a fully qualified absolute path. The solution in http://ask.electric-cloud.com/questions/2192/how-to-configure-which-directory-the-artifact-repository-uses will not work until you change the `DATA_DIRECTORY` as described in this procedure.

1. Publish the artifact under the `/opt/electriccloud/electriccommander/repository-data/<artifact_name>/<version>` directory:

   ```
   ectool publishArtifactVersion --artifactName <artifact_name> --version <version>
   --fromDirectory <directory> --compress 0
   ```

   Example: To publish an artifact under the `/opt/electriccloud/electriccommander/repository-data/jdoe/2001/1.0.0` directory, enter:

   ```
   ectool publishArtifactVersion --artifactName "jdoe:2001" --version "1.0.0"
   --fromDirectory /tmp/job1 --compress 0
   ```

   The `from Directory` (`/tmp/job1`) contains the files to be uploaded.

2. Stop the repository server:

   ```
   /etc/init.d/commanderRepository stop
   ```

3. Move the artifact repository data from the `/DATA_DIRECTORY` directory to the `/NEW_DATA_DIRECTORY` directory:

   ```
   mv /opt/electriccloud/electriccommander/repository-data/NEW_DATA_DIRECTORY
   ```

4. In `/opt/electriccloud/electriccommander/conf/repository/wrapper.conf`, change `set.default.DATA_DIRECTORY=/opt/electriccloud/electriccommander` to `set.default.DATA_DIRECTORY=/NEW_DATA_DIRECTORY`.

   Example: Change `set.default.DATA_DIRECTORY=/opt/electriccloud/electriccommander` to `set.default.DATA_DIRECTORY=/vagrant_data`.

5. Create a `logs/repository/` subdirectory under `NEW_DATA_DIRECTORY`.

   Example: Create `/vagrant_data/logs/repository` using the following commands:

   ```
   mkdir /vagrant_data/logs

   mkdir /vagrant_data/logs/repository
   ```

6. Copy `/opt/electriccloud/electriccommander/conf/repository/server.properties` and `/opt/electriccloud/electriccommander/conf/repository/keystore` to the `NEW_DATA_DIRECTORY/conf/repository` directory.

   Example: Copy the directories to the `/vagrant_data/conf/repository` using the following commands.

   ```
   mkdir /vagrant_data/conf

   mkdir /vagrant_data/conf/repository

   cp /opt/electriccloud/electriccommander/conf/repository/server.properties/
   vagrant_data/conf/repository

   cp /opt/electriccloud/electriccommander/conf/repository/keystore/
   vagrant_data/conf/repository
   ```

7. In `/etc/init.d/commanderRepository`, change `DATADIR=/opt/electriccloud/electriccommander` to `DATADIR=/NEW_DATA_DIRECTORY`.

   Example: Change `DATADIR=/opt/electriccloud/electriccommander` to `DATADIR=/vagrant_data`.

8. Start the repository server:

   ```
   /etc/init.d/commanderRepository start
   ```

9. Check if ElectricFlow has started using the artifact repository in the new network location:

   ```
   netstat -aon | grep 8200
   ```

   Example:

   ```
   vagrant@commander1:~$ netstat -aon | grep 8200

   tcp        0      0 0.0.0.0:8200            0.0.0.0:*               LISTEN
   off (0.00/0/0)
   ```

   This shows that the `DATA_DIRECTORY` repository has moved to `/vagrant_data`.

   The repository `service.log`, `repository.log` and `repository.pid` files will be created in the `/vagrant_data/logs/repository` directory.

10. Publish a new artifact and check if it is published:

    ```
    ectool getArtifactVersions | grep artifactVersionName
    ```

Moving the Artifact Repository in Windows

In this scenario, the DATA directory definition in `C:\ProgramData\Electric Cloud\ElectricCommander\conf\repository\wrapper.conf` or `C:/Program Files/Electric Cloud/ElectricCommander/repository/bin/wrapper-windows-x86-64.exe` (depending on where ElectricFlow is installed) has these settings:

- `set.default.INSTALL_DIRECTORY=C:/Electric Cloud/ElectricCommander`
- `set.default.DATA_DIRECTORY=C:/Electric Cloud/ElectricCommander`

If the Windows server where the current artifact repository is stored is full and you want to move it to a new device with more disk space, map the artifact repository to a new network location.

In this procedure, the set.default.DATA_DIRECTORY value will change to the NEW_DATA_DIRECTORY value. The REPOSITORY_BACKING_STORE value in C:\ProgramData\Electric Cloud\ElectricCommander\conf\repository\server.properties will remain the same, relative to the DATA_DIRECTORY, which will be NEW_DATA_DIRECTORY.

In the example, the DATA_DIRECTORY will be changed to d:\ecdata, and the REPOSITORY_BACKING_STORE value will be relative to this directory.

> **Note:** The REPOSITORY_BACKING_STORE value is always relative to the DATA_DIRECTORY and cannot be set to a fully qualified absolute path. The solution in http://ask.electric-cloud.com/questions/2192/how-to-configure-which-directory-the-artifact-repository-uses will not work until you change the DATA_DIRECTORY as described in this procedure.

1. Publish the artifact under the current directory (C:\ProgramData\Electric Cloud\ElectricCommander\repository-data\<artifact_name>\<version>):

   ```
   ectool publishArtifactVersion --artifactName <artifact_name> --version <version>
   --fromDirectory <directory> --compress 0
   ```

   Example: To publish an artifact under the current directory (C:\ProgramData\Electric Cloud\ElectricCommander\repository-data\jdoe\2001\1.0.0), enter:

   ```
   ectool publishArtifactVersion --artifactName "jdoe:2001" --version "1.0.0"
   --fromDirectory d:/temp/artest --compress 0
   ```

   The from Directory (d:/temp/artest) contains the files to be uploaded.

2. Stop the repository server one of these ways:

   - Use the Windows service pane.
   - If you have admin user permissions, enter **net stop CommanderRepository**.

3. Move the artifact repository data from the current /DATA_DIRECTORY directory to the /NEW_DATA_DIRECTORY directory using one of these methods:

   - Use Windows Explorer.
   - Enter **move *** <destinationDirectory> where the <destinationDirectory> is the /NEW_DATA_DIRECTORY directory.

   Example: To move the data to D:\ECDATA, enter **move * d:\ecdata**.

4. In C:\ProgramData\Electric Cloud\ElectricCommander\conf\repository\wrapper.conf, change set.default.DATA_DIRECTORY=C:/ProgramData/Electric Cloud/ElectricCommander to set.default.DATA_DIRECTORY=/NEW_DATA_DIRECTORY.

   Example: Change set.default.DATA_DIRECTORY=C:/ProgramData/Electric Cloud/ElectricCommander to set.default.DATA_DIRECTORY=D:/ECDATA.

5. Create a `logs/repository/` subdirectory under the `NEW_DATA_DIRECTORY`using one of these methods:
   - Use Windows Explorer.
   - Enter the `mkdir` command as in the following example:

     Example: Create `D:\ECDATA\logs\repository` using the following commands.

     ```
     mkdir D:\ECDATA\logs

     mkdir D:\ECDATA\logs\repository
     ```

6. Copy `C:\ProgramData\Electric Cloud\ElectricCommander\conf\repository\server.properties` and `C:\ProgramData\Electric Cloud\ElectricCommander\conf\repository\keystore` to the `NEW_DATA_DIRECTORY/conf/repository` directory using one of these methods:
   - Use Windows Explorer.
   - Enter the `mkdir` command as in the following example:

     Copy the directories to the `/ECDATA/conf/repository` using the following commands:

     ```
     mkdir /ECDATA/conf

     mkdir /ECDATA/conf/repository

     copy C:\ProgramData\Electric Cloud\ElectricCommander\conf\repository\
     server.properties D:\ECDATA\conf\repository\

     copy C:\ProgramData\Electric Cloud\ElectricCommander\conf\repository\
     keystore D:\ECDATA\conf\repository\
     ```

7. Start the repository server using Windows Service pane.
8. Check if ElectricFlow has started using the artifact repository in the new network location:

   ```
   netstat -aon | find “8200”
   ```

   Example:

   ```
   C:\windows\system32> netstat -aon | find "8200"

   TCP    0.0.0.0:8200           0.0.0.0:0              LISTENING       22868
   ```

   This shows that the `DATA_DIRECTORY` repository has moved to `D:\ECDATA`.

   The repository `service.log`, `repository.log` and `repository.pid` files will be created in the `D:\ECDATA \logs\repository` directory.
9. Publish a new artifact and check if it is published:

   ```
   ectool getArtifactVersions | grep artifactVersionName
   ```

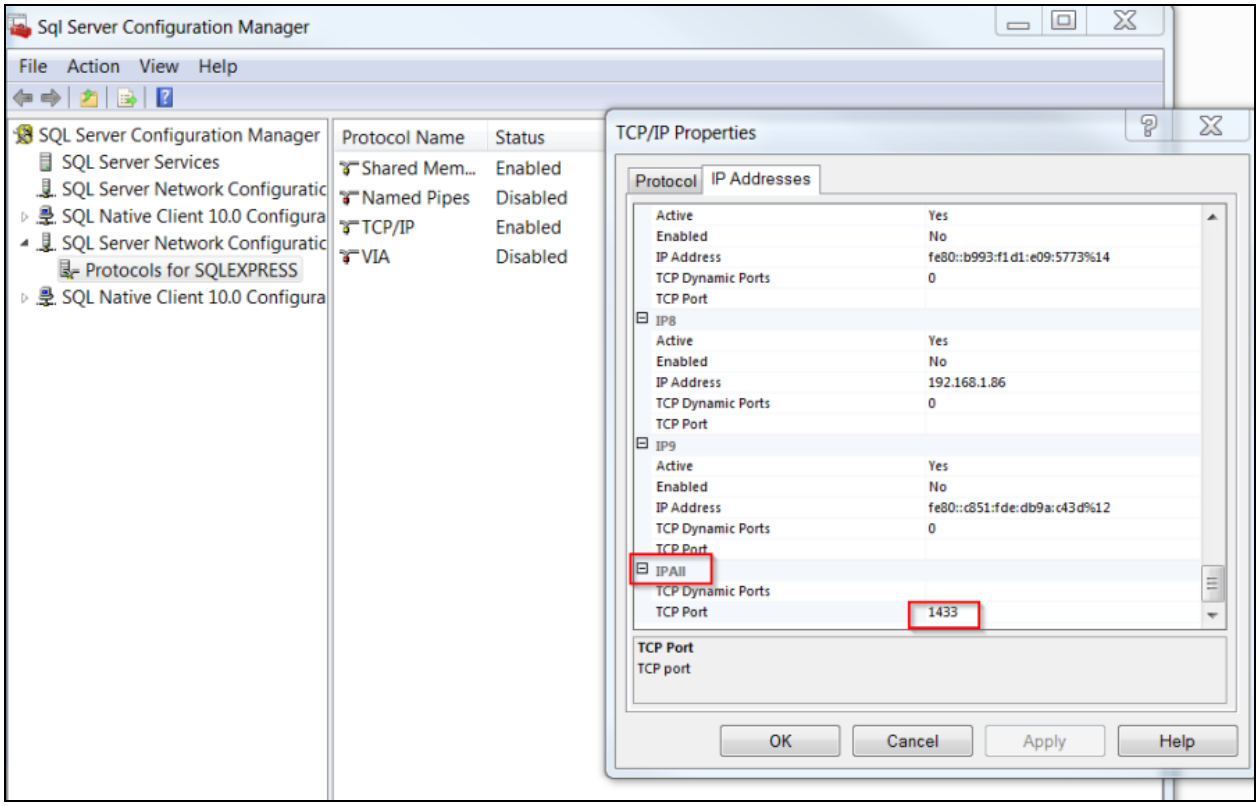
Connecting ElectricFlow to a Microsoft SQL Server Named Instance

If you are using a named instance of the SQL server, you need to configure the SQL server to listen on specific port so that the ElectricFlow server can connect to it.

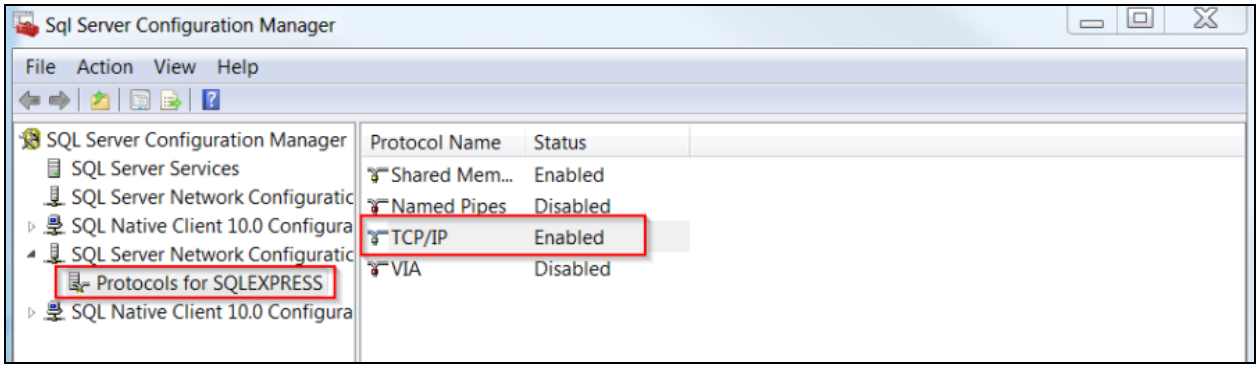
1. In SQL Server Configuration Manager, expand the **SQL Server Network Configuration**, and click on the server instance that you want to configure.
2. In the right pane, double-click **TCP/IP**.
3. In the **TCP/IP Properties** dialog box, click the **IP Addresses** tab.
4. In the IPAll section, enter an available port number in the **TCP Port** field.

   For example, enter **1433**.
5. Click **OK** to close the dialog box.
6. When a prompt appears that the server must be restarted, click **OK**.
7. In the left pane, click **SQL Server Services**.
8. In the right pane, right-click the instance of SQL Server that you selected in Step 1, and click **Restart**.

   When the Database Engine restarts, it will listen on port 1433.



9. Enable **TCP/IP**.



Installing the MySQL JDBC Driver

ElectricFlow does not include the MySQL JDBC driver. If you performed a clean ElectricFlow server installation that will connect to a MySQL database, you must obtain and install the MySQL JDBC driver. To do so, follow these steps:

1. Install ElectricFlow.

   During the installation, make sure that you choose *not* to install the built-in ElectricFlow database.

2. After the installation completes, download the correct MySQL JDBC driver from http://dev.mysql.com/downloads/connector/j/ as follows.

   - MySQL:5.6.10 and older: Use the latest MySQL Connector/J 5.1.*nn* driver from https://downloads.mysql.com/archives/c-j/.

     For example, download mysql-connector-java-5.1.46.jar.

   - MySQL 5.6.18 and newer: Use either the latest MySQL Connector/J 5.1.*nn* driver from https://downloads.mysql.com/archives/c-j/ or the MySQL Connector/J 8.0 driver from https://dev.mysql.com/downloads/connector/j/.

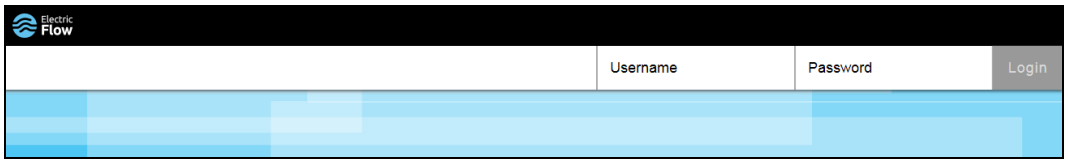
3. Install the downloaded driver into the *INSTALL_DIR*`/server/lib` directory.
4. Restart the server.
5. Open the home page of the Automation Platform web UI by browsing to https://*ElectricFlow_server*/commander/ and logging in.
6. Go to the **Administration** > **Database Configuration** page and configure the server to use a MySQL database.

Logging Into the ElectricFlow Web Interface

This task describes how to log into the ElectricFlow web interface. If you chose during installation to configure an external database, you will not be able to log into ElectricFlow until you set up a database configuration.

- Enter the URL of the ElectricFlow server in a browser window. For example, https://123.123.1.222

  The login screen appears.



Important: For a new installation, the default admin account user name is *admin* and the password is *changeme.* You should change the default admin account as soon as possible.

- Enter a user name and password, and click **Login**.

Chapter 4: Creating a Server Cluster for ElectricFlow or DevOps Insight

This chapter provides you with guidelines and procedures for adding horizontal scalability and high availability to your ElectricFlow environment. Horizontal scalability and high availability are achieved by adding additional machines to an ElectricFlow configuration to create a server cluster. A clustered configuration of ElectricFlow servers also requires these software components:

- Apache ZooKeeper, which is a centralized service for maintaining and synchronizing group services in a cluster.
- A software or hardware load balancer for routing work to machines in the cluster.

These components typically need to be managed by your IT department.

This chapter includes the Creating a DevOps Insight Server Cluster on page 4-36 section, which describes the overall steps for adding machines to create a DevOps Insight server cluster. This section also includes information about cluster upgrades, cluster reconfigurations, and configuring a cluster.

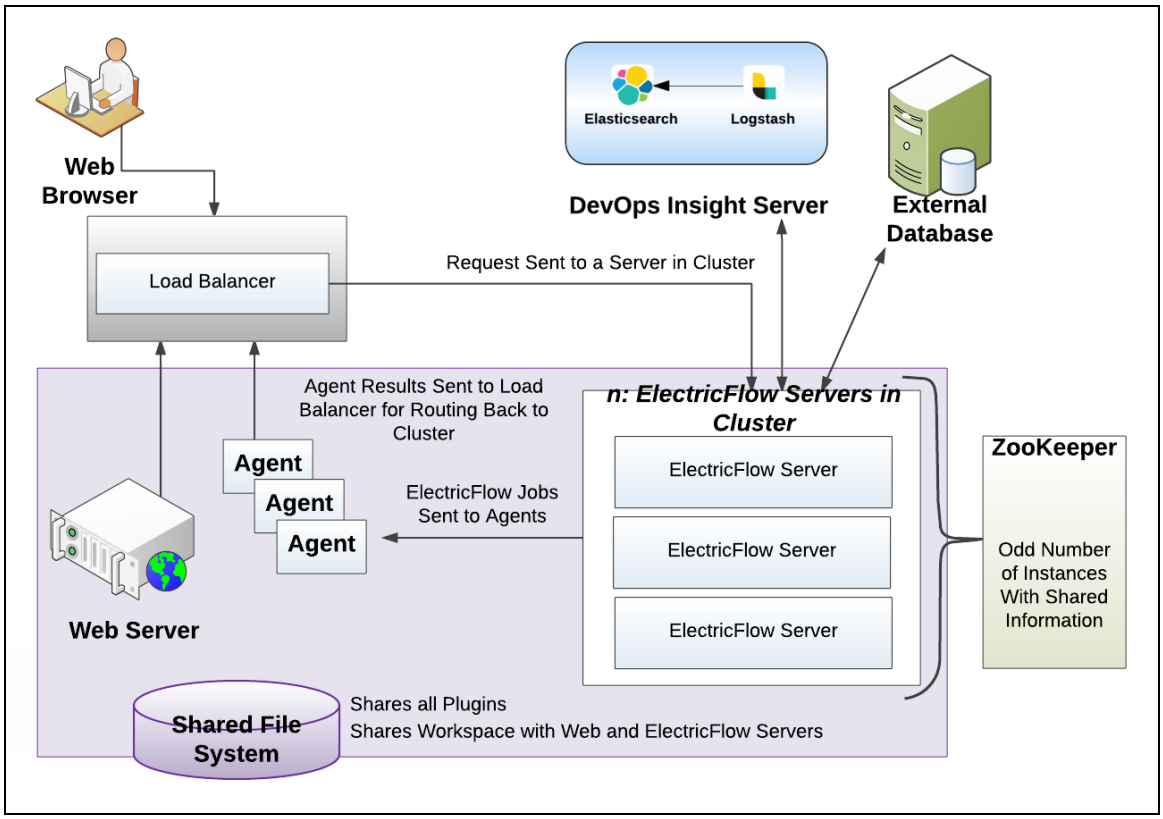
Benefits from Clustering

A clustered ElectricFlow configuration has the following benefits:

- Add fault tolerance by rerouting jobs and API requests to running ElectricFlow servers
- Increase the supported number of simultaneous jobs and API requests
- Distribute API requests across multiple ElectricFlow servers
- Expand capacity over time by adding additional ElectricFlow servers
- Distribute ElectricFlow requests across multiple web servers

Architecture of an ElectricFlow Cluster

The following figure shows an example of the ElectricFlow architecture in a clustered configuration.



In the clustered configuration, a node refers to the machine on which an agent is installed, and a server refers to the ElectricFlow server cluster. In the *server status*, the load balancer reports only status and error prompts about a specific server.

As shown in the previous diagram, we strongly recommend that you point the web browser to the load balancer, not to the web server.

Resource, Agent, and Procedure Configuration Considerations

This section describes various ElectricFlow behaviors and possible modifications that you may want to make in your ElectricFlow configuration.

Default Local Resource Use

In a default installation, an agent is installed alongside the ElectricFlow server, and a resource named *local* is automatically created for the agent. In a cluster, the local resource points to only one node. By default, some of the plugins and sample code installed with ElectricFlow automatically use the local resource because it is usually present.

Many ElectricFlow users also write procedures that use the local resource for these reasons:

- A user knows the local resource exists.
- The user needs local access to the file system, such as access to the log files or configuration files.

- The user needs local access to resources on the ElectricFlow server to perform tasks such as checking the CPU, memory, or disk space usage.

Important: Any existing procedures for local access to ElectricFlow server file systems or resources need to be modified when you change your installation to run in a server cluster. There are now multiple ElectricFlow server nodes, each with their own log files, configuration files, and local resources.

The name of the ElectricFlow log file on an ElectricFlow server node in a cluster has also been changed to `commander-<hostname>.log` to facilitate collecting the multiple logs together without them overwriting each other.

Unsupported Host

ElectricFlow does not support the local host IP address (127.0.0.1) for any ElectricFlow configuration, because it is ambiguous when multiple servers are used.

Separate Local Agents For Improved Performance

If you are using clustering for performance reasons, you must manage your machine resources efficiently. You use additional machine resources if you run an ElectricFlow agent and ElectricFlow server on the same machine. If you are concerned about performance, remove any agents from your ElectricFlow server node machines. You should also verify that any agents that may have been installed as part of an earlier configuration are also removed. *An agent is installed automatically with any ElectricFlow service by default.*

Pool Local Agents For Improved Reliability

If you are using clustering only for reliability reasons, you must reduce or eliminate single points of failure. In this configuration, having a single server node that runs the local agent is counterproductive. If you need a more reliable configuration, you can install agents on all of your ElectricFlow server node machines and put them in a resource pool named *local*. Local agents should be used for broadcast and maintenance type work. See Agent Resource Strategies on page 4-3 for more information.

Procedure Strategies

If you have a local agent on each ElectricFlow server, it may be appropriate for some procedures to have one or more procedure steps that are broadcast across all the resources in the local resource pool. These procedure steps are probably followed by a step that aggregates the resulting data in an appropriate way. For other procedures, it may be more appropriate to use the shared file system to which all the ElectricFlow servers have access, and still have the step run on a single agent.

Agent Resource Strategies

If you do not have any agents local to any of your server machines, a local resource is not automatically created. There are two possible strategies to handle this:

- Do not have a local resource and remove any local resource on your system. For everything that is currently configured to use the local resource, reconfigure them to use other resources.

- Create a resource pool called *local* for agents on non-ElectricFlow-server machines. Everything that is configured to use the local resource now runs on an agent of the local pool. However, this agent is no longer local to any of your ElectricFlow server nodes.

If anything relies on the local resource being local to the server, it must be modified to work across multiple server nodes, to function remotely from an agent running on another machine, or to be both. In some configurations, you may need to use the shared file system between the ElectricFlow servers and the remote agent.

For example, if you want to write log-parsing procedures using nonlocal agents, you can configure all of your ElectricFlow servers to write their logs to different locations in this shared file system. A procedure step running on a remote agent machine with access to this shared file system can then parse all the server logs.

Database Restriction

The built-in database is not supported in a clustered ElectricFlow configuration. You must use an alternate database listed in Supported Alternate Databases on page 2-12: Oracle, MS SQL Server, or MySQL.

broker-data Directory Restriction

The contents of the DATA_DIRECTORY/broker-data directory can never be shared between nodes in an ElectricFlow cluster.

This may occur when a virtual machine running an ElectricFlow server is cloned, and the DATA_DIRECTORY/broker-data directory is also cloned as part of the cloning operation.

If this occurs, remove the DATA_DIRECTORY/broker-data directory from the new virtual machine (VM) as follows:

1. Shut down the ElectricFlow server on the new VM.
2. Delete the DATA_DIRECTORY/broker-data directory on the new VM.
3. Restart the server on the new VM.

Software for Clustering

Apache Zookeeper

Apache ZooKeeper is a centralized service required for clustering.

- Apache ZooKeeper is a critical part of the clustering architecture. You must use ZooKeeper Version 3.4.6 or later to maintain and synchronize group services in the ElectricFlow cluster. ElectricFlow includes a tool called ZKConfigTool, which you can use to populate ZooKeeper quickly with ElectricFlow configuration information.
- Exhibitor can be used with ZooKeeper to monitor the synchronization between the ZooKeeper nodes. This software is not required to implement an ElectricFlow cluster configuration, but can provide instance monitoring, backup, recovery, cleanup and visualization services. For details, see the Exhibitor documentation.

Load Balancer

You must use a load balancer in an ElectricFlow cluster. You can use any load balancer or load-balancing software for a cluster as long as the load balancer acts as an SSL endpoint and supports returning HTTP

redirections.

> **Note:** Transport Layer Security (TLS) has replaced Secure Sockets Layer version 3.0 (SSLv3) on the ElectricFlow server and the ElectricFlow web server.

Dependencies for Clustering

A clustered configuration has the following minimum requirements:

- Two or more copies of the ElectricFlow server. Clustering is supported starting with ElectricFlow 5.0.
- Two or more ElectricFlow agents.
- At least one ElectricFlow web server on its own machine, or two or more servers if you are using clustering for reliability improvements.
- An enterprise license. The license is required by ElectricFlow to connect to an external database.
- Apache ZooKeeper as the centralized service for maintaining configuration information.
  - ZooKeeper should be installed on a machine without an ElectricFlow server or load balancer. This separation of services is advised to optimize the performance and reliability of your configuration.
  - ZooKeeper must be installed on an odd number of machines. For example, you may need 1, 3, 5, or more instances of the software depending on your environment.
- The ElectricFlow servers must be configured to appear as a single instance in ZooKeeper.
- A hardware load balancer or load-balancing software installed on one machine.
- The web servers, agents, and ElectricFlow servers should share a common file system for plugin information.

Electric Cloud recommends at least one ElectricFlow web server on its own machine or two or more servers if you are using clustering for reliability improvements.

> **Note:** Multiple ElectricFlow clusters can use the same database server, but not the same database schema instance.

Configuring Clustering

There are two different approaches you can take when you configure your ElectricFlow software for horizontal scalability. The approach you choose depends on the needs for your particular ElectricFlow environment for reliability versus performance.

> **Note:** Whichever of the following approaches you choose, you should install multiple ElectricFlow services (agent, web server, ElectricFlow server, and repository) on more than one physical machine (for example, not just virtual machines) to eliminate single points of failure.

- Reliability—Choose this type of configuration if your only concern is redundancy for the ElectricFlow application. This approach requires only the addition of multiple ElectricFlow machines to the server cluster. Multiple ElectricFlow services can reside on a machine, but multiple instances of the service

software should exist. For example, an ElectricFlow server and agent can reside on the same machine as long as other instances of the components exist on different physical machines.

- Performance—Choose this type of configuration if your ElectricFlow cluster will be in a high load environment. This approach requires the installation of ElectricFlow services on a sufficient number of dedicated machines. You should install the ElectricFlow server, repository, web server, and agent services on separate machines from the server nodes to avoid competition for system resources.

> **Note:** You can change a reliability configuration to a performance configuration at a later time, but additional configuration of your ElectricFlow software will be required. For more information, see Separating Agents from ElectricFlow Servers on page 4-34.

Separating Agents from ElectricFlow Servers

Use this procedure if you need to separate ElectricFlow services and agents. By default, an ElectricFlow agent is installed with the ElectricFlow server, web server, and repository. For more information, see Resource, Agent, and Procedure Configuration Considerations on page 4-2 and Verifying ElectricFlow Services on page 4-35.

1. Verify that no ElectricFlow agents are installed on any of the ElectricFlow server nodes. If necessary, remove the agent software from the ElectricFlow server nodes.
2. Verify that none of the ElectricFlow utilities use a local resource. If you are not sure if a local resource is in use, create an agent resource called *local* and monitor the system.
3. Remove the local resource.
4. Create a new agent resource with a new name for each agent on each ElectricFlow server node machine.
5. Create a resource pool named *local* containing all these resources.

Preparing Your Cluster Resources

Before you install any ElectricFlow software, you must complete the following tasks:

1. Identify all machines to be used in the horizontally scalable configuration. It is helpful to have all the network information and machine descriptions available before you begin any work so IP addresses can be used consistently throughout a cluster.
    1. Identify which systems will have a new installation of ElectricFlow and which pre-existing systems will be converted to operate in a cluster. Because traffic between the load balancer and the ElectricFlow server nodes is not encrypted, for security reasons all the ElectricFlow server nodes should be located on the same private network as the load balancer, preferably in the same data center.
    2. Record the IP addresses of:
        - The load balancer machine
        - The machines that will run ZooKeeper
        - The web server machines
        - The ElectricFlow server you will use to import configuration information into ZooKeeper
        - The remaining ElectricFlow server machines that will make up the cluster

3. Record the fully qualified domain name of the load balancer machine. This name will be used in several stages of the configuration process, and should be used consistently throughout the process.

2. Install the load balancer on a machine. For more information, see the instructions from the manufacturer.

3. Install ZooKeeper on an odd number of machines. To eliminate a single point of failure, three or five instances of the software are recommended. For more information, go to Installing ZooKeeper on page 4-9.

Installing and Configuring a Load Balancer

ElectricFlow Server Load Balancer

You must use a load balancer in an ElectricFlow cluster. You can choose to use any hardware load balancer or load-balancing software for a cluster configuration as long as the load balancer can act as an SSL endpoint and support returning HTTP re-directions.

> **Note:** Transport Layer Security (TLS) has replaced Secure Sockets Layer version 3.0 (SSLv3) on the ElectricFlow server and the ElectricFlow web server.

When configuring your load balancer, follow these general guidelines.

- You must configure a load-balancer IP address for each node in your cluster.
- Load-balance traffic on port 8000 across the ElectricFlow servers on port 8000
- Act as an SSL endpoint for port 8443 and load-balance the traffic on that port across the ElectricFlow servers on port 8000.
- Stomp Client URI: ElectricFlow uses STOMP for the following purposes:
  - Transferring log files when you use the EC-FlowLogCollector plugin.
  - Preflights, such as in a CI scenario.
  - Certain API commands such as `waitforjob`

  If you enter `stomp+ssl://FLOW_SERVER_LOAD_BALANCER_OR_IP:61613` into the **Stomp Client URI** field in the ElectricFlow server settings page, this property is used as the URI for stomp clients. For example, `stomp+ssl://myef-lb.electric-cloud.com:61613`. If not present, a default value is calculated using the server's host name.

  The server must be restarted for this setting to take effect.

  > **Tip:**
  > You can also enable this functionality via `ectool`:
  >
  > ```
  > ectool --server localhost setProperty /server/settings/stompClientUri
  > stomp+ssl://EF_SERVER_LOAD_BALANCER_OR_IP:61613
  >
  > ectool --server localhost setProperty /server/settings/stompSecure true
  > ```

- SSL for STOMP:
  - If the load balancer does not act as a SSL endpoint for STOMP port 61613 but instead does SSL pass-through, SSL/TLS bridging or re-encryption, make sure the **Use SSL for Stomp** checkbox (in the **Edit Server Settings** dialog box) is *checked*. This ensures that Commander knows that the STOMP packets are encrypted and will decrypt them.
  - If the load balancer acts as an SSL endpoint (meaning that it does SSL termination) for STOMP port 61613, make sure the **Use SSL for Stomp** checkbox is *unchecked*. This ensures that Commander knows that the load balancer is forwarding STOMP packets unencrypted.
- The load balancer must be configured to perform frequent health check HTTP GET requests for a specific URL and take servers temporarily out of rotation if they receive an HTTP status 503 response. For example, for the ElectricFlow server, this URL can be used: `http://<server-host-name>:<server-http-port>/commanderRequest/health`.
- There are no requirements for the state associated with a user session to be replicated across the cluster.

For an example of how to configure a widely used load balancer such as HAProxy, see the *KBEC-00281 - Configuring Load Balancers in ElectricFlow Clusters* Knowledge Base article.

You can use the previous example as a model for the load balancer configuration in your system and modify it to meet the system requirements of your particular model of load balancer and system configuration.

Electric Cloud does not support any specific load balancer. For information using about HAProxy, go to the HAProxy website. For more information about using another load balancer, go to the website for that load balancer.

> **Note:** If you are using HAProxy and are exporting or importing data in an large XML file, a *504 Gateway Timeout* error, also called an *HTTP 504* error, may occur. You should change the timeout value in the `/etc/haproxy/haproxy.cfg` configuration file from 50 seconds
>
> ```
> clitimeout 50000
> srvtimeout 50000
> ```
>
> to
>
> 10 minutes
>
> ```
> clitimeout 600000
> srvtimeout 600000
> ```

Web Server Load Balancer

The load balancer needs to contain cookie-based sticky sessions to retain the session for the user when accessing a specific web server. This way, future requests from the user will go to the same web server, locking the session to that web server. This will avoid redirect and session issues. For information about setting up sticky sessions, see your load balancer documentation.

Installing ZooKeeper

Use Apache Zookeeper Version 3.4.6 or later to maintain and synchronize group services in a clustered ElectricFlow configuration. For more information, go to the Apache ZooKeeper website.

For your convenience, ZooKeeper 3.4.6 is bundled in your ElectricFlow installation here: `<install dir>/utils`.

To install ZooKeeper:

1. Use either the bundled ZooKeeper or download ZooKeeper from the ZooKeeper website.
2. Extract and install the files into an appropriate location. For example, `/opt/zookeeper-<release_version>`

   > **Note:** You must install ZooKeeper on an odd number of machines. The number of machines will determine if you install ZooKeeper in standalone mode (for one machine) or in replicated mode (for three or five machines).

3. Create a `zoo.cfg` configuration file for each machine with an instance of ZooKeeper. For example, `zookeeper-<release_version>/conf/zoo.cfg`.

   - For standalone mode on a single ZooKeeper machine, the file has these values:

     ```
     tickTime=2000
     dataDir=/var/lib/zookeeper
     clientPort=2181
     ```

   - For replicated mode across multiple ZooKeeper machines, the `zoo.cfg` file on each server should have these values:

     ```
     tickTime=2000
     dataDir=/var/lib/zookeeper
     clientPort=2181
     initLimit=5
     syncLimit=2
     server.1=<ZooKeeper_hostname_1>:2888:3888
     server.2=<ZooKeeper_hostname_2>:2888:3888
     server.3=<ZooKeeper_hostname_3>:2888:3888
     server.4=<ZooKeeper_hostname_4>:2888:3888
     server.5=<ZooKeeper_hostname_5>:2888:3888
     ```

     Where *<ZooKeeper_hostname_1>* through *<ZooKeeper_hostname_5>* are the hostnames of the servers for the ZooKeeper service.

     > **Note:** The file has only three server value lines for a three-ZooKeeper configuration.

   Create `/var/lib/zookeeper/myid` files on each ZooKeeper server, with each containing a single ASCII digit: 1 for the first server, 2 for the second server, and so on, corresponding to their server.`<digit>` values in the `zoo.cfg` files

4. Create `/var/lib/zookeeper/myid` files on each ZooKeeper server, with each containing a single ASCII digit as follows:

- 1 for the first server,
- 2 for the second server
- Up to 5, corresponding to the appropriate server.<*digit*> value in the `zoo.cfg` file.

Running ZooKeeper as a Service on Linux

Install ZooKeeper on machines other than those running ElectricFlow. ZooKeeper must not be run on the same machines as those running the ElectricFlow servers.

To run ZooKeeper as a service, follow these steps:

1. For each machine on which you want to install ZooKeeper, navigate to the `zookeeper-wrapper.zip` file in the ElectricFlow <*install dir*>`/utils` directory.
2. Copy or move the `zookeeper-wrapper.zip` file to each of the machines that you want to use for ZooKeeper.
3. Extract `zookeeper-wrapper.zip` to a directory of your choice.

   For example, `/opt/zookeeper-<release_version>`.
4. Using a text editor, open the `wrapper.conf` file in the extracted `zookeeper-wrapper/conf` directory.

   For example, `/opt/zookeeper-<release_version>/zookeeper-wrapper/conf`.
5. Add the path to the ZooKeeper home directory. For example:

   ```
   # Path to unpacked zookeeper

   set.default.ZOOKEEPER_HOME=/opt/zookeeper-<release_version>
   ```
6. Add the path to the JRE. For example:

   ```
   set.default.JAVA_HOME=/usr/lib/jvm/java-7-openjdk-amd64/jre
   ```

   Now you are ready to start ZooKeeper as a service.
7. Run `zookeeper start`

Running ZooKeeper as a Service on Windows

Install ZooKeeper on machines other than those running ElectricFlow. ZooKeeper must not be run on the same machines as those running the ElectricFlow servers.

To run ZooKeeper as a service, follow these steps:

1. For each machine on which you want to install ZooKeeper, navigateto the `zookeeper-wrapper.zip` file in the ElectricFlow <*install dir*>`\utils` directory.
2. Copy or move the `zookeeper-wrapper.zip` file to each of the machines that you want to use for ZooKeeper.
3. Extract `zookeeper-wrapper.zip` to a directory of your choice.

   For example, `C:\Users\Administrator\zooservice`

4. Using a text editor, open the `wrapper.conf` file located in the extracted `zookeeper-wrapper\conf` directory.

   For example, `C:\Users\Administrator\zooservice\zookeeper-wrapper\zookeeper-wrapper\conf`

5. Add the path to the ZooKeeper home directory.

   For example:

   ```
   # Path to unpacked zookeeper

   set.default.ZOOKEEPER_HOME=C:\Users\Administrator\zooservice\zookeeper-3.4.6
   ```

   Now you are ready to install and start ZooKeeper as a service.

6. Navigate to and click `InstallZooKeeper-NT.bat` to install ZooKeeper as a service.

   The file is in the `zookeeper-wrapper\bin` directory.

   For example, `C:\Users\Administrator\zooservice\zookeeper-wrapper\zookeeper-wrapper\conf`.

7. Navigate to and click `StartZooKeeper-NT.bat` to start ZooKeeper as a service.

   The file is located in the `zookeeper-wrapper\bin` directory.

   For example, `C:\Users\Administrator\zooservice\zookeeper-wrapper\zookeeper-wrapper\conf`.

   If you choose to use the command- line interface or a script to start the service, enter `ZooKeeperCommand.bat start`.

Ensuring that ZooKeeper Can Locate Java

Because ZooKeeper is a Java application, ensure Java is installed and ZooKeeper can locate it. The default value for the `JAVA_HOME` setting (in the `zookeeper-wrapper/conf/wrapper.conf` file) is

```
set.default.JAVA_HOME=/opt/electriccloud/electriccommander/jre
```

If ElectricFlow is not installed or is not installed in its default directory, set `JAVA_HOME` in `wrapper.conf` to the location for Java. For example:

```
set.default.JAVA_HOME=/usr/lib/jvm/java-7-openjdk-amd64
```

Verifying that ZooKeeper is in Standalone Mode

The following example shows how to verify that the Zookeeper service is running in standalone mode:

```
/opt/zookeeper-3.4.5/bin$sudo ./zkServer.sh status
JMX enabled by default
Using config: /opt/zookeeper-3.4.5/bin/../conf/zoo.cfg
Mode: standalone
```

> **Tip:** You can find the Zookeeper `bin` directory by running `ps -ef grep "zoo.cfg"`. This command will display the location of the `zoo.cfg` file. For example, `zookeeper-<release_version>/conf/zoo.cfg`.

If you do not see the status above but see the error below, then ZooKeeper might be configured in replication mode and therefore cannot connect to the other nodes in its ensemble:

```
JMX enabled by default
Using config: /opt/zookeeper-3.4.5/bin/../conf/zoo.cfg
Error contacting service. It is probably not running.
```

The status when ZooKeeper is in replication mode will look something like:

```
Zookeeper version: 3.4.5-1392090, built on 09/30/2012 17:52 GMT
Clients:
10.168.33.13.35821[0](queued=0,recved=1, sent=0)
10.68.33.13.35748[1](queued=0,recved=2189,sent=2189)

Latency min/avg/max: 0/0/86
Received: 2198
Sent: 2197
Connections: 2
Outstanding: 0
Zxid: Ox27758
Mode: standalone
Node count: 29
```

In this case, you must configure Zookeeper in standalone mode and then restart the ZooKeeper service as in the following example:

```
/opt/zookeeper-3.4.5/bin$sudo./zkServer.sh stop
JMX enabled by default
Using config: /opt/zookeeper-3.4.5/bin/../conf/zoo.cfg
Stopping zookeeper ..../zkServer.sh: line 143: kill: (1776) - No such process
STOPPED
```

```
/opt/zookeeper-3.4.5/bin$sudo./zkServer.sh start
JMX enabled by default
Using config: /opt/zookeeper-3.4.5/bin/../conf/zoo.cfg
Starting zookeeper ... STARTED
```

Verifying that ZooKeeper is Running

To check that the ZooKeeper software is running, follow these steps:

1. Log in to each ZooKeeper machine and enter:

   ```
   echo ruok | nc 127.0.0.1 2181
   ```

2. Confirm that you get the following response from each ZooKeeper instance by entering:

   ```
   imok
   ```

   If no response appears or a `broken pipe` error appears, then ZooKeeper is not running.

3. Obtain more information about the status of Zookeeper by logging into each ZooKeeper machine and entering:

   ```
   echo status | nc 127.0.0.1 2181
   ```

Exhibitor Software

After installing ZooKeeper you might want to install the optional Exhibitor software on every machine with an instance of Zookeeper. The Exhibitor software provides a web interface that allows you to

monitor the status of ZooKeeper. It also keeps the configurations of all the ZooKeeper nodes in sync when any of them are changed, and provides tools to rotate and prune the ZooKeeper logs, to prevent them from growing indefinitely. For more information, see the Exhibitor documentation.

If you choose to install Exhibitor, you must configure your ElectricFlow server nodes so they know how to contact Exhibitor. If you have already set up your ElectricFlow server cluster and ZooKeeper servers and are later adding Exhibitor to it, this can be done using the `ecconfigure` tool, which is normally found at `/opt/electriccloud/electriccommander/bin/ecconfigure` on Linux or `C:\Program Files\Electric Cloud\ElectricCommander\bin\ecconfigure.exe` on Windows. Bring down all nodes in your cluster, and run `ecconfigure` on each ElectricFlow node in the cluster with the option (as the user that ElectricFlow runs as, or with administrative privileges):

```
ecconfigure --serverExhibitorConnection <Exhibitor_servers>
```

where `<Exhibitor_servers>` is a comma-separated (no spaces) list of the `IP_address_or_FQDN:port_number` of your three or five (or for a test system, possibly just one) Exhibitor servers (the port number of Exhibitor is normally 8080). For example, 10.0.2.1:8080,10.0.2.2:8080,10.0.2.3:8080 for a three-ZooKeeper/Exhibitor cluster.

It is not necessary to use the command with a single exhibitor and a single ZooKeeper server.

Configuring a Multi-ZooKeeper Cluster

If you plan to use a multi-ZooKeeper cluster, you must configure each ZooKeeper with a unique number from the range 1, 2, 3 for a 3-ZooKeeper cluster or 1, 2, 3, 4, 5 for a 5-ZooKeeper cluster. You must include this number in the following file:

```
<dataDir>/myid
```

where <*dataDir*> is the path you set in your zoo.cfg file.

For example, you can run these commands:

```
sudo touch /var/lib/zookeeper/myid
```

```
sudo -- sh -c 'echo <number> > /var/lib/zookeeper/myid'
```

where <*number*> is the appropriate number between 1 and 3, or 1 and 5.

ZooKeeper Requires a Majority of Nodes to Be Up

ZooKeeper requires a majority of its nodes to be up in order for it to be functional. A majority is:

- 1 of 1
- 2 of 3
- 3 of 5

If a majority of nodes is not up, the expected behavior is a "not currently serving requests" error from ZooKeeper.

Installing ElectricFlow Software

You must install ElectricFlow components on all the nodes in your ElectricFlow cluster. Where you install the individual components depends on the type of cluster configuration you need to create. For more information on how to install ElectricFlow, see Installing ElectricFlow on page 3-1.

Use the reliability approach if you want to minimize single points of failure in your ElectricFlow installation; use the performance approach if (in addition to minimizing single points of failure), you want to maximize throughput of your ElectricFlow server at the cost of using more hardware.

In the reliability approach, other ElectricFlow components such as agents, repositories, and web servers are placed on the same machine as a node of the ElectricFlow server cluster; in the performance approach, they are placed on other servers to leave as many resources as possible available for the ElectricFlow server node.

Choose one of the following four installation approaches for your environment:

- New ElectricFlow Installation for Reliability on page 4-14
- New ElectricFlow Installation for Performance on page 4-15
- Converting an Existing ElectricFlow Installation for Reliability on page 4-16
- Converting an Existing ElectricFlow Installation for Performance on page 4-16

For any of these approaches, when you install agent, repository, and web server services, you can save time by configuring the software to point to a remote server location. You must also register all of these service agents as resources on the ElectricFlow server. For more information, see Duplicating Repository Contents to a New Repository Server on page 4-18.Duplicating Repository Contents to a New Repository Server on page 4-18

- For a command-line installation, set the option `--remoteServer <load_balancer_FQDN>`. If you are doing an advanced installation on Linux, when prompted for the remote ElectricFlow server, enter the <*load_balancer_FQDN*>.
- For a graphical user interface installation, set the Server Host Name field in the "Remote ElectricFlow server" installer page to <*load_balancer_FQDN*>:8000.
  The *load_balancer_FQDN* is the fully qualified domain name of your ElectricFlow server's load balancer machine.

For details about the overall steps for installing DevOps Insight on a group of servers to create a DevOps Insight server cluster, see Creating a DevOps Insight Server Cluster on page 4-36.

New ElectricFlow Installation for Reliability

The reliability approach allows multiple ElectricFlow services to run on a machine, but multiple instances of the service should exist to prevent single points of failure.

1. Install the ElectricFlow server and agent software on one node in the ElectricFlow cluster.

   Note: If you do not already have an ElectricFlow web server that you can temporarily point at this ElectricFlow server node, you might want to also install an ElectricFlow web server that can be used for the following two steps in this section. Before you install the ElectricFlow server and agent software on the remaining nodes in the ElectricFlow cluster, turn off the web server. You turn off the web server on Linux by using the command `/etc/init.d/commanderApache stop`, or on Windows by stopping the service and setting the Startup Type to Manual.

2. Configure ElectricFlow to use an external database. At this time, the ElectricFlow node is in a single-server configuration.

   For more information, see Switching to an Alternate Database from the Built-In Database on page 12-10.

3. Move the plugins directory on the ElectricFlow server software node to a location on the shared file system.

   For more information, see Universal Access to the Plugins Directory on page 5-21.

4. Install the ElectricFlow server and agent software on the remaining nodes in the ElectricFlow cluster.
5. Install the ElectricFlow repository service on one or more machines.
6. Register agents on these machines as resources on the ElectricFlow server.
7. Install the ElectricFlow web server service on one or more machines.

New ElectricFlow Installation for Performance

The performance approach requires separate machines for each ElectricFlow service.

1. Install just the ElectricFlow server software on all the nodes in the ElectricFlow cluster.

   For more information, see Silent Unattended Installation Method on page 3-112.

   > **Note:** If you do not already have an ElectricFlow web server that you can temporarily point at this ElectricFlow server node, you may also want to install an ElectricFlow web server that can be used for the following two steps in this section. Before you install the ElectricFlow server and agent software on the remaining nodes in the ElectricFlow cluster, turn off the web server . You turn off the web server on Linux by using the command `/etc/init.d/commanderApache stop`, or on Windows by stopping the service and setting the Startup Type to Manual.

2. Configure one instance of the ElectricFlow server software to use an external database.

   At this time, the ElectricFlow node is in a single-server configuration.

   For more information, see Switching to an Alternate Database from the Built-In Database on page 12-10.

3. Move the plugins directory on the ElectricFlow server software node to a location on the shared file system.

   For more information, see Moving the Plugins Directory to a Pre-Configured Network Location on page 5-21.

4. Install the following software services on one or more individual machines.

   Each service should not be installed with any other ElectricFlow software components.

   - ElectricFlow agent
   - ElectricFlow repository server
   - ElectricFlow web server

5. Remove any agents that were automatically installed with the ElectricFlow server.

   For more information, see Separating Agents from ElectricFlow Servers on page 4-34 and Verifying ElectricFlow Services on page 4-35.

Converting an Existing ElectricFlow Installation for Reliability

Because this is a conversion of an existing ElectricFlow system, one or more machines with the ElectricFlow server, agent, web server, and repository software already exist. The reliability approach allows multiple ElectricFlow services to run on a machine, but multiple instances of the service should exist to prevent single points of failure.

1. Upgrade the existing ElectricFlow software according to the instructions in Roadmap for Upgrading ElectricFlow on page 6-1.

   Horizontal scalability is supported starting with ElectricFlow 5.0.

2. Verify that ElectricFlow is pointing to an external database.

   To verify which database is in use:

   1. Log in to ElectricFlow.
   2. Select **Administration** > **Database Configuration** to see the current database.

      The database connection is successfully configured if you can log into ElectricFlow.

      See Switching to an Alternate Database from the Built-In Database on page 12-10 if additional configuration is required.

3. Verify that ElectricFlow is configured to use a plugins directory located on the shared file system.

   For more information, see Universal Access to the Plugins Directory.

4. Install the ElectricFlow server and agent software on the remaining nodes for the ElectricFlow cluster.
5. Install the ElectricFlow repository server on one or more machines.
6. Install the ElectricFlow web server on one or more machines.
7. Register the machine agents as resources on the ElectricFlow server.

Converting an Existing ElectricFlow Installation for Performance

Because this is a conversion of an existing ElectricFlow system, one or more machines with the ElectricFlow server, agent, web server, and repository software already exist. The performance approach requires separate machines for each ElectricFlow service.

1. Upgrade the existing ElectricFlow software according to the instructions in Roadmap for the Upgrade Process.

   Horizontal scalability is supported starting with ElectricFlow 5.0.

2. Verify that ElectricFlow is pointing to an external database.

   To verify which database is in use:

   - Log in to ElectricFlow.
   - Select **Administration** > **Database Configuration** to see the current database.

     The database connection is successfully configured if you can log into ElectricFlow.

     See Switching to an Alternate Database from the Built-In Database on page 12-10 if additional configuration is required.

3. Verify that ElectricFlow is configured to use a plugins directory located on the shared file system.

   For more information, see Universal Access to the Plugins Directory.

4. Remove any web server or agent software that is installed with the original ElectricFlow machine.

   This software will be reinstalled on a separate system.

5. Install the ElectricFlow server software on the nodes for the ElectricFlow cluster.

6. Install the following software services on one or more individual machines.

   Each machine should not be installed with any other ElectricFlow software services.

   - ElectricFlow agent
   - ElectricFlow repository server
   - ElectricFlow web server

7. Remove any agents that were automatically installed with the ElectricFlow server, web server, and repository services.

   The original ElectricFlow machine as well as the new installations should be checked to verify that the agent software is removed.

   For more information, see Separating Agents from ElectricFlow Servers on page 4-34 and Verifying ElectricFlow Services on page 4-35

8. Remove any repository server software that is installed with the original ElectricFlow machine after you duplicate the repository server contents.

   For more information, see Duplicating Repository Contents to a New Repository Server on page 4-18.

   If necessary, install the repository software on additional machines.

Configuring Repository Servers

This section describes how to configure repository servers for high availability.

Overall Steps for Configuring Repository Servers

The overall steps for configure repository servers for high availability are as follows:

1. Duplicating Repository Contents to a New Repository Server on page 4-18
2. Configuring Each Existing Repository Server for Clustered Operation on page 4-18

3. Pointing Each Repository Server to the Repository Backingstore on page 4-19
4. Opening the 7800 and 8200 Firewall Inbound Ports on Each Repository Server on page 4-20
5. Registering the Repository to the ElectricFlow Cluster on page 4-20
6. Registering the Repository Local Agents to the ElectricFlow Cluster on page 4-20

To set up repository servers for a typical cluster deployment, complete the following steps.

Duplicating Repository Contents to a New Repository Server

To duplicate the contents of an existing repository server to a new repository server:

1. Stop both repository servers.
2. Copy the entire contents of the repository backingstore directory from the existing repository server to the corresponding location on the new repository server.

   The default location for the backingstore directory (*DATA_DIR*`/repository-data`) is:

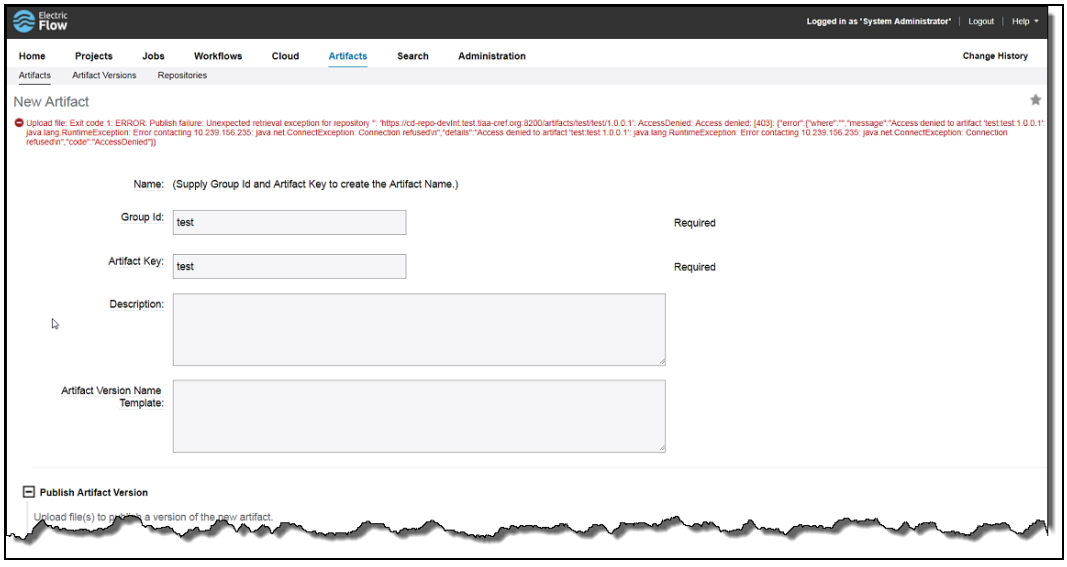
   - UNIX—`/opt/electriccloud/electriccommander/repository-data`
   - Windows—`C:\ProgramData\Electric Cloud\ElectricCommander\repository-data`
3. Restart both repository servers.

Configuring Each Existing Repository Server for Clustered Operation

Additional configuration is required for any repository server that existed before being converted to operate in a cluster or was not configured to point to a remote server location during installation. You must point each of these repository servers to the ElectricFlow server's load balancer machine instead of directly to an ElectricFlow server. To do so, enter the following command on each repository server:

`/opt/electriccloud/electriccommander/bin/ecconfigure --repositoryTargetHostName` *`load_balancer_FQDN`*

This avoids the following AccessDenied error when an artifact is uploaded from the Automation Platform UI:



In the example above, `10.239.156.235` is the IP address of one of the two load-balanced ElectricFlow servers.

To point a repository server to the ElectricFlow server's load balancer machine:

1. Locate the `ecconfigure` tool.
   - On Linux, it is usually at `/opt/electriccloud/electriccommander/bin/ecconfigure`.
   - On Windows, it is usually at `C:\Program Files\Electric Cloud\ElectricCommander\bin\ecconfigure.exe`.
2. Run the tool with the following option on the repository server.

   You might need to do this as root or with administrator privileges:

   `ecconfigure --repositoryTargetHostName` *`load_balancer_FQDN`*

   where *`load_balancer_FQDN`* is the fully-qualified domain name of your ElectricFlow server's load balancer machine.

Pointing Each Repository Server to the Repository Backingstore

In a non-clustered configuration, the repository server is configured to store artifact versions in a directory called the repository backingstore. By default, the backingstore is the *`DATA_DIR`*`/repository-data` directory in the repository installation. In a clustered configuration, you must point each repository server to a common backingstore location.

Windows

In `C:\ProgramData\Electric Cloud\ElectricCommander\conf\repository\server.properties` on each repository server, set `REPOSITORY_BACKING_STORE` to a UNC path to a network share on the file server, and then restart that repository server.

For example, set:

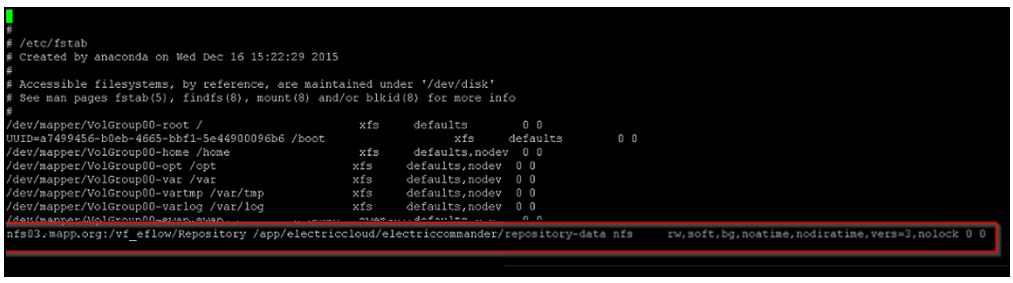
```
REPOSITORY_BACKING_STORE=//10.0.109.72/repo_data/repository-data
```

You can also configure this by running the `ecconfigure --repositoryStorageDirectory` command on each repository server. For example, enter:

```
ecconfigure --repositoryStorageDirectory //10.0.109.72/repo_data/repository-data
```

Linux

If the network file share is Linux, mount it to the *`DATA_DIR`*`/repository-data` on all repository servers. For example, you can mount the file share for *`DATA_DIR`*`/repository-data` in the `/etc/fstab` file on each Linux repository server as follows:



This means that you do not need to run the `ecconfigure –repositoryStorageDirectory` command to change `REPOSITORY_BACKING_STORE` from its `REPOSITORY_BACKING_STORE= repository-data` default value.

Opening the 7800 and 8200 Firewall Inbound Ports on Each Repository Server

On each repository server, make sure that the firewall inbound ports for 7800 and 8200 are open. This allows the load balancer to balance these ports.

Load balancing the repository local agent port 7800 is optional. If you do not do so, you must register each repository local agent in the **Cloud > Resources** page in the Automation Platform UI on the ElectricFlow server.

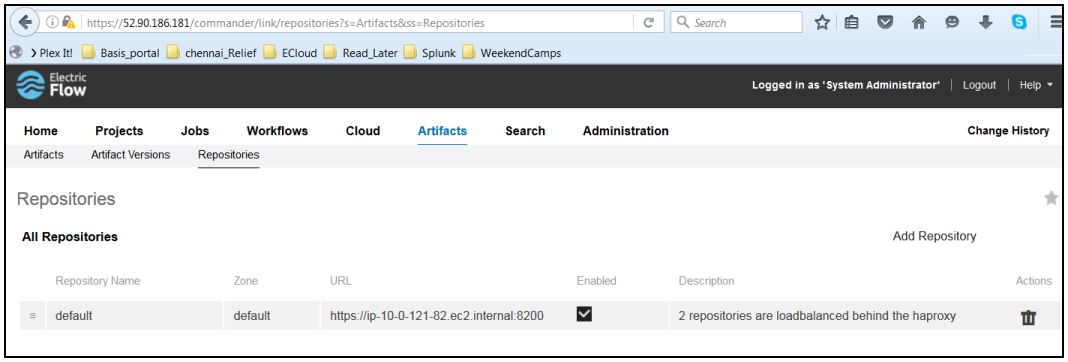
If you load balance the repository local agent port 7800, you can register just the repository local agent load balancer port 7800 in the **Cloud > Resources** page on the ElectricFlow server, but you must "ping" it from the Automation Platform UI several times so that all the agents behind the load balancer are pinged, and then they will know how to contact the ElectricFlow server.

Registering the Repository to the ElectricFlow Cluster

You must use the load balancer URL to register the repository to the ElectricFlow cluster so that the ElectricFlow server can find the repository server load balancer. To do so:

1. In the Automation Platform on the ElectricFlow server, go to the **Artifacts > Repositories** tab.
2. Verify that the repository server URL points to the load balancer machine.

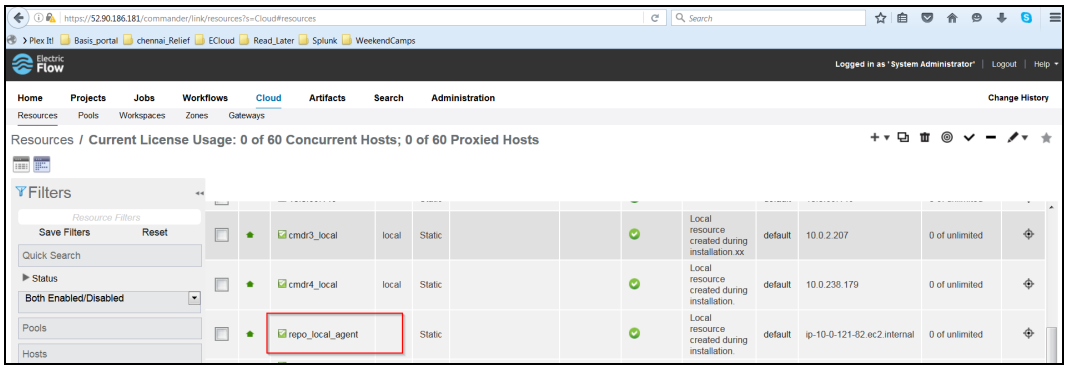
   The following example uses `https://ip-10-0-121-82.ec2.internal:8200` as the load balancer URL:



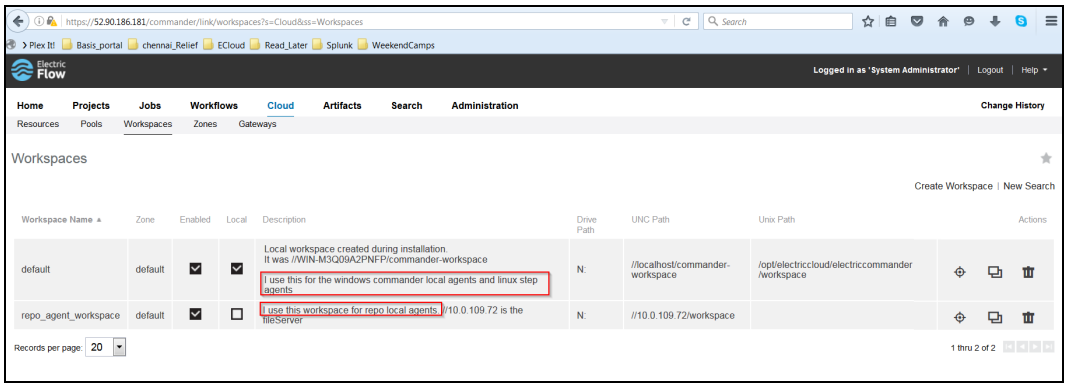
Registering the Repository Local Agents to the ElectricFlow Cluster

You must register the repository local agents to the ElectricFlow cluster. You can do this by registering only the load balancer as the agent resource if you already load-balanced port 7800 of the repository local agents.

If you did not load balance port 7800, you must register each repository local agent in the **Cloud > Resources** page in the Automation Platform on the ElectricFlow server as in the following example:



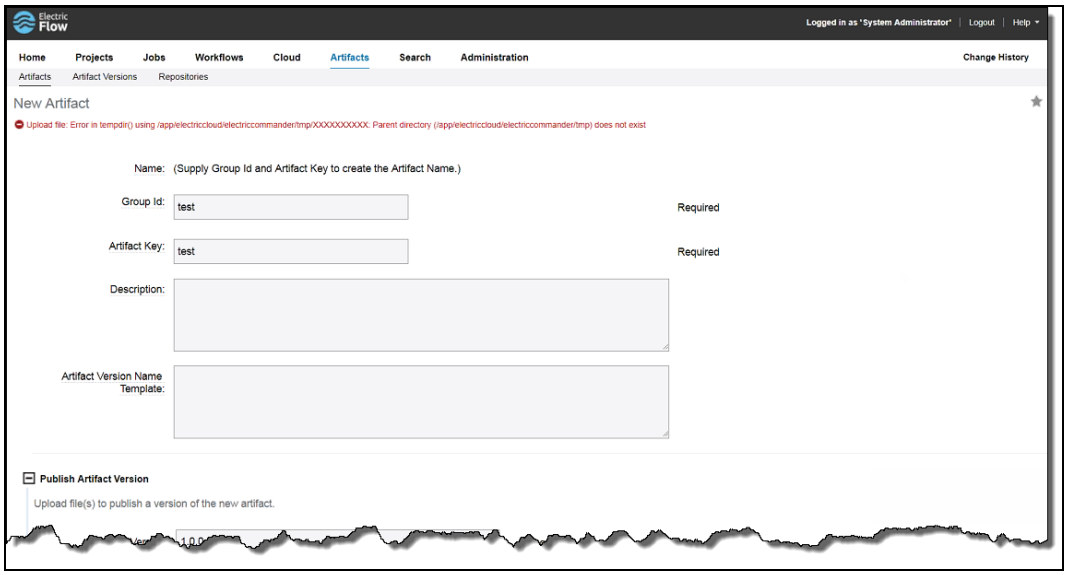
In the example below, a network share workspace for the repository agents on the network file server is registered as a workspace in ElectricFlow (to use it as the workspace for the repository agent). In this example, the windows UNC path is set to `//10.0.109.72/workspace`:



Re-Creating a Deleted *DATA_DIR*/tmp Directory on an ElectricFlow Web Server

Each ElectricFlow web server that is load balanced requires a *DATA_DIR*/tmp directory. When you upload artifacts using the Automation Platform UI, that web server uses this folder as an intermediate location to upload the artifacts to the repository. This means that if the following error appears on a web server during file upload from the UI, you must re-create this folder on that machine:

Upload file: Error in tempdir() using /app/electriccloud/electriccommander/tmp/XXXXXXXXXX: Parent directory (/app/electriccloud/electriccommander/tmp) does not exist



Running a Cluster in Clustered Mode

You must configure the machines installed with ElectricFlow to operate as a cluster.

To configure them in clustered mode:

1. Share the plugins directory across the ElectricFlow servers, agents, and web servers if you have not done so already. For more information, see Universal Access to the Plugins Directory on page 5-21.

2. Configure all the ElectricFlow servers for clustered mode operation, giving them a unique identifier that points to the load balancer, the location of the ZooKeeper servers, and optionally also the Exhibitor servers if you are using Exhibitor.

   You must use the ecconfigure commands on each ElectricFlow server, web server, and agent on which the software component is installed. You must enter the commands on each server that you add. There are no problems if you rerun the commands on a machine. Use one of these methods:

   - Use the ecconfigure tool.
     - Locate the ecconfigure tool.

       For Linux, it is usually at `/opt/electriccloud/electriccommander/bin/ecconfigure`.

       For Windows, it is usually at `C:\Program Files\Electric Cloud\ElectricCommander\bin\ecconfigure.exe`.

     - Run it with the following options on each ElectricFlow server node (as the user that ElectricFlow runs as, or with administrative privileges):

       ```
       ecconfigure --serverName <load_balancer_FQDN>
       --serverZooKeeperConnection <ZooKeeper_servers>
       ```

   - Use Exhibitor and enter:

     ```
     ecconfigure --serverName <load_balancer_FQDN>
     --serverZooKeeperConnection <ZooKeeper_servers>
     --serverExhibitorConnection <Exhibitor_servers>
     ```

   where

   - *<load_balancer_FQDN>* is the fully-qualified domain name of your load balancer machine.
   - *<ZooKeeper_servers>* is a comma-separated (no spaces) list of the IP_address_or_FQDN:port_number for each of your three or five (or for a test system, possibly just one) ZooKeeper servers (the port number for ZooKeeper is normally 2181).
   - If you are using Exhibitor, *<Exhibitor_servers>* is a comma-separated (no spaces) list of the IP addresses or fully-qualified domain names of your three or five (or for a test system possibly just one) Exhibitor servers . The port number that ElectricFlow uses to connect to Exhibitor is always port 80.) For example, 10.0.2.1,10.0.2.2,10.0.2.3 for a three-ZooKeeper/Exhibitor cluster.

   Example for a three-ZooKeeper cluster:

   ```
   ecconfigure --serverName machine.company.com --serverZooKeeperConnection
   10.0.2.1:2181,10.0.2.2:2181,10.0.2.3:2181
   ```

   These ecconfigure commands start the ElectricFlow server nodes.

- Configure the load balancing software for the ElectricFlow server cluster and the ElectricFlow web servers.
    - Stop all ElectricFlow server nodes that you want to cluster.
    - Back up the `<data_dir>\conf` directory on all ElectricFlow servers nodes.
    - Choose an ElectricFlow server node (usually the first node) from which to copy the `<data_dir>\conf` directory to the other nodes.
    - Empty the contents of the `<data_dir>\conf` directory on the other nodes.
    - Copy the contents of `<data_dir>\conf` from the first node to `<data_dir>\conf` on the other nodes. This ensures that `<data_dir>\conf` on all clustered ElectricFlow servers is identical.

Running a Cluster in Single-Server Mode

A couple of rarely-used ElectricFlow operations are not supported in clustered mode. If you need to perform any of the following operations, you must restart the server in single-server mode:

- Changing the database configuration
  This operation can be done in the web UI using **Administration** > **Database Configuration** or with ectool. After performing this operation, you must rerun ZKConfigTool to upload the changed configuration from the local `database.properties` file to ZooKeeper before switching back to clustered mode.
- Doing a full import using ectool
- Doing a full export using the safe mode with either of these options:
    - Restart
    - Shutdown

To run an ElectricFlow machine in single-server mode:

1. Identify the machine you need to work on and shut down the other machines in the ElectricFlow cluster.
2. Verify that the database.properties, keystore, passkey, and commander.properties configuration files on the machine you are working on are up to date and match those that were uploaded to ZooKeeper.
3. Switch this remaining ElectricFlow machine to single-server mode by running the following command:

   ```
   ecconfigure --serverEnableClusteredMode=0
   ```

   The commanderAgent and commanderServer services restart.
4. Complete your work on the ElectricFlow machine.

5. If you changed the database configuration, use ZKConfigTool to upload the updated database.properties file to ZooKeeper.

   If you used the `eccert` tool (for added trusted agents, revoked certificates, and so on), use ZKConfigTool to upload the updated keystore file and/or `conf/security` folder to ZooKeeper.

   If you updated the `commander.properties` file, use ZKConfigTool to upload it to ZooKeeper.

   See Uploading Configuration Files to ZooKeeper on page 4-25 for more information.

6. Switch this remaining ElectricFlow machine back to clustered mode by running the following command:

   ```
   ecconfigure --serverEnableClusteredMode=1
   ```

7. Start all the other nodes in the ElectricFlow cluster back up

Adding the Configuration to ZooKeeper

After performing the steps in Running a Cluster in Clustered Mode on page 4-22, complete the following steps:

1. Confirm that the *<data_dir>*`\conf` directories on all ElectricFlow servers to be clustered are identical.
2. Use ZKConfigTool to upload configuration files from the existing ElectricFlow server to ZooKeeper by using the steps in Uploading Configuration Files to ZooKeeper on page 4-25.

Uploading Configuration Files to ZooKeeper

Before starting the ElectricFlow server cluster, you must populate your Apache ZooKeeper server with ElectricFlow database configuration information that all ElectricFlow server nodes will use in the cluster. You use `ZKConfigTool` to import this information into your ZooKeeper server.

> **Note:** You must run `ZKConfigTool` after changing any updatable configuration file. Even with multiple ZooKeeper machines, you must do this only once, and the data is propagated to all of those machines.

The following minimum set of files from the *<data_dir>*`\conf` directory is imported:

- `database.properties`
- `keystore`
- `passkey`
- `commander.properties`

Prerequisites

- The ElectricFlow server package must be installed on the system.
- The system must be running a version of Java supported by ElectricFlow. Java is automatically installed with ElectricFlow as part of the Tools installation.
- The ZooKeeper software must be installed on the network.

Location

ElectricFlow includes `ZKConfigTool`. It is installed in the following default locations:

- Windows:
  `C:\Program Files\Electric Cloud\ElectricCommander\server\bin\zk-config-tool-jar-with-dependencies.jar`
- Linux:
  `/opt/electriccloud/electriccommander/server/bin/zk-config-tool-jar-with-dependencies.jar`

Command

`ZKConfigTool` is best run from the `<install_dir>/conf` directory.

```
$ java -jar zk-config-tool-jar-with-dependencies.jar <options>
```

Option	Description
`--commanderPropertiesFile <path_to_file>`	Import the ElectricFlow Server `commander.properties` file.
`--confSecurityFolder <path_to_folder>`	Import the ElectricFlow Server `conf/security` folder.
`--databasePropertiesFile <path_to_file>`	Import the ElectricFlow Server `database.properties` file.
`--help`	Show the command help.
`--keystoreFile <path_to_file>`	Import the ElectricFlow Server `keystore` file.
`--<path_to_file>`	Import the ElectricFlow Server `passkey` file.
`--readFile<path_on_zookeeper><path_to_file>`	Read the specified file from the ZooKeeper server.
`--readFolder <path_on_zookeeper> <path_to_folder>`	Read the specified folder from the ZooKeeper server.
`--version`	Show the version number of the `ZKConfigTool` tool.

Option	Description
`--writeFile <path_on_ zookeeper><path_to_file>`	Write the specified file to the ZooKeeper server.
`--writeFolder <path_on_zookeeper> <path_to_folder>`	Write the specified folder to the Zookeeper server.

Import Files

Run `ZKConfigTool` to populate the ZooKeeper server with configuration information. The system must have the ElectricFlow tools installed and must be able to communicate with ZooKeeper.

The following command shows how to run `ZKConfigTool` from the `<data_dir>\conf` directory:

Linux:

```
../jre/bin/java -DCOMMANDER_ZK_CONNECTION=<ZooKeeper_Server_IP>:2181 -jar
../server/bin/zk-config-tool-jar-with-dependencies.jar
com.electriccloud.commander.cluster.ZKConfigTool --databasePropertiesFile
database.properties --keystoreFile keystore --passkeyFile passkey --
commanderPropertiesFile commander.properties --confSecurityFolder security
```

Windows:

```
"C:\Program Files\Electric Cloud\ElectricCommander\jre\bin\java.exe" -DCOMMANDER_ZK_
CONNECTION=<ZooKeeper_Server_IP>:2181 -jar "C:\Program Files\Electric
Cloud\ElectricCommander\server\bin\zk-config-tool-jar-with-dependencies.jar"
com.electriccloud.commander.cluster.ZKConfigTool --databasePropertiesFile
database.properties --keystoreFile keystore --passkeyFile passkey --
commanderPropertiesFile commander.properties --confSecurityFolder security
```

Next Steps

After you upload the new configuration files:

1. Shut down all running ElectricFlow servers in the cluster.
2. Start one ElectricFlow server in the cluster.
3. Check if the ElectricFlow server is fully running by using the following ectool command:

   ```
   ectool --server localhost --timeout 900 getServerStatus --block 1 --
   serverStateOnly 1
   ```

   This command runs for 900 seconds (15 minutes) or until `getServerStatus` displays either `bootstrap` or `running`.
4. If the output says `bootstrap`, enter the command again until it says `running`.

   You can also "tail" the `<data_dir>\logs\commander-<hostname>.log` file to check for errors that could prevent the ElectricFlow server from going to `running` state.
5. After the first ElectricFlow server is in `running` state, start the other ElectricFlow servers so that they can join the cluster.

After all ElectricFlow servers are in `running` state, you can check the "view" of the cluster.

- On Linux platforms, enter:

  ```
  <install_dir>/bin/ectool --server localhost getServerStatus --diagnostics 1 |
  grep -i "\(<view>\|<service_name>\|<participants>\)"
  ```

- On Windows platforms, enter:

  ```
  <install_dir>\bin\ectool --server localhost getServerStatus --diagnostics 1 |
  findstr "<view> <service_name> <participants>" 2>NULL
  ```

Getting information on the ElectricFlow Server Cluster from ZooKeeper

Use `ClusterInfoTool` to get information on the running ElectricFlow server cluster from ZooKeeper.

Prerequisites

- The ElectricFlow server cluster must be installed and running on the network.
- Configuration files that all ElectricFlow server nodes will use in a clustered configuration must be uploaded to Apache ZooKeeper server using the `ZKConfigTool`.
- The ZooKeeper cluster must be running an odd number of Zookeeper nodes and there must be a leader node.
- The system must be running a version of Java supported by ElectricFlow. Java is automatically installed on a system with the ElectricFlow software as part of the Tools installation.

Locations

The ElectricFlow installer adds the `ClusterInfoTool` to the following default locations:

- Windows: `C:\Program Files\Electric Cloud\ElectricCommander\server\bin\cluster-info-tool-jar-with-dependencies.jar`
- Linux: `/opt/electriccloud/electriccommander/server/bin/cluster-info-tool-jar-with-dependencies.jar`

Command

```
$ java -DCOMMANDER_ZK_CONNECTION=<ZooKeeper_Server1_IP>:2181,<ZooKeeper_Server2_
IP>:2181,<ZooKeeper_Server3_IP>:2181 -jar cluster-info-tool-jar-with-dependencies.jar
```

where `DCOMMANDER_ZK_CONNECTION` must point to the ZooKeeper system to which the ElectricFlow server cluster is connected.

Output

This is sample output generated by `ClusterInfoTool`:

```
Checking /commander/jgroups/hornetq:

        582d3642-9736-e87f-4e19-c22d7776ccab ->

                WIN-M3Q09A2PNFP-21066   6c80e9a9-2fed-6352-e437-fe76b65aa80d
                10.0.175.78:5446    F
                WIN-M3Q09A2PNFP-28728   0afa39df-d072-b05a-034a-aed74b7a39ee
                10.0.238.179:5446   F
                WIN-M3Q09A2PNFP-28295   c69f0430-df04-bf24-1294-b471a4a3f151
                10.0.2.207:5446     F
```

```
                WIN-M3Q09A2PNFP-15869   582d3642-9736-e87f-4e19-c22d7776ccab
                10.0.2.206:5446     T


Checking /commander/jgroups/commander:

        0c32103a-d4e7-da91-f8c4-1fd8e125c156 ->

                WIN-M3Q09A2PNFP-19713   0c32103a-d4e7-da91-f8c4-1fd8e125c156
                10.0.2.206:5447     T
                WIN-M3Q09A2PNFP-14432   75afe3eb-7a04-d160-c8d0-d64f8ac3c796
                10.0.175.78:5447    F
                WIN-M3Q09A2PNFP-33530   1cf1795f-e658-3cae-c515-546082bffec9
                10.0.238.179:5447   F
                WIN-M3Q09A2PNFP-60743   cf8b4201-3fac-71c4-18da-0a885b3e1e61
                10.0.2.207:5447     F
```

How to interpret `ClusterInfoTool` output:

- The nodes `/commander/jgroups/hornetq` and `/commander/jgroups/commander` contain information on these JGroups clusters:
  - `commander` for the ElectricFlow server cluster
  - `hornetq` for the HornetQ cluster
- The children nodes under each of the JGroups nodes represent the participating ElectricFlow servers in the cluster. Each child node entry is in this form:

  <Logical_Name> *<UUID>* *<IP_address>*:*<port>* T|F
- The number of entries in both JGroups nodes should be the same, with matching IP addresses but with different port numbers and distinct logical names and UUIDs. The coordinator node in each JGroups cluster is identified with a 'T' against its entry.

Configuring Web Server Properties

You must update the `httpd.conf` file on each web server in the cluster. The `httpd.conf` file is usually in `apache/conf` on a Linux machine and `ProgramData\Electric Cloud\ElectricCommander\apache\conf` on a Windows machine.

To configure all the web servers for clustered mode operation and give them the name of the load balancer:

1. Locate the `ecconfigure` tool.

   On Linux, it is usually at `/opt/electriccloud/electriccommander/bin/ecconfigure`.

   On Windows, it is usually at `C:\Program Files\Electric Cloud\ElectricCommander\bin\ecconfigure.exe`.

2. Run the tool with the following option on each web server:

   ```
   ecconfigure --webTargetHostName <load_balancer_FQDN>
   ```

   where <*load_balancer_FQDN*> is the fully qualified domain name of your ElectricFlow server's load balancer machine.

   The `--webTargetHostName` argument modifies the ElectricFlow web server configuration and therefore also attempts to restart the ElectricFlow web server.

3. (Optional additional step for load-balanced web servers) Run the tool with the following option on each load-balanced web server:

   ```
   ecconfigure --webHostName <load_balancer_of_webserversFQDN>
   ```

   where <*load_balancer_ofwebservers_FQDN*> is the fully qualified domain name of your web server's load balancer machine.

   The `--webHostName` argument modifies the ElectricFlow web server configuration and therefore also attempts to restart the ElectricFlow web server.

4. On each web server or load-balanced web server, if you used the `ecconfigure` command without `sudo` as recommended, the `commanderApache` service will not start and produces an error. Therefore, restart it manually afterward using `sudo`.

   You can also use the `--skipServiceRestart` argument to avoid the `ecconfigure` command's restart attempt and the restart error message. But you must separately restart each web server as `sudo` to effect these changes.

5. After configuring the web servers, configure the web server load balancer to use "sticky sessions." For details, see Installing and Configuring a Load Balancer on page 4-7.

Configuring ElectricFlow Agents

You must configure ElectricFlow agents to function within a resource pool.

> **Note:** Transport Layer Security (TLS) has replaced Secure Sockets Layer version 3.0 (SSLv3) on the ElectricFlow server and the ElectricFlow web server.

1. Start and log in to ElectricFlow.
2. Go to the **Cloud** > **Resources** page.
3. Delete any resource named *local*.
4. Create a resource pool named *local*.
5. Create resources for all the machines that had the ElectricFlow agent software installed.

6. Add agent resources to the local resource pool.

   Choose the appropriate step for your approach.

   - If you are creating a reliability configuration where each ElectricFlow server machine also has an agent installed, the local resource pool should consist of the set of agents local to the ElectricFlow server machines.
   - If you are creating a performance configuration where none of the ElectricFlow server machines have agents installed, the local resource pool (which in this configuration is not actually local to the servers) should contain several agents to handle any work that may be assigned. For example, work may be assigned because of old default resource settings.

7. Verify that the default resource pool contains two or more resources for reliability.

   You must create and add resources to this pool if none exist.

8. Go to the **Administration** > **Server** page.
9. Click on the **Settings** link near the top right to open the Edit Server Settings form.
10. Set the **Server IP address** entry in the form to the fully qualified domain name of your ElectricFlow server's load balancer.

    This setting controls how agents contact the ElectricFlow server when they send results from jobs and similar prompts.

Configuring the Cluster Workspace

You must edit the default workspace for log files across the ElectricFlow servers, agents, and web servers.

> **Important:** As you need and create more workspaces over time, each workspace should be in a shared network location that all machines in the ElectricFlow cluster can access.

To edit the default workspace:

- Select **Cloud** > **Workspaces**.
- Edit the default workspace entry to reference a shared network location.

  For more information, see the "Workspaces and Disk Space Management" section in the "Automation Platform" chapter of the *ElectricFlow User Guide* at http://docs.electric-cloud.com/eflow_doc/FlowIndex.html.

Adding Trusted Agents to Clusters

Perform the following procedures in these situations:

- Every time that you create trusted agents.
- Every time that you revoke the certificates of existing trusted agents.
- To create a gateway to a zone with trusted agents at both ends. You have to do this only a few times.
- When you want all agents in cluster to be trusted.

You can select one of these methods to add a trusted agent to a cluster:

- Before adding the trusted agent, shut down all but one node in the cluster. Go to Method 1 on page 4-33.
- Keep most of the nodes up while adding trusted agents. Go to Method 2 on page 4-34.

Preparing Your Cluster Environment

Perform the following steps one time when configuring the cluster to work with trusted agents. You can skip these steps if you have already done them.

1. Select the server node. This should be the node you used to upload configuration files to ZooKeeper while performing the procedure in Uploading Configuration Files to ZooKeeper on page 4-25.
2. Confirm that the certificate files from the node you selected and from ZooKeeper match by using the `verifyClusterCertificate.pl` script. For detailed instructions and the script output, see the *KBEC-00342 - Using the verifyClusterCertificate.pl script for Trusted Agents* Knowledge Base article.

   In the script output, look for `SUCCESS` to confirm that certificate files match. If this fails, you might be on the wrong node.
3. Replace the `conf/security` folder on all nodes with the `conf/security` folder from node selected in Step 1.
4. If you want to add a trusted agent to a cluster and it is OK to shut down all but one node in the cluster, go to Method 1 on page 4-33.
5. If you want to keep most of the nodes up while adding trusted agents, perform the rest of the steps in this procedure.
6. (Windows platforms) Change directories to the `<data_dir>/conf/` folder.
7. Upload the `conf/security` folder from the node you selected to ZooKeeper by using the following command:

   - Linux:

     ```
     COMMANDER_ZK_CONNECTION=<ZooKeeper_Server_IP>:2181 <install_
     dir>/jre/bin/java -cp <install_dir>/server/bin/zk-config-tool-jar-with-
     dependencies.jar com.electriccloud.commander.zkconfig.ZKConfigTool --
     writeFolder /commander/conf/security <data_dir>/conf/security
     ```

   - Windows:

     ```
     "C:\Program Files\Electric Cloud\ElectricCommander\jre\bin\java.exe" -
     DCOMMANDER_ZK_CONNECTION=<ZooKeeper_Server_IP>:2181 -jar "C:\Program
     Files\Electric Cloud\ElectricCommander\server\bin\zk-config-tool-jar-with-
     dependencies.jar" com.electriccloud.commander.cluster.ZKConfigTool --
     confSecurityFolder security
     ```

8. On *all* cluster nodes:
    1. Open `wrapper.conf` in the `<data_dir>/conf` directory.
    2. Uncomment the following line:

       ```
       wrapper.java.additional.603=-DCLUSTER_CERTIFICATE_SERVICE_USE_
       ZOOKEEPER=true
       ```

    3. Make sure that `DCLUSTER_CERTIFICATE_SERVICE_USE_ZOOKEEPER` is set to `true`.

    Ensure that there is no conflict with number 603 and that it is not already used in the system.
9. Restart the nodes with updated configurations.
10. Go to Method 2 on page 4-34.

Method 1

Follow these steps to add a trusted agent to a cluster by first shutting down all but one node in the cluster. Perform the tasks from Step 2 to Step 4 on the agent machine.

1. Shut down all but one node in the cluster.
2. On the machine with an agent that you want to make a trusted agent, enter commands such as the following to create the trusted agent:
    1. `ectool --server <Server_host> login admin changeme`
       to log into the server with the specified hostname of IP address and save the session ID.
    2. To make a remote agent trusted, enter:

       `<install_dir>/bin/eccert initAgent --remote --force`

       to generate a certificate request for this agent, send a certificate authority (CA) request to the ElectricFlow server (the CA), receive a signed certificate from the CA for this agent, and add the CA certificate and the agent's private key (also signed by the CA) to the agent's keystore.

       To make a local agent (that is, local to the ElectricFlow server) trusted, enter:

       `<install_dir>/bin/eccert initAgent --remote --force`

       The ElectricFlow server keeps a copy of the signed agent certificate in the `$install_dir/conf/security/certs` directory.

       Do not use `eccert` as `sudo`, which would change the ownership of the configuration files such as the keystore file to the root user. These files must be owned by the user who starts the ElectricFlow services.
3. On the agent machine, enter

   `ectool createResource <agent_name> --hostName <agent_FQDN_or_IP> --trusted true`

   to add the agent as a trusted agent to the ElectricFlow server in the previous step, where `<agent_FQDN_or_IP>` is the fully-qualified domain name or IP address of the agent.
4. Restart the agent on the agent machine.

Method 2

Perform the following steps on an agent machine to add a trusted agent to a cluster without shutting down server nodes. This procedure works only in ElectricFlow 6.3 or later.

Due to limitations in ZooKeeper, using this method imposes a maximum of around 500 signed certificates. If you want to use this method and are likely to need more than 500 trusted agents, we recommend re-using a certificate across multiple trusted agents.

1. On the machine with an agent that you want to make a trusted agent, enter commands such as the following to create the trusted agent:

    1. `ectool --server <Server_host> login admin changeme`
    to log into the server with the specified hostname of IP address and save the session ID.

    2. To make a remote agent trusted, enter:

        `<install_dir>/bin/eccert initAgent --remote --force`

        to generate a certificate request for this agent, send a certificate authority (CA) request to the ElectricFlow server (the CA), receive a signed certificate from the CA for this agent, and add the CA certificate and the agent's private key (also signed by the CA) to the agent's keystore.

        To make a local agent (that is, local to the ElectricFlow server) trusted, enter:

        `<install_dir>/bin/eccert initAgent --remote --force`

        The ElectricFlow server keeps a copy of the signed agent certificate in the `$install_dir/conf/security/certs` directory.

2. On the agent machine, enter

    `ectool createResource <agent_name> --hostName <agent_FQDN_or_IP> --trusted true`

    to add the agent as a trusted agent to the ElectricFlow server in the previous step, where `<agent_FQDN_or_IP>` is the fully-qualified domain name or IP address of the agent.

3. Restart the agent on the agent machine.

Separating Agents from ElectricFlow Servers

Use this procedure if you need to separate ElectricFlow services and agents. By default, an ElectricFlow agent is installed with the ElectricFlow server, web server, and repository. For more information, see Resource, Agent, and Procedure Configuration Considerations on page 4-2 and Verifying ElectricFlow Services on page 4-35.

1. Verify that no ElectricFlow agents are installed on any of the ElectricFlow server nodes. If necessary, remove the agent software from the ElectricFlow server nodes.
2. Verify that none of the ElectricFlow utilities use a local resource. If you are not sure if a local resource is in use, create an agent resource called *local* and monitor the system.
3. Remove the local resource.
4. Create a new agent resource with a new name for each agent on each ElectricFlow server node machine.
5. Create a resource pool named *local* containing all these resources.

Verifying ElectricFlow Services

You can verify what services are on a machine installed with ElectricFlow software by the following methods:

- If you have a Linux system
  1. Go to the `/etc/init.d/` directory.
  2. Look for scripts starting with "commander". For example, commanderAgent, commanderApache, commanderRepository, commanderServer.
  3. As root, from any directory, use this command format:

     `/etc/init.d/<service_name> status`

     where <service_name> is the ElectricFlow service you are interested in, such as

     `/etc/init.d/commanderServer status`
- If you have a Windows system:
  1. Go to the Services control panel.
  2. Look for services starting with the name "ElectricFlow". For example, ElectricFlow Agent, ElectricFlow Database, or ElectricFlow Server.
  3. If the services have a status of Started, they are installed and running.

Accessing ElectricFlow with Clustering

You access an ElectricFlow server in a clustered configuration the same way you would for a single-server configuration.

Enter the address of one of the web servers into your browser address bar.

If you are using ectool, use the `--server` option to direct your request to the fully qualified domain name of the load balancer.

Health Check for the ElectricFlow Cluster

In a clustered configuration, it is important that all ElectricFlow servers that are set up to participate in the cluster can communicate with ZooKeeper and with each other though JGroups. So each ElectricFlow server runs a periodic critical services health check to test that it has a valid session with ZooKeeper and that the server is part of the JGroups cluster for ElectricFlow.

The health check is run every minute by default. If it fails after five repeated attempts, the server goes into bootstrap mode so that it cannot serve any requests. Once in bootstrap mode, the server will periodically attempt to reinitialize its services. If it is successful and is able to join the JGroups cluster, it can serve incoming requests again.

Additional Ways to Improve an ElectricFlow Cluster

Clustering your ElectricFlow configuration does not necessarily remove all single points of failure or potential performance bottlenecks from your system. There are other components that are part of the ElectricFlow environment that you must consider to eliminate single points of failure or prevent performance bottlenecks.

Third-Party Software

The following items are widely-used third-party commercial products that are used in conjunction with ElectricFlow. A variety of solutions and strategies to increase the reliability and scalability of these products and eliminate remaining single-points-of-failure are available from other vendors and sources.

- Network
- Load balancer
- External database
- File server used for the shared file system

ElectricFlow Components

You can address some or all of the following potential issues with the following ElectricFlow components. The issues you address depends on the level of reliability and performance you need for your system.

- Repository server—You can mirror your artifacts across multiple repository servers.
- ElectricFlow procedures or steps—You can specify a resource pool of agents rather than a single agent.

Creating a DevOps Insight Server Cluster

The DevOps Insight server uses the Elasticsearch search engine and the Logstash data-collection and log-parsing engine to gather data from the ElectricFlow server for use in the DevOps Insight dashboards such as the Deployments, Releases, and Release Command Center dashboards. Elasticsearch is a powerful search and analysis engine, and part of this power lies in the ability to scale for better performance and stability.

An Elasticsearch cluster is a collection of one or more nodes (servers) that holds your entire data and provides federated indexing and search capabilities across all nodes. A cluster is identified by a unique name, which is set using the **Elasticsearch Cluster name** provided to the DevOps Insight server installer. This name is important, because a node can be part of a cluster only if the node is set up to join the cluster by its name.

A node is a single server that is part of your cluster, stores your data, and participates in the cluster's indexing and search capabilities. Just like a cluster, a node is identified by a name, which is set using the **Elasticsearch Node name** provided to the DevOps Insight server installer.

A cluster can have any number of nodes. At least three nodes are recommended for a cluster.

Overall Steps for Creating a Typical DevOps Insight Cluster

The overall steps for creating a typical DevOps Insight cluster are as follows:

1. Planning the Total Number of Master-Eligible Nodes on page 4-37
2. Choosing the Security Mode for the Cluster on page 4-37
3. Installing the First Node in the Cluster on page 4-37
4. Installing Each Additional Node on page 4-37
5. Configuring the Load Balancer and the ElectricFlow Server on page 4-38

To set up a typical DevOps Insight cluster deployment, complete the following steps.

1. Planning the Total Number of Master-Eligible Nodes

The master node is responsible for lightweight cluster-wide actions such as creating or deleting an index, tracking which nodes are part of the cluster, and deciding which shards to allocate to which nodes. It is important for the cluster health to have a stable master node. Any master-eligible node may be elected to become the master node.

To prevent data loss in case of network failure, the minimum number of master-eligible nodes that must be visible in the cluster must be set to a quorum of master-eligible nodes:

(Number of master-eligible nodes in the cluster / 2) + 1

For example, in a cluster with three master-eligible nodes, the minimum number of master-eligible nodes should be set to 2.

This value must be used during installation for every node in the cluster, and it must be the same for all nodes.

2. Choosing the Security Mode for the Cluster

If the mode will be password protected, then you must choose a password. This password *must* be used during installation for every node in cluster and it *must* be the same for all nodes. If the cluster will not be password protected, then all nodes *must* be installed using the same mode.

3. Installing the First Node in the Cluster

Use one of the following installer modes:

- GUI mode—In the **Cluster Settings** installer screen, check the **This is the first node in the cluster** checkbox.

  For details about this screen and when it appears during the installation session, see Running an Advanced Graphical User Interface Installation on page 3-23.

- Console mode—Answer **Yes** at the **Do you want to specify additional Elasticsearch cluster mode settings?** prompt and at the **Is this node the first node to be installed in the Elasticsearch cluster?** prompt.

  For details about when these prompts appear during the installation session, see Running an Advanced Command-Line Installation on page 3-89.

- Silent mode—Additional arguments are not needed (cluster mode is the default).

Make sure that you specify the minimum number of master nodes in the cluster as well as the password from 1. Planning the Total Number of Master-Eligible Nodes on page 4-37 and 2. Choosing the Security Mode for the Cluster on page 4-37.

This node will be a master node and a data node automatically.

Retrieve the certificate file containing a CA-signed certificate for the ElectricFlow DevOps Insight server, which is located at `<DATA_DIR>/conf/reporting/elasticsearch/signing-ca.p12`. This file is needed for installing the DevOps Insight server on all other cluster nodes.

4. Installing Each Additional Node

Perform this step on each additional node. These settings should be the same across all additional nodes. Use one of the following installer modes:

- GUI mode—In the **Cluster Settings** pane, uncheck the **This is the first node in the cluster** checkbox. For details about this screen and when it appears during the installation session, see Running an Advanced Graphical User Interface Installation on page 3-23.

  Then specify the certificate file containing a CA-signed certificate retrieved from the first node on the **Advanced Settings** pane.

- Console mode—Answer **Yes** at the **Do you want to specify additional Elasticsearch cluster mode settings?** prompt and **No** at the **Is this node the first node to be installed in the Elasticsearch cluster?** prompt.

  For details about when these prompts appear during the installation session, see Running an Advanced Command-Line Installation on page 3-89.

  Then specify the certificate file containing a CA-signed certificate retrieved from the first node at the **PKCS#12 file containing a CA-signed certificate for the ElectricFlow DevOps Insight Server** prompt.

  > **Note:** You can leave this entry blank for a new installation in non-clustered mode or for the first node in clustered mode. In this case, the installer will generate a new self-signed certificate and will use it to sign other TLS certificates.

- Silent mode—Use the `--elasticsearchNodeAdditional` command line argument and specify the certificate file containing a CA-signed certificate retrieved from the first node using the `--elasticsearchCACertificateFile` command line argument.

Make sure that you specify the minimum number of master nodes in the cluster as well as the password from 1. Planning the Total Number of Master-Eligible Nodes on page 4-37 and 2. Choosing the Security Mode for the Cluster on page 4-37.

> **Note:** The cluster name *must* be the same for all nodes.
>
> **Note:** During node installation, the list of other nodes should be specified. It is mandatory for additional nodes and optional for the first node. You should specify all available master nodes in this list.

5. Configuring the Load Balancer and the ElectricFlow Server

You must configure a load balancer to balance two ports for the ElectricFlow DevOps Insight services:

- TCP port of the Elasticsearch service. The default port number is 9200.
- TCP port of the Logstash service. The default port number is 9500.

You must configure the ElectricFlow server to use the load balancer host by using the instructions in the "DevOps Insight Server Configuration" section in the *ElectricFlow User Guide* at http://docs.electric-cloud.com/eflow_doc/FlowIndex.html.

DNS Issue and Publish Host Setting

The Publish host setting is critical for nodes in a cluster. By default, this setting contains the value of the node's hostname. The cluster will not work if this hostname resolves as an internal IP address that is

unreachable by other nodes or as the IP address of the loopback interface (such as 127.0.0.1). If this is the case, then messages similar to the following will be in the log file:

```
[2018-08-30T09:49:23,907][DEBUG][o.e.a.a.c.h.TransportClusterHealthAction] [node0] no known master node, scheduling a retry
```

```
[2018-08-30T09:49:25,449][INFO ][o.e.d.z.ZenDiscovery ] [node0] failed to send join request to master [{node1}{-xG7eVHnRbGvGVgAYwoukA}{nshut08WQ9iRoPwpJomomA}{node1}{127.0.1.1:9300}], reason [RemoteTransportException[[node0][127.0.1.1:9300][internal:discovery/zen/join]]; nested: NotMasterException[Node [{node0}{rVpbjN3VQyGv3mf-FG7bOA}{--V1HU-qToeAMkxPENGeNw}{node0}{127.0.1.1:9300}] not master for join request]; ], tried [3] times
```

To fix this issue, you must specify a hostname that resolves to a real IP address by setting `--elasticsearchPublishHost` in console or silent install modes or in the corresponding field in GUI installation mode. Also, the exact IP address can be passed to this setting.

Ensuring a Healthy Cluster Before Upgrade or Reconfiguration Operations

The upgrade or reconfiguration processes require that the settings be written to the cluster data, but this is impossible if the cluster is unhealthy. In this case, the installation pauses and waits for a healthy cluster. Therefore, you should upgrade every node separately and one by one.

Make sure that the cluster has green or yellow status before and after the upgrade or reconfiguration of each node. To do so, use the `curl` utility:

```
$ curl -k 'https://reportuser:<YOUR PASSWORD>@localhost:9200/_cluster/health?pretty'
{
  "cluster_name" : "elasticsearch",
  "status" : "green",
  "timed_out" : false,
  "number_of_nodes" : 5,
  "number_of_data_nodes" : 4,
  "active_primary_shards" : 1,
  "active_shards" : 1,
  "relocating_shards" : 0,
  "initializing_shards" : 0,
  "unassigned_shards" : 0,
  "delayed_unassigned_shards" : 0,
  "number_of_pending_tasks" : 0,
  "number_of_in_flight_fetch" : 0,
  "task_max_waiting_in_queue_millis" : 0,
  "active_shards_percent_as_number" : 100.0
}
```

If the cluster is unhealthy, then the command returns the following result:

```
$ curl -k 'https://reportuser:<YOUR_PASSWORD>@localhost:9200/_cluster/health?pretty'
Search Guard not initialized (SG11). See https://github.com/floragunncom/search-guard-docs/blob/master/sgadmin.md
```

At first, you should upgrade all master nodes. Then you can start the upgrade on the other nodes.

The exception is changing the cluster name setting. When this reconfiguration is performed over the first nodes, then the cluster cannot be formed, because other nodes have a different cluster name. In this case, the upgrade will pause at the "Starting services" stage. At this point, you can start the upgrades for the other master nodes. When the number of upgraded nodes equals or exceeds the

minimum number of master-eligible nodes, then the cluster will be formed, and all paused upgrades will complete successfully.

Upgrading from Version 8.4 and Earlier

When you generate the signing CA certificate file, all certificates are regenerated during the upgrade from DevOps Insight version 8.4 and earlier to version 8.5 and later. The old certificates are saved with the `*.backup` extension in the configuration directories.

If you used custom certificates, you can restore them by renaming `*.pem.backup` and `*.jks.backup` in the `DATA_DIR/conf/reporting/elasticsearch` and `DATA_DIR/conf/reporting/logstash` directories. You must restart the DevOps Insight services when you rename certificate files.

Changing the Password for Secure Access to a DevOps Insight Cluster

To change the password for secure access to a DevOps Insight cluster, you must perform the reconfiguration with the corresponding changes over each node in the cluster.

Chapter 5: Configuring ElectricFlow

This section contains the configuration tasks you must perform after you install ElectricFlow.

> **Important:** The following situation might occur when the workspace files are in a directory other than the default *workspace* directory and the ElectricFlow configuration links to it. When you install a new version, ElectricFlow creates a workspace directory in the default location. It does not recognize the preconfigured workspace link in the previous configuration.
>
> When configuring ElectricFlow after an upgrade, you cannot use ecconfigure to move the workspace directory to the preconfigured network location. You must manually specify the link to the workspace directory in the new configuration.

The Default Zone and Gateways to Remote Zones

The ElectricFlow server is a member of the `default` zone (created during ElectricFlow installation). To ensure that the ElectricFlow server can reach remote zones, you must establish a gateway or a gateway chain to reach each one either directly or indirectly. Also, to preserve this reachability, do not rename the `default` zone.

Applying an Enterprise License Key

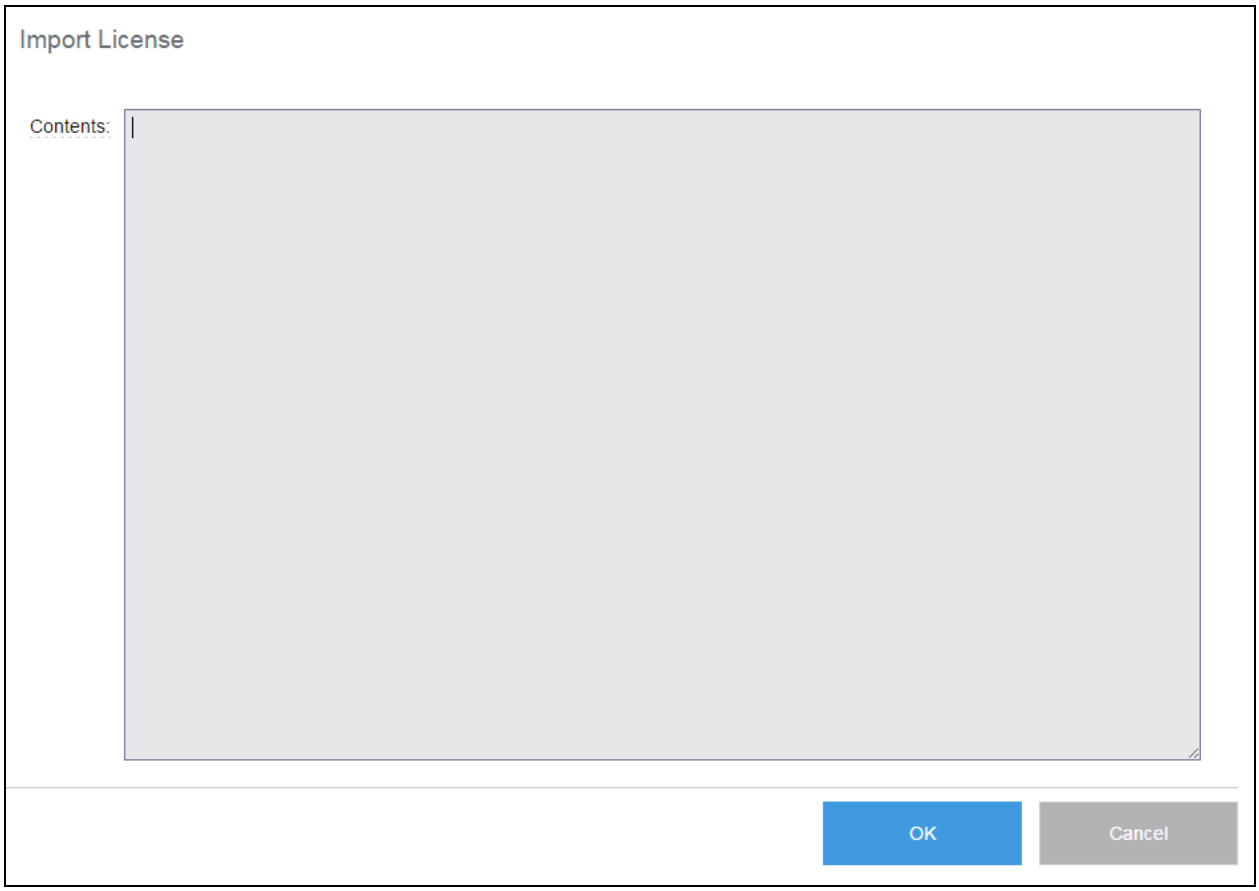
Use the following task to add an enterprise license to an ElectricFlow server.

1. Log into the ElectricFlow server.

   For more information, see Logging Into the ElectricFlow Web Interface on page 3-181.

2. Go to **Administration** > **Licenses**.
3. Click **Import License**.

   The Import License text box appears.



4. Open the license file in a text editor and copy and paste the entire license text in the Import License text box.
5. Click **OK** to import the ElectricFlow license.

External Database Configuration

During the ElectricFlow server installation, if you elected not to install the built-in (default) database, you need to configure an alternate database. An ElectricFlow enterprise license is required to configure an alternate database. For more information about supported databases, see Database Requirements on page 2-11. For installation instructions, see Configuring ElectricFlow to Use an Alternate Database on page 5-4.

Database Interactions

Your database administrator (DBA) must create a database for use specifically by ElectricFlow. The ElectricFlow server interacts with the database using a JDBC driver for each of the databases that ElectricFlow supports.

The first step in any interaction is to present user credentials to the database. This information is stored in the ElectricFlow `database.properties` file as a user name plus a password. The password is stored as an encrypted string, using the "passkey" generated by the server.

Database User

For MS SQL Server and MySQL, your DBA should create a database user for use specifically by ElectricFlow. For Oracle, your DBA *must* create a database user for use specifically by ElectricFlow.

The ElectricFlow database user must have permissions to add or delete rows from the database at all times. The database user must also have rights to create or delete tables, and add or remove a columns, indices, and constraints to a table at certain defined times.

When the ElectricFlow server first starts up, ElectricFlow creates a schema in the specified database, so the database user should be the owner of the ElectricFlow database. This allows ElectricFlow to make the required schema changes.

If the ElectricFlow server cannot connect to the database, it continues to wait for a valid database configuration. Check the log files for a successful database connection.

Default Database Ports

The supported external databases use the following ports:

Port	Used by
1521	Oracle
1433	Microsoft SQL Server
3306	MySQL

MySQL Prerequisites

Ensuring Database User Permission to Create and Deleta Schema During an Upgrade

Electric Cloud supports installations only where the database user has rights to create and delete tables at all times. To upgrade the MySQL databases, the database user must also have permissions to create and delete schema (databases) for the duration of the upgrade. For security reasons, permissions granted to database users to create and delete schema in multi-tenant MySQL databases may be revoked after the upgrade process is complete.

Installing the JDBC Driver

For MySQL, the JDBC driver is not installed by the ElectricFlow installer (for licensing reasons). It must be downloaded and installed separately from the MySQL website. For more information, go to Installing the MySQL JDBC Driver on page 3-181.

Setting the Default MySQL Server Timezone

By default, a MySQL instance uses the timezone from the operating system, which might cause the following bootstrap failure if the instance is using the MySQL JDBC Connector/J driver version 8.0 or later:

```
{noformat}
2018-07-27T03:47:35.070 | ERROR | bootstrap                          |          |
                                |
                    | HikariPool
| ElectricFlow - Exception during pool initialization.
```

```
java.sql.SQLException: The server time zone value 'PDT' is unrecognized or represents
more than one time zone. You must configure either the server or JDBC driver (via the
serverTimezone configuration property) to use a more specific time zone value if you
want to utilize time zone support.
at com.mysql.cj.jdbc.exceptions.SQLError.createSQLException(SQLError.java:127)
at com.mysql.cj.jdbc.exceptions.SQLError.createSQLException(SQLError.java:95)

…
```

If this failure occurs, apply the fix to the MySQL instance (not to the ElectricFlow server) that is described at *https://bugs.mysql.com/bug.php?id=86425*. For more information about this issue, see *https://bugs.mysql.com/bug.php?id=85816* and *https://stackoverflow.com/a/47108511*.

Setting Character Encoding to UTF-8 and Default Collation to Case-Insensitive

Since release 8.0.1, Electric Cloud has instructed customers using a MySQL database to ensure that the default collation for their database schema is set to `utf8_unicode_ci` or `utf8_general_ci` and that no table in their schema overrides this. As of release 8.5, the ElectricFlow server checks this configuration on startup and logs errors in the server log if it is not set correctly.

If the collation is not configured correctly, then entering non-ASCII text into ElectricFlow might cause errors. For example, setting a release name to a non-ASCII value and attempting a search causes an exception.

If your MySQL database schema or any tables in it are set to a non-UTF-8 collation order, see Knowledge Base article *KBEC-00385 - Converting a MySQL Database From Latin-1 to UTF-8 Before Upgrading to ElectricFlow 8.0.1* for detailed instructions about safely converting your schema to UTF-8. (NMB-26521)

Tuning Memory Allocation

To tune memory allocation, see the *KBEC-00038 - Improving ElectricCommander server performance by tuning memory allocation* Knowledge Base article.

Configuring ElectricFlow to Use an Alternate Database

If you deselected the "database" check box during installation, you cannot log into ElectricFlow until you set up a database configuration pointing to an external database. An ElectricFlow enterprise license is required; for details, see External Database Configuration on page 5-2. You can change the database configuration through the ElectricFlow web interface or the ElectricFlow command-line tool.

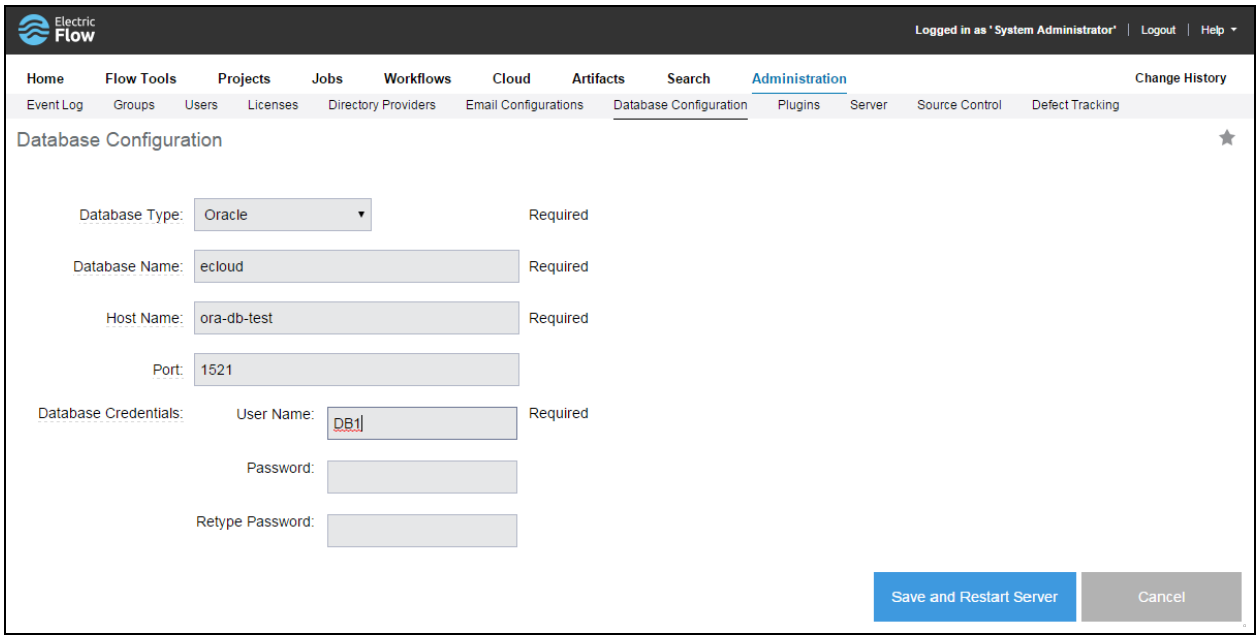
To change the database configuration in an ElectricFlow cluster, you must reset ElectricFlow to single-server mode beforehand. For details, see Running a Cluster in Single-Server Mode on page 4-24.

Setting the Database with the Web Interface

Use this procedure to set the database with the ElectricFlow web interface. You cannot log into ElectricFlow until you set up a database configuration pointing to an external database, but you can use the ElectricFlow web interface to connect to an external database.

1. Go to the **Administration** tab in the ElectricFlow UI and select **Database Configuration**.

   The **Database Configuration** screen appears.



2. Select your **Database Type** from the drop-down menu.

> **Note:** The **Built-in (MariaDB)** option is supported only for the built-in database that is installed by ElectricFlow. Any other MariaDB database is not supported; that is, you cannot install another MariaDB instance and use it with ElectricFlow. Also, changing the database configuration options (such as the database name, host name, and credentials) to use any other database such as MySQL when using the **Built-in (MariaDB)** option is not supported.

3. Enter your **Database Name**.
4. Enter the **Host Name** for your database server.
5. Accept the default **Port** or supply the port number you need for your database.
6. Supply the database **User Name** the ElectricFlow server will use to access your database.
7. Enter and confirm the **Password** for the database user you specified.
8. Click **Save and Restart Server** after supplying information in all fields.

Setting the Database from a Command Line

This section contains topics related to setting an alternate database for ElectricFlow through a command line.

SQL Server Authentication

SQL Server supports two types of user authentication:

- SQL Server Authentication
- Windows Authentication

You must find out from your DBA which authentication type is required for the ElectricFlow user because, the authentication type influences how information is provided in the ectool command `setDatabaseConfiguration`. See setDatabaseConfiguration Command Examples on page 5-7 for example command syntax.

Setting the Database with ectool

You use the ectool `setDatabaseConfiguration` command to change the database configuration from the command line.

1. Determine if your database is SQL Server. The type of user authentication used by the database impacts the syntax of the `setDatabaseConfiguration` command that you use in this procedure.

   For more information, see these topics:

   - SQL Server Authentication on page 5-5
   - setDatabaseConfiguration Command Examples on page 5-7
   - Setting the Database as a SQL Server with SSO Login on Windows on page 5-8

2. (ElectricFlow 5.0 and later) Set the database configuration based on the type of user authentication used by the database:

   - For SQL server authentication or a SQL Server with NTLM login, enter

     ```
     setDatabaseConfiguration <--options>
     ```

     where `<--options>` are the options that you specify based on the type of user authentication.

     For more information, go to setDatabaseConfiguration Command Options on page 5-9.

   - For a SQL server with the SSO login on Windows, see Setting the Database as a SQL Server with SSO Login on Windows on page 5-8.

   After you change the ElectricFlow database configuration, the server attempts to connect to the database to do the initial schema setup.

   > **Note:** Do not restart the ElectricFlow server at this time. (You should restart the server only if it was already connected to a built-in or external database and the ectool `setDatabaseConfiguration` command has been used to connect the server to an entirely new database.)

3. Enter the following command and wait for the output:

   ```
   ectool --server localhost --timeout 900 getServerStatus --block 1 --
   serverStateOnly 1
   ```

   This command runs for 900 seconds (15 minutes), or until Commander finishes creating all the schema objects, or until `getServerStatus` displays either `bootstrap` or `running`.

4. If the output says `bootstrap`, enter the command again until it says `running`.

   The commander.log file shows `commanderServer is running` as in the following snippet:

   ```
   2016-02-10T19:19:06.582 | 10.0.2.206 | INFO | bootstrap |  |  | ServerStatus
   | commanderServer is running
   ```

   Failing to wait and restarting Commander while it is creating schema objects will cause it to fail to start (during the next manual start) with an error:

   ```
   2016-09-07T15:08:55.825 | DEBUG | bootstrap |  |  | upgradeData |
   OperationInvoker | Exception: InvalidSchema: Unable to validate the database
   schema: could not extract ResultSet 2016-09-07T15:08:55.831 | ERROR |
   bootstrap  |  |  |  | BootstrapCommanderServerImpl | Unable to validate the
   database schema: could not extract ResultSet
   com.electriccloud.errors.EcException: Unable to validate the database schema:
   could not extract ResultSet
   at com.electriccloud.errors.EcException.create(EcException.java:165)
   at com.electriccloud.errors.EcExceptionBuilder.build
   (EcExceptionBuilder.java:34)
   at com.electriccloud.upgrade.UpgradeManagerImpl.doDataMaintenance
   (UpgradeManagerImpl.java:672)
   at com.electriccloud.upgrade.UpgradeDataOperation.perform
   (UpgradeDataOperation.java:50)
   at com.electriccloud.upgrade.UpgradeDataOperation.perform
   (UpgradeDataOperation.java:26)
   ```

5. Use `getServerStatus` to look for problems logging into the database or creating the schema. This command shows log prompts from the server bootstrap process.

   > **Note:** Before the server is completely up, `getServerStatus` does not require a login session, but after the server is up, it does. Thus, if you enter the `getServerStatus` call and get a "session expired" error, the server is up.

setDatabaseConfiguration Command Examples

Following are examples of how to use the `setDatabaseConfiguration` command.

SQL Server Authentication:

```
ectool setDatabaseConfiguration
--databaseType sqlserver
--databaseName commander
--hostName localhost
--port 1433
--userName commander
--password commander
```

The `--userName` and `--password` options must be included in the `setDatabaseConfiguration` command.

SQL Server with NTLM login:

```
ectool setDatabaseConfiguration
--databaseType sqlserver
--databaseName commander
--hostName localhost
--port 1433
--userName commander@domain.com
--password commander
```

The user name must include the domain name. For example, *user@domain.com* or *domain\user.*

Setting the Database as a SQL Server with SSO Login on Windows

1. Download the JDBC driver from Microsoft.

   Select the appropriate JDBC driver version from the Microsoft JDBC driver download page (https://www.microsoft.com/en-us/download/details.aspx?id=11774). See the information in the "Details" and "System Requirements" sections to help select the correct driver.

   For example, if you download the sqljdbc_4.0.2206.100_enu.tar.gz file and unzip it, you get this file:

   sqljdbc_4.0.2206.100_enu.tar.gz\sqljdbc_4.0.2206.100_enu.tar\sqljdbc_4.0\enu\auth\x64\sqljdbc_auth.dll

2. Check whether you already have the JDBC driver file, such as sqljdbc4.jar, in the C:\Program Files\Electric Cloud\ElectricCommander\server\wars\commander-server.war\WEB-INF\lib\ directory.

   This file ships with ElectricFlow and should be version 4.0 or later.

3. Copy the sqljdbc_auth.dll file from Step 1 to the C:/Program Files/Electric Cloud/ElectricCommander/server/lib directory, which is the same folder set as the java.library.path property.

4. Enter the following command to update the `COMMANDER_CUSTOM_DB_URL` and `COMMANDER_DB_URL` properties in the C:\ProgramData\Electric Cloud\ElectricCommander\conf\database.properties file with the JDBC URL specified in the customDatabaseUrl argument:

   ```
   ectool setDatabaseConfiguration --databaseType sqlserver --databaseName
   commander --hostName localhost --port 1433 --customDatabaseUrl
   "jdbc:sqlserver://localhost:1433;integratedSecurity=true;databaseName=command
   er;applicationName=ElectricCommander;selectMethod=cursor"
   ```

   > **Note:** ElectricFlow uses `COMMANDER_DB_URL` for information only. If `COMMANDER_CUSTOM_DB_URL` is set, this value is used instead of `COMMANDER_DB_URL`.

5. Make sure that the ElectricCommander Server service is set to run as a user who can log into the SQL Server database using Windows authentication.

6. If you needed to set the ElectricCommander Server service to run as a user who can log into the SQL Server database using Windows authentication, restart the ElectricFlow server. Otherwise, check the commander.log file to verify that the ElectricFlow server connects to the SQL Server database with SSO login.

   The ElectricFlow server connects to the SQL Server with SSO login. If this is successful, the following lines appear in the commander.log file:

   ```
   USERDNSDOMAIN=ELECTRIC €LOUD.COM

   USERDOMAIN=ELECTRIC €LOUD

   USERNAME=<user who can log into the SQL Server database using Windows
   authentication>

   USERPROFILE=C:\Users\<user who can log into the SQL Server database using
   Windows authentication>

   ..

   java.library.path=C:/Program Files/Electric
   Cloud/ElectricCommander/server/lib

   ...

   2015 Ð7 Ð7T18:08:13.255 | INFO | bootstrap | | | 2015 Ð7 Ð7T18:08:13.319 | WARN
   | bootstrap | | |

   ...

   2015 Ð7 Ð7T18:08:14.479 | INFO | bootstrap | | | 2015 Ð7 Ð7T18:08:14.479 | INFO
   | bootstrap | | |
   ```

setDatabaseConfiguration Command Options

The following options are available for use with the ectool command `setDatabaseConfiguration`. The option command syntax is:

```
setDatabaseConfiguration
      [--databaseType <mysql|sqlserver|Oracle|builtin>]
      [--databaseName <database name>]
      [--hostName <host name>]
      [--ignorePasskeyMismatch <Boolean flag>]
      [--ignoreServerMismatch <Boolean flag>]
      [--password <password>]
      [--port <port number>]
      [--preserveSessions <Boolean flag>]
      [--userName <user name>]
      [--customDatabaseDialect <custom database dialect>]
      [--customDatabaseDriver <custom database driver>]
      [--customDatabaseUrl <custom database URL>]
```

The following table describes the command options:

Option	Description
`databaseType`	Selects the database type–supported options are `mysql|sqlserver|Oracle|builtin` **Note:** The `builtin` option is supported only for the built-in MariaDB database that is installed by ElectricFlow. Any other MariaDB database is not supported; that is, you cannot install another MariaDB instance and use it with ElectricFlow. Also, changing the database configuration options (such as the database name, host name, and credentials) to use any other database such as MySQL when using the `builtin` option is not supported.
`databaseName`	The name of your alternate database–this is not the host name, but the name the DBA gave the database object.
`hostName`	The host name where your database is running
`ignorePasskeyMismatch`	<*Boolean flag* - `0|1|true|false`> - If the server is started with a different passkey, ignore the mismatch if “true”. **Note:** This action discards all saved passwords.
`ignoreServerMismatch`	<*Boolean flag* - `0|1|true|false`> - If the server is started on a different host than where the server previously started, ignore the mismatch if “true”.
`port`	The port number used by the database Server–if omitted, port 1433 is used
`preserveSessions`	<*Boolean flag* - `0|1|true|false`> - If ignoring a server mismatch, default behavior invalidates all sessions. Setting this flag to “true” saves all sessions, allowing the server to reconnect to running jobs. This option is used in combination with `ignoreServerMismatch`.
`userName`	The user name to use when connecting to the database

Option	Description
password	The password to use to connect to the database
customDatabaseDialect	Internal option–use only at the request of Electric Cloud support
customDatabaseDriver	Internal option–use only at the request of Electric Cloud support
customDatabaseUrl	Internal option–use only at the request of Electric Cloud support

Enabling Full Database URL Control

If you want to enable full control over the database URL:

1. Go to the database.properties file.
2. Create a property called COMMANDER_CUSTOM_DB_URL.
3. Add the explicit connect string to that property.

> **Note:** COMMANDER_DB_URL is a "write-only" property that it is never read by ElectricFlow. It is generated based on other settings in that file and serves as information only.

If you were using ElectricFlow in clustered mode, you should now upload your modified database.properties.file to ZooKeeper and then change ElectricFlow back to clustered mode. For more information, go to Running a Cluster in Clustered Mode on page 4-22.

Configuring Services Autostart for Non-Root/Non-sudo Linux Installations

Linux installations that you perform as a non-root user or without sudo permissions cannot automatically start the ElectricFlow server, web server, repository server, or agents. This means that you must set up service autostart after installation is complete. This section describes how to set up autostart and also how to disable it if needed.

Setting Up Services Autostart

This section describes how to set up autostart of services for all ElectricFlow components as follows:

- Setting Up Autostart for ElectricFlow Server Services on page 5-12
- Setting Up Autostart for ElectricFlow Web Server Services on page 5-12
- Setting Up Autostart for ElectricFlow Repository Server Services on page 5-13
- Setting Up Autostart for ElectricFlow Built-In Database Server Services on page 5-14
- Setting Up Autostart for ElectricFlow Agent Services on page 5-15
- Setting Up Autostart for ElectricFlow DevOps Insight Server Services on page 5-16

Setting Up Autostart for ElectricFlow Server Services

1. Provide a service control script in the `/etc/init.d` directory by using one of the following methods.

   Note: *<install_dir>* is `/opt/electriccloud/electriccommander` by default.

   - Secure (preferred) method:

     ```
     $ sudo cp -v <install_dir>/startup/commanderServer /etc/init.d/ef-user-server

     '<install_dir>/startup/commanderServer' -> '/etc/init.d/ef-user-server'

     $ sudo chown -v root:root /etc/init.d/ef-user-server

     ownership of '/etc/init.d/ef-user-server' retained as root:root

     $ sudo chmod -v 0755 /etc/init.d/ef-user-server

     mode of '/etc/init.d/ef-user-server' retained as 0755 (rwxr-xr-x)
     ```

   - Less secure (alternative) method:

     ```
     $ sudo ln -sv <install_dir>/startup/commanderServer /etc/init.d/ef-user-server

     '/etc/init.d/ef-user-server' -> '<install_dir>/startup/commanderServer'
     ```

2. Create links in the "rc" directories.

   The exact commands as well as the prompts displayed in response will vary with the specific distribution. Following are several examples:

   - RHEL 5.*x*, 6.*x*, or 7.*x*; CentOS 5.*x*, 6.*x*, or 7.*x*; SLES 11 or 12:

     ```
     $ sudo /sbin/chkconfig --add ef-user-server
     $ sudo /sbin/chkconfig ef-user-server on
     ```

   - Ubuntu 14.*x*, 16.*x*, or 18.*x*:

     ```
     $ sudo /usr/sbin/update-rc.d ef-user-server defaults

     Adding system startup for /etc/init.d/ef-user-server ...
     /etc/rc0.d/K20ef-user-server -> ../init.d/ef-user-server
     /etc/rc1.d/K20ef-user-server -> ../init.d/ef-user-server
     /etc/rc6.d/K20ef-user-server -> ../init.d/ef-user-server
     /etc/rc2.d/S20ef-user-server -> ../init.d/ef-user-server
     /etc/rc3.d/S20ef-user-server -> ../init.d/ef-user-server
     /etc/rc4.d/S20ef-user-server -> ../init.d/ef-user-server
     /etc/rc5.d/S20ef-user-server -> ../init.d/ef-user-server
     ```

Setting Up Autostart for ElectricFlow Web Server Services

1. Provide a service control script in the `/etc/init.d` directory by using one of the following methods.

> **Note:** <*install_dir*> is /opt/electriccloud/electriccommander by default.

- Secure (preferred) method:

```
$ sudo cp -v <install_dir>/startup/commanderApache /etc/init.d/ef-user-apache

'<install_dir>/startup/commanderApache' -> '/etc/init.d/ef-user-apache'

$ sudo chown -v root:root /etc/init.d/ef-user-apache

ownership of '/etc/init.d/ef-user-apache' retained as root:root

$ sudo chmod -v 0755 /etc/init.d/ef-user-apache

mode of '/etc/init.d/ef-user-apache' retained as 0755 (rwxr-xr-x)
```

- Less secure (alternative) method:

```
$ sudo ln -sv <install_dir>/startup/commanderApache /etc/init.d/ef-user-apache

'/etc/init.d/ef-user-apache' -> '<install_dir>/startup/commanderApache'
```

2. Create links in the "rc" directories.

   The exact commands as well as the prompts displayed in response will vary with the specific distribution. Following are several examples:

   - RHEL 5.*x*, 6.*x*, or 7.*x*; CentOS 5.*x*, 6.*x*, or 7.*x*; SLES 11 or 12:

```
$ sudo /sbin/chkconfig --add ef-user-apache
$ sudo /sbin/chkconfig ef-user-apache on
```

   - Ubuntu 14.*x*, 16.*x*, or 18.*x*:

```
$ sudo /usr/sbin/update-rc.d ef-user-apache defaults

Adding system startup for /etc/init.d/ef-user-apache ...
/etc/rc0.d/K20ef-user-apache -> ../init.d/ef-user-apache
/etc/rc1.d/K20ef-user-apache -> ../init.d/ef-user-apache
/etc/rc6.d/K20ef-user-apache -> ../init.d/ef-user-apache
/etc/rc2.d/S20ef-user-apache -> ../init.d/ef-user-apache
/etc/rc3.d/S20ef-user-apache -> ../init.d/ef-user-apache
/etc/rc4.d/S20ef-user-apache -> ../init.d/ef-user-apache
/etc/rc5.d/S20ef-user-apache -> ../init.d/ef-user-apache
```

Setting Up Autostart for ElectricFlow Repository Server Services

1. Provide a service control script in the /etc/init.d directory by using one of the following methods.

> **Note:** <*install_dir*> is /opt/electriccloud/electriccommander by default.

- Secure (preferred) method:

```
$ sudo cp -v <install_dir>/startup/commanderRepository /etc/init.d/ef-user-repository

'<install_dir>/startup/commanderRepository' -> '/etc/init.d/ef-user-repository'

$ sudo chown -v root:root /etc/init.d/ef-user-repository

ownership of '/etc/init.d/ef-user-repository' retained as root:root

$ sudo chmod -v 0755 /etc/init.d/ef-user-repository

mode of '/etc/init.d/ef-user-repository' retained as 0755 (rwxr-xr-x)
```

- Less secure (alternative) method:

```
$ sudo ln -sv <install_dir>/startup/commanderRepository /etc/init.d/ef-user-repository

'/etc/init.d/ef-user-repository' -> '<install_
dir>/startup/commanderRepository'
```

2. Create links in the "rc" directories.

   The exact commands as well as the prompts displayed in response will vary with the specific distribution. Following are several examples:

   - RHEL 5.*x*, 6.*x*, or 7.*x*; CentOS 5.*x*, 6.*x*, or 7.*x*; SLES 11 or 12:

```
$ sudo /sbin/chkconfig --add ef-user-repository
$ sudo /sbin/chkconfig ef-user-repository on
```

   - Ubuntu 14.*x*, 16.*x*, or 18.*x*:

```
$ sudo /usr/sbin/update-rc.d ef-user-repository defaults

Adding system startup for /etc/init.d/ef-user-repository ...
/etc/rc0.d/K20ef-user-repository -> ../init.d/ef-user-repository
/etc/rc1.d/K20ef-user-repository -> ../init.d/ef-user-repository
/etc/rc6.d/K20ef-user-repository -> ../init.d/ef-user-repository
/etc/rc2.d/S20ef-user-repository -> ../init.d/ef-user-repository
/etc/rc3.d/S20ef-user-repository -> ../init.d/ef-user-repository
/etc/rc4.d/S20ef-user-repository -> ../init.d/ef-user-repository
/etc/rc5.d/S20ef-user-repository -> ../init.d/ef-user-repository
```

Setting Up Autostart for ElectricFlow Built-In Database Server Services

1. Provide a service control script in the `/etc/init.d` directory by using one of the following methods.

   > **Note:** *<install_dir>* is `/opt/electriccloud/electriccommander` by default.

- Secure (preferred) method:

```
$ sudo cp -v <install_dir>/startup/commanderDatabase /etc/init.d/ef-user-database

'<install_dir>/startup/commanderDatabase' -> '/etc/init.d/ef-user-database'

$ sudo chown -v root:root /etc/init.d/ef-user-database

ownership of '/etc/init.d/ef-user-database' retained as root:root

$ sudo chmod -v 0755 /etc/init.d/ef-user-database

mode of '/etc/init.d/ef-user-database' retained as 0755 (rwxr-xr-x)
```

- Less secure (alternative) method:

```
$ sudo ln -sv <install_dir>/startup/commanderDatabase /etc/init.d/ef-user-database

'/etc/init.d/ef-user-database' -> '<install_
dir>/startup/commanderDatabase'
```

2. Create links in the "rc" directories.

   The exact commands as well as the prompts displayed in response will vary with the specific distribution. Following are several examples:

   - RHEL 5.*x*, 6.*x*, or 7.*x*; CentOS 5.*x*, 6.*x*, or 7.*x*; SLES 11 or 12:

     ```
     $ sudo /sbin/chkconfig --add ef-user-database
     $ sudo /sbin/chkconfig ef-user-database on
     ```

   - Ubuntu 14.*x*, 16.*x*, or 18.*x*:

     ```
     $ sudo /usr/sbin/update-rc.d ef-user-database defaults

     Adding system startup for /etc/init.d/ef-user-database ...
     /etc/rc0.d/K20ef-user-database -> ../init.d/ef-user-database
     /etc/rc1.d/K20ef-user-database -> ../init.d/ef-user-database
     /etc/rc6.d/K20ef-user-database -> ../init.d/ef-user-database
     /etc/rc2.d/S20ef-user-database -> ../init.d/ef-user-database
     /etc/rc3.d/S20ef-user-database -> ../init.d/ef-user-database
     /etc/rc4.d/S20ef-user-database -> ../init.d/ef-user-database
     /etc/rc5.d/S20ef-user-database -> ../init.d/ef-user-database
     ```

Setting Up Autostart for ElectricFlow Agent Services

1. Provide a service control script in the `/etc/init.d` directory by using one of the following methods.

   Note: *<install_dir>* is `/opt/electriccloud/electriccommander` by default.

- Secure (preferred) method:

```
$ sudo cp -v <install_dir>/startup/commanderAgent /etc/init.d/ef-user-agent

'<install_dir>/startup/commanderAgent' -> '/etc/init.d/ef-user-agent'

$ sudo chown -v root:root /etc/init.d/ef-user-agent

ownership of '/etc/init.d/ef-user-agent' retained as root:root

$ sudo chmod -v 0755 /etc/init.d/ef-user-agent

mode of '/etc/init.d/ef-user-agent' retained as 0755 (rwxr-xr-x)
```

- Less secure (alternative) method:

```
$ sudo ln -sv <install_dir>/startup/commanderAgent /etc/init.d/ef-user-agent

'/etc/init.d/ef-user-agent' -> '<install_dir>/startup/commanderAgent'
```

2. Create links in the "rc" directories.

   The exact commands as well as the prompts displayed in response will vary with the specific distribution. Following are several examples:

   - RHEL 5.*x*, 6.*x*, or 7.*x*; CentOS 5.*x*, 6.*x*, or 7.*x*; SLES 11 or 12:

     ```
     $ sudo /sbin/chkconfig --add ef-user-agent
     $ sudo /sbin/chkconfig ef-user-agent on
     ```

   - Ubuntu 14.*x*, 16.*x*, or 18.*x*:

     ```
     $ sudo /usr/sbin/update-rc.d ef-user-agent defaults

     Adding system startup for /etc/init.d/ef-user-agent ...
     /etc/rc0.d/K20ef-user-agent -> ../init.d/ef-user-agent
     /etc/rc1.d/K20ef-user-agent -> ../init.d/ef-user-agent
     /etc/rc6.d/K20ef-user-agent -> ../init.d/ef-user-agent
     /etc/rc2.d/S20ef-user-agent -> ../init.d/ef-user-agent
     /etc/rc3.d/S20ef-user-agent -> ../init.d/ef-user-agent
     /etc/rc4.d/S20ef-user-agent -> ../init.d/ef-user-agent
     /etc/rc5.d/S20ef-user-agent -> ../init.d/ef-user-agent
     ```

Setting Up Autostart for ElectricFlow DevOps Insight Server Services

DevOps Insight requires autostart for two services: Elasticsearch and Logstash. The DevOps Insight server uses the Elasticsearch search engine and the Logstash data-collection and log-parsing engine to gather data from the ElectricFlow server for use in the DevOps Insight dashboards such as the Deployments, Releases, and Release Command Center dashboards.

Setting Up Autostart for Elasticsearch

1. Provide a service control script in the `/etc/init.d` directory by using one of the following methods.

Note: *<install_dir>* is /opt/electriccloud/electriccommander by default.

- Secure (preferred) method

```
$ sudo cp -v <install_dir>/reporting/startup/commanderElasticsearch
/etc/init.d/ef-user-elasticsearch

'<install_dir>/reporting/startup/commanderElasticsearch' ->
'/etc/init.d/ef-user-elasticsearch'

$ sudo chown -v root:root /etc/init.d/ef-user-elasticsearch

ownership of '/etc/init.d/ef-user-elasticsearch' retained as root:root

$ sudo chmod -v 0755 /etc/init.d/ef-user-elasticsearch

mode of '/etc/init.d/ef-user-elasticsearch' retained as 0755 (rwxr-xr-x)
```

- Less secure (alternative) method:

```
$ sudo ln -sv <install_dir>/reporting/startup/commanderElasticsearch
/etc/init.d/ef-user-elasticsearch

'/etc/init.d/ef-user-elasticsearch' -> '<install_
dir>/reporting/startup/commanderElasticsearch'
```

2. Create links in the "rc" directories.

   The exact commands as well as the prompts displayed in response will vary with the specific distribution. Following are several examples:

   - RHEL 5.*x*, 6.*x*, or 7.*x*; CentOS 5.*x*, 6.*x*, or 7.*x*; SLES 11 or 12:

```
$ sudo /sbin/chkconfig --add ef-user-elasticsearch
$ sudo /sbin/chkconfig ef-user-elasticsearch on
```

   - Ubuntu 14.*x*, 16.*x*, or 18.*x*:

```
$ sudo /usr/sbin/update-rc.d ef-user-elasticsearch defaults

Adding system startup for /etc/init.d/ef-user-elasticsearch ...
/etc/rc0.d/K20ef-user-elasticsearch -> ../init.d/ef-user-elasticsearch
/etc/rc1.d/K20ef-user-elasticsearch -> ../init.d/ef-user-elasticsearch
/etc/rc6.d/K20ef-user-elasticsearch -> ../init.d/ef-user-elasticsearch
/etc/rc2.d/S20ef-user-elasticsearch -> ../init.d/ef-user-elasticsearch
/etc/rc3.d/S20ef-user-elasticsearch -> ../init.d/ef-user-elasticsearch
/etc/rc4.d/S20ef-user-elasticsearch -> ../init.d/ef-user-elasticsearch
/etc/rc5.d/S20ef-user-elasticsearch -> ../init.d/ef-user-elasticsearch
```

Setting Up Autostart for Logstash

1. Provide a service control script in the /etc/init.d directory by using one of the following methods.

Note: *<install_dir>* is /opt/electriccloud/electriccommander by default.

- Secure (preferred) method

```
$ sudo cp -v <install_dir>/reporting/startup/commanderLogstash
/etc/init.d/ef-user-logstash

'<install_dir>/reporting/startup/commanderLogstash' -> '/etc/init.d/ef-
user-logstash'

$ sudo chown -v root:root /etc/init.d/ef-user-logstash

ownership of '/etc/init.d/ef-user-logstash' retained as root:root

$ sudo chmod -v 0755 /etc/init.d/ef-user-logstash

mode of '/etc/init.d/ef-user-logstash' retained as 0755 (rwxr-xr-x)
```

- Less secure (alternative) method:

```
$ sudo ln -sv <install_dir>/reporting/startup/commanderLogstash
/etc/init.d/ef-user-logstash

'/etc/init.d/ef-user-logstash' -> '<install_
dir>/reporting/startup/commanderLogstash'
```

2. Create links in the "rc" directories.

   The exact commands as well as the prompts displayed in response will vary with the specific distribution. Following are several examples:

   - RHEL 5.*x*, 6.*x*, or 7.*x*; CentOS 5.*x*, 6.*x*, or 7.*x*; SLES 11 or 12:

     ```
     $ sudo /sbin/chkconfig --add ef-user-logstash
     $ sudo /sbin/chkconfig ef-user-logstash on
     ```

   - Ubuntu 14.*x*, 16.*x*, or 18.*x*:

     ```
     $ sudo /usr/sbin/update-rc.d ef-user-logstash defaults

     Adding system startup for /etc/init.d/ef-user-logstash ...
     /etc/rc0.d/K20ef-user-logstash -> ../init.d/ef-user-logstash
     /etc/rc1.d/K20ef-user-logstash -> ../init.d/ef-user-logstash
     /etc/rc6.d/K20ef-user-logstash -> ../init.d/ef-user-logstash
     /etc/rc2.d/S20ef-user-logstash -> ../init.d/ef-user-logstash
     /etc/rc3.d/S20ef-user-logstash -> ../init.d/ef-user-logstash
     /etc/rc4.d/S20ef-user-logstash -> ../init.d/ef-user-logstash
     /etc/rc5.d/S20ef-user-logstash -> ../init.d/ef-user-logstash
     ```

Disabling Services Autostart

This section describes how to disable autostart of services for all ElectricFlow components as follows:

- Disabling Autostart for ElectricFlow Server Services on page 5-19
- Disabling Autostart for ElectricFlow Web Server Services on page 5-19
- Disabling Autostart for ElectricFlow Repository Server Services on page 5-19
- Disabling Autostart for ElectricFlow Built-In Database Server Services on page 5-19
- Disabling Autostart for ElectricFlow Agent Services on page 5-20
- Disabling Autostart for ElectricFlow DevOps Insight Server Services on page 5-20

Disabling Autostart for ElectricFlow Server Services

Enter the following command:

```
$ sudo rm -fv /etc/init.d/ef-user-server /etc/rc?.d/*ef-user-server
/etc/init.d/rc?.d/*ef-user-server
```

```
removed '/etc/init.d/ef-user-server'
removed '/etc/rc0.d/K20ef-user-server'
removed '/etc/rc1.d/K20ef-user-server'
removed '/etc/rc2.d/S20ef-user-server'
removed '/etc/rc3.d/S20ef-user-server'
removed '/etc/rc4.d/S20ef-user-server'
removed '/etc/rc5.d/S20ef-user-server'
removed '/etc/rc6.d/K20ef-user-server'
```

The prompts displayed in response to these commands might vary with the specific distribution and version.

Disabling Autostart for ElectricFlow Web Server Services

Enter the following command:

```
$ sudo rm -fv /etc/init.d/ef-user-apache /etc/rc?.d/*ef-user-apache
/etc/init.d/rc?.d/*ef-user-apache
```

```
removed '/etc/init.d/ef-user-apache'
removed '/etc/rc0.d/K20ef-user-apache'
removed '/etc/rc1.d/K20ef-user-apache'
removed '/etc/rc2.d/S20ef-user-apache'
removed '/etc/rc3.d/S20ef-user-apache'
removed '/etc/rc4.d/S20ef-user-apache'
removed '/etc/rc5.d/S20ef-user-apache'
removed '/etc/rc6.d/K20ef-user-apache'
```

The prompts displayed in response to these commands might vary with the specific distribution and version.

Disabling Autostart for ElectricFlow Repository Server Services

Enter the following command:

```
$ sudo rm -fv /etc/init.d/ef-user-repository /etc/rc?.d/*ef-user-repository
/etc/init.d/rc?.d/*ef-user-repository
```

```
removed '/etc/init.d/ef-user-repository'
removed '/etc/rc0.d/K20ef-user-repository'
removed '/etc/rc1.d/K20ef-user-repository'
removed '/etc/rc2.d/S20ef-user-repository'
removed '/etc/rc3.d/S20ef-user-repository'
removed '/etc/rc4.d/S20ef-user-repository'
removed '/etc/rc5.d/S20ef-user-repository'
removed '/etc/rc6.d/K20ef-user-repository'
```

The prompts displayed in response to these commands might vary with the specific distribution and version.

Disabling Autostart for ElectricFlow Built-In Database Server Services

Enter the following command:

```
$ sudo rm -fv /etc/init.d/ef-user-database /etc/rc?.d/*ef-user-database
/etc/init.d/rc?.d/*ef-user-database

removed '/etc/init.d/ef-user-database'
removed '/etc/rc0.d/K20ef-user-database'
removed '/etc/rc1.d/K20ef-user-database'
removed '/etc/rc2.d/S20ef-user-database'
removed '/etc/rc3.d/S20ef-user-database'
removed '/etc/rc4.d/S20ef-user-database'
removed '/etc/rc5.d/S20ef-user-database'
removed '/etc/rc6.d/K20ef-user-database'
```

The prompts displayed in response to these commands might vary with the specific distribution and version.

Disabling Autostart for ElectricFlow Agent Services

Enter the following command:

```
$ sudo rm -fv /etc/init.d/ef-user-agent /etc/rc?.d/*ef-user-agent
/etc/init.d/rc?.d/*ef-user-agent

removed '/etc/init.d/ef-user-agent'
removed '/etc/rc0.d/K20ef-user-agent'
removed '/etc/rc1.d/K20ef-user-agent'
removed '/etc/rc2.d/S20ef-user-agent'
removed '/etc/rc3.d/S20ef-user-agent'
removed '/etc/rc4.d/S20ef-user-agent'
removed '/etc/rc5.d/S20ef-user-agent'
removed '/etc/rc6.d/K20ef-user-agent'
```

The prompts displayed in response to these commands might vary with the specific distribution and version.

Disabling Autostart for ElectricFlow DevOps Insight Server Services

DevOps Insight requires that you disable autostart for two services: Elasticsearch and Logstash.

Disabling Autostart for Elasticsearch

```
$ sudo rm -fv /etc/init.d/ef-user-elasticsearch /etc/rc?.d/*ef-user-elasticsearch
/etc/init.d/rc?.d/*ef-user-elasticsearch

removed '/etc/init.d/ef-user-elasticsearch'
removed '/etc/rc0.d/K20ef-user-elasticsearch'
removed '/etc/rc1.d/K20ef-user-elasticsearch'
removed '/etc/rc2.d/S20ef-user-elasticsearch'
removed '/etc/rc3.d/S20ef-user-elasticsearch'
removed '/etc/rc4.d/S20ef-user-elasticsearch'
removed '/etc/rc5.d/S20ef-user-elasticsearch'
removed '/etc/rc6.d/K20ef-user-elasticsearch'
```

The prompts displayed in response to these commands might vary with the specific distribution and version.

Disabling Autostart for Logstash

```
$ sudo rm -fv /etc/init.d/ef-user-logstash /etc/rc?.d/*ef-user-logstash
/etc/init.d/rc?.d/*ef-user-logstash

removed '/etc/init.d/ef-user-logstash'
removed '/etc/rc0.d/K20ef-user-logstash'
```

```
removed '/etc/rc1.d/K20ef-user-logstash'
removed '/etc/rc2.d/S20ef-user-logstash'
removed '/etc/rc3.d/S20ef-user-logstash'
removed '/etc/rc4.d/S20ef-user-logstash'
removed '/etc/rc5.d/S20ef-user-logstash'
removed '/etc/rc6.d/K20ef-user-logstash'
```

The prompts displayed in response to these commands might vary with the specific distribution and version.

Universal Access to the Plugins Directory

A plugin is a collection of one or more features, or a third-party integration or tool that can be added to ElectricFlow. The ElectricFlow server installs all plugins into a configurable location named the plugins directory. This directory must be readable by the web server and any agents that need access to the content of one or more plugins. There are two ways to make the plugins directory readable by the web server and any agents. You can configure the ElectricFlow server, agents, and web servers to point to a central network location, or you can replicate the contents of the plugins directory on remote agents and web servers.

Configuring Universal Access for a Network Location

Use these procedures to configure ElectricFlow server, agents, and web servers to point to a universally accessible network location. This is the recommended approach because newly installed plugins are immediately available to all agents and web servers. We strongly recommend that you do this when you are running ElectricFlow in clustered mode, because it allows all the server nodes to share a common plugins directory. You also avoid the overhead of managing multiple plugins directories.

Note:
The Windows local system account cannot access network resources such as shared file systems used for plugins or workspaces. Therefore, do not use this option for a clustered server deployment, which requires a shared file system for plugins. This option is typically used only for installing agents on numerous machines, which would otherwise require that you create a new account on each of those machines.

Ways to Configure Universal Access to Plugins

A network location for the plugins directory can be set up in one of two ways:

- Moving the Plugins Directory to a Pre-Configured Network Location on page 5-21—This approach is recommended if you already have a network file system accessible to the ElectricFlow server and all remote agents and web servers.
- Leaving the Plugins Directory on the ElectricFlow Server on page 5-22—This approach leaves the plugins directory in the current location on the ElectricFlow server and shares the location across the network. *This approach is only recommended if you do not already have a network location available.*

Moving the Plugins Directory to a Pre-Configured Network Location

Use this task to move the plugins directory to a pre-configured network location. You must have a network file system accessible to the ElectricFlow server and all remote agents and web servers to perform this task.

Important: You must have root privileges to use the `--webPluginsDirectory` option.

1. Create an empty directory in the network accessible location.
2. Move the contents of the plugins subdirectory from the ElectricFlow server's data directory to this new directory.
3. Run the following commands on the ElectricFlow server:
   1. `ectool setProperty /server/settings/pluginsDirectory "<PLUGINS>"`

      This command gives the ElectricFlow server the location of `pluginsDirectory`.
   2. Run this command *only* if a web server was installed (by default, during an Express Server installation):
      `ecconfigure --webPluginsDirectory "<PLUGINS>"`

      This command modifies Apache web server configuration files (`ectool` lacks this ability). This command properly configures the Apache web server for the new plugins directory.

   Where `<PLUGINS>` is either the Windows path to the shared directory for Windows machines, or the UNIX path to the shared directory for UNIX machines.

   Note: When installing remote agents or web servers, you will be prompted to enter information about the ElectricFlow server. Select the check box to "discover the plugins directory", and the correct location is automatically picked up from the server and set during installation.
4. Run the following command on remote web servers that were already installed:
   `ecconfigure --webPluginsDirectory "<PLUGINS>"`

   Where `<PLUGINS>` is either the Windows path to the shared directory for Windows machines, or the UNIX path to the shared directory for UNIX machines.

Leaving the Plugins Directory on the ElectricFlow Server

Use this task to leave the plugins in the current location on the ElectricFlow server and share that location across the network so remote agents and web servers can obtain access.

Important: This approach is recommended only if you do not already have a network location available to the ElectricFlow server and all remote agents and web servers. See Moving the Plugins Directory to a Pre-Configured Network Location on page 5-21 if you do have a universally available network location.

Important: You must have root privileges to use the `--webPluginsDirectory` option.

1. Choose the appropriate step based on the ElectricFlow server platform:
   - If your ElectricFlow server is a Windows machine, the plugins directory is automatically shared by the name "commander-plugins" during installation. When you install remote agents and web servers on Windows, they will discover this location and be configured to use

it.

- If you are installing remote agents or web servers on UNIX machines, follow these steps:

  1. Create a Samba mount on a UNIX machine pointing to the plugins share on the Windows machine.
     `//<COMMANDER_SERVER_HOST_NAME>/commander-plugins`
  2. Export the Samba mount as a network file system share on the same UNIX machine used in the previous step.
     1. Add the following entry to `/etc/exports`:
        `/opt/electriccloud/electriccommander/plugins by default): <EXPORT> * (ro,sync)`
        Where `<EXPORT>` is the directory you want to export.
     2. Start/restart the NFS server.
  3. Mount the network file system share to an available directory before installation on UNIX remote agents and web servers. Make sure to mount the share to the same directory across all machines, henceforth referred to as `<UNIX_PLUGINS>`.
     1. Create `<UNIX_PLUGINS>`.
     2. Add the following entry to `/etc/fstab`:
        `<HOST>:<EXPORT> <UNIX_PLUGINS> nfs defaults 0 0`
        Where `<HOST>` is the host name of the machine on which the directory has been exported and `<EXPORT>` is the directory being exported on that machine.
     3. Call: `mount -a`.

2. Perform the following steps if your ElectricFlow server is a Linux machine and you are installing remote agents or web servers on other UNIX machines. The plugins directory is not automatically shared as on Windows.
   1. Export the local plugins directory as a network file system share on the ElectricFlow server machine:
      1. Add the following entry to `/etc/exports` (`/opt/electriccloud/electriccommander/plugins` by default):
         `<EXPORT> *(ro,sync)`
         Where `<EXPORT>` is the directory you want to export.
      2. Start/restart the NFS server.
   2. Mount the network file system share to an available directory before installation on UNIX remote agents and web servers. You must mount the share to the same directory across all machines. This share is referred to as `<UNIX_PLUGINS>` in the following steps:
      1. Create `<UNIX_PLUGINS>`.
      2. Add the following entry to `/etc/fstab` (replace `<HOST>` with the host name of the machine on which the directory has been exported and `<EXPORT>` with the directory being exported on that machine):

         ```
         <HOST>:<EXPORT> <UNIX_PLUGINS> nfs defaults 0 0
         ```

      3. Call: `mount -a`.
3. Choose the appropriate step to configure remote agents or web servers.

- If you need to install the software for a new remote agent or web server, you must select the "**discover the plugins directory**" option. This will allow the installer to automatically detect and set the correct location from the server.
- If you need to configure a remote web servers that were already installed, run the following command:
  `ecconfigure --webPluginsDirectory "<PLUGINS>"`
  Where `<PLUGINS>` is the Windows path to the shared directory for Windows machines, or the UNIX path to the shared directory for UNIX machines.

Replicating the Plugin Directory on Remote Systems

Use this procedure to keep the Plugins directory in its default server location and replicate the contents to remote agents and web servers.

> **Important:** This approach requires you to manage multiple plugin directories. Every time a new plugin is installed on the ElectricFlow server, you must synchronize the changes across all remote copies of the directory. Only replicate the Plugins directory if you cannot use or configure a central network location.

- Copy the plugins directory to remote agents and web servers using any file copy mechanism.
  - The copied plugins directories must be readable by only the remote agents and web servers.
  - Plugins should be copied to a plugins subdirectory within the data directory for each remote agent and web server.
  - Every time the ElectricFlow server Plugins directory is updated, you must synchronize the changes across all remote copies.

Network Plugins Shares for High-Availability ElectricFlow Components

This section explains how to mount network plugins share files when you are setting up ElectricFlow components for high availability. This section contains the steps to set up an ElectricFlow server cluster (two load-balanced ElectricFlow servers and two load-balanced ElectricFlow web servers) sharing the same plugin folders (a "network plugins share").

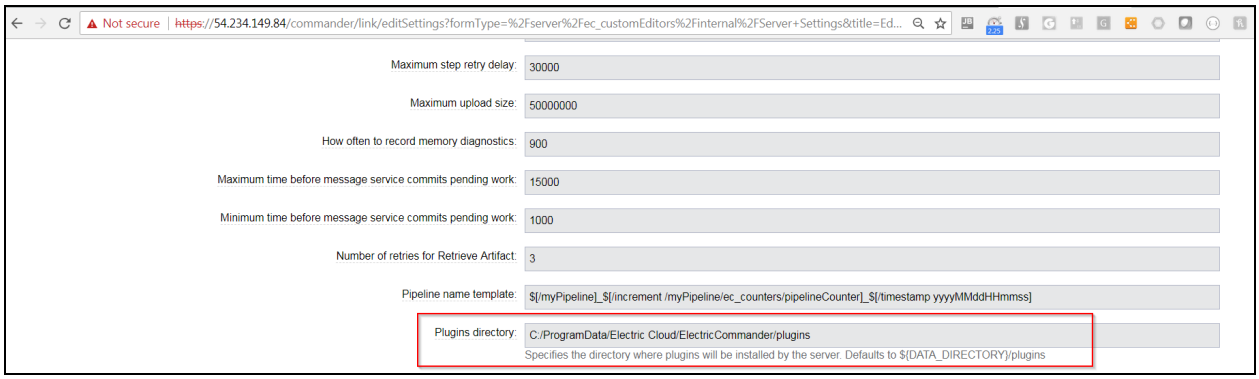
This setup consists of the following main steps:

1. Making the ElectricFlow Components Use the Remote /mymountpoint/plugins as the Plugins Folder on page 5-25
2. Using the Same Network Plugins Share for the ElectricFlow Web Server and the ElectricFlow Server on page 5-26
3. Network Plugins Shares for High-Availability ElectricFlow Components on page 5-24
4. Network Plugins Shares for High-Availability ElectricFlow Components on page 5-24
5. Restarting the ElectricFlow Web Server on page 5-27

Making the ElectricFlow Components Use the Remote /mymountpoint/plugins as the Plugins Folder

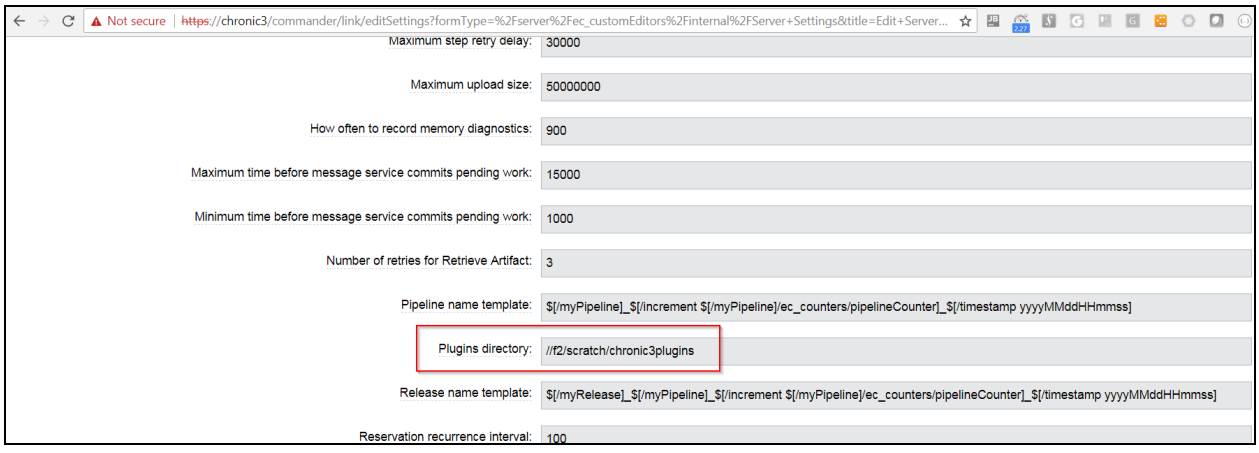
This section describes how to make the ElectricFlow components (ElectricFlow server, ElectricFlow web server, and agent) use the remote `/mymountpoint/plugins` as the plugins folder as mentioned in Moving the Plugins Directory to a Pre-Configured Network Location).

The plugins directory in the Automation Platform **Administration > Server > Settings** page (such as `C:/ProgramData/Electric Cloud/ElectricCommander/plugins` on Windows or `/opt/electriccloud/electriccommander/plugins` on Linux) need not be changed if the ElectricFlow server is standalone and is using its local plugins folder. This is the default location where the plugins are installed by the ElectricFlow server (the *DATA_DIR*`/plugins` folder by default).



But if the ElectricFlow server is clustered, you should use the plugins directory from a network plugins share and copy the contents of the *DATA_DIR*`/plugins` folder (and subfolders) from the ElectricFlow node 1 to this network plugins share.

Below is an example where an ElectricFlow server (on Windows) is using the network plugins share from a network UNC path `//f2/scratch/chronic3plugins` as seen in the Automation Platform **Administration > Server > Settings** page:



You can also set this network plugins share using the following `ectool` commands (after logging in via `ectool`):

```
ectool --server FLOW_SERVER_LOAD_BALANCER login admin changeme
```

```
ectool setProperty "/server/settings/pluginsDirectory" "/mymountpoint/plugins"
```

Note:

Never mount the plugins folder used by the CloudBees Flow server and web server to the CloudBees Flow *DATA_DIR*/plugins folder, although you might be tempted to do this to avoid changing the /server/settings/pluginsDirectory server property (via the setting in the **Administration>Server>Settings>Plugins** dialog in the Automation Platform (which defaults to *DATA_DIR*/plugins). If you do so, then when doing an upgrade of the CloudBees Flow server (say from version 9.0.1.135920 to 9.1.0.135893) the installer will try to rename *DATA_DIR* (for example, it will try to rename /opt/electriccloud/electriccommander to /opt/electriccloud/electriccommander-9.0.1.135920) and then the new /opt/electriccloud/electriccommander for 9.1.0.135893 will be created. If the *DATA_DIR*/plugins folder is NFS-mounted, the installer cannot access it and will fail.

The CloudBees Flow server user must have read and write access to the "network plugins share" (/*mymountpoint*/plugins). During a CloudBees Flow server upgrade, because the /server/settings/pluginsDirectory is set to /*mymountpoint*/plugins, when the CloudBees Flow server starts, it installs the new plugins in the /*mymountpoint*/plugins network share folder, which will then be available to the remaining HA nodes and the web servers.

Using the Same Network Plugins Share for the ElectricFlow Web Server and the ElectricFlow Server

If the local or remote web server will use the same network plugins share as the ElectricFlow server, then run the command described below on the web server.

The web server can be local or remote to the ElectricFlow server. This updates the *DATA_DIR*/apache/conf/httpd.conf file (for example, on Linux, it is /opt/electriccloud/electriccommander/apache/conf/httpd.conf by default).

Linux Web Servers

```
ecconfigure --webPluginsDirectory PLUGINS_SHARE_FOLDER
```

For example:

```
ecconfigure --webPluginsDirectory /pluginmount/plugins
```

Windows Web Servers

If the ElectricFlow web servers are on Windows, you must use the UNC path. For example:

```
ecconfigure --webPluginsDirectory //f2/scratch/chronic3plugins
```

Running the ecconfigure –webPluginsDirectory command updates the *DATA_DIR*/apache/conf/httpd.conf file as shown below:

```
ecbuild@perftest1:/opt/electriccloud/electriccommander/apache/conf$ grep plugin httpd.conf
# Commander plugins cgi-bin
# Change this location if plugins are not in the default location.
<Directory "/mymountpoint/plugins/*/cgi-bin">
ScriptAliasMatch ^/commander/plugins/([^/]*)/cgi-bin/(.*)$ "/mymountpoint/plugins/$1/cgi-bin/$2"
# Commander plugins htdocs
# Change this location if plugins are not in the default location.
<Directory "/mymountpoint/plugins/*/htdocs">
AliasMatch ^/commander/plugins/([^/]*)/(.*)$ "/mymountpoint/plugins/$1/htdocs/$2"
    RewriteRule ^/commander/pages/([^/]+)/(.*)$ "%{DOCUMENT_ROOT}/commander/componentContainer.php?pluginName=${escape:$1}&fileName=${escape:$2}.xml&%{QUERY_STR
ING}"
    # Handle unresolved plugins style urls
    RewriteRule ^/commander/plugins/([^/]+)/(.*)$ "%{DOCUMENT_ROOT}/commander/pluginHandler.php/$1/$2?%{QUERY_STRING}"
SetEnv COMMANDER_PLUGINS "/mymountpoint/plugins"
```

The `ecconfigure --agentPluginsDirectory` command updates the *DATA_DIR*`/conf/agent.conf` file and the *DATA_DIR*`/conf/agent/agent.properties` file as in the following screenshot. But note that these entries are no longer used for most plugins.

```
ecbuild@perftest1:/opt/electriccloud/electriccommander/conf$ grep plugin agent.conf
# Directory containing installed plugins.
pluginsPath = /mymountpoint/plugins
```

Restarting the ElectricFlow Web Server

On each ElectricFlow web server, if you run any of the above `ecconfigure` commands, you must restart it as `sudo` using:

```
sudo /etc/init.d/commanderApache restart
```

Running `commanderApache` using `sudo` does not change the permissions of the files in *DATA_DIR*`/apache/logs` such as the `access.log`, `error.log`, and `httpd.pid` files to `root`. They will still be owned by the user specified in the `--unixServerUser` option (the service user) used when the web server was installed.

Configuring Single Sign-On

You can integrate ElectricFlow with the Kerberos enterprise single sign-on (SSO) solutions, so that users are not presented with the ElectricFlow login screen when they want to take action in ElectricFlow. SSO improves the end user experience, so that users do not need to remember multiple credentials or log in multiple times. SSO also improves overall enterprise security, because most SSO solutions support features such as multi-factor authentication and stricter password policies.

SSO setup is recommended for occasional ElectricFlow users (that is, users who do not regularly spend long periods of time in ElectricFlow). These end users typically use ElectricFlow to start or check progress of releases, pipelines, or deployments, approve gates or complete manual tasks, look at dashboards, and so on.

Prerequisites

Before you configure single sign-on, make sure that:

- Kerberos software is installed and configured on the server and client nodes.
- Your system administrator has set up one or more Kerberos Key Distribution Centers (KDCs), and that each KDC is accessible from every node in your environment.
- Kerberos client software is installed on all hosts that are involved in Kerberos authentication.

  This software is required to communicate with the KDC server but is not included in ElectricFlow. A valid Kerberos configuration (such as a `krb5.conf` file) that includes information for how to connect to the KDC, realm, and domain must be provided for the client.

  Note that Windows clients have Kerberos authentication built into the authentication process, so there is no need for additional software.

Example Data for Kerberos Configuration

Component	Value
Active Directory domain	`example.com`
Kerberos realm	`EXAMPLE.COM`
ElectricFlow web server service	`efwebserver.example.com`
ElectricFlow web server Service Principal Name	`HTTP/efwebserver.example.com@EXAMPLE.COM`
ElectricFlow web server service account	`efweb-krbsvc`
ElectricFlow server service	`efserver.example.com`
ElectricFlow server Service Principal Name	`HTTP/efserver.example.com@EXAMPLE.COM`
ElectricFlow server service account	`efserver-krbsvc`
End-user account	`bob`

Configuring Kerberos with Active Directory

Before setting up your ElectricFlow servers and ElectricFlow web servers with single sign-on, Kerberos principals that are required for authentication must be configured with Active Directory. Administrator privileges are needed for the following procedures.

Creating an End-User Account in the Active Directory Domain

To create this account:

1. Log into the domain controller as administrator.
2. Create an account with a username (for example, `bob`) and a password.

   An organization typically already has its end user set up in Active Directory.

Creating Service Accounts in the Active Directory Domain

Service accounts are used to run the services for the ElectricFlow server and the ElectricFlow web server. See Example Data for Kerberos Configuration on page 5-28 above. You can run both services under the same account.

To set up these service accounts:

1. Login to the domain controller as administrator.
2. Create a user with a username and a password.

   Select the **User cannot change password** and **Password never expires** check boxes.

Configuring Service Accounts for Kerberos Delegation

Kerberos delegation allows an application (in this case, ElectricFlow services) to reuse the end-user credentials to access resources hosted on a different server. Delegation is not enabled by default in

Active Directory. This means that service accounts for the ElectricFlow web server and server services need to be set up and trusted for delegation.

To set up Kerberos delegation:

1. Login to the domain controller as administrator.
2. Open the Windows Active Directory Users and Computers tool.
3. Find the service account that you created (for example, `efweb-krbsvc`).
4. Open user properties and navigate to the **Delegation** tab.
5. Select the **Trust this user for delegation to any service (Kerberos only)** option.
6. Repeat these steps for the other service account that you created (for example, `efserver-krbsvc`).

Mapping the User Accounts to Service Principal Names

Kerberos service principals for the ElectricFlow web server and ElectricFlow server services need to be created and associated with the service user accounts in Active Directory. This operation can be performed either through the Windows Active Directory Users and Computers administrator tool or through the `setspn` utility.

Using Windows Active Directory Users and Computers

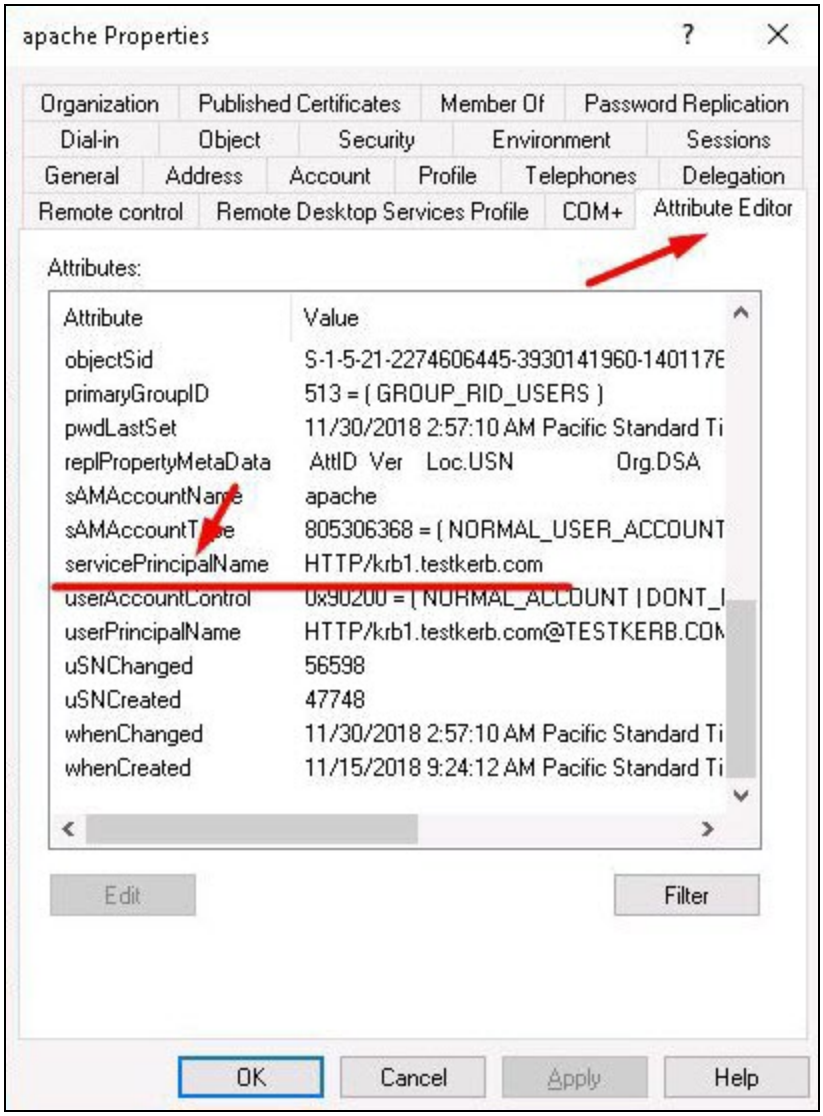
1. Log into the domain controller as administrator.
2. Open the Windows Active Directory Users and Computers tool.
3. Find the user account that you created (for example, `efweb-krbsvc`).
4. Open user properties and navigate to the Attribute Editor.

   If the Attribute Editor does not appear in the UI, open **View** > **Advanced Features** and then enable it. For details, see https://activedirectoryfaq.com/2014/10/ad-attribute-editor-missing-make-search-visible/.
5. Select the **servicePrincipalName** attribute.
6. Enter the value `HTTP/efwebserver.example.com@EXAMPLE.COM`, where `efwebserver.example.com` is the fully qualified domain name (FQDN) for the web server node.

   When a load balancer is used for the web server, this value should be the FQDN of the load balancer.

7. Repeat these steps for the ElectricFlow server to associate the user account `efserver-krbsvc` to the service principal `HTTP/efserver.example.com@EXAMPLE.COM`.

   In the following example, the Service Principal Name is `HTTP/krb1.testkerb.comand`, and the user account is named `apache`:



Using the setspn Utility

The `setspn` utility lets you manipulate Service Principal Names (SPNs) in Active Directory.

1. Associate the Service Principal Name for the web server to its service account by entering:

   ```
   C:\ setspn -s HTTP/efwebserver.example.com@EXAMPLE.COM efweb-krbsvc
   ```

2. Associate the Service Principal Name for the ElectricFlow server to its service account by entering:

   ```
   C:\ setspn -s HTTP/efserver.example.com@EXAMPLE.COM efserver-krbsvc
   ```

Generating a Keytab File for ElectricFlow Services

Keytab files need to be generated for the ElectricFlow server and ElectricFlow web server services. A keytab file stores the encryption keys for the Kerberos Services Principals (for example, `HTTP/efwebserver.example.com@EXAMPLE.COM` for a web server). The service account's password is used to encrypt the entries in this file.

> **Tip:** You can also generate a keytab file by using the Automation Platform UI. For details, see the "Single Sign-On" section in the "Automation Platform" chapter of the *ElectricFlow User Guide* at http://docs.electric-cloud.com/eflow_doc/FlowIndex.html.

To generate keytab files for the ElectricFlow server and ElectricFlow web server services:

1. Create a keytab file for the ElectricFlow server that will store the credentials or encryption keys for its Service Principal Name (`HTTP/efserver.example.com@EXAMPLE.COM`) that is associated with the service account (`efserver-krbsvc`).

   To create a keytab file, enter the following `ktpass` command on domain controller node with administrative privileges:

   ```
   C:\ ktpass -princ HTTP/efserver.example.com@EXAMPLE.COM -mapuser
   EXAMPLE\efserver-krbsvc -pass password -setpass -setupn +dumpsalt -crypto all -
   ptype KRB5_NT_PRINCIPAL -out efserver.keytab
   ```

   The password must match the one used to create the service account for the ElectricFlow server.

2. Create a keytab file for the ElectricFlow web server that will store the credentials or encryption keys for its Service Principal Name (`HTTP/efwebserver.example.com@EXAMPLE.COM`) that is associated with the service account (`efweb-krbsvc`).

   To create a keytab file, enter the following `ktpass` command on the domain controller node with administrator privileges:

   ```
   C:\ ktpass -princ HTTP/efwebserver.example.com@EXAMPLE.COM -mapuser
   EXAMPLE\efweb-krbsvc -pass password -setpass -setupn +dumpsalt -crypto all -
   ptype KRB5_NT_PRINCIPAL -out efwebserver.keytab
   ```

   The password must match the one used to create the service account for the ElectricFlow web server.

Configuring Web Browsers for Single Sign-On

This section shows individual end users how to configure their web browsers for Kerberos. So that users can access server resources that require Kerberos authentication, their browser must be enabled to send Kerberos credentials. ElectricFlow supports the following browsers:

- Google Chrome
- Microsoft Internet Explorer
- Mozilla Firefox

Configuring Chrome for Kerberos

Windows

Chrome on Windows should use Internet Explorer settings. For instructions, see Configuring Internet Explorer for Kerberos on page 5-33.

Ensure that the registry on your machine is properly set up. The recommended way to configure a policy on Windows is by using a Group Policy Object (GPO). However, on machines that are joined to an Active Directory domain, policy settings might also be stored in the registry under `HKEY_LOCAL_MACHINE` or `HKEY_CURRENT_USER` in the following paths:

- `Software\Policies\Google\Chrome\AuthServerWhitelist = efwebserver.example.com`
- `Software\Policies\Google\Chrome\AuthNegotiateDelegateWhitelist = efwebserver.example.com`

For details, see https://dev.chromium.org/administrators/policy-list-3#AuthServerWhitelist and https://dev.chromium.org/administrators/policy-list-3#AuthNegotiateDelegateWhitelist.

MacOS

To configure Chrome for Kerberos on macOS, complete the following steps. The example below is for a user named `bob`:

1. Enter:

   ```
   /usr/bin/defaults write /Users/bob/Library/Preferences/com.google.Chrome.plist
   AuthNegotiateDelegateWhitelist "efwebserver.example.com"
   ```

2. Enter:

   ```
   /usr/bin/defaults write /Users/bob/Library/Preferences/com.google.Chrome.plist
   AuthServerWhitelist "efwebserver.example.com"
   ```

3. Log out of your user account.
4. Log back in, and run `kinit bob@EXAMPLE.COM`.

   This generates a Kerberos ticket on your computer.

5. Confirm by running the `klist` command.

   The list of Kerberos tickets appears.

6. Shut down and restart Chrome.

For more information about Chrome setup for Kerberos, see https://www.jamf.com/jamf-nation/discussions/10688/chrome-and-kerberos-single-sign-on.

Following are common problems with Kerberos setup on Chrome on macOS:

- SSO login no longer works.

  Follow the steps below:

  1. Log out of your user account and log back in.
  2. On macOS, enter `kinit bob@EXAMPLE.COM` and generate a ticket for the `krbtgt` account again.

3. Ensure that the Kerberos ticket is not expired on your local machine.

   On macOS, run the `klist` command and see the Kerberos ticket expiry information.

- SSO does not work on Chrome.

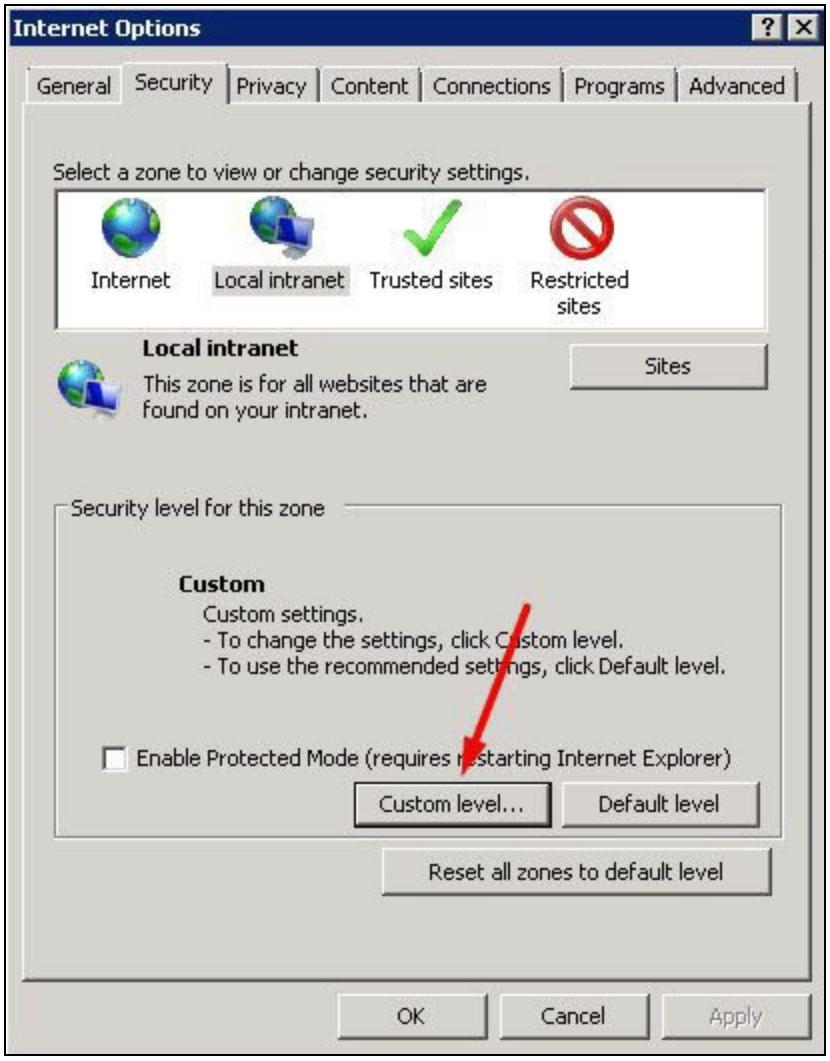
  Follow the steps below:

  1. Ensure that registry entries are correct as stated above.
  2. Ensure that there are no spaces before or after the name of the registry entry `AuthNegotiateDelegateWhitelist`.

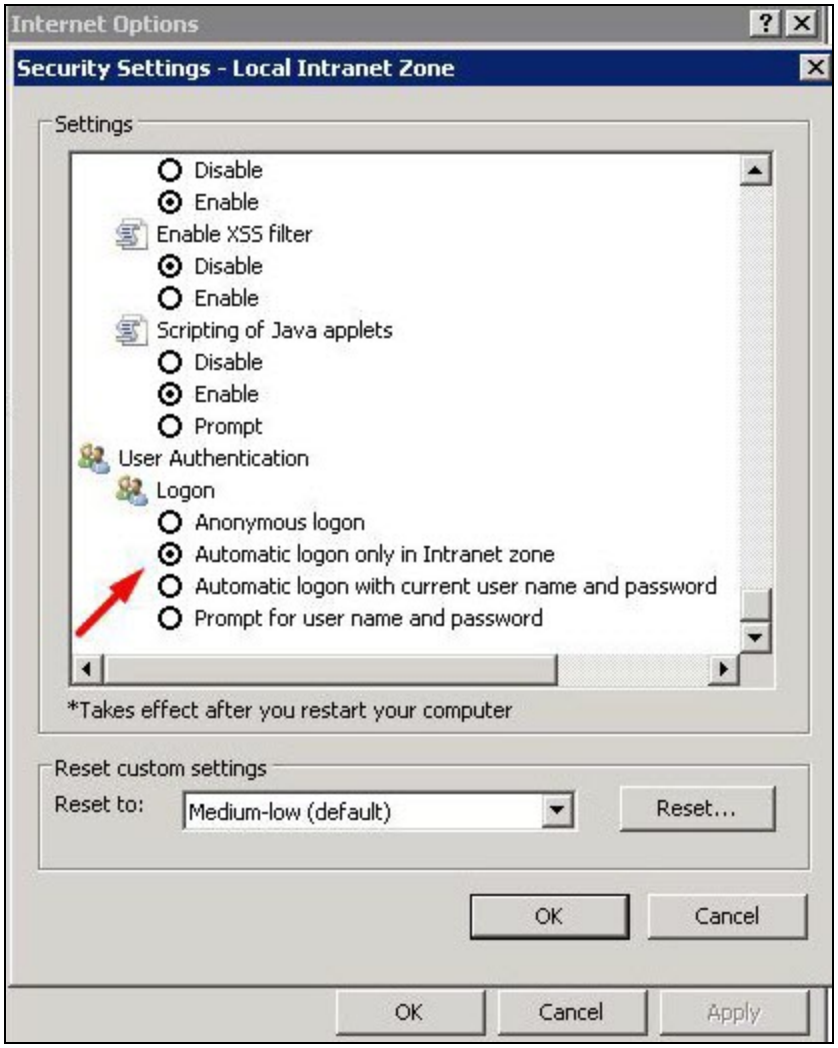
Configuring Internet Explorer for Kerberos

To configure internet Explorer for Kerberos:

1. Navigate to **Settings** > **Internet Options** > **Security**.
2. Click the **Custom Level** button:

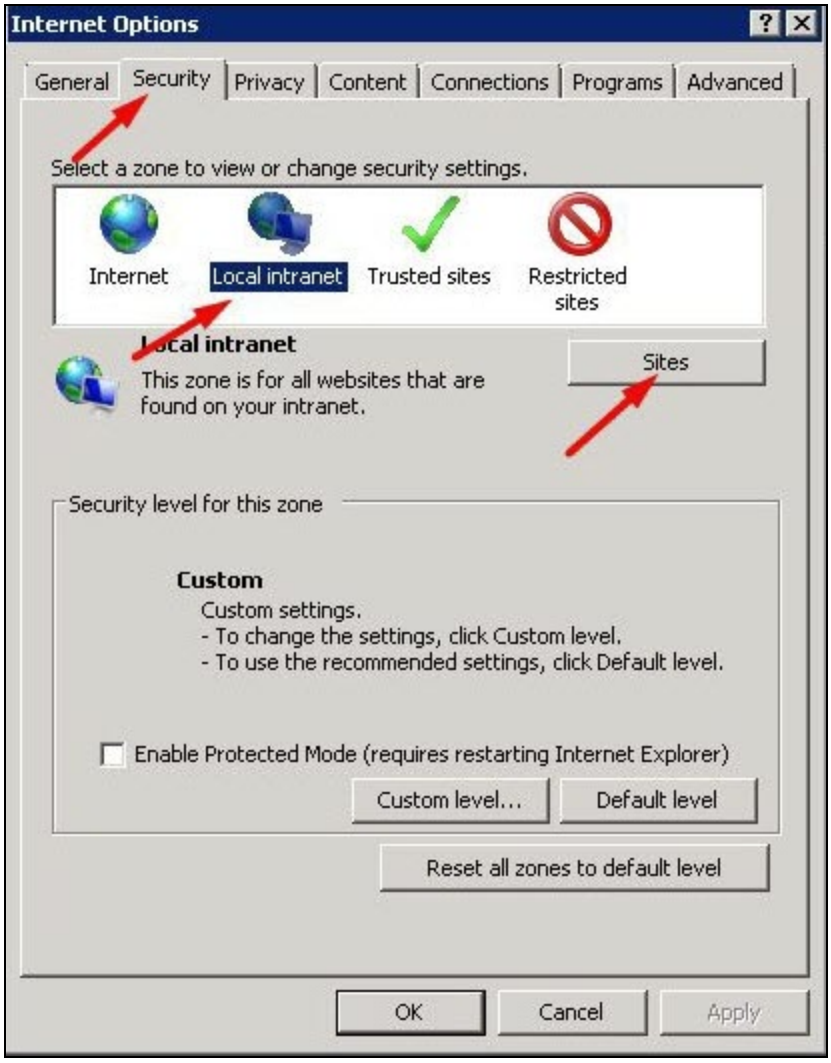


3. Under **User Authentication** > **Logon**, select **Automatic logon only in Intranet zone**:

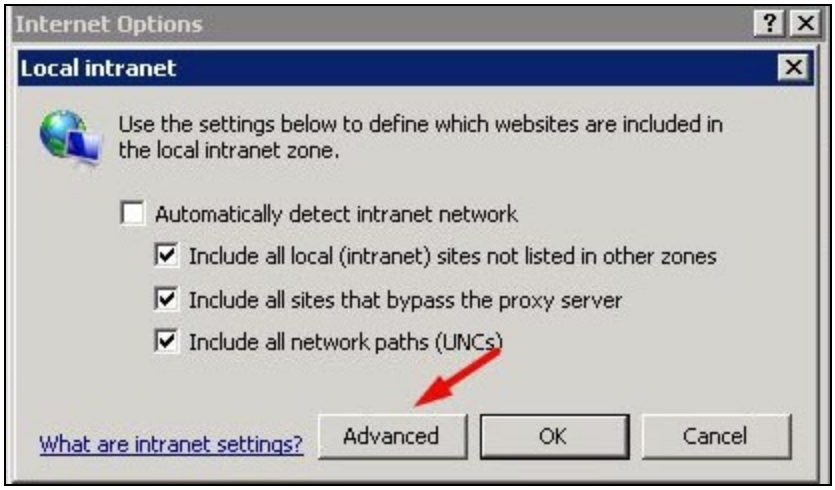


4. Click **OK**.

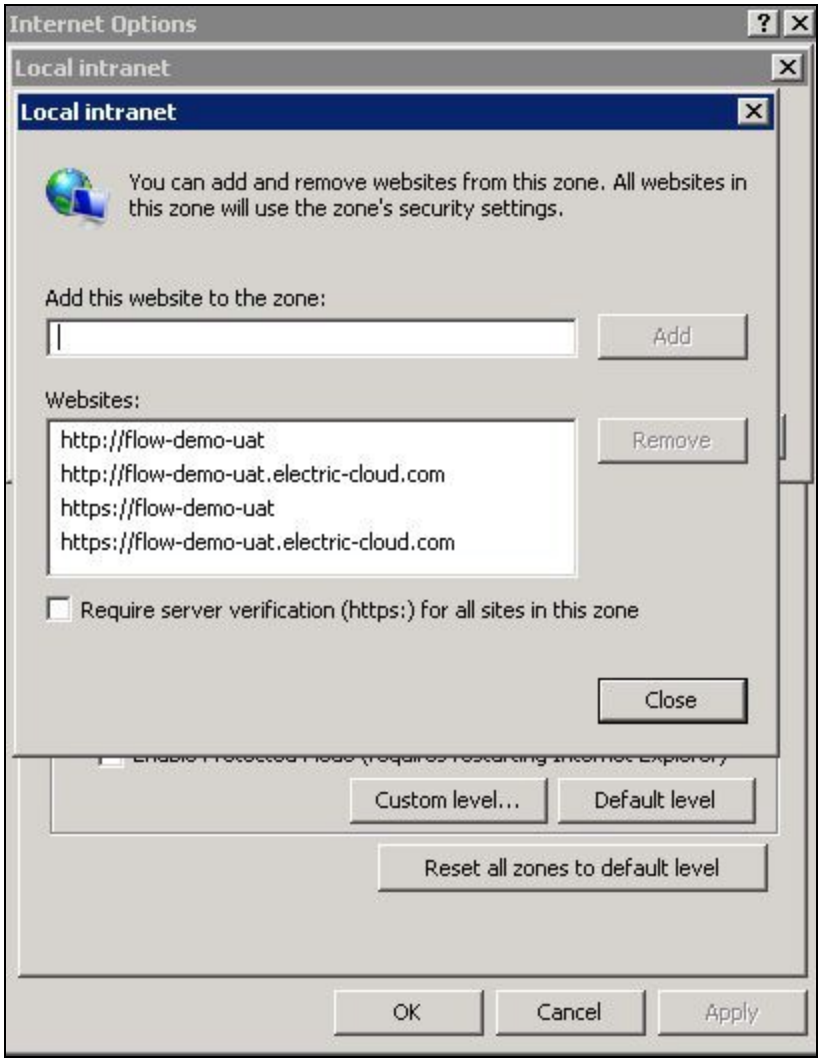
5. On the **Security** tab, select **Local Intranet**, and then click **Sites**:



6. On the **Sites** popup window, select all options (this is default):



7. Click the **Advanced** button and add the ElectricFlow web server FQDN to your local intranet zone:



8. Click **Add**, and then **Close**.

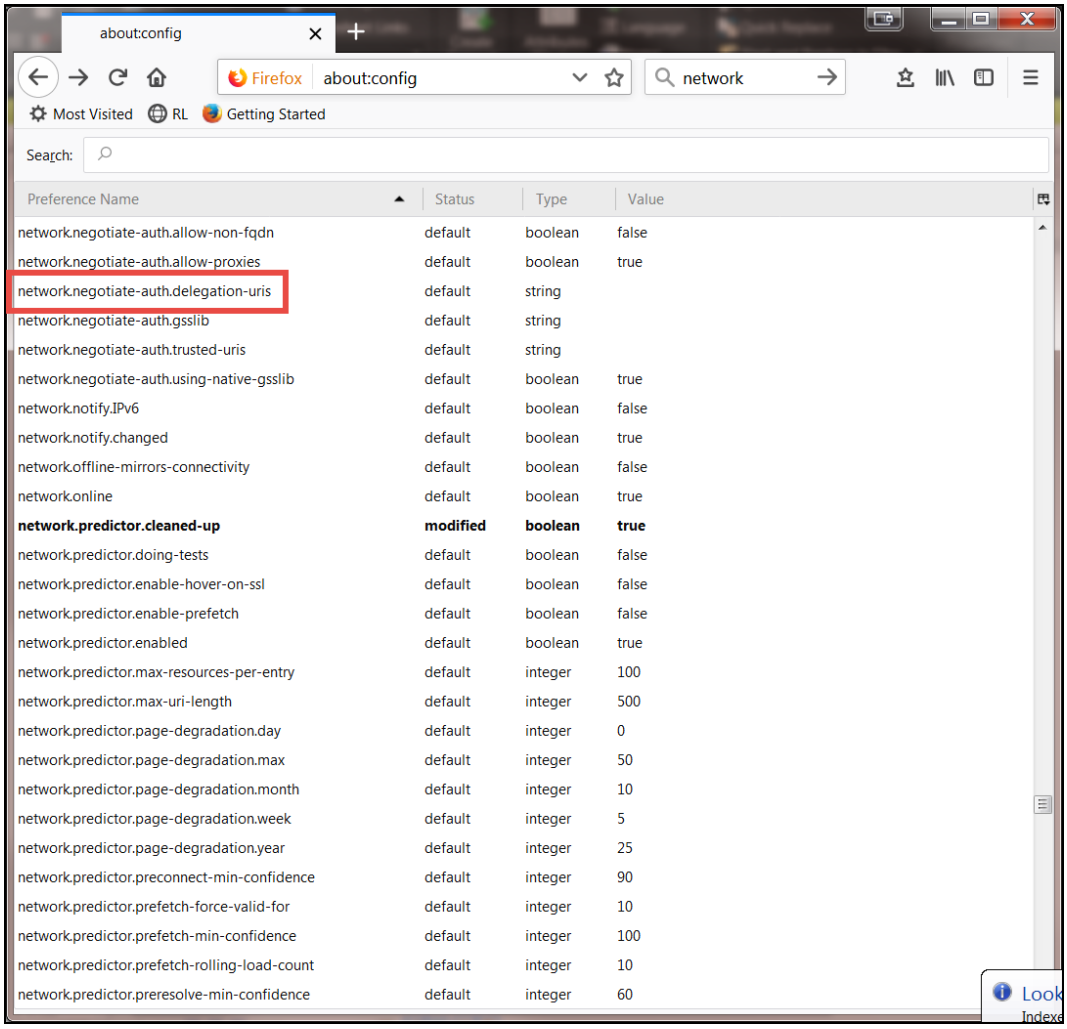
Configuring Firefox for Kerberos

To configure Firefox for Kerberos:

1. Navigate to about:config.
2. Click the **I accept the risk!** button.

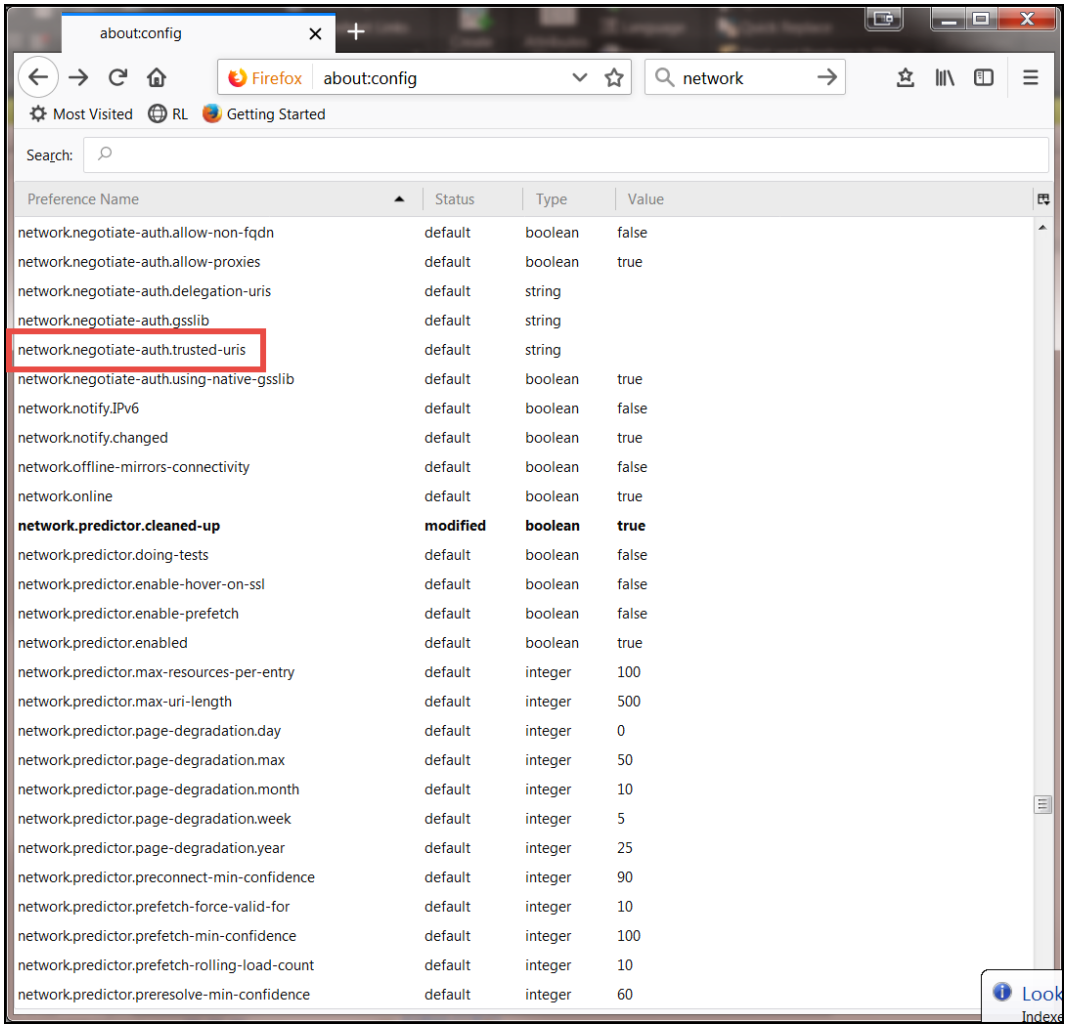
   The list of preferences appears.

3. Double-click the **network.negotiate-auth.delegation-uris** preference:



And then enter the value for the ElectricFlow web server FQDN. For example, enter `efwebserver.example.com`. This preference lists the sites for which the browser may delegate user authorization to the server.

4. Double-click the **network.negotiate-auth.trusted-uris** preference:



And then enter host or domain names (delimited by commas). Note that you can use a wildcard for domain names by prefixing them with a dot. For example, `.example.com`. For the ElectricFlow web server FQDN, you can provide the `efwebserver.example.com` value for this preference.

5. Click **OK**, and then close the browser window.

Configuring ElectricFlow for Single Sign-On

After Kerberos is installed and configured, and service accounts and Service Principal Names are created in Active Directory, you must enable single sign-on in ElectricFlow. For details, see the "Single Sign-On" section in the "Automation Platform" chapter of the *ElectricFlow User Guide* at http://docs.electric-cloud.com/eflow_doc/FlowIndex.html.

End-User Login Flow for Single Sign-On

For information about how end users will log into ElectricFlow using single sign-on, see the "Logging in to ElectricFlow" section in the "Introduction to ElectricFlow" chapter of the *ElectricFlow User Guide* at

http://docs.electric-cloud.com/eflow_doc/FlowIndex.html.

Configuring the Web and Repository Servers to Use IPv4

The ElectricFlow web server and the ElectricFlow repository server use IPv6 IP addresses by default. This section describes how to restrict these services to use IPv4 IP addresses regardless of what your platform and Apache Portable Runtime (APR) will support. For more information about binding to addresses and ports, see https://httpd.apache.org/docs/2.4/bind.html.

Configuring the Web Server to Use IPv4

To restrict the ElectricFlow web server to handle IPv4 connections only:

1. Open the *DATA_DIR*/apache/conf/httpd.conf file.

   The default location is:

   - (Linux) /opt/electriccloud/electriccommander/apache/conf/httpd.conf
   - (Windows) C:\ProgramData\Electric Cloud\ElectricCommander\apache\conf\httpd.conf

2. Specify an IPv4 address on the Listen directive by changing the following line:

   ```
   Listen 80
   ```

   to

   ```
   Listen 0.0.0.0:80
   ```

3. Open the *DATA_DIR*/apache/conf/ssl.conf file.

   The default location is:

   - (Linux) /opt/electriccloud/electriccommander/apache/conf/ssl.conf
   - (Windows) C:\ProgramData\Electri Cloud\ElectricCommander\apache\conf\ssl.conf

4. Specify an IPv4 address on the Listen directive by changing the following line:

   ```
   Listen 443
   ```

   to

   ```
   Listen 0.0.0.0:443
   ```

5. Restart the ElectricFlow web service (CommanderApache to make the change take effect.

   For instructions, see Starting All ElectricFlow Server Services on page 12-21.

Configuring the Repository Server to Use IPv4

To restrict the ElectricFlow repository server to handle IPv4 connections only:

1. Open the *DATA_DIR*/conf/repository/wrapper.conf file.

   The default location is:

- (Linux) `/opt/electriccloud/electriccommander/conf/repository/wrapper.conf`
- (Windows) `C:\ProgramData\Electric Cloud\ElectricCommander\conf\repository\wrapper.conf`

2. At the end of the file, add the following line:

   `wrapper.java.additional.nnn=-Djava.net.preferIPv4Stack=true`

   For example:

   `wrapper.java.additional.300=-Djava.net.preferIPv4Stack=true`

   nnn must be a unique number in that file.

3. Restart the ElectricFlow repository service (`commanderRepository`) to make the change take effect.

   For instructions, see Starting All ElectricFlow Server Services on page 12-21.

Configuring ElectricFlow Server-to-Agent Communication to Use Trusted HTTPS

Communication between the ElectricFlow server and agents can be configured as HTTP, HTTPS, or Trusted HTTPS. Trusted HTTPS implements mutual authentication between the server and the agent. Use the following procedures to configure Trusted HTTPS between the ElectricFlow server and agent using internal CA-signed certificates.

Prerequisites

The procedures below assume that:

- You have the following files (in PEM format) for the ElectricFlow server and all ElectricFlow agent systems where you want to configure ElectricFlow server-to-agent communication as Trusted HTTPS:
  - The individual server private key file
  - A CA-signed server certificate
  - The root CA certificate
  - The intermediate CA certificate
- The above files reside in the `/opt/electriccloud/electriccommander/conf` directory.
- Java keytool is in the path of the user running the commands which follow. If not, run the following command to set it for the current shell:

  ```
  # export PATH=${PATH}:/opt/electriccloud/electriccommander/jre/bin
  ```

Configuring the ElectricFlow Server

On the ElectricFlow server, do the following:

1. Concatenate the intermediate and root CA certificates (in this order) into one file:

   ```
   # cat intermediate_cert.crt root_cert.crt  bundle.crt
   ```

2. Save a copy of the original self-signed keystore:

   ```
   # cp -p /opt/electriccloud/electriccommander/conf/keystore
   /opt/electriccloud/electriccommander/conf/keystore.orig
   ```

3. Create a PKCS12 keystore incorporating the certificate chain along with the SSL private key and server certificate:

   ```
   # openssl pkcs12 -export -chain -in electricflow_server_fqdn.cer -inkey
   electricflow_server_fqdn.key -out electricflow_server.p12 -name jetty -CAfile
   bundle.crt

   Enter pass phrase for electricflow_server.key: password

   Enter Export Password: password

   Verifying - Enter Export Password: password
   ```

4. Convert the PKCS12 keystore to a Java Keystore (JKS):

   ```
   # keytool -importkeystore -destkeystore keystore -srckeystore electricflow_
   server.p12 -alias jetty
   ```

5. Since the keystore also acts as the truststore, import the *root* CA certificate into the keystore (with an alias prefixed with `CA:`):

   ```
   # keytool -importcert -keystore keystore -alias CA:root_cert_alias -file root_
   cert.crt
   ```

6. Edit the server's properties file:

   ```
   #
   # SSL settings
   #
   COMMANDER_HTTPS_PORT=8443
   COMMANDER_KEYSTORE=conf/keystore
   COMMANDER_CRL_FILE=conf/security/crl.pem
   COMMANDER_KEY=conf/security/ca_pk.pem
   COMMANDER_CERT=conf/security/ca.pem
   COMMANDER_KEYSTORE_PASSWORD=keystore_password
   ```

7. Disable the Certificate Revocation List (CRL).

   This is necessary because the ElectricFlow Jetty server supports only one CRL and, in this environment, there are both root and intermediate CAs (meaning that two CA CRLs would need to be checked). Rename the default `crl.pem` file so that the one expected by the `agent.properties` file does not exist:

   ```
   # mv /opt/electriccloud/electriccommander/conf/security/crl.pem
   /opt/electriccloud/electriccommander/conf/security/crl.pem.orig
   ```

8. Regenerate the `/opt/electriccloud/electriccommander/conf/security/crl.pem` file using the new *root* CA:

   ```
   openssl ca -config config_file.cnf -gencrl -keyfile keyfile.pem -cert ca_
   certfile.pem -out crl.pem
   ```

9. Copy the `crl.pem` file to the `/opt/electriccloud/electriccommander/conf/agent/crl.pem` file of the Trusted agent.

   After you restart both the ElectricFlow server and the agent machine, the resource will be pingable as a Trusted agent.

10. Set the ownership and permissions on the keystore file:

    ```
    # chown eflow_server_user:eflow_server_user keystore
    # chmod 0600 keystore
    ```

11. Restart the ElectricFlow server while "tailing" the server log file to ensure the server starts correctly:

    a. Open a second SSH session to the ElectricFlow server and run the following command:

       ```
       # tail -f /opt/electriccloud/electriccommander/logs/commander.log
       ```

    b. In the first SSH session to the agent server, run the following commands:

       ```
       # /etc/init.d/commanderServer stop
       # /etc/init.d/commanderServer start
       ```

    c. Verify that no errors are reported in the server startup.

Configuring Each Agent Machine

On *each* ElectricFlow agent machine where you want to set up Trusted HTTPS, do the following:

1. Concatenate the intermediate and root CA certificates into one file:

   ```
   # cat intermediate_cert.crt root_cert.crt > bundle.crt
   ```

2. Save a copy of the original self-signed keystore:

   ```
   # cp -p /opt/electriccloud/electriccommander/conf/agent/keystore
   /opt/electriccloud/electriccommander/conf/agent/keystore.orig
   ```

3. Create a PKCS12 keystore incorporating the certificate chain along with the SSL private key and server certificate:

   ```
   # openssl pkcs12 -export -chain -in agent_hostname_fqdn.cer -inkey agent_
   hostname_fqdn.key -out agent_hostname.p12 -name jetty -CAfile bundle.crt

   Enter pass phrase for agent_hostname_fqdn.key: password

   Enter Export Password: password

   Verifying - Enter Export Password: password
   ```

4. Convert the PKCS12 keystore to a Java Keystore (JKS):

   ```
   # keytool -importkeystore -destkeystore keystore -srckeystore agent_hostname.p12
   -alias jetty
   ```

5. Because the keystore also acts as the truststore, import the *root* CA certificate into the keystore (with an alias prefixed with `CA:`):

   ```
   # keytool -importcert -keystore keystore -alias CA:root_cert_alias -file root_
   cert.crt
   ```

6. Edit the agent's properties file (`/opt/electriccloud/electriccommander/conf/agent/agent.properties`) and set the keystore password (if different from the default):

   ```
   AGENT_KEYSTORE=conf/agent/keystore
   AGENT_KEYSTORE_PASSWORD=keystore_password
   AGENT_CRL_FILE=conf/agent/crl.pem
   ```

7. Disable the Certificate Revocation List (CRL).

   This is necessary because the ElectricFlow Jetty server supports only one CRL and, in this environment, there are both root and intermediate CAs (meaning that two CA CRLs would need to be checked). Rename the default `crl.pem` file so that the one expected by the `agent.properties` file does not exist:

   ```
   # mv /opt/electriccloud/electriccommander/conf/agent/crl.pem
   /opt/electriccloud/electriccommander/conf/agent/crl.pem.orig
   ```

8. Regenerate the `/opt/electriccloud/electriccommander/conf/security/crl.pem` file using the new *root* CA:

   ```
   openssl ca -config config_file.cnf -gencrl -keyfile keyfile.pem -cert ca_
   certfile.pem -out crl.pem
   ```

9. Copy the `crl.pem` file to the `/opt/electriccloud/electriccommander/conf/agent/crl.pem` file of the Trusted agent.

   After you restart both the ElectricFlow server and the agent machine, the resource will be pingable as a Trusted agent.

10. Set the ownership and permissions on the keystore file:

    ```
    # chown eflow_agent_user:eflow_agent_user keystore
    # chmod 0600 keystore
    ```

11. Restart the ElectricFlow agent while "tailing" the agent log file to ensure the agent starts correctly:

    a. Open a second SSH session to the agent server and run the following command:

       ```
       # tail -f /opt/electriccloud/electriccommander/logs/agent/agent.log
       ```

    b. In the first SSH session to the agent server, run the following commands:

       ```
       # /etc/init.d/commanderAgent stop
       # /etc/init.d/commanderAgent start
       ```

    c. Verify that no errors are reported in the agent startup.

12. In the Automation Platform web interface, while logged in as an admin user, do the following to change the agent communication to Trusted and verify communication:
    a. Select **Cloud > Resources**.
    b. Locate the desired agent and click the its name (in the **Name** column).
    c. In the **Resource Details** window that appears, go to the top of the window and click **Edit**.
    d. In the **Edit Resource** window, scroll to the bottom, change the **Connection Type** from **HTTPS** to **Trusted HTTPS**, and click **OK**.
    e. In the **Cloud Resources** page, check the checkbox to the left of the agent you just configured.
    f. Click the bullseye icon on the button bar to ping the node.
    g. Verify that the **Job Status** shows a checkbox inside a green circle.

Environment Proxy Server Configuration

In your environment proxy servers might exist between an intranet and internet. Because proxy servers can inhibit certain types of internet access, you will need to set proxy settings for each impacted machine in your installation. ElectricFlow servers or web servers can be deployed behind a proxy server.

It is not a common practice to place repository servers or agent systems behind a proxy server since these systems communicate with ElectricFlow through an intranet connection.

Configuring Proxy Settings for Servers

Use `ecconfigure` to set proxy settings for any web server or ElectricFlow server in your configuration that is deployed behind a proxy server. Repository servers are not typically placed behind a proxy server.

1. Select the appropriate perl scripts to run depending on the server type.
    - To set ElectricFlow server proxy settings, enter:

      ```
      ec-perl src/ecconfigure.pl
      --serverProxyHost <IP_ADDRESS_PROXY>
      --serverProxyPort <PORT>
      --serverNoProxyHosts "<HOST1,HOST2>"
      ```

    - To set web server proxy settings, enter:

      ```
      ec-perl src/ecconfigure.pl
      --webProxyUrl http://<IP_ADDRESS:PORT>
      --webNoProxyHosts <HOST1,HOST2,HOST3>
      ```

      Where:

      <IP_ADDRESS_PROXY> is the IP address of the proxy server,

      <PORT> is the server port for the proxy server, and

      <HOST1,HOST2> is one or more comma separated host names for the servers in the configuration.

2. Restart all the servers where you have applied a proxy setting

Important: If you do not restart the servers, the proxy settings will not work.

Testing Server Proxy Settings

Use the following task to verify your proxy server settings.

- Perform the following steps depending on your server type.
  - If you have a web server:
    1. Go to the **Plugin Manager** web page.
    2. Verify the catalog can be viewed and no errors are reported when accessing the catalog URL.
  - If you have an ElectricFlow server:
    1. Go to the **Plugin Manager** web page.
    2. Verify you can install a plugin from the catalog.

Configuring Proxy Agents

Use `ecconfigure` to set proxy settings for any agent system that is deployed behind a proxy server. A proxy server is not usually placed between agents and an ElectricFlow server.

Important: When you use a proxy agent, the proxy target must run an SSH v2 server.

1. Run the following command to set Agent proxy settings:
   ```
   ec-perl src/ecconfigure.pl
   --agentProxyHost <IP_ADDRESS_PROXY>
   --agentProxyPort <PORT>
   --agentNoProxyHosts "<HOST1,HOST2>"
   ```

   Where:

   <IP_ADDRESS_PROXY> is the IP address of the proxy server,

   <PORT> is the server port for the proxy server, and

   <HOST1,HOST2> is one or more comma separated host names for the servers in the configuration.
2. Set the cygwin 1.7 privilege by running the following commands.

Note: Certain commands require administrator privileges to run (for example, `net stop xxx`) using cygwin 1.7 `sshd`. These commands can fail with "access denied" errors. These errors did not occur in cygwin 1.5. The ElectricFlow proxy agent relies on `sshd` being privileged. To set this privilege on cygwin 1.7, you must run an additional setup script (in addition to `ssh-host-config`).

1. `cyglsa-config`
2. `reboot`

Increasing File Descriptors for Linux and Linux Docker Containers

A file descriptor is an object that a process uses to read or write to an open file and open network sockets (although there are other uses).

Operating systems place limits on the number of file descriptors that a process can open. In addition to per-process limits, an OS also has a global limit on the number of file descriptors that all its processes, together, might consume.

A common bottleneck in the default Linux operating system configuration is a lack of file descriptors.

ElectricFlow Server

An ElectricFlow server uses approximately one file descriptor per running job step and three per uncompleted job.

The following example configures ElectricFlow to use a new limit of 32768:

1. Add the following line to the `init` script for the ElectricFlow Server (in `/etc/init.d/commander`) before the `su -` command:

   ```
   ulimit -n 32768
   ```

2. Restart the ElectricFlow server:

   ```
   /etc/init.d/commanderServer restart
   ```

ElectricFlow Agent

An ElectricFlow agent uses at least two file descriptors per running job step.

It is important to make sure that operating systems on high traffic sites are configured to provide sufficient numbers of file descriptors to ElectricFlow.

The following example describes how to raise the maximum number of file descriptors to 32768 for the ElectricFlow process on the Red Hat Linux distribution:

1. Allow all users to modify their file descriptor limits from an initial value of 1024 up to the maximum permitted value of 32768 by changing `/etc/security/limits.conf`. The following two lines should be part of the file contents:

   ```
   soft nofile 1024
   ```

   ```
   hard nofile 32768
   ```

2. In `/etc/pam.d/login`, add the following line if it does not already exist:

   ```
   session required pam_limits.so
   ```

3. Configure ElectricFlow to use the new limits. Add the following line to the `init` script for the ElectricFlow Agent (in `/etc/init.d/ecmdrAgent` or `/etc/init.d/commanderAgent`):

   ```
   ulimit -n 32768
   ```

4. Restart the ElectricFlow agent:

   ```
   /etc/init.d/commanderAgent restart
   ```

Adjusting Swappiness on Linux

For Java-based machines (ElectricFlow server, repository server, agent), You should adjust the swappiness kernel parameter to favor applications over disk cache.

- To favor applications 100% over disk cache use:

  ```
  sysctl -w vm.swappiness=0
  ```

  The default of 60 can result in significant delays during garbage collection if any i/o intensive process runs on the machine.

Chapter 6: Roadmap for Upgrading ElectricFlow

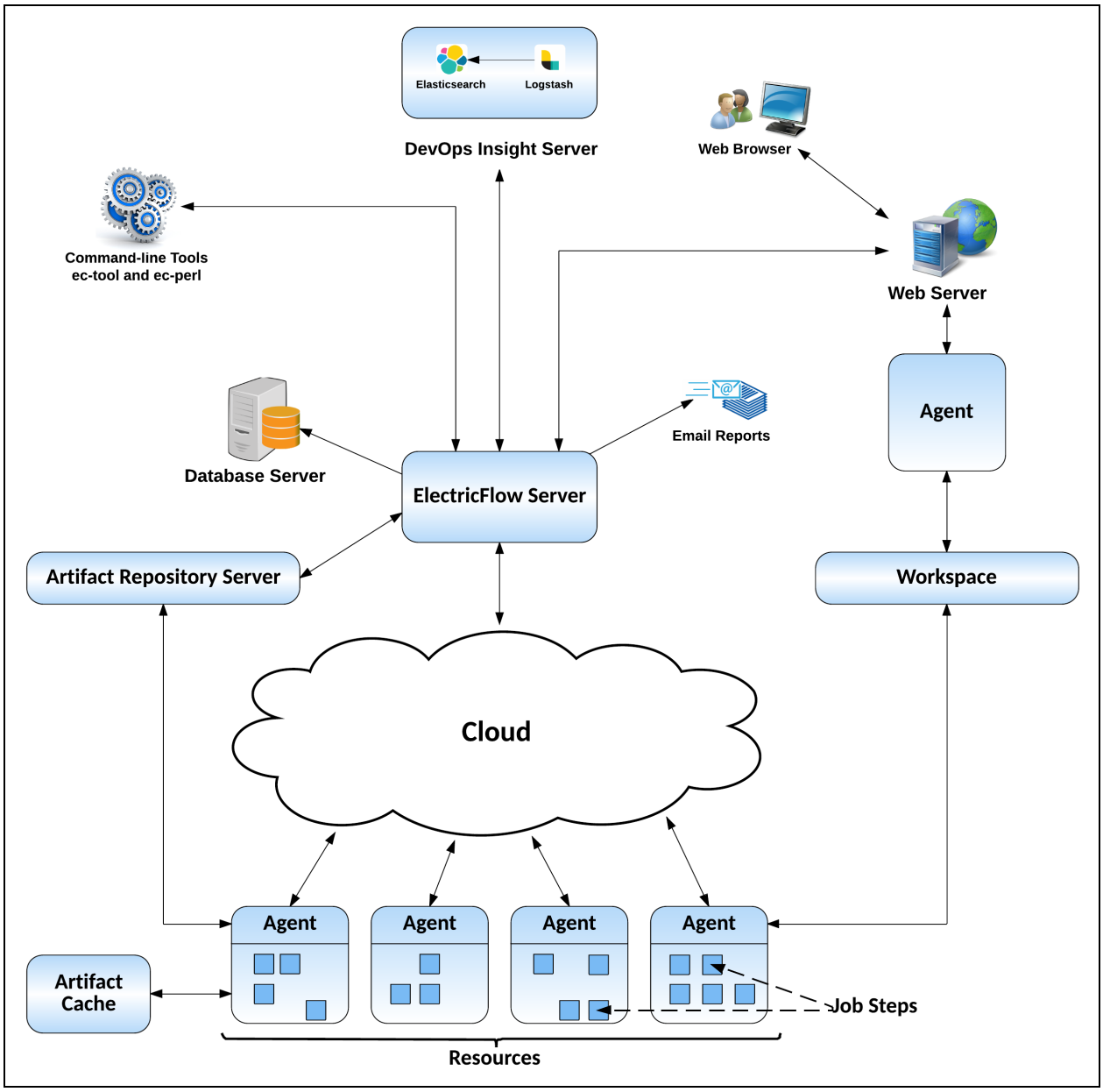
Note: Upgrading ElectricFlow 5.x, 6.x, 7.x, 8.0.x, and versions earlier than 8.3 that are using the built-in (default) database is not supported. This means that you must switch to an alternate ElectricFlow-supported database before your upgrade. For instructions, see the "Switching to an Alternate Database from the Built-In Database" section in the ElectricFlow Installation Guide at http://docs.electric-cloud.com/eflow_doc/FlowIndex.html.

Upgrade		File to Download	Run the Installer	Go to
From	To			
ElectricFlow 5.x	ElectricFlow 9.x, the latest version	`ElectricFlow-<version>`	Yes	Upgrading from ElectricFlow 5.x, 6.x, 7.x, or 8.x to 9.x on page 7-1
ElectricFlow 5.x for a clustered environment	ElectricFlow 9.x, the latest version, for a clustered environment	`ElectricFlow-<version>`	Yes	Upgrading from ElectricFlow 5.x, 6.x, 7.x, or 8.x to 9.x in a Clustered Environment on page 8-1
ElectricFlow 6.x	ElectricFlow 9.x, the latest (newer) version	`ElectricFlow-<version>`	Yes	Upgrading from ElectricFlow 5.x, 6.x, 7.x, or 8.x to 9.x on page 7-1
ElectricFlow 6.x for a clustered environment	ElectricFlow 9.x, the latest (newer) version, for a clustered environment	`ElectricFlow-<version>`	Yes	Upgrading from ElectricFlow 5.x, 6.x, 7.x, or 8.x to 9.x in a Clustered Environment on page 8-1
ElectricFlow 7.x	ElectricFlow 9.x, the latest (newer) version	`ElectricFlow-<version>`	Yes	Upgrading from ElectricFlow 5.x, 6.x, 7.x, or 8.x to 9.x on page 7-1

Upgrade		File to Download	Run the Installer	Go to
From	To			
ElectricFlow 7.x for a clustered environment	ElectricFlow 9.x, the latest (newer) version, for a clustered environment	`ElectricFlow-<version>`	Yes	Upgrading from ElectricFlow 5.x, 6.x, 7.x, or 8.x to 9.x in a Clustered Environment on page 8-1
ElectricFlow 8.x	ElectricFlow 9.x, the latest (newer) version	`ElectricFlow-<version>`	Yes	Upgrading from ElectricFlow 5.x, 6.x, 7.x, or 8.x to 9.x on page 7-1
ElectricFlow 8.x for a clustered environment	ElectricFlow 9.x, the latest (newer) version, for a clustered environment	`ElectricFlow-<version>`	Yes	Upgrading from ElectricFlow 5.x, 6.x, 7.x, or 8.x to 9.x in a Clustered Environment on page 8-1

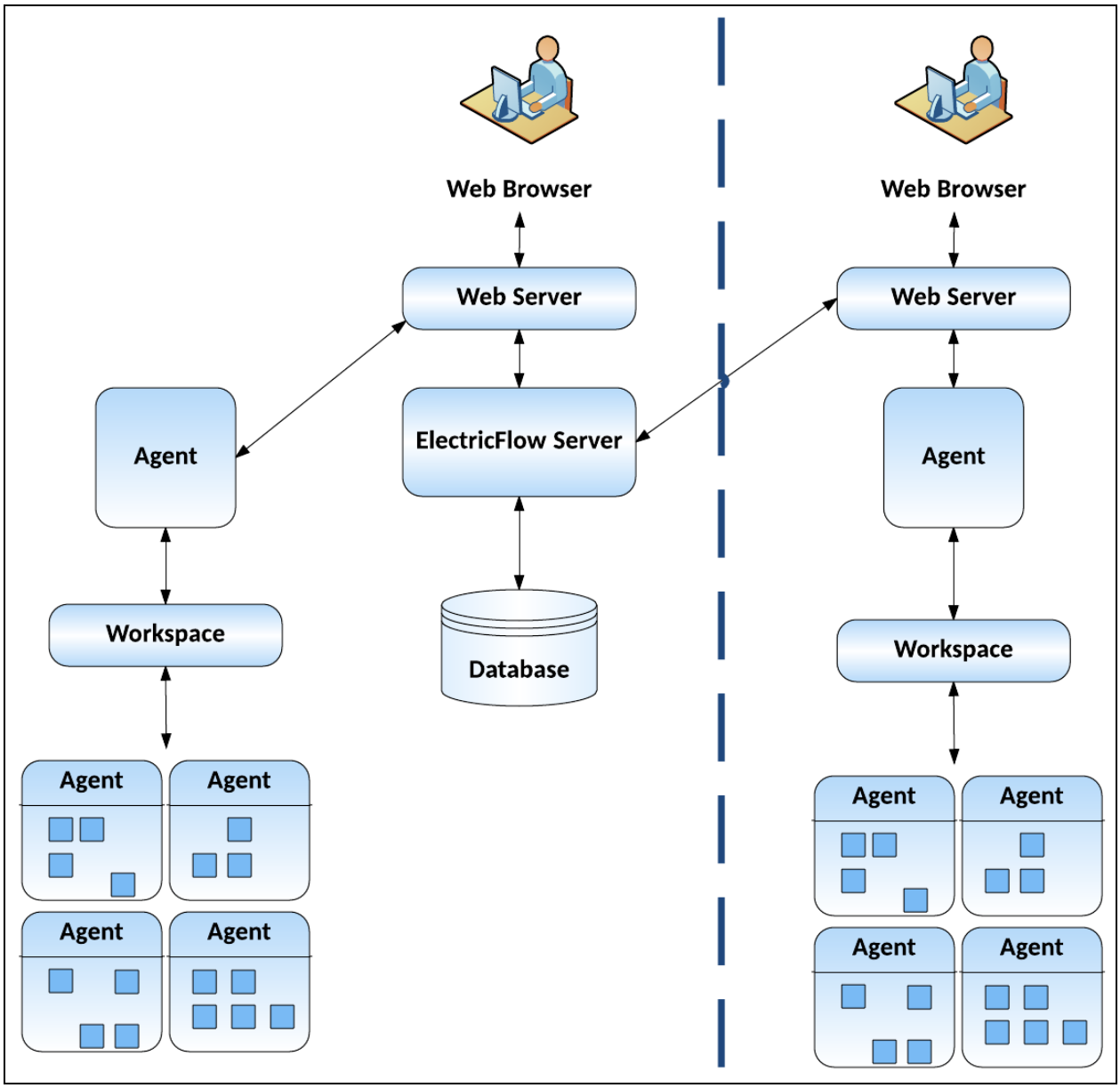
These are sample ElectricFlow configurations:

Single-Site Architecture



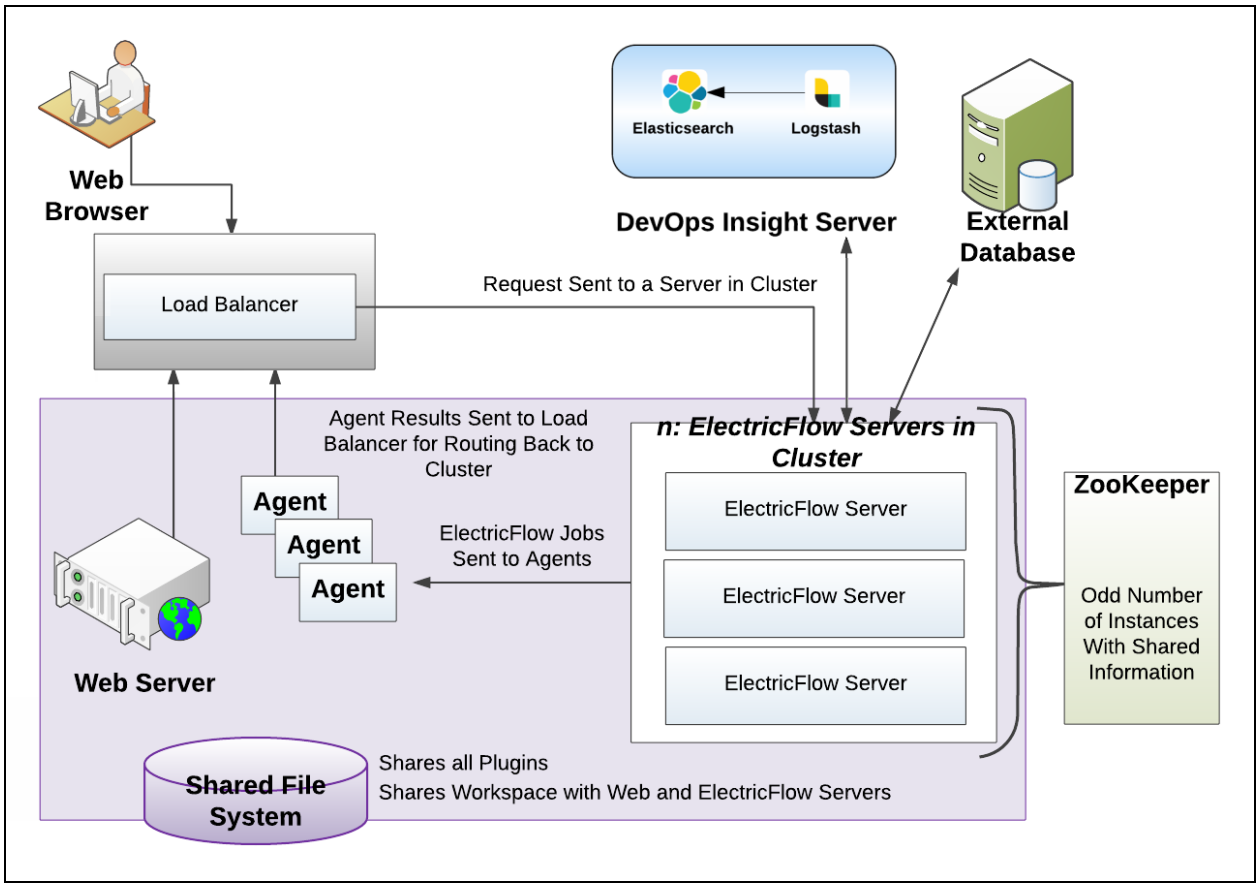
See Architecture for a complete description of this configuration.

Remote Web Server Configuration



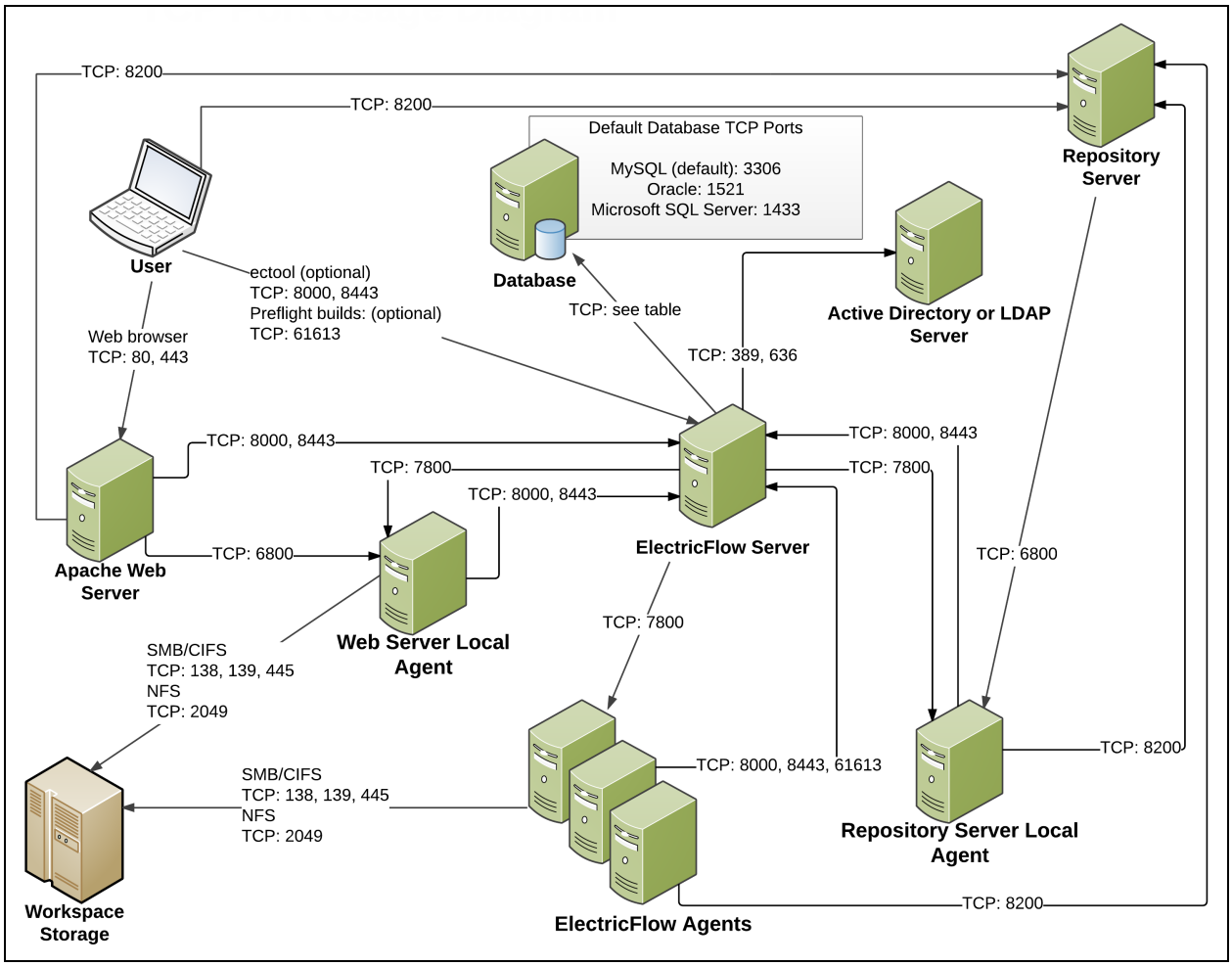
See Architecture for a complete description of this configuration.

Clustered Configuration



See Architecture of an ElectricFlow Cluster for a detailed description of this configuration.

Server Components in ElectricFlow



For a detailed description, see the *KBEC-00041 - ElectricFlow TCP port usage - diagram and descriptions* KB article.

Chapter 7: Upgrading from ElectricFlow 5.x, 6.x, 7.x, or 8.x to 9.x

This section describes how to upgrade the software from ElectricFlow 5.x, 6.x, 7.x, or 8.x to 9.x (a newer version). The procedure is the same as when you upgrade from ElectricFlow 5.x, 6.x, 7.x, or 8.x to 9.x (a newer version) and upgrade cluster configurations at the same time, except that you do not need to perform additional tasks to upgrade the cluster.

To upgrade from ElectricFlow 5.x, 6.x, 7.x, or 8.x to 9.x (a newer version), use the `ElectricFlow-<version>` installer, which collects the ElectricFlow service account credentials, uninstalls the current release, installs the latest ElectricFlow release, configures the system with all property values mined, and restores custom files and data.

After preparing for the upgrade, make sure to shut down the ElectricFlow server service before installing ElectricFlow 9.x.

Configuration Settings Preserved After an Upgrade

The following configuration settings are saved during a software upgrade. These settings are reloaded into ElectricFlow after the upgrade.

Agent Configuration Settings

Properties in <data_dir>/conf/agent.conf

artifactCache	caFile
caPath	certFile
duplicateDetectionListSize	idleOutboundConnectionTimeout
idlePostRunnerTimeout	idleServerRequestWorkerTimeout
idleWorkerTimeout	keyFile
loadProfile	logFile
logLevel	logMaxFiles
logMaxSize	outboundRequestInitialRetryInterval
outboundRequestMaxRetryInterval	outboundRequestTimeout
pluginsPath	port
proto	serverConnectTimeout
serverReadTimeout	unixShellPattern
verifyPeer	

Properties in <data_dir>/conf/agent/wrapper.conf

set.ECWRAPPER_WRITE_MAX_ATTEMPTS	set.ECWRAPPER_WRITE_RETRY_INTERVAL
wrapper.console.format	wrapper.java.additional.<n> where n must be ≥ 10000 (custom parameter)
wrapper.java.additional.701	wrapper.java.additional.702
wrapper.java.additional.703	wrapper.java.classpath.<n> (where n must be ≥ 1)
wrapper.java.initmemory	wrapper.java.initmemory.percent
wrapper.java.library.path.<n> (where n must be ≥ 1)	wrapper.java.maxmemory
wrapper.java.maxmemory.percent	wrapper.logfile
wrapper.logfile.format	wrapper.logfile.loglevel
wrapper.logfile.maxfiles	wrapper.logfile.maxsize
wrapper.ntservice.dependency.<n>	wrapper.ntservice.interactive
wrapper.ntservice.starttype	wrapper.ping.interval
wrapper.ping.timeout	wrapper.request_thread_dump_on_failed_jvm_exit
wrapper.shutdown.timeout	wrapper.startup.timeout
wrapper.successful_invocation_time	wrapper.syslog.loglevel

Properties in <data_dir>/conf/agent/agent.properties

AGENT_ACCEPT_QUEUE_SIZE	AGENT_CRL_FILE
AGENT_DOMAIN_NAME	AGENT_KEYSTORE
AGENT_KEYSTORE_PASSWORD	AGENT_LOCAL_PORT
AGENT_MAX_HTTP_THREADS	AGENT_PORT
AGENT_PROTOCOL	AGENT_SERVER_SESSIONS_FILE
IDLE_CONNECTION_TIMEOUT	MAX_CONNECTIONS
MAX_CONNECTIONS_PER_ROUTE	MAX_LOGGED_prompt_LENGTH
OUTBOUND_CONNECT_TIMEOUT	

ElectricFlow Server Configuration Settings

Properties in <data_dir>/conf/commander.properties

COMMANDER_ACCEPT_QUEUE_SIZE	COMMANDER_BATCH_DB_REQUESTS_OVERRIDE
COMMANDER_CERT	COMMANDER_CRITICAL_SERVICES_MAX_ATTEMPTS_TO_BE_IN_PRIMARY_CLUSTER
COMMANDER_CRITICAL_SERVICES_MONITORING_ENABLED	COMMANDER_CRITICAL_SERVICES_MONITORING_FREQUENCY
COMMANDER_CRL_FILE	COMMANDER_DATA_DIR_MONITORING_ENABLED
COMMANDER_FORCE_ENABLE_ADMIN	COMMANDER_HTTPS_PORT
COMMANDER_KEY	COMMANDER_KEYSTORE
COMMANDER_KEYSTORE_PASSWORD	COMMANDER_LOG_DIR_MONITORING_ENABLED
COMMANDER_MAX_API_THREADS	COMMANDER_MAX_DISPATCH_THREADS
COMMANDER_MAX_HTTP_THREADS	COMMANDER_MAX_QUARTZ_THREADS
COMMANDER_MAX_WORKFLOW_THREADS	COMMANDER_MQ_DATADIR
COMMANDER_MQ_DIR_MONITORING_ENABLED	COMMANDER_MQ_DISK_SPACE_MONITORING_ENABLED
COMMANDER_MQ_DISK_SPACE_MONITORING_IN_CLUSTER_ONLY	COMMANDER_MQ_HARD_DISK_SPACE_LIMIT
COMMANDER_MQ_SOFT_DISK_SPACE_LIMIT	COMMANDER_NESTED_LDAP_GROUPS_MAXIMUM_DEPTH_LIMIT
COMMANDER_PASSWORD_KEYFILE	COMMANDER_PORT
COMMANDER_SERVER_NAME	COMMANDER_STOMP_PORT
org.apache.coyote.USE_CUSTOM_STATUS_MSG_IN_HEADER	

Properties in <data_dir>/conf/wrapper.conf

set.default.COMMANDER_HTTPS_PORT	wrapper.syslog.loglevel
set.default.COMMANDER_XML_READER_STRIP_WHITESPACE_TEXT	set.default.COMMANDER_PORT

set.default.INSTALL_DIRECTORY	set.default.DATA_DIRECTORY
wrapper.java.additional.<n> where n must be ≥ 10000 (custom parameter)	wrapper.console.format
wrapper.java.additional.250	wrapper.java.additional.240
wrapper.java.additional.350	wrapper.java.additional.260
wrapper.java.additional.601	wrapper.java.additional.600
wrapper.java.additional.603	wrapper.java.additional.602
wrapper.java.additional.702	wrapper.java.additional.701
wrapper.java.additional.800	wrapper.java.additional.703
wrapper.java.additional.802	wrapper.java.additional.801
wrapper.java.additional.901	wrapper.java.additional.803
wrapper.java.additional.903	wrapper.java.additional.902
wrapper.java.additional.950	wrapper.java.additional.1600
wrapper.java.additional.1601	wrapper.java.classpath.<n> (where n must be ≥ 1)
wrapper.java.initmemory	wrapper.java.initmemory.percent
wrapper.java.library.path.<n> (where n must be ≥ 1)	wrapper.java.maxmemory
wrapper.java.maxmemory.percent	wrapper.logfile
wrapper.logfile.format	wrapper.logfile.loglevel
wrapper.logfile.maxfiles	wrapper.logfile.maxsize
wrapper.ping.interval	wrapper.ping.timeout
wrapper.request_thread_dump_on_failed_jvm_exit	wrapper.shutdown.timeout
wrapper.startup.timeout	wrapper.successful_invocation_time

Repository Server Configuration Settings

Properties in <data_dir>/conf/repository/server.properties

AGENT_URL	COMMANDER_HOST
IDLE_CONNECTION_TIMEOUT	MAX_CONNECTIONS
MAX_CONNECTIONS_PER_ROUTE	REPOSITORY_ACCEPT_QUEUE_SIZE
REPOSITORY_BACKING_STORE	REPOSITORY_KEYSTORE
REPOSITORY_KEYSTORE_PASSWORD	REPOSITORY_MAX_HTTP_THREADS
REPOSITORY_PORT	REPOSITORY_PROTOCOL
VALIDATE_FROM_DISK	

Properties in <data_dir>/conf/repository/wrapper.conf

set.default.DATA_DIRECTORY	set.default.INSTALL_DIRECTORY
set.default.REPOSITORY_PORT	set.default.REPOSITORY_PROTOCOL
wrapper.console.format	wrapper.java.additional.400
wrapper.java.additional.401	wrapper.java.additional.402
wrapper.java.additional.701	wrapper.java.additional.702
wrapper.java.additional.703	wrapper.java.classpath.<n> (where n must be ≥ 1)
wrapper.java.initmemory	wrapper.java.initmemory.percent
wrapper.java.library.path.<n> (where n must be ≥ 1)	wrapper.java.maxmemory
wrapper.java.maxmemory.percent	wrapper.logfile
wrapper.logfile.format	wrapper.logfile.loglevel
wrapper.logfile.maxfiles	wrapper.logfile.maxsize
wrapper.ping.interval	wrapper.ping.timeout
wrapper.request_thread_dump_on_failed_jvm_exit	wrapper.shutdown.timeout
wrapper.startup.timeout	wrapper.successful_invocation_time
wrapper.syslog.loglevel	

Web Server Configuration Settings

Properties in <data_dir>/apache/conf/httpd.conf

Listen	RewriteCond %{HTTPS}
ServerName	SetEnv CGI_HTTP_PROXY
SetEnv COMMANDER_DLC	SetEnv COMMANDER_HTTPS_PORT
SetEnv COMMANDER_PLUGINS	SetEnv COMMANDER_PORT
SetEnv COMMANDER_SERVER	SetEnv no_proxy

Properties in <data_dir>/apache/conf/extra/auth-kerberos.conf

KrbConstrainedDelegation	webEnableKrb5Trace
KrbServiceName	

Built-In Database Configuration Settings

Properties in <data_dir>conf/mariadb/mariadb.conf

port (under sections [mysqld] and [client])	innodb_buffer_pool_size

Properties in <data_dir>/apache/conf/php.ini

date.timezone

Properties in <data_dir>/apache/conf/ssl.conf

Listen

Properties in <data_dir>/apache/htdocs/commander/config.php

csrfProtection	ssoEnabledKerberos

Use Cases

The actual steps that you perform to upgrade from ElectricFlow 5.x, 6.x, 7.x, or 8.x to the latest ElectricFlow release are based on your ElectricFlow environment.

Review the information in the following table, and select the use case that best matches your ElectricFlow environment to go to the detailed upgrade process steps.

Clustered Environment	Link to the Upgrade Process Steps
Yes	Upgrading from ElectricFlow 5.x, 6.x, 7.x, or 8.x to 9.x in a Clustered Environment on page 8-1
No	Upgrading from ElectricFlow 5.x, 6.x, 7.x, or 8.x to 9.x on page 7-1

Preparing for Your Upgrade

Review the following information before you upgrade ElectricFlow.

Upgrade Testing

In most implementations, ElectricFlow is being used in an environment that affects many users. We recommend that you test your upgrade on a separate test server to understand all aspects of the upgrade process. This minimizes the potential impacts to downstream users.

Backing Up Your Existing ElectricFlow Data

If you are upgrading an ElectricFlow server, it is *extremely* important that you back up your existing ElectricFlow data before upgrading. See ElectricFlow Server Backups on page 12-1for more information about backups.

- Always back up the Plugins Directory. The default location is the `plugins` subdirectory within the data directory.
- Always back up the files that contain your configuration and custom settings. To ensure that all important settings are saved, back up the following subdirectories in *<data_dir>*:
  - The entire `conf` subdirectory (which contains the ElectricFlow server and agent configuration files)
  - Apache web server configuration files in the `apache/conf` subdirectory
- Always back up any other files where you have created custom configurations, specified other custom information, or created any type of modification.

  Note: The ElectricFlow files you might have modified are too numerous to list, so you should back up the entire data directory and other miscellaneous files that might have changed.

- If you use an artifact repository, back up your ElectricFlow repository configuration files in the `conf/repository` subdirectory.
- Determine if any changes were made to the custom editor or preflight driver script properties (installed by default). Back up those files if changes were made.

  These properties are stored in the server-level property sheet, which can be viewed in the web UI by accessing the Administration tab/ Server subtab.

  Custom editors are stored in the nested sheet named `ec_customEditors`. Preflight driver scripts are stored in the nested sheet named `ec_preflight`. The upgrade process overwrites default custom editor and preflight driver scripts with current versions. We recommend backing up any

custom properties you created by renaming the property. For example, change `ec_preflight/clientDrivers/perforce` to `ec_preflight/clientDrivers/perforce_modified`.

Upgrade Installer Preservation

After you back up your ElectricFlow server, create a folder where you can download the `ElectricFlow-<version>` installation file.

MySQL Upgrades

ElectricFlow upgrades involving a MySQL database can take several hours to complete if you have a significant data set. *Do not interrupt the upgrade process.* You can corrupt your database if the upgrade process is interrupted. A restore from a previous database backup will be required.

Use the `ectool` to view the upgrade progress. On a command line, enter:

```
ectool getServerStatus
```

An install/upgrade log file named `installer.log` is created in the `logs` subdirectory in the data directory.

Choosing the Correct Upgrade Method

This section describes the various upgrade methods and options for specific platform configurations. For information about supported server platforms and supported non-server platforms, see Supported Server Platforms on page 2-1 and Supported Agent Platforms on page 2-2.

User Interface Upgrade

This method provides a wizard for upgrading ElectricFlow on a supported server platform. This upgrade method is generally preferred by Windows users, but is supported on Linux platforms with the X Window System installed. See User Interface Upgrade Method on page 8-12 for more details.

Upgrade options:

- **Upgrade Existing Installation**

  This option uninstalls the current release, installs the latest ElectricFlow release, collects the ElectricFlow service account credentials, configures the system with all property values, and restores custom files and data.

- **Clean Install**

  This option allows you to specify a different installation directory for the new version. The files from your previous ElectricFlow version will not be removed or modified and will remain in their original directories.

  > **Note:** On Linux, when ElectricFlow is already installed and you want to use the clean install upgrade method, you must do an advanced installation.

  > **Note:** On Windows, a clean installation replaces the registry entries of the current installation. On Linux, a clean installation replaces the files in the `/etc/init.d` directory. The result is that only one instance of ElectricFlow (the new version) is running.

Interactive Command-Line Upgrade

This method provides an interactive command-line for upgrading ElectricFlow on a supported server platform. This upgrade method is only available for Linux platforms. See Interactive Command-Line Upgrade Method on page 8-16 for more details.

Upgrade options:

- **Upgrade Existing Installation**

  This option uninstalls the current release, installs the latest ElectricFlow release, collects the ElectricFlow service account credentials, configures the system with all property values, and restores custom files and data.

- **Clean Install**

  This option allows you to specify a different installation directory for the new version. The files from your previous ElectricFlow version will not be removed or modified and will remain in their original directories.

  Note: On Linux, when ElectricFlow is already installed and you want to use the clean install upgrade method, you must do an advanced installation.

  Note: On Windows, a clean installation replaces the registry entries of the current installation. On Linux, a clean installation replaces the files in the `/etc/init.d` directory. The result is that only one instance of ElectricFlow (the new version) is running.

Silent Unattended Upgrade

This method provides a non-interactive command-line upgrade for supported server platforms. You may find this installation method preferable for upgrading multiple remote agents and servers. See Silent (Unattended) Upgrade Method on page 8-17 for more details.

Upgrade options:

- **Upgrade Existing Installation**

  This option uninstalls the current release, installs the latest ElectricFlow release, collects the ElectricFlow service account credentials, configures the system with all property values, and restores custom files and data.

Important: You cannot add a new repository server with this upgrade method.

Repository Server With an ElectricFlow Upgrade

The only way to install a repository server on the same machine as other services is to uninstall and reinstall ElectricFlow. You can install the repository server on a different machine to avoid uninstalling and reinstalling ElectricFlow.

Non-Server Platform Agent Upgrade

You cannot directly upgrade a non-server platform agent (that is, an agent on a machine that is not a supported ElectricFlow server platform). You must uninstall and then reinstall these machines using the ElectricFlow installer. For more information, see Uninstalling ElectricFlow on page 10-1 and Non-Server Platform Installation Method for UNIX Agents on page 14-30.

Stand-Alone Repository Server or Web Server Upgrade

You cannot directly upgrade a standalone repository server or standalone web server. You must uninstall and then reinstall these servers using the ElectricFlow installer. The uninstall and reinstall process is required to install an agent on the server machine. An agent is required on the machine with the standalone repository server or web server. For more information, see Uninstalling ElectricFlow on page 10-1, Installing ElectricFlow on page 3-1, and Copying Repository Contents on page 8-18.

User Interface Upgrade Method

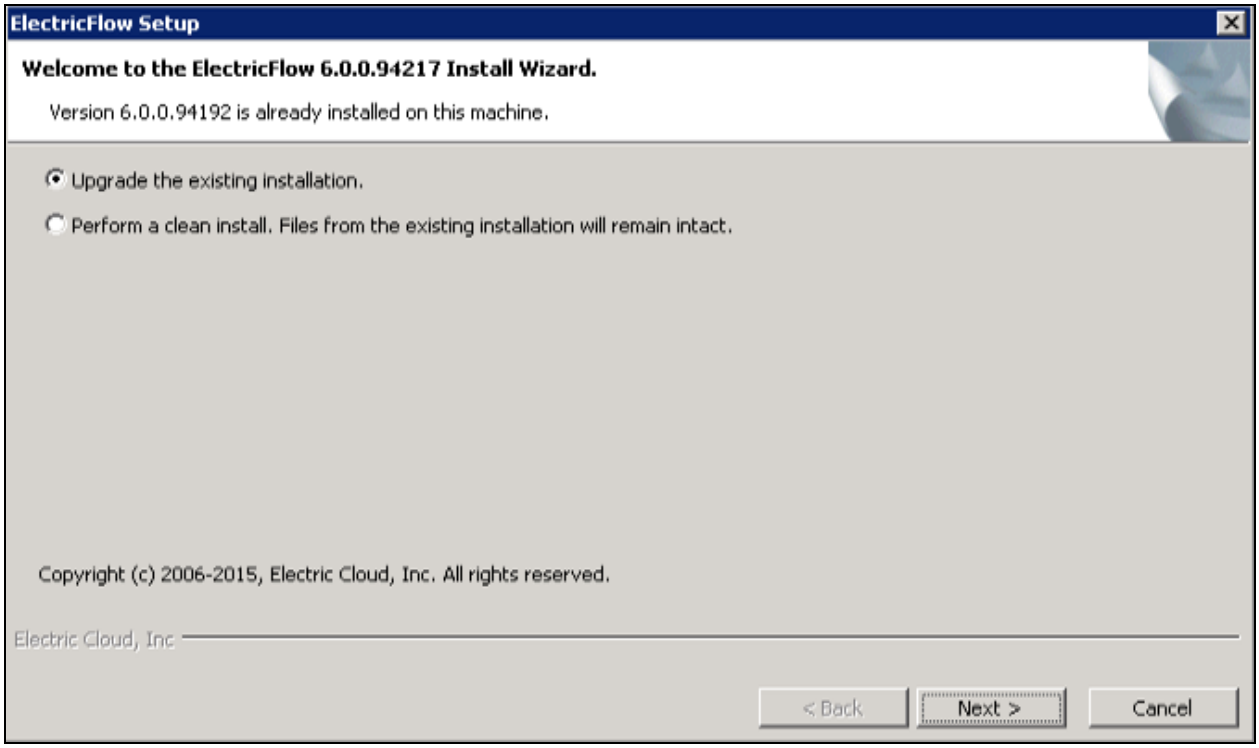
Use this procedure to upgrade ElectricFlow. Review Preparing for Your Upgrade on page 7-7 before performing this procedure.

> **Important:** When upgrading the nodes in an ElectricFlow cluster, you must keep the other nodes stopped until the primary node upgrade is complete.

1. Enter the following command to make the installer file executable:

   ```
   chmod +x ./ElectricFlow-<version>
   ```

2. Double-click the `ElectricFlow-<version>` file to begin installation. The **Welcome to the ElectricFlow Install Wizard** screen appears.



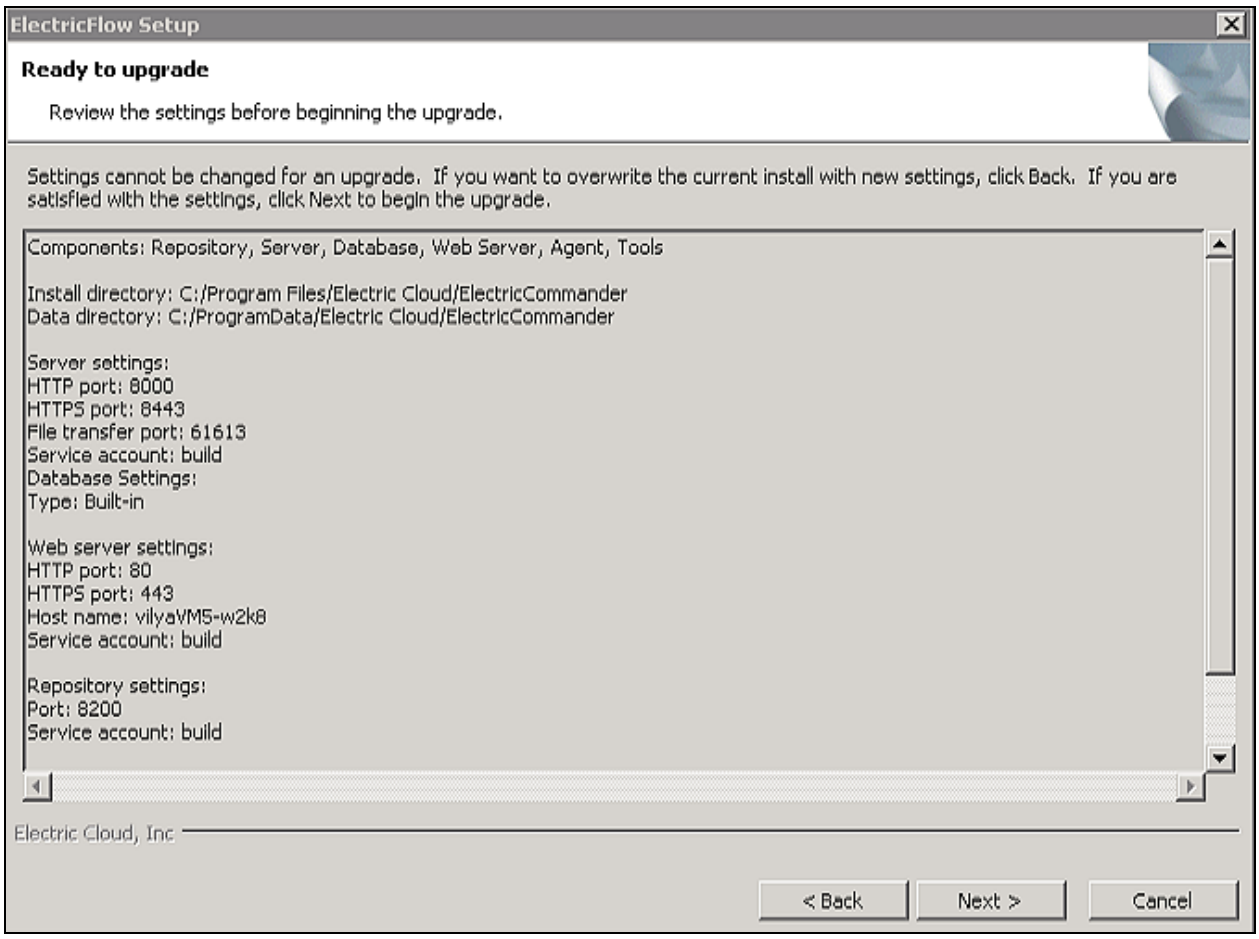
3. Choose one of the following options:

   - Select **Upgrade the existing installation** if you want to upgrade your current ElectricFlow installation directory.
   - Select **Perform a clean install** if you want to specify a different installation directory for the new version.

> **Note:** During a clean installation, current services remain running until you click **Next** on the Ready to Install screen. This means the new installation cannot use the same ports and directories as the existing installation. To use the same ports and directories, you must manually stop the existing services. This will free the existing ports and directories.

4. Click **Next** to upgrade the existing installation.

   The **Ready to upgrade** screen appears.

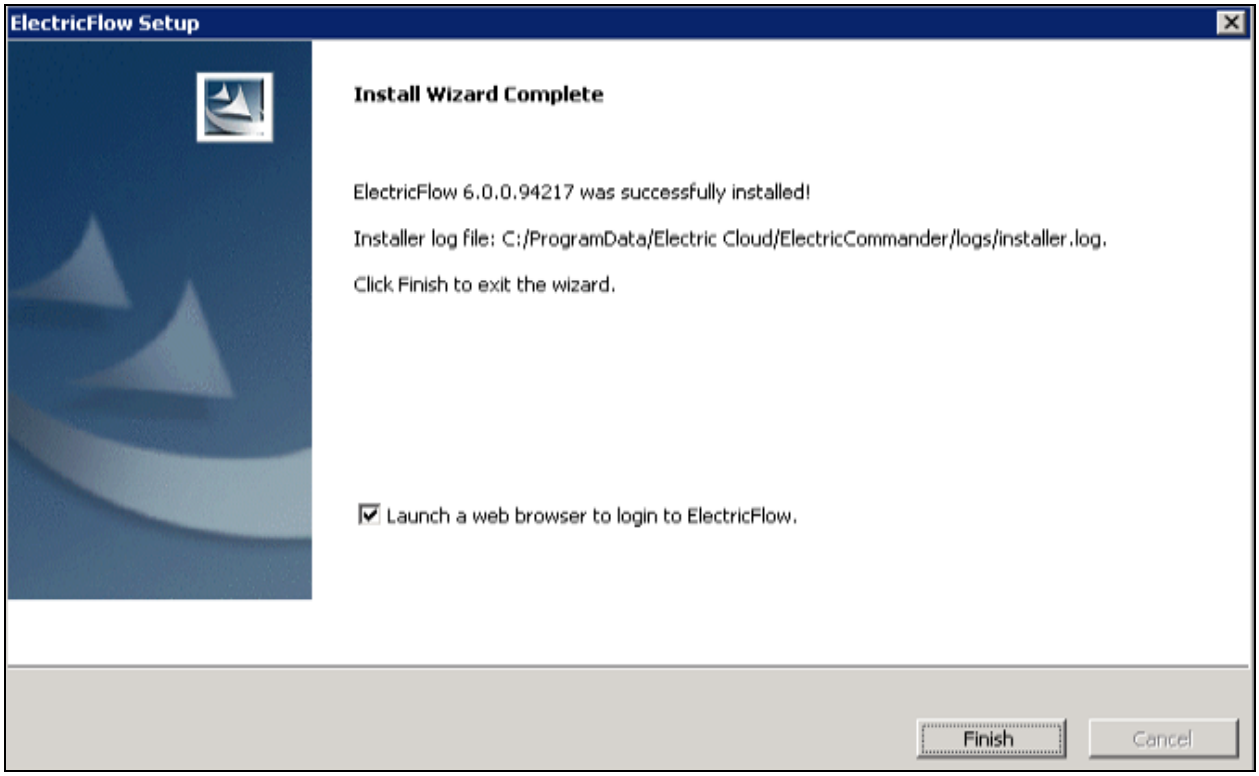


5. Review the upgrade settings.

   Use the **Back** button to change your selections if necessary.

6. Click **Next** to continue.

   The installer displays a status bar to show the progress of the upgrade process. You can also view the `installer.log` file to see progress. The time that it takes to complete this process depends on the size of the database. It may take fifteen minutes or longer to complete. Once this process is complete, the new ElectricFlow version is installed.



7. Select the **Launch a web browser to login to ElectricFlow** check box if you want ElectricFlow to open the login screen now.

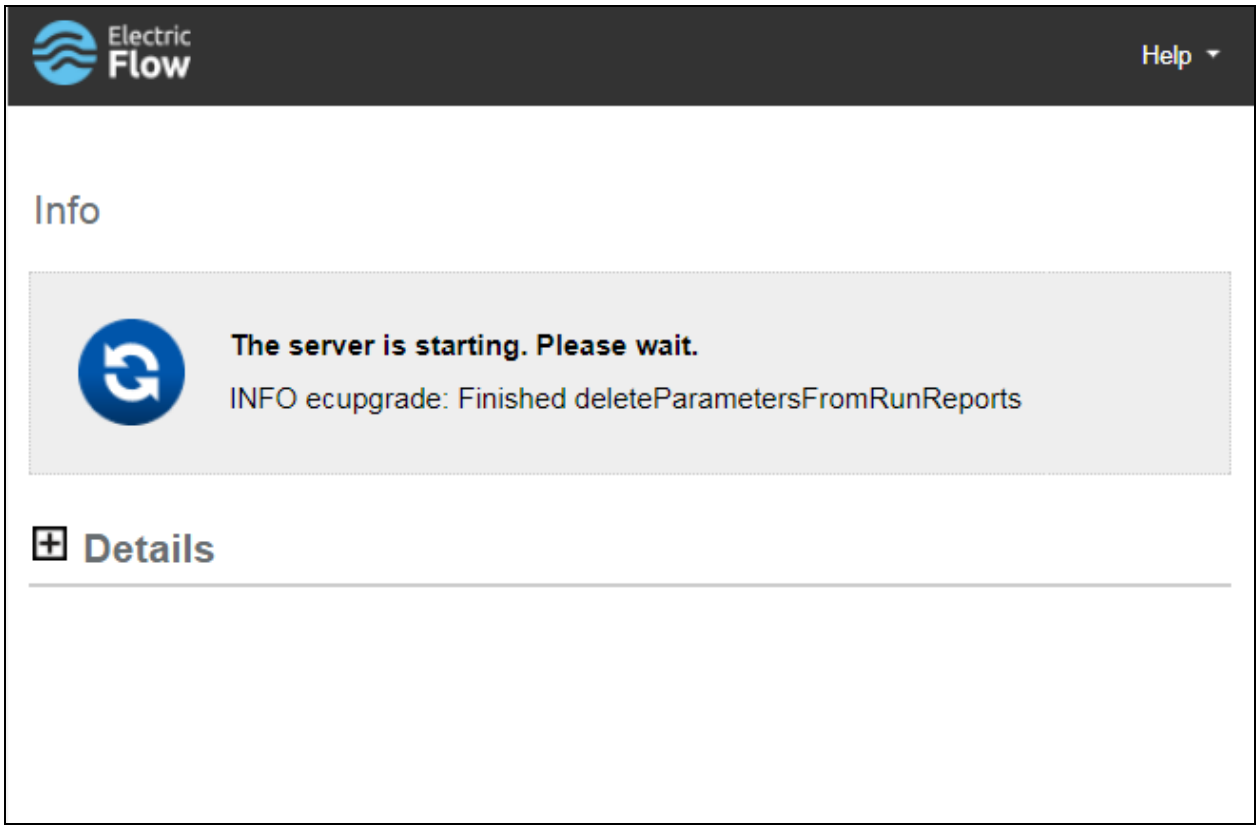
8. Click **Finish** to complete the upgrade.

   If an ElectricFlow server is being upgraded, when the installation is complete, the server continue to upgrade the database (if applicable). You cannot log in to the ElectricFlow server until the database upgrade finishes. You can view the upgrade status by using `ectool` from a command line:

   ```
   ectool getServerStatus
   ```

   However, for upgrades involving large databases, the output from `ectool getServerStatus` might remain unchanged for long periods. In this case, you can see more granular database update activity by following the procedure in the *KBEC-00086 - Enabling and collecting voluminous JDBC logging* KB article to add SQL logging to the `<data_dir>/logs/commander.log` file. You can view recent SQL logging updates to the file by using the Linux `tail <data_dir>/logs/commander.log` command.

   After clicking **Finish**, you might see a web page similar to the following screen if the upgrade is still in progress:



Interactive Command-Line Upgrade Method

Use the following procedure to complete a command-line upgrade of a Linux platform. Review Preparing for Your Upgrade on page 7-7 before performing this procedure.

Important: When upgrading the nodes in an ElectricFlow cluster, you must keep the other nodes stopped until the primary node upgrade is complete.

1. Enter the following command to make the installer file executable:

   ```
   chmod +x ./ElectricFlow-<version>
   ```

2. Choose one of the following commands to begin the upgrade:

   - If you have a Linux platform, enter `./ElectricFlow-<version>`.
   - If you have a Linux platform with the X Window System, the installer will automatically bring up the graphical user interface.

     To override this behavior, enter `./ElectricFlow-<version> --mode console`.

   The following prompt appears:

   ```
   Copyright (c) 2006-2018, Electric Cloud, Inc. All rights reserved.

   This will install ElectricFlow on your computer. Continue? [n/Y]
   ```

3. Enter: `y`

   The following prompt appears:

   ```
   Upgrade the existing <version> installation to version <version>? [n/Y
   ```

4. Choose one of the following options:

   - If you want to upgrade your current ElectricFlow installation directory, enter `y`.
   - Enter `n` to exit the installer.

   The following prompt appears:

   ```
   Installing agent...

   Installing server...

   Copied log file to "/opt/electriccloud/electriccommander/logs"

   ElectricFlow <version> was successfully installed!

   Installer log file: /opt/electriccloud/electriccommander/logs/installer.log.
   ```

Silent (Unattended) Upgrade Method

You can run the ElectricFlow upgrade in unattended (silent) mode with no user interface for either Windows or Linux.

Important: When upgrading the nodes in an ElectricFlow cluster, you must keep the other nodes stopped until the primary node upgrade is complete.

1. (Linux only) Enter the following command to make the installer file executable:

   ```
   chmod +x ./ElectricFlow-<version>
   ```

2. Enter the following command from a command line to begin a silent upgrade:

   ```
   ./ElectricFlow-<version> --mode silent
   ```

Copying Repository Contents

Perform the following steps to copy the contents of an existing repository server into a newly installed repository server:

1. Install the new repository server software.
2. Stop the existing and new repository servers.
3. Copy the entire contents of the repository backingstore directory from the existing repository server to the corresponding location on the newly installed repository server.

   The default location for the backingstore directory (`<datadir>/repository-data`) is:

   - **UNIX**—`/opt/electriccloud/electriccommander/repository-data`
   - **Windows**—`C:\ProgramData\Electric Cloud\ElectricCommander\repository-data`

Chapter 8: Upgrading from ElectricFlow 5.x, 6.x, 7.x, or 8.x to 9.x in a Clustered Environment

This section describes how to upgrade the software from ElectricFlow 5.x, 6.x, 7.x, or 8.x to 9.x (a newer version) and upgrade cluster configurations at the same time. The procedure is the same as when you upgrade from ElectricFlow 5.x, 6.x, 7.x, or 8.x to 9.x (a newer version) except that you need to perform additional tasks to upgrade the cluster.

To upgrade from ElectricFlow 5.x, 6.x, 7.x, or 8.x to 9.x (a newer version), use the `ElectricFlow-<version>` installer, which collects the ElectricFlow service account credentials, uninstalls the current release, installs the latest ElectricFlow release, configures the system with all property values mined, and restores custom files and data.

After preparing for the upgrade, make sure to shut down the ElectricFlow server service before installing ElectricFlow 9.x.

Configuration Settings Preserved After an Upgrade

The following configuration settings are saved during a software upgrade. These settings are reloaded into ElectricFlow after the upgrade.

Agent Configuration Settings

Properties in <data_dir>/conf/agent.conf

artifactCache	caFile
caPath	certFile
duplicateDetectionListSize	idleOutboundConnectionTimeout
idlePostRunnerTimeout	idleServerRequestWorkerTimeout
idleWorkerTimeout	keyFile
loadProfile	logFile
logLevel	logMaxFiles
logMaxSize	outboundRequestInitialRetryInterval
outboundRequestMaxRetryInterval	outboundRequestTimeout
pluginsPath	port

proto	serverConnectTimeout
serverReadTimeout	unixShellPattern
verifyPeer	

Properties in <data_dir>/conf/agent/wrapper.conf

set.ECWRAPPER_WRITE_MAX_ATTEMPTS	set.ECWRAPPER_WRITE_RETRY_INTERVAL
wrapper.console.format	wrapper.java.additional.<n> where n must be ≥ 10000 (custom parameter)
wrapper.java.additional.701	wrapper.java.additional.702
wrapper.java.additional.703	wrapper.java.classpath.<n> (where n must be ≥ 1)
wrapper.java.initmemory	wrapper.java.initmemory.percent
wrapper.java.library.path.<n> (where n must be ≥ 1)	wrapper.java.maxmemory
wrapper.java.maxmemory.percent	wrapper.logfile
wrapper.logfile.format	wrapper.logfile.loglevel
wrapper.logfile.maxfiles	wrapper.logfile.maxsize
wrapper.ntservice.dependency.<n>	wrapper.ntservice.interactive
wrapper.ntservice.starttype	wrapper.ping.interval
wrapper.ping.timeout	wrapper.request_thread_dump_on_failed_jvm_exit
wrapper.shutdown.timeout	wrapper.startup.timeout
wrapper.successful_invocation_time	wrapper.syslog.loglevel

Properties in <data_dir>/conf/agent/agent.properties

AGENT_ACCEPT_QUEUE_SIZE	AGENT_CRL_FILE
AGENT_DOMAIN_NAME	AGENT_KEYSTORE
AGENT_KEYSTORE_PASSWORD	AGENT_LOCAL_PORT
AGENT_MAX_HTTP_THREADS	AGENT_PORT
AGENT_PROTOCOL	AGENT_SERVER_SESSIONS_FILE

IDLE_CONNECTION_TIMEOUT	MAX_CONNECTIONS
MAX_CONNECTIONS_PER_ROUTE	MAX_LOGGED_prompt_LENGTH
OUTBOUND_CONNECT_TIMEOUT	

ElectricFlow Server Configuration Settings

Properties in <data_dir>/conf/commander.properties

COMMANDER_ACCEPT_QUEUE_SIZE	COMMANDER_BATCH_DB_REQUESTS_ OVERRIDE
COMMANDER_CERT	COMMANDER_CRITICAL_SERVICES_MAX_ ATTEMPTS_TO_BE_IN_PRIMARY_CLUSTER
COMMANDER_CRITICAL_SERVICES_ MONITORING_ENABLED	COMMANDER_CRITICAL_SERVICES_ MONITORING_FREQUENCY
COMMANDER_CRL_FILE	COMMANDER_DATA_DIR_MONITORING_ ENABLED
COMMANDER_FORCE_ENABLE_ADMIN	COMMANDER_HTTPS_PORT
COMMANDER_KEY	COMMANDER_KEYSTORE
COMMANDER_KEYSTORE_PASSWORD	COMMANDER_LOG_DIR_MONITORING_ENABLED
COMMANDER_MAX_API_THREADS	COMMANDER_MAX_DISPATCH_THREADS
COMMANDER_MAX_HTTP_THREADS	COMMANDER_MAX_QUARTZ_THREADS
COMMANDER_MAX_WORKFLOW_THREADS	COMMANDER_MQ_DATADIR
COMMANDER_MQ_DIR_MONITORING_ENABLED	COMMANDER_MQ_DISK_SPACE_MONITORING_ ENABLED
COMMANDER_MQ_DISK_SPACE_MONITORING_ IN_CLUSTER_ONLY	COMMANDER_MQ_HARD_DISK_SPACE_LIMIT
COMMANDER_MQ_SOFT_DISK_SPACE_LIMIT	COMMANDER_NESTED_LDAP_GROUPS_ MAXIMUM_DEPTH_LIMIT
COMMANDER_PASSWORD_KEYFILE	COMMANDER_PORT
COMMANDER_SERVER_NAME	COMMANDER_STOMP_PORT
org.apache.coyote.USE_CUSTOM_STATUS_MSG_ IN_HEADER	

Properties in <data_dir>/conf/wrapper.conf

set.default.COMMANDER_HTTPS_PORT	wrapper.syslog.loglevel
set.default.COMMANDER_XML_READER_STRIP_WHITESPACE_TEXT	set.default.COMMANDER_PORT
set.default.INSTALL_DIRECTORY	set.default.DATA_DIRECTORY
wrapper.java.additional.<n> where n must be ≥ 10000 (custom parameter)	wrapper.console.format
wrapper.java.additional.250	wrapper.java.additional.240
wrapper.java.additional.350	wrapper.java.additional.260
wrapper.java.additional.601	wrapper.java.additional.600
wrapper.java.additional.603	wrapper.java.additional.602
wrapper.java.additional.702	wrapper.java.additional.701
wrapper.java.additional.800	wrapper.java.additional.703
wrapper.java.additional.802	wrapper.java.additional.801
wrapper.java.additional.901	wrapper.java.additional.803
wrapper.java.additional.903	wrapper.java.additional.902
wrapper.java.additional.950	wrapper.java.additional.1600
wrapper.java.additional.1601	wrapper.java.classpath.<n> (where n must be ≥ 1)
wrapper.java.initmemory	wrapper.java.initmemory.percent
wrapper.java.library.path.<n> (where n must be ≥ 1)	wrapper.java.maxmemory
wrapper.java.maxmemory.percent	wrapper.logfile
wrapper.logfile.format	wrapper.logfile.loglevel
wrapper.logfile.maxfiles	wrapper.logfile.maxsize
wrapper.ping.interval	wrapper.ping.timeout
wrapper.request_thread_dump_on_failed_jvm_exit	wrapper.shutdown.timeout
wrapper.startup.timeout	wrapper.successful_invocation_time

Repository Server Configuration Settings

Properties in <data_dir>/conf/repository/server.properties

AGENT_URL	COMMANDER_HOST
IDLE_CONNECTION_TIMEOUT	MAX_CONNECTIONS
MAX_CONNECTIONS_PER_ROUTE	REPOSITORY_ACCEPT_QUEUE_SIZE
REPOSITORY_BACKING_STORE	REPOSITORY_KEYSTORE
REPOSITORY_KEYSTORE_PASSWORD	REPOSITORY_MAX_HTTP_THREADS
REPOSITORY_PORT	REPOSITORY_PROTOCOL
VALIDATE_FROM_DISK	

Properties in <data_dir>/conf/repository/wrapper.conf

set.default.DATA_DIRECTORY	set.default.INSTALL_DIRECTORY
set.default.REPOSITORY_PORT	set.default.REPOSITORY_PROTOCOL
wrapper.console.format	wrapper.java.additional.400
wrapper.java.additional.401	wrapper.java.additional.402
wrapper.java.additional.701	wrapper.java.additional.702
wrapper.java.additional.703	wrapper.java.classpath.<n> (where n must be ≥ 1)
wrapper.java.initmemory	wrapper.java.initmemory.percent
wrapper.java.library.path.<n> (where n must be ≥ 1)	wrapper.java.maxmemory
wrapper.java.maxmemory.percent	wrapper.logfile
wrapper.logfile.format	wrapper.logfile.loglevel
wrapper.logfile.maxfiles	wrapper.logfile.maxsize
wrapper.ping.interval	wrapper.ping.timeout
wrapper.request_thread_dump_on_failed_jvm_exit	wrapper.shutdown.timeout
wrapper.startup.timeout	wrapper.successful_invocation_time
wrapper.syslog.loglevel	

Web Server Configuration Settings

Properties in <data_dir>/apache/conf/httpd.conf

Listen	RewriteCond %{HTTPS}
ServerName	SetEnv CGI_HTTP_PROXY
SetEnv COMMANDER_DLC	SetEnv COMMANDER_HTTPS_PORT
SetEnv COMMANDER_PLUGINS	SetEnv COMMANDER_PORT
SetEnv COMMANDER_SERVER	SetEnv no_proxy

Properties in <data_dir>/apache/conf/extra/auth-kerberos.conf

KrbConstrainedDelegation	webEnableKrb5Trace
KrbServiceName	

Built-In Database Configuration Settings

Properties in <data_dir>conf/mariadb/mariadb.conf

port (under sections [mysqld] and [client])	innodb_buffer_pool_size

Properties in <data_dir>/apache/conf/php.ini

date.timezone

Properties in <data_dir>/apache/conf/ssl.conf

Listen

Properties in <data_dir>/apache/htdocs/commander/config.php

csrfProtection	ssoEnabledKerberos

Use Cases

The actual steps that you perform to upgrade from ElectricFlow 5.x, 6.x, 7.x, or 8.x to the latest ElectricFlow release are based on your ElectricFlow environment.

Review the information in the following table, and select the use case that best matches your ElectricFlow environment to go to the detailed upgrade process steps.

Clustered Environment	Link to the Upgrade Process Steps
Yes	Upgrading from ElectricFlow 5.x, 6.x, 7.x, or 8.x to 9.x in a Clustered Environment on page 8-1
No	Upgrading from ElectricFlow 5.x, 6.x, 7.x, or 8.x to 9.x on page 7-1

Upgrading Clusters

In a cluster configuration, make sure to review the tasks in Preparing for Your Upgrade on page 8-8. To ensure that all important settings are saved, back up the following subdirectories in *DATA_DIR*:

- The entire `conf` subdirectory, which contains the ElectricFlow server and agent configuration files
- Apache web server configuration files in the `apache/conf` subdirectory

Before the upgrade, you must shut down the ElectricFlow server service before installing ElectricFlow 9.x.

During the upgrade, ElectricFlow is not available after the web server is stopped or after the server service on the last ElectricFlow server node is stopped.

- Perform either of these actions:
  - Stop the ElectricFlow server service on all nodes.
  - Stop the ElectricFlow server service on all nodes *except* on the primary ElectricFlow server node.

    You can keep the server service on your primary ElectricFlow server node up, because the installer stops the server service automatically on that node when it upgrades the node.

    > **Important:** When upgrading the nodes in an ElectricFlow cluster, you must keep the other nodes stopped until the primary node upgrade is complete.

- On the primary ElectricFlow server, do the following:
  1. Upgrade the ElectricFlow server. This will also do the following:
     - Connects the server to the database
     - Upgrades the plugins
     - Starts the server
  2. Check and restore the `wrapper.conf` settings. For example, the settings for the line "`wrapper.java.additional.600=`".
  3. Restart the ElectricFlow servers service.
- On the ElectricFlow web server, do the following:
  1. Upgrade the node on the web server.
  2. Check and restore the `httpd.conf` settings.

- Upgrade any nodes on the repository servers.

  Your ElectricFlow system is now available.

- Upgrade the remaining ElectricFlow server nodes.

During the upgrade, some settings may be lost. Verify the following settings before connecting to the ElectricFlow system:

- `httpd.conf` settings for redirecting—These lines should be commented out:

  ```
  # Redirect http to https

  # RewriteCond %{HTTPS} !=on

  # RewriteRule ^/commander/(.*) https://%{SERVER_NAME}:443%{REQUEST_URI} [NC,R,L]
  ```

- `httpd.conf` setting for `COMMANDER_SERVER`–This should point to the load balancer:

  ```
  SetEnv COMMANDER_SERVER "<FQDN of your load balancer>"
  ```

- `wrapper.conf` contains the line that points to your Zookeeper instances.

  For example:

  ```
  wrapper.java.additional.600=-DCOMMANDER_ZK_CONNECTION=192.168.7.20:2181
  ```

Preparing for Your Upgrade

Review the following information before you upgrade ElectricFlow.

Upgrade Testing

In most implementations, ElectricFlow is used in an environment that affects many users. You should test your upgrade on a separate test server to understand all aspects of the upgrade process. This minimizes the potential impact on downstream users.

Backing Up Your Existing ElectricFlow Data

> **Important:** Before upgrading an ElectricFlow server, you must back up your existing ElectricFlow data. See ElectricFlow Server Backups on page 12-1 for more information about backups.

Backing Up Commander Cluster Configuration Files

The configuration files for the Commander cluster are in `<data_dir>\conf`. The default location is:

- Linux: `/opt/electriccloud/electriccommander/conf/`
- Windows: `C:\ProgramData\Electric Cloud\ElectricCommander\conf`

> **Note:** Although the Commander cluster configuration files such as `commander.properties`, `database.properties`, `keystore`, and `passkey` are present in one of the directories above, they are not actually used by the cluster during runtime.

These files were uploaded to Apache ZooKeeper from the first node that was clustered as described in Uploading Configuration Files to ZooKeeper. You can download these files from ZooKeeper to a temporary folder and then compare them with those in the `\conf` folder. You can do so by using the

ElectricFlow `ZKConfigTool`, which is discussed in Uploading Configuration Files to ZooKeeper on page 4-25.

For example, complete the following steps to download these files to `C:\temp` on Windows, where `<install_dir>` is `C:\Program Files\Electric Cloud\ElectricCommander`.

1. Download the files from ZooKeeper by entering the following commands:

   ```
   cd C:\temp

   "C:\Program Files\Electric Cloud\ElectricCommander\jre\bin\java" -DCOMMANDER_ZK_
   CONNECTION=<ZooKeeper_Server_IP>:2181 -jar "C:\Program Files\Electric
   Cloud\ElectricCommander\server\bin\zk-config-tool-jar-with-dependencies.jar"
   com.electriccloud.commander.cluster.ZKConfigTool --readFile
   /commander/conf/database.properties database.properties

   "C:\Program Files\Electric Cloud\ElectricCommander\jre\bin\java" -DCOMMANDER_ZK_
   CONNECTION=<ZooKeeper_Server_IP>:2181 -jar "C:\Program Files\Electric
   Cloud\ElectricCommander\server\bin\zk-config-tool-jar-with-dependencies.jar"
   com.electriccloud.commander.cluster.ZKConfigTool --readFile
   /commander/conf/keystore keystore

   "C:\Program Files\Electric Cloud\ElectricCommander\jre\bin\java" -DCOMMANDER_ZK_
   CONNECTION=<ZooKeeper_Server_IP>:2181 -jar "C:\Program Files\Electric
   Cloud\ElectricCommander\server\bin\zk-config-tool-jar-with-dependencies.jar"
   com.electriccloud.commander.cluster.ZKConfigTool --readFile
   /commander/conf/passkey passkey

   "C:\Program Files\Electric Cloud\ElectricCommander\jre\bin\java" -DCOMMANDER_ZK_
   CONNECTION=<ZooKeeper_Server_IP>:2181 -jar "C:\Program Files\Electric
   Cloud\ElectricCommander\server\bin\zk-config-tool-jar-with-dependencies.jar"
   com.electriccloud.commander.cluster.ZKConfigTool --readFile
   /commander/conf/commander.properties
   ```

2. Make sure that the four files in `C:\temp` are the same as the ones in `<data_dir>\conf`. (You can use a file diff tool to make this easier.) If any file in `<data_dir>\conf` is different, then back up that file and replace it with the one that you downloaded from ZooKeeper.

Backing Up Other Files

The ElectricFlow files that might have been modified are too numerous to list, so you should back up the entire ElectricFlow data directory and other miscellaneous files that might have changed. But at a minimum, you must back up the following files:

- The plugins directory. The default location is the `plugins` subdirectory within `<data_dir>`.
- Files that contain your configuration and custom settings. To ensure that all important settings are saved, back up the following subdirectories in `<data_dir>`:
  - The entire `conf` subdirectory, which contains the ElectricFlow server and agent configuration files.
  - Apache web server configuration files in the `apache/conf` subdirectory.
  - (If applicable) The local MySQL database configuration file, `my.ini`, in the `mysql` subdirectory.
- Any other files where you created custom configurations, specified other custom information, or created any type of modification.

- (If you use an artifact repository) The ElectricFlow repository configuration files in the `conf/repository` subdirectory.
- (If modified) The custom editor or preflight driver script properties (installed by default).

  These properties are stored in the server-level property sheet, which you can view in the web UI in the **Administration** > **Server** subtab.

  Custom editors are stored in the nested sheet named `ec_customEditors`. Preflight driver scripts are stored in the nested sheet named `ec_preflight`. The upgrade process overwrites default custom editor and preflight driver scripts with current versions. You should back up any custom properties that you created by renaming those properties. For example, change `ec_preflight/clientDrivers/perforce` to `ec_preflight/clientDrivers/perforce_modified`.

Upgrade Installer Preservation

After you back up your ElectricFlow server, create a folder where you can download the `ElectricFlow-<version>` installation file.

MySQL Upgrades

ElectricFlow upgrades involving a MySQL database can take several hours to complete if you have a significant data set.

> **Important:** To avoid corrupting your database, do not interrupt the upgrade process. A restore from a previous database backup would be required if this occurs.

You can use ectool to view the upgrade progress. On a command line, enter

```
ectool getServerStatus
```

An install or upgrade log file named `installer.log` is created in the `logs` subdirectory in `<data_dir>`.

Choosing the Correct Upgrade Method

This section describes the various upgrade methods and options for specific platform configurations. For information about supported server platforms and supported non-server platforms, see Supported Server Platforms on page 2-1 and Supported Agent Platforms on page 2-2.

User Interface Upgrade

This method provides a wizard for upgrading ElectricFlow on a supported server platform. This upgrade method is generally preferred by Windows users, but is supported on Linux platforms with the X Window System installed. See User Interface Upgrade Method on page 8-12 for more details.

Upgrade options:

- **Upgrade Existing Installation**

  This option uninstalls the current release, installs the latest ElectricFlow release, collects the ElectricFlow service account credentials, configures the system with all property values, and restores custom files and data.

- **Clean Install**

  This option allows you to specify a different installation directory for the new version. The files from your previous ElectricFlow version will not be removed or modified and will remain in their original directories.

  > **Note:** On Linux, when ElectricFlow is already installed and you want to use the clean install upgrade method, you must do an advanced installation.

  > **Note:** On Windows, a clean installation replaces the registry entries of the current installation. On Linux, a clean installation replaces the files in the `/etc/init.d` directory. The result is that only one instance of ElectricFlow (the new version) is running.

Interactive Command-Line Upgrade

This method provides an interactive command-line for upgrading ElectricFlow on a supported server platform. This upgrade method is only available for Linux platforms. See Interactive Command-Line Upgrade Method on page 8-16 for more details.

Upgrade options:

- **Upgrade Existing Installation**

  This option uninstalls the current release, installs the latest ElectricFlow release, collects the ElectricFlow service account credentials, configures the system with all property values, and restores custom files and data.

- **Clean Install**

  This option allows you to specify a different installation directory for the new version. The files from your previous ElectricFlow version will not be removed or modified and will remain in their original directories.

  > **Note:** On Linux, when ElectricFlow is already installed and you want to use the clean install upgrade method, you must do an advanced installation.

  > **Note:** On Windows, a clean installation replaces the registry entries of the current installation. On Linux, a clean installation replaces the files in the `/etc/init.d` directory. The result is that only one instance of ElectricFlow (the new version) is running.

Silent Unattended Upgrade

This method provides a non-interactive command-line upgrade for supported server platforms. You may find this installation method preferable for upgrading multiple remote agents and servers. See Silent (Unattended) Upgrade Method on page 8-17 for more details.

Upgrade options:

- **Upgrade Existing Installation**

  This option uninstalls the current release, installs the latest ElectricFlow release, collects the ElectricFlow service account credentials, configures the system with all property values, and restores custom files and data.

Important: You cannot add a new repository server with this upgrade method.

Repository Server With an ElectricFlow Upgrade

The only way to install a repository server on the same machine as other services is to uninstall and reinstall ElectricFlow. You can install the repository server on a different machine to avoid uninstalling and reinstalling ElectricFlow.

Non-Server Platform Agent Upgrade

You cannot directly upgrade a non-server platform agent (that is, an agent on a machine that is not a supported ElectricFlow server platform). You must uninstall and then reinstall these machines using the ElectricFlow installer. For more information, see Uninstalling ElectricFlow on page 10-1 and Non-Server Platform Installation Method for UNIX Agents on page 14-30.

Stand-Alone Repository Server or Web Server Upgrade

You cannot directly upgrade a standalone repository server or standalone web server. You must uninstall and then reinstall these servers using the ElectricFlow installer. The uninstall and reinstall process is required to install an agent on the server machine. An agent is required on the machine with the standalone repository server or web server. For more information, see Uninstalling ElectricFlow on page 10-1, Installing ElectricFlow on page 3-1, and Copying Repository Contents on page 8-18.

User Interface Upgrade Method

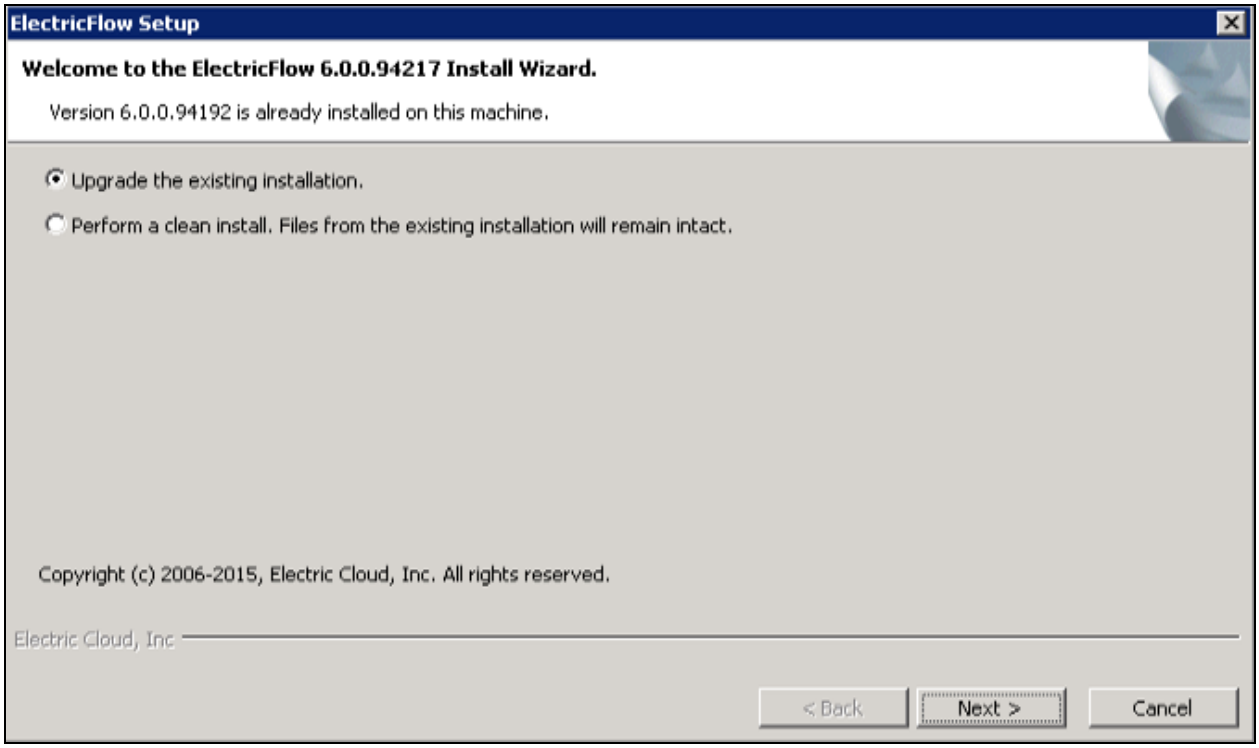
Use this procedure to upgrade ElectricFlow. Review Preparing for Your Upgrade on page 7-7 before performing this procedure.

Important: When upgrading the nodes in an ElectricFlow cluster, you must keep the other nodes stopped until the primary node upgrade is complete.

1. Enter the following command to make the installer file executable:

   ```
   chmod +x ./ElectricFlow-<version>
   ```

2. Double-click the `ElectricFlow-<version>` file to begin installation. The **Welcome to the ElectricFlow Install Wizard** screen appears.



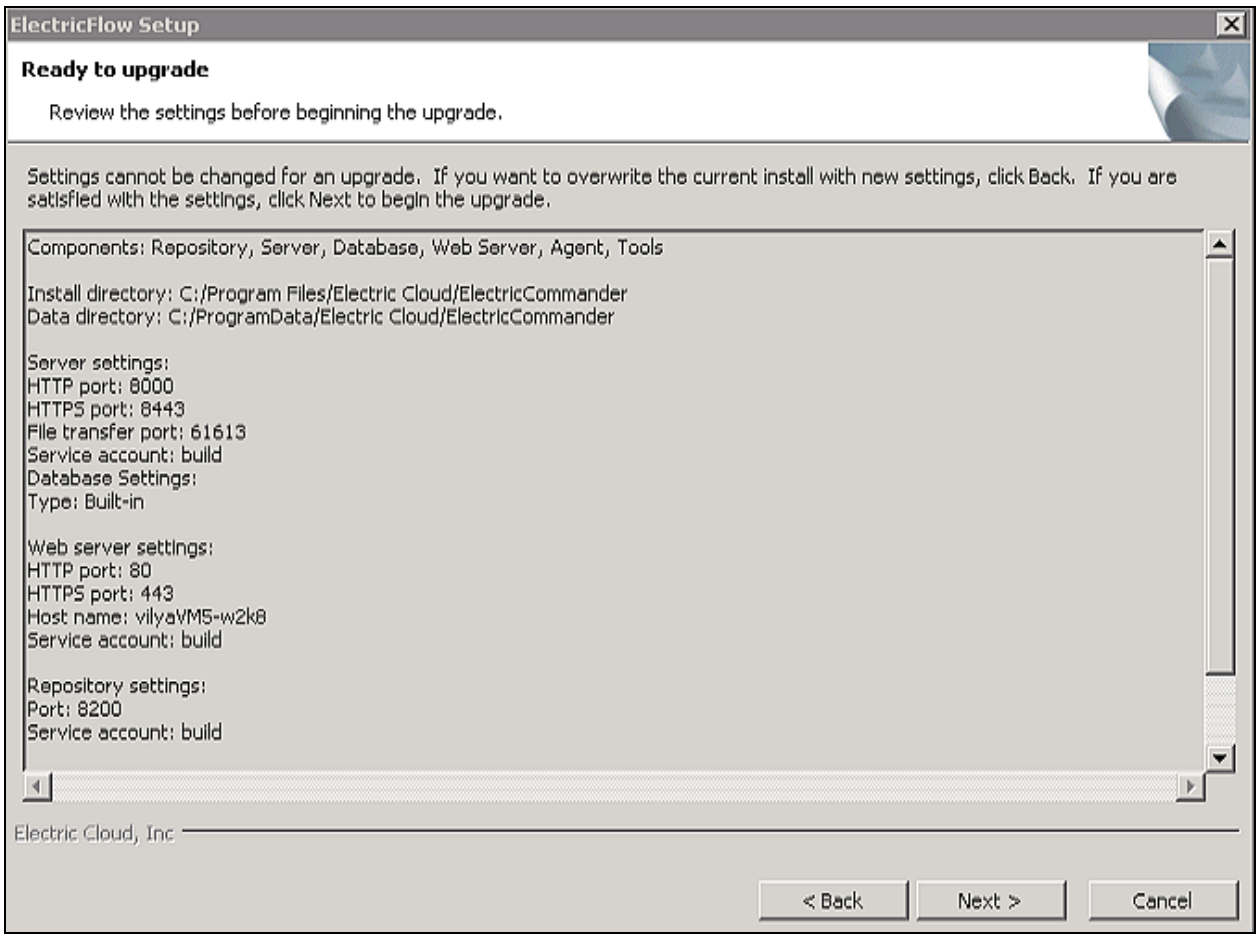
3. Choose one of the following options:

   - Select **Upgrade the existing installation** if you want to upgrade your current ElectricFlow installation directory.
   - Select **Perform a clean install** if you want to specify a different installation directory for the new version.

> **Note:** During a clean installation, current services remain running until you click **Next** on the Ready to Install screen. This means the new installation cannot use the same ports and directories as the existing installation. To use the same ports and directories, you must manually stop the existing services. This will free the existing ports and directories.

4. Click **Next** to upgrade the existing installation.

   The **Ready to upgrade** screen appears.

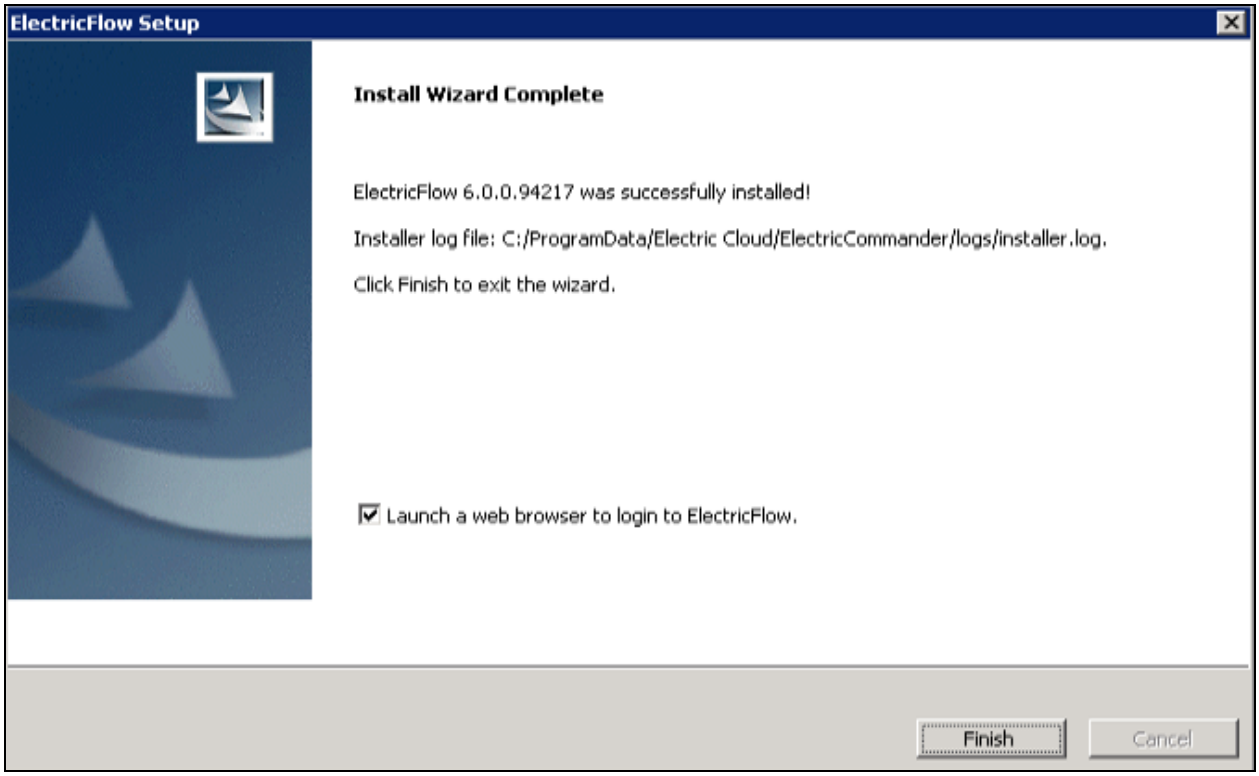


5. Review the upgrade settings.

   Use the **Back** button to change your selections if necessary.

6. Click **Next** to continue.

   The installer displays a status bar to show the progress of the upgrade process. You can also view the `installer.log` file to see progress. The time that it takes to complete this process depends on the size of the database. It may take fifteen minutes or longer to complete. Once this process is complete, the new ElectricFlow version is installed.



7. Select the **Launch a web browser to login to ElectricFlow** check box if you want ElectricFlow to open the login screen now.

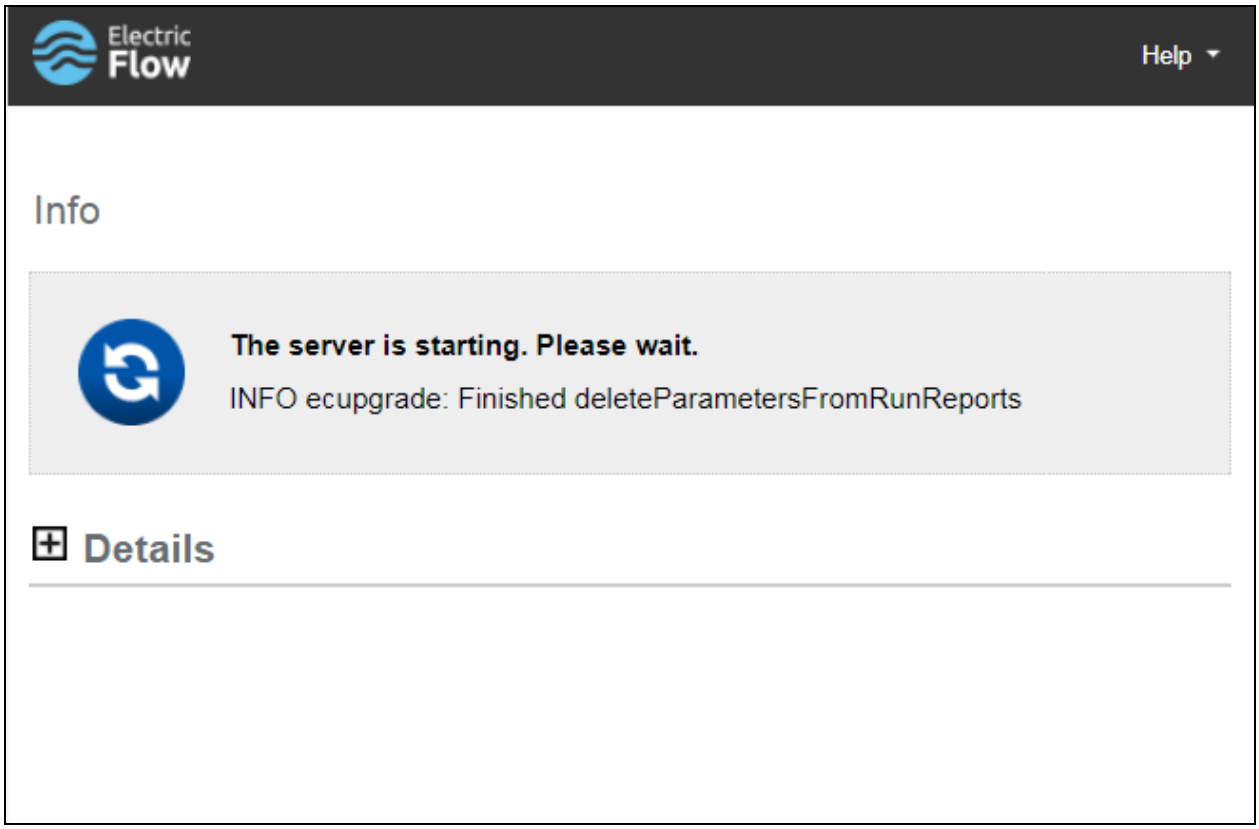
8. Click **Finish** to complete the upgrade.

   If an ElectricFlow server is being upgraded, when the installation is complete, the server continue to upgrade the database (if applicable). You cannot log in to the ElectricFlow server until the database upgrade finishes. You can view the upgrade status by using `ectool` from a command line:

   ```
   ectool getServerStatus
   ```

   However, for upgrades involving large databases, the output from `ectool getServerStatus` might remain unchanged for long periods. In this case, you can see more granular database update activity by following the procedure in the *KBEC-00086 - Enabling and collecting voluminous JDBC logging* KB article to add SQL logging to the `<data_dir>/logs/commander.log` file. You can view recent SQL logging updates to the file by using the Linux `tail <data_dir>/logs/commander.log` command.

   After clicking **Finish**, you might see a web page similar to the following screen if the upgrade is still in progress:



Interactive Command-Line Upgrade Method

Use the following procedure to complete a command-line upgrade of a Linux platform. Review Preparing for Your Upgrade on page 7-7 before performing this procedure.

Important: When upgrading the nodes in an ElectricFlow cluster, you must keep the other nodes stopped until the primary node upgrade is complete.

1. Enter the following command to make the installer file executable:

   ```
   chmod +x ./ElectricFlow-<version>
   ```

2. Choose one of the following commands to begin the upgrade:

   - If you have a Linux platform, enter `./ElectricFlow-<version>`.
   - If you have a Linux platform with the X Window System, the installer will automatically bring up the graphical user interface.

     To override this behavior, enter `./ElectricFlow-<version> --mode console`.

   The following prompt appears:

   ```
   Copyright (c) 2006-2018, Electric Cloud, Inc. All rights reserved.

   This will install ElectricFlow on your computer. Continue? [n/Y]
   ```

3. Enter: `y`

   The following prompt appears:

   ```
   Upgrade the existing <version> installation to version <version>? [n/Y
   ```

4. Choose one of the following options:

   - If you want to upgrade your current ElectricFlow installation directory, enter `y`.
   - Enter `n` to exit the installer.

   The following prompt appears:

   ```
   Installing agent...

   Installing server...

   Copied log file to "/opt/electriccloud/electriccommander/logs"

   ElectricFlow <version> was successfully installed!

   Installer log file: /opt/electriccloud/electriccommander/logs/installer.log.
   ```

Silent (Unattended) Upgrade Method

You can run the ElectricFlow upgrade in unattended (silent) mode with no user interface for either Windows or Linux.

Important: When upgrading the nodes in an ElectricFlow cluster, you must keep the other nodes stopped until the primary node upgrade is complete.

1. (Linux only) Enter the following command to make the installer file executable:

   ```
   chmod +x ./ElectricFlow-<version>
   ```

2. Enter the following command from a command line to begin a silent upgrade:

   ```
   ./ElectricFlow-<version> --mode silent
   ```

Copying Repository Contents

Perform the following steps to copy the contents of an existing repository server into a newly installed repository server:

1. Install the new repository server software.
2. Stop the existing and new repository servers.
3. Copy the entire contents of the repository backingstore directory from the existing repository server to the corresponding location on the newly installed repository server.

   The default location for the backingstore directory (`<datadir>/repository-data`) is:

   - **UNIX**—`/opt/electriccloud/electriccommander/repository-data`
   - **Windows**—`C:\ProgramData\Electric Cloud\ElectricCommander\repository-data`

Uploading Configuration Files to ZooKeeper if Needed

After you upgrade the ElectricFlow server node, you must again compare the <*data_dir*>\conf\commander.properties file with the file that you downloaded from ZooKeeper (which you saved to c:\temp). To do so, complete the following steps.

1. Open the <*data_dir*>\conf\commander.properties file.
2. Make sure that the COMMANDER_SERVER_NAME property is set to <*load_balancer_FQDN*>.
3. If the following line exists, remove it:

   ```
   COMMANDER_MQ_DISK_SPACE_CHECK_FREQUENCY=60
   ```

4. Check whether the following lines exist. If they do not exist, add them:

   ```
   COMMANDER_CRITICAL_SERVICES_MONITORING_FREQUENCY=60
   COMMANDER_CRITICAL_SERVICES_MONITORING_ENABLED=true
   COMMANDER_CRITICAL_SERVICES_MAX_ATTEMPTS_TO_BE_IN_PRIMARY_CLUSTER=5
   ```

   > **Note:** Ensure that these properties are not duplicated in the file.

5. Upload the new file to ZooKeeper as described in the Uploading Configuration Files to ZooKeeper on page 4-25 section in the "Clustering" chapter.

Chapter 9: Upgrading the ElectricFlow DevOps Insight Server

Before You Upgrade

Upgrading the ElectricFlow Server

Before you upgrade the DevOps Insight server, make sure that the ElectricFlow server is upgraded to the corresponding version.

For details about the overall steps for installing DevOps Insight on a group of servers to create a DevOps Insight server cluster, see Creating a DevOps Insight Server Cluster on page 4-36.

Upgrading the DevOps Insight Server on a System with Other ElectricFlow Components

For a production environment, Electric Cloud recommends that you run the DevOps Insight server on a system other than systems running other ElectricFlow components (such as the ElectricFlow server, web server, repository server, or agent). If you have installed it on the same system (such as for testing or other non-production or trial-basis situations), use the following upgrade process.

1. Uninstall the ElectricFlow DevOps Insight server from the system.
2. Upgrade the other ElectricFlow components on the system.
3. Install the new version of the ElectricFlow DevOps Insight server on the system.

Preserving Non-DevOps Insight Custom Settings in the elasticsearch.yml File

The DevOps Insight installer overwrites the `elasticsearch.yml` configuration file with a new file. As of DevOps Insight version 8.3, the file includes a `Custom Settings` section, which lets you add Elasticsearch settings not managed by the DevOps Insight server without being lost during an upgrade. If you added settings to this file in version 8.2 or earlier that you want to preserve, you must back up the file to a separate location *before* upgrading to version 8.3 or newer versions and then add the settings to the `Custom Settings` section after the upgrade. During future upgrades, the installer will preserve the settings in the `Custom Settings` section.

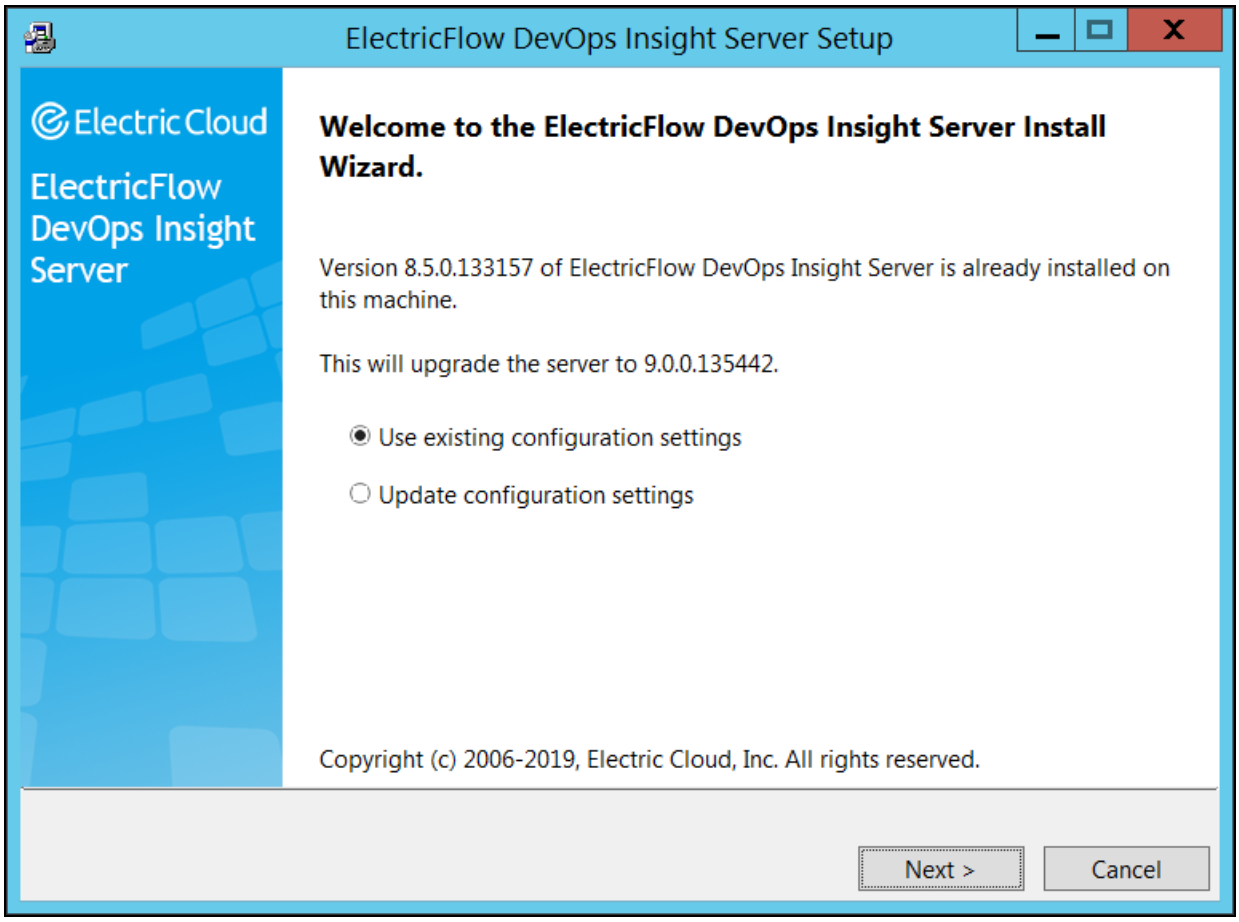
User Interface Upgrade Method

The graphical user interface installation method is supported by Windows platforms and Linux platforms running the X Window System.

Use this procedure to upgrade the ElectricFlow DevOps Insight server.

1. Double-click the following file to run the installer.
   - Linux: `ElectricFlowDevOpsInsightServer-x64-<version>`
   - Windows: `ElectricFlowDevOpsInsightServer-x64-<version>.exe`

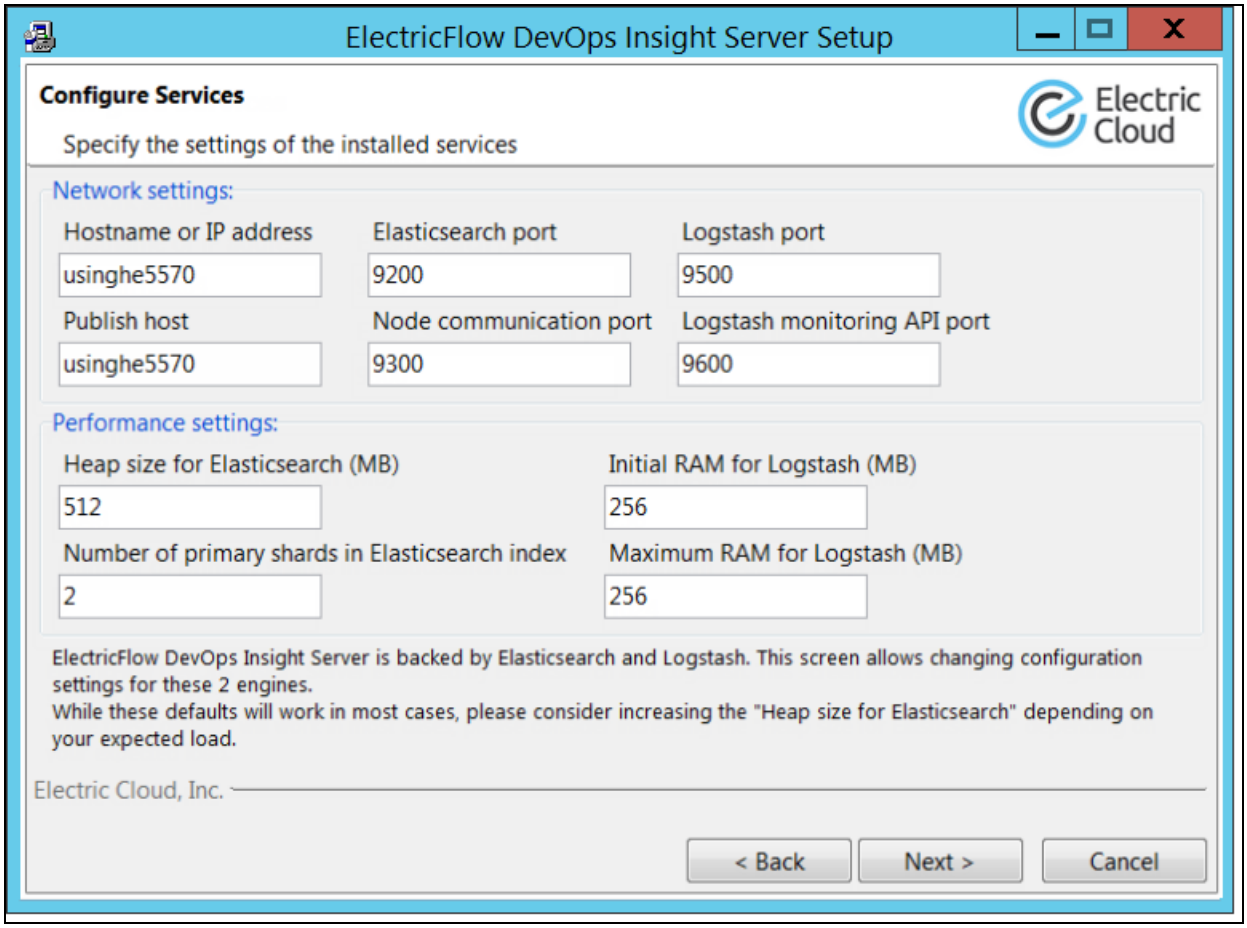
   The **Welcome to the DevOps Insight ElectricFlow Installer** screen appears:



2. Choose one of the following options:
   - Select **Use existing configuration settings** to upgrade your current installation without changing the settings.
   - Select **Update configuration settings** to specify new parameters for the upgraded software.

3. Click **Next** to continue.

The **Configure Services** screen appears:



Hostname or IP address—Name of the host that will be used to access the installed ElectricFlow DevOps Insight server.

Publish host—The network address that the Elasticsearch node advertises to other nodes in the cluster, so that those nodes can connect to it.

Elasticsearch port—Port number to be used to access Elasticsearch.

The DevOps Insight server uses the Elasticsearch search engine and the Logstash data-collection and log-parsing engine to gather data from the ElectricFlow server for use in the Deployments, Releases, and Release Command Center dashboards.

Node communication port—Port number used for internal communication between nodes within the Elasticsearch cluster.

Logstash port—Port number to be used to store information in Logstash.

Logstash monitoring API port—Port number used by the Logstash monitoring APIs that provide runtime metrics about Logstash.

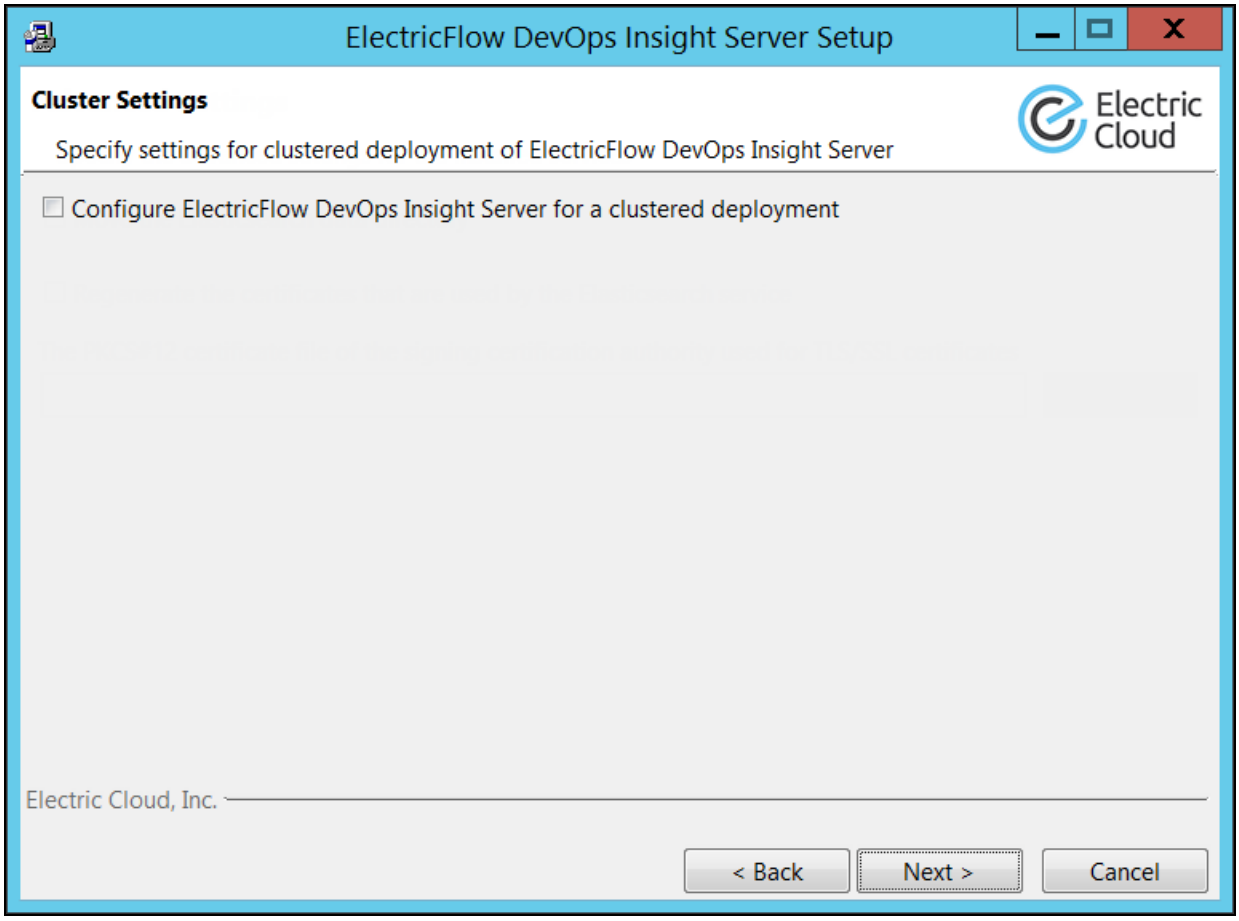
Heap size for Elasticsearch (MB)—Heap size for Elasticsearch in megabytes.

Number of primary shards in Elasticsearch index—Number of primary shards in the Elasticsearch index.

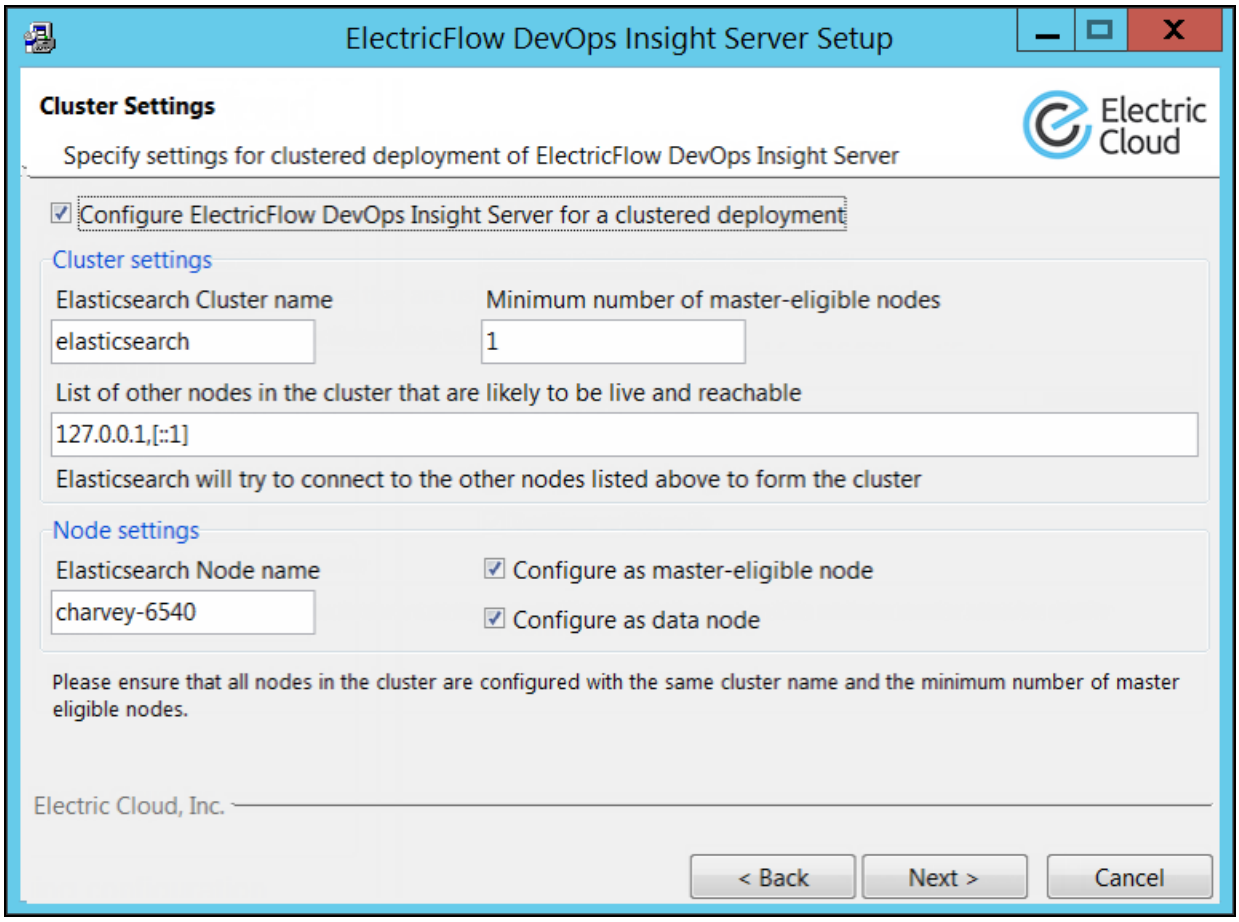
Initial RAM for Logstash (MB)—Initial heap size for Logstash in megabytes.

Maximum RAM for Logstash (MB)—Maximum heap size for Logstash in megabytes.

4. Complete the information on the **Configure Services** screen, and click **Next** to continue. The **Cluster Settings** screen appears:



Configure ElectricFlow DevOps Insight Server for a clustered deployment—Check this field if you want to add this system to a DevOps Insight server cluster. If you do so, additional fields appear to let you enter the details about this node and the cluster:

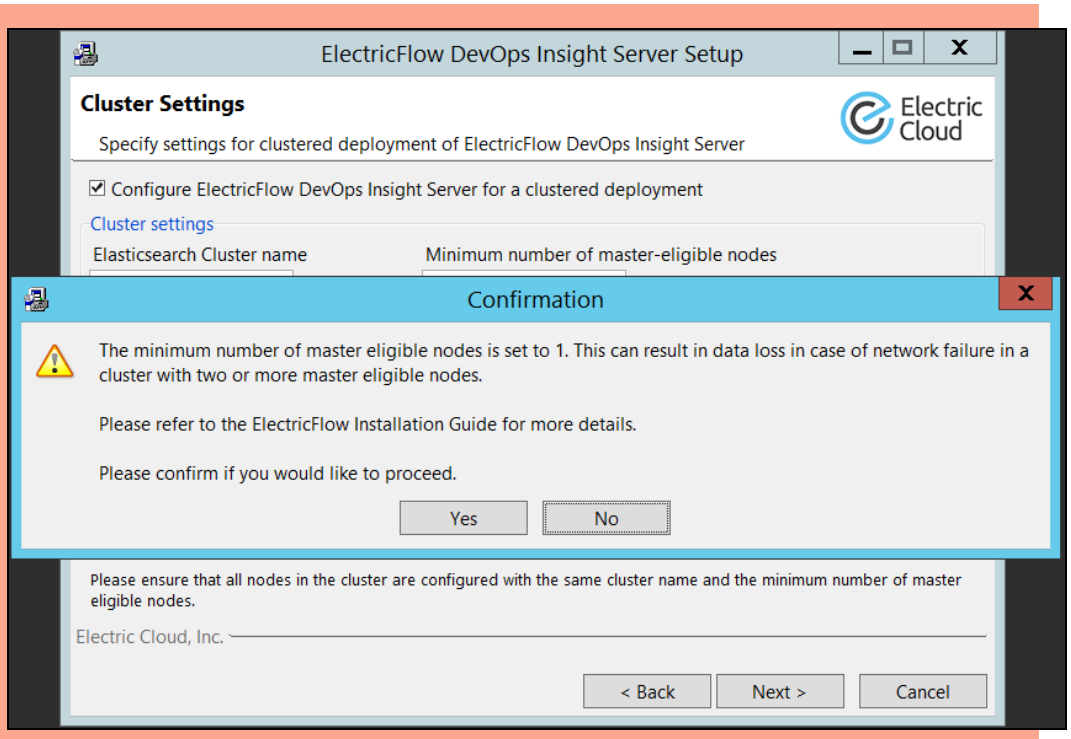


Elasticsearch Cluster name—Name of the cluster. The cluster name must be unique across all Elasticsearch clusters in the network.

Minimum number of master-eligible nodes—Minimum number of master-eligible nodes that must be visible in order to form a cluster. For details about how to determine how many master-eligible nodes you need for your cluster, see 1. Planning the Total Number of Master-Eligible Nodes on page 4-37. The master node will be elected from the list of master-eligible nodes.

For details about master-eligible nodes, see the Node module in the *Elasticsearch Reference*. For details about master elections, see the Zen Discovery module in the *Elasticsearch Reference*.

> **Important:**
> If you specify `1`, you are asked to confirm this number in the following warning:



To prevent data loss in case of network failure, the minimum number of master-eligible nodes that must be visible in the cluster must be set to a quorum of master-eligible nodes:

(Number of master-eligible nodes in the cluster / 2) + 1

For example, in a cluster with three master-eligible nodes, minimum number of master-eligible nodes should be set to 2.

The minimum number of master-eligible nodes should be set to 1 only if you intend to run a single-node cluster. For a multi-node cluster, the minimum number of master-eligible nodes must be set to a quorum as described above.

List of other nodes in the cluster that are likely to be live and reachable—Additional nodes that are running DevOps Insight and can become part of the cluster. These can be any nodes (whether they are master-eligible or not). You can enter any combination of IP addresses or host names.

This is mandatory for additional nodes and optional for the first node. You should specify in this list all available master nodes.

Elasticsearch Node name—Name of this node in the cluster. This serves as a unique identifier and therefore must be a unique name in the cluster.

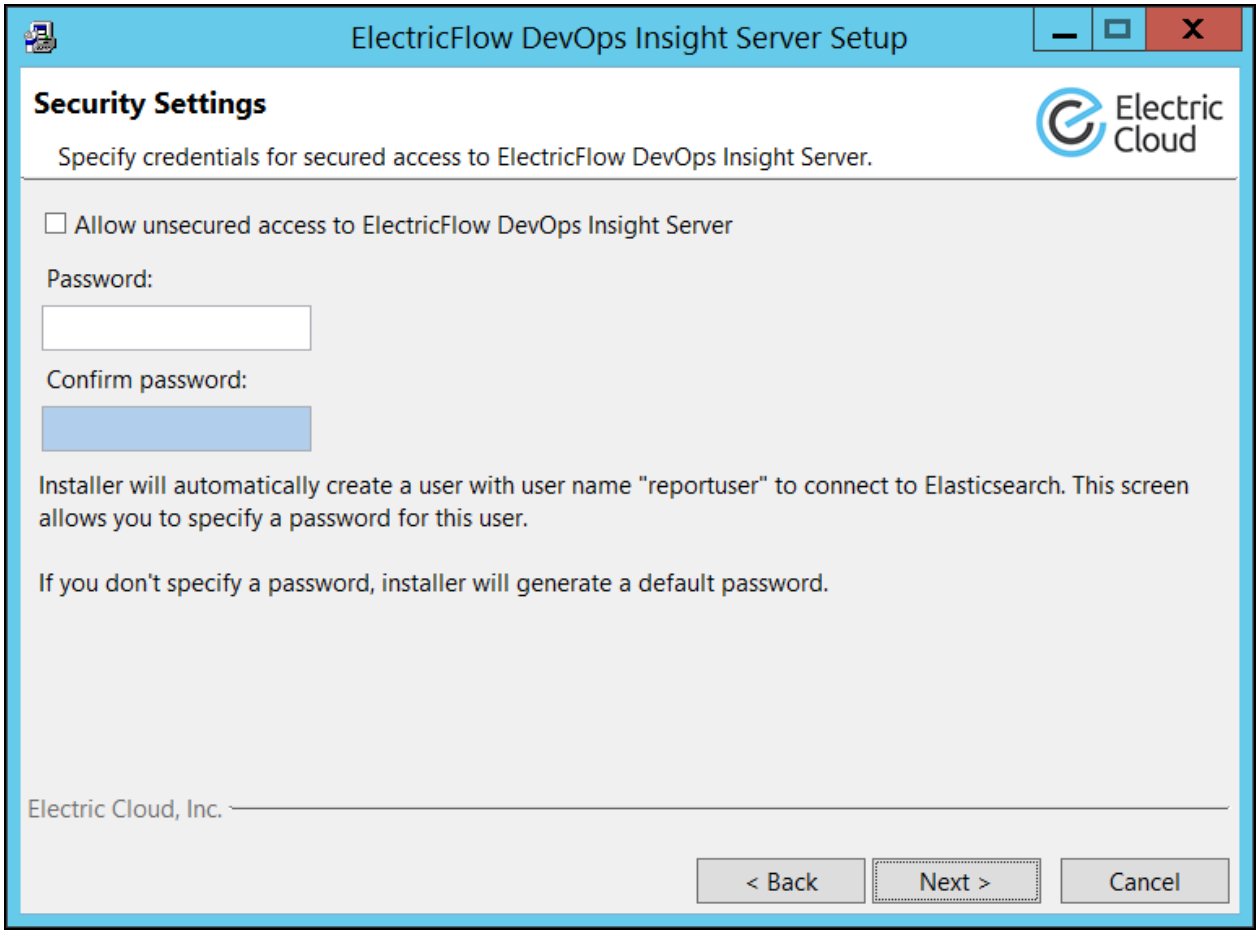
This is the first node in the cluster—Check this checkbox if this is the first node that you are adding to the cluster.

Configure as master-eligible node—Makes this node eligible to be elected as a master node. Master-eligible nodes participate in updating the cluster state as well as elections of the master node. A master-eligible node can also be a data node. The first node that you add to a cluster is always a master-eligible node (and also a data node).

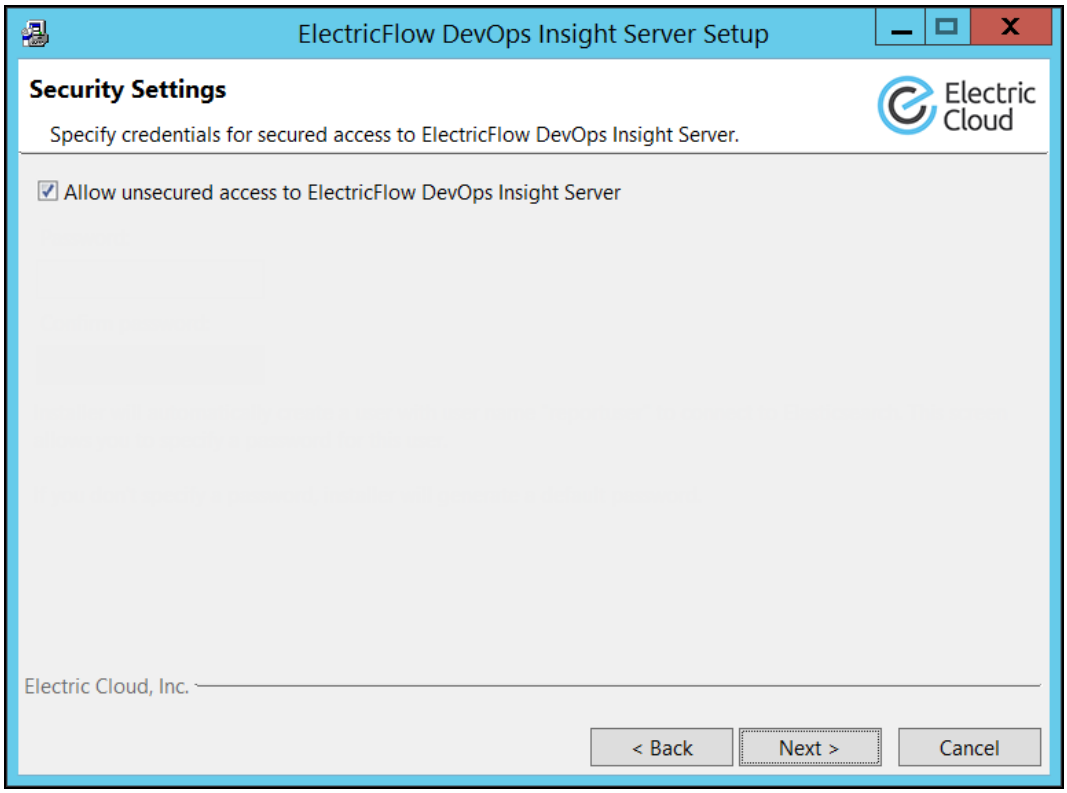
Configure as data node—Determines whether this node will be a data node. A data node stores data that is indexed into Elasticsearch and performs data-related operations such as CRUD, search, and aggregations. A data node can also be a master-eligible node. The first node that you add to a cluster is always a data node (and also a master-eligible node).

5. Complete the information on the **Cluster Settings** screen.

6. Click **Next** to continue.
   The **Security Settings** screen appears:



Allow unsecured access to ElectricFlow DevOps Insight Server—Check this field if you do *not* want to use a secure protocol and authentication when accessing the DevOps Insight server:



Otherwise, the **Password** and **Confirm password** fields let you enter the server password:

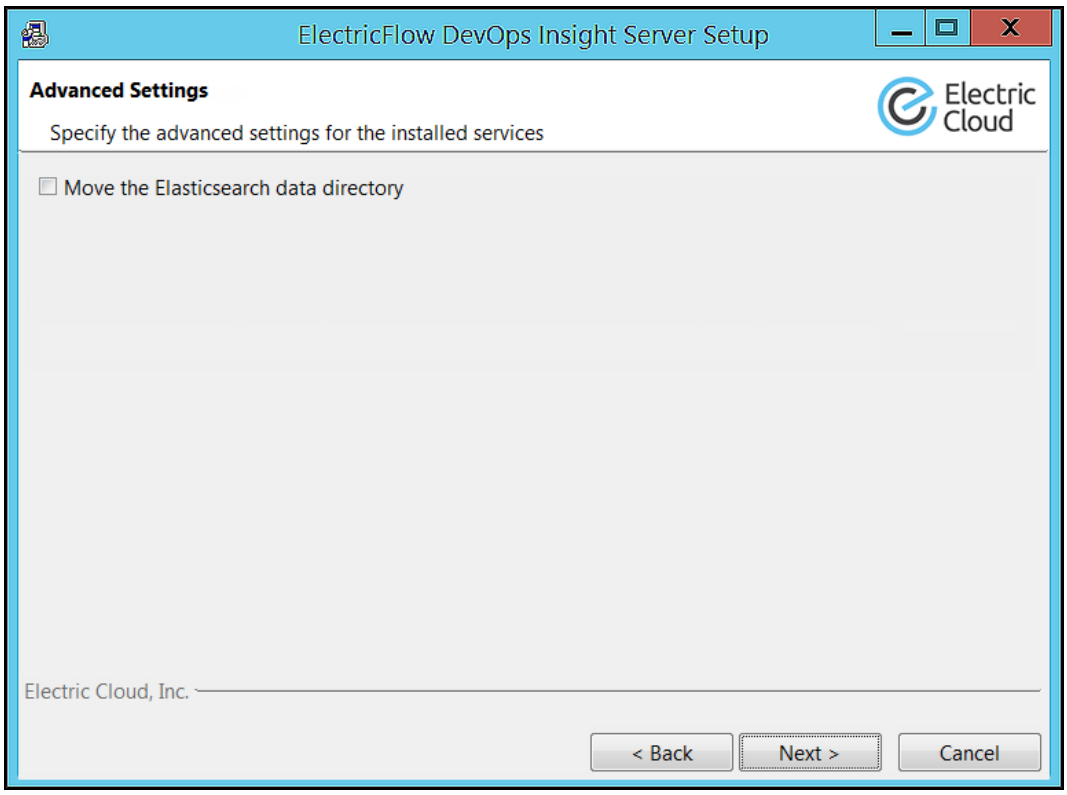
Password—Password to be used to access the server. The installer will automatically create a user with user name `reportuser` and the password that you specified. If you do not specify a password, the installer will generate a default password. (Electric Cloud recommends that you change this password.)

Confirm password—Confirm the password. Enter the same password in this field as in the previous field.

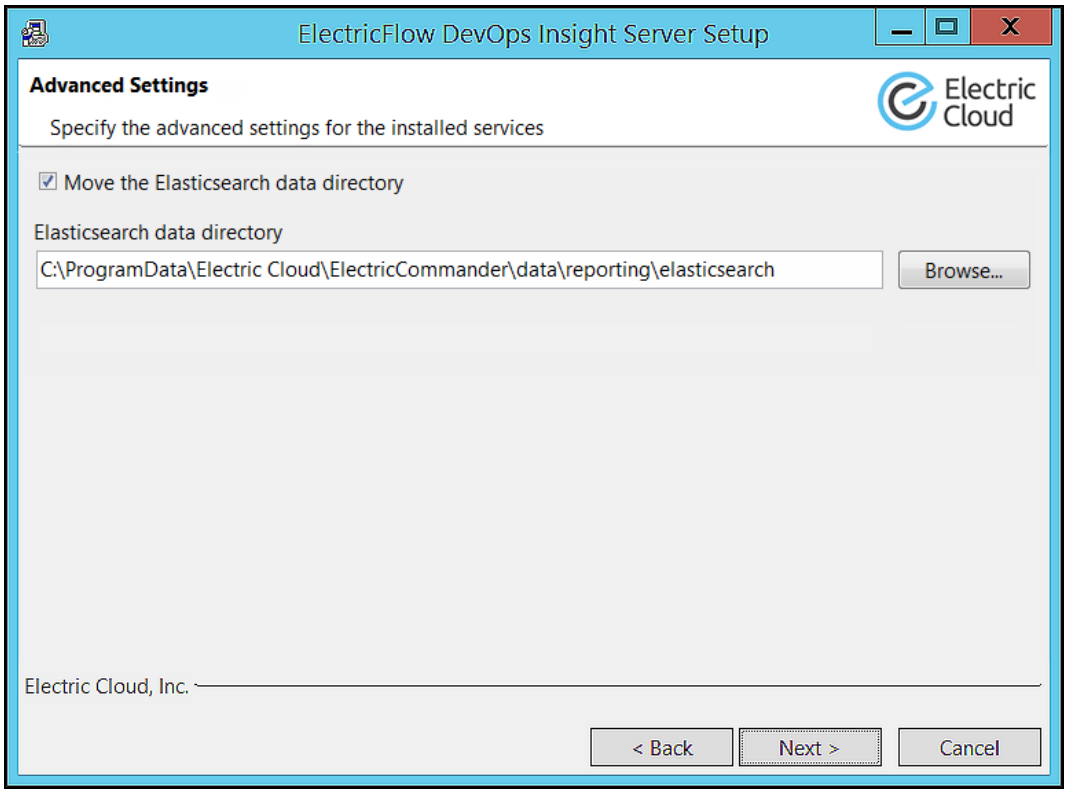
> **Important:** Unsecured access is not recommended for use in a production environment.

7. Click **Next** to continue.

   The following upgrades-only screen appears during the upgrade:



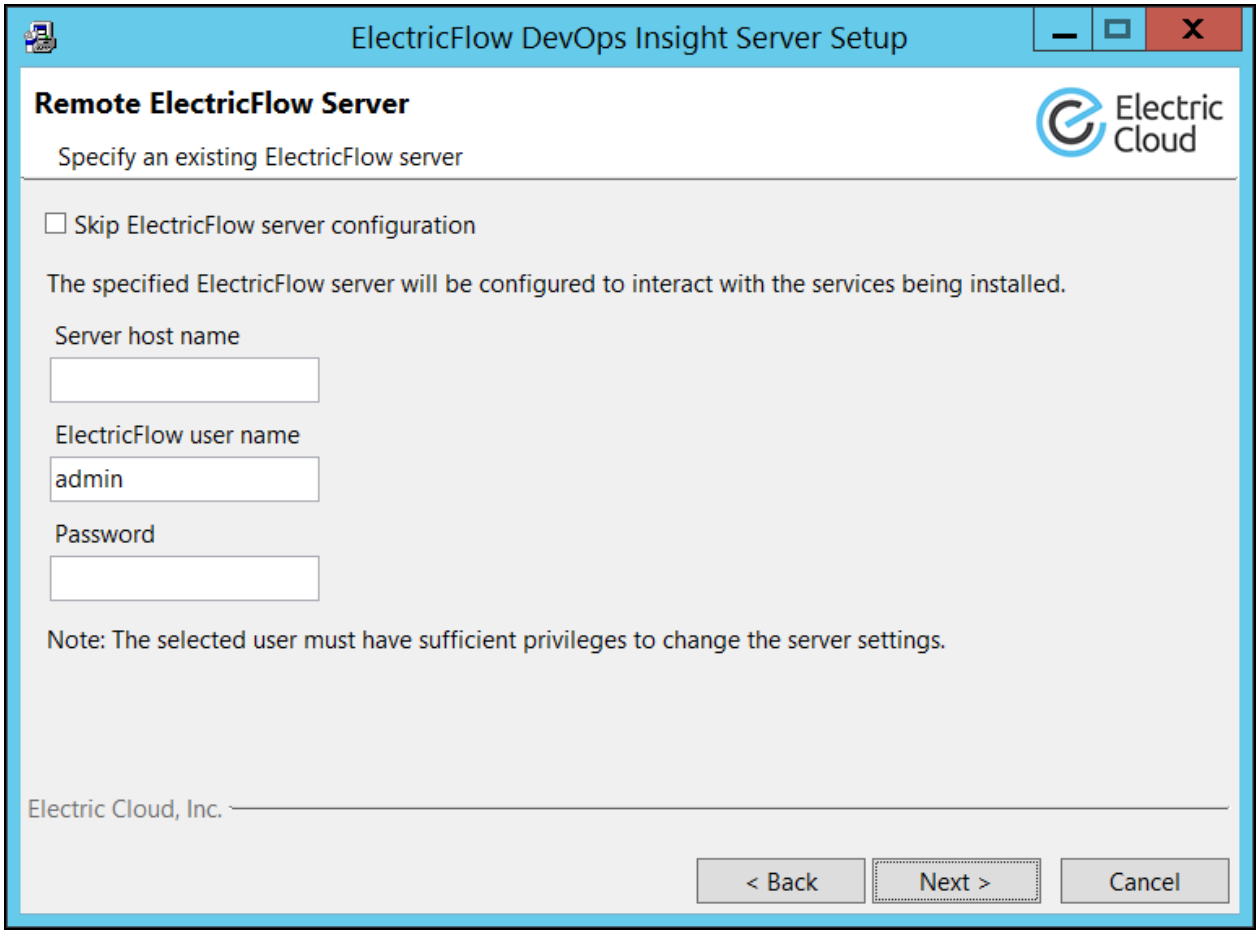
The **Move the Elasticsearch data directory** checkbox lets you change the location of the Elasticsearch index data. This option is useful if the system ran low on disk space because the data files outgrew the space available on the volume where the data directory is configured. If you check the checkbox, the **Elasticsearch data directory** field appears:


ElectricFlow DevOps Insight Server Setup
Advanced Settings
Specify the advanced settings for the installed services
Electric Cloud
Move the Elasticsearch data directory
Elasticsearch data directory
C:\ProgramData\Electric Cloud\ElectricCommander\data\reporting\elasticsearch
Browse...
Electric Cloud, Inc.
< Back
Next >
Cancel

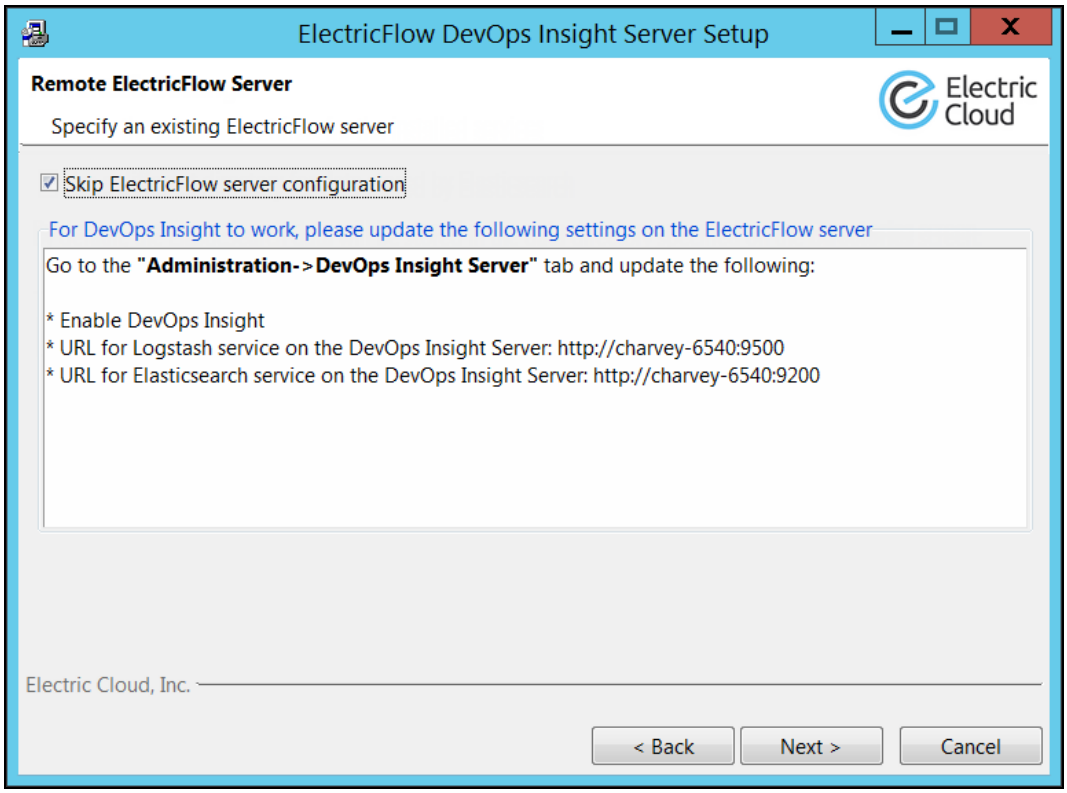
8. Complete the information on the **Advanced Settings** screen, and click **Next** to continue.

> **Note:** If you are currently using the default Elasticsearch password for the `reportuser` user for regular access to the DevOps Insight server services, Electric Cloud recommends that you enter `y` so that you can change that password.

The **Remote ElectricFlow Server** screen appears:



Skip ElectricFlow server configuration—Determines whether to skip the automatic configuration of the remote ElectricFlow server with the services being installed. If you choose this option, the screen changes as follows:



If you choose this option, fill in the fields in the screen as follows:

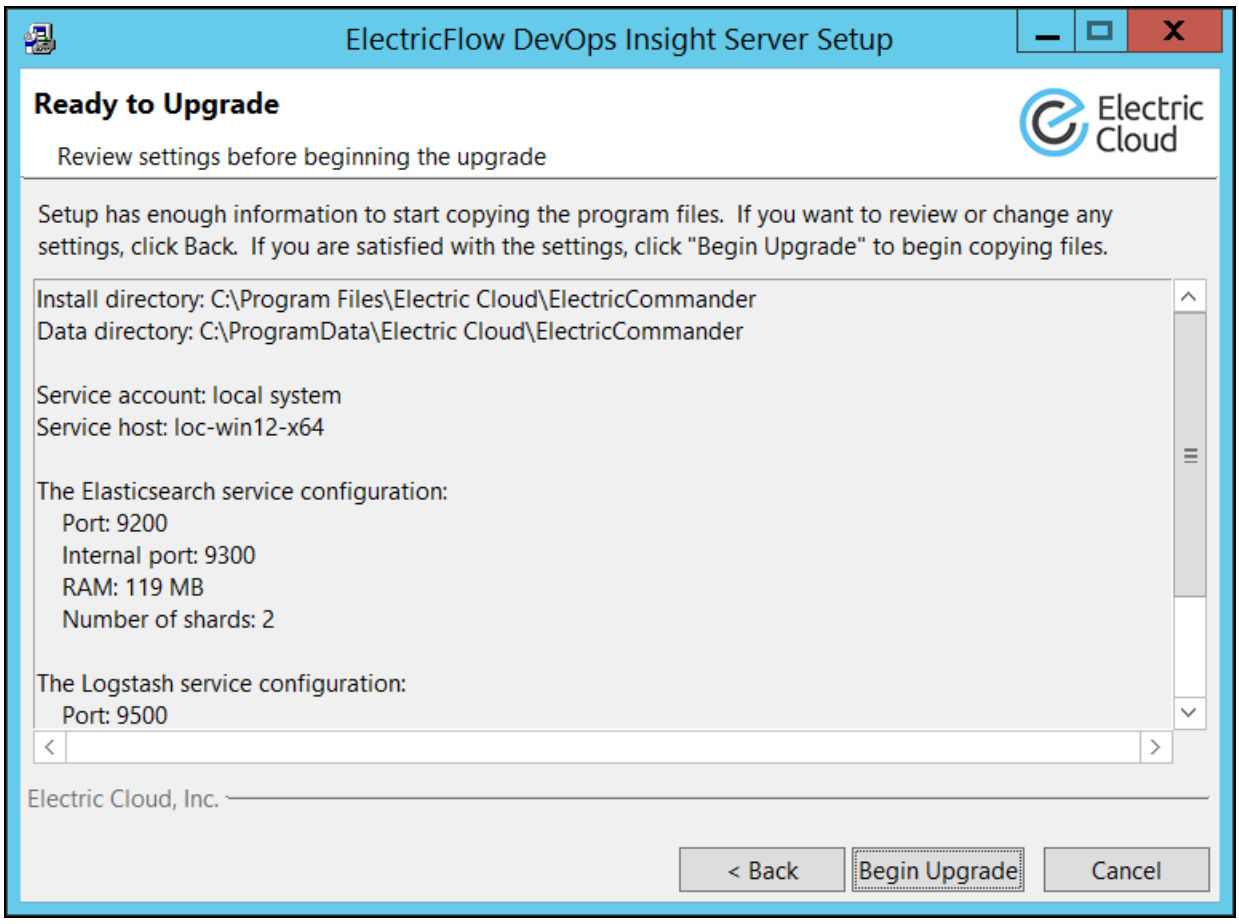
Server host name—Name of the ElectricFlow server that will communicate with this DevOps Insight server. If the remote server is using a non-default HTTPS port, you must enter *<host>*:*<port>*.

ElectricFlow User Name—Name of an ElectricFlow user on the ElectricFlow server who has sufficient privileges to edit server settings. This field defaults to the ElectricFlow-supplied `admin` user.

Password—Password for the ElectricFlow user. The default password for the `admin` user is `changeme`.

9. Click **Next**.

   The **Ready to Upgrade** screen appears:

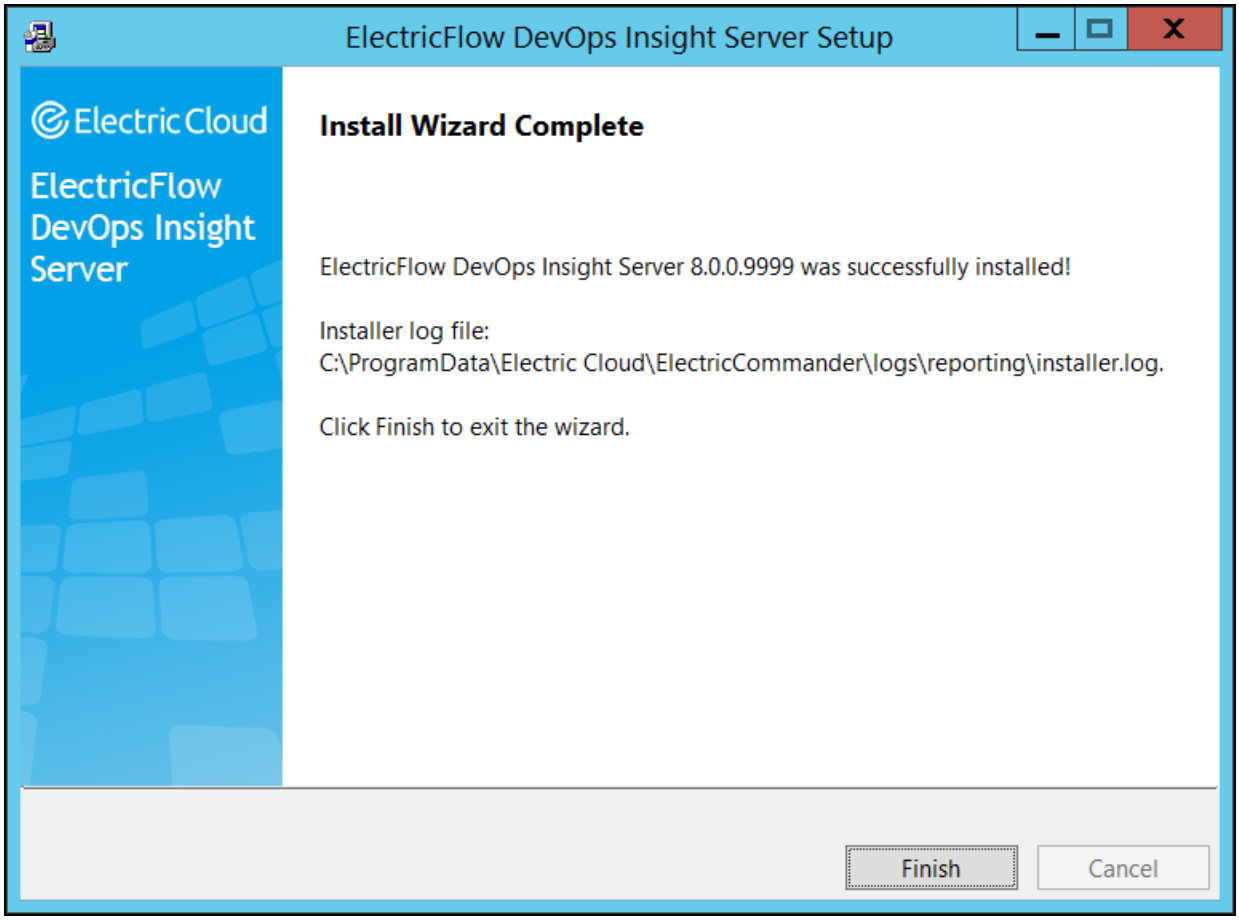


10. Review your upgrade settings.

    Use the **Back** button to change your selections if necessary.

11. Click **Begin Upgrade** to continue.

    The installer displays a status bar to show the progress of the upgrade process. You can also view the `installer-EFlowReportServ.log` file to see the upgrade progress. Once this process is complete, the new ElectricFlow DevOps Insight server version is installed:



12. Click **Finish** to complete the upgrade.

Interactive Command-Line Upgrade Method

The command-line user interface upgrade method is supported only by Linux platforms. In this mode, additional command line parameters that are listed in Windows or Linux DevOps Insight Server Silent Unattended Installation Example on page 3-130 can be used.

Use the following procedure to complete an interactive command-line upgrade of a Linux platform.

1. Choose one of the following commands:

   - On Linux *without* the X Window System, enter `sudo ./ElectricFlowDevOpsInsightServer-x64-<version>`

- On Linux *with* the X Window System, enter `sudo ./ElectricFlowDevOpsInsightServer-x64-<version> --mode console`

  This command prevents the installer from automatically invoking the installation graphical user interface.

The following prompt appears:

```
Logging to "/tmp/ijtmp_C75F6886-BE25-D84C-BB8F-56EC5C16DBC1/installer-
EFlowReportServ.log"

Installing temporary...
Copyright (c) 2006-2019, Electric Cloud, Inc. All rights reserved.

Version <version> of ElectricFlow DevOps Insight Server is already installed on
this machine.

Upgrade the server to <version>? [n/Y]
```

2. Enter `y`.

   The following prompt appears:

   ```
   Do you want to update configuration settings? [y/N]
   ```

3. Choose one of the following options:

   - To upgrade your current installation without changing the settings, enter `n`.
   - To specify new parameters for the upgraded software, enter `y`.

   The following prompt appears:

   ```
   Choose the port which will be used by Elasticsearch [9200]
   ```

   The DevOps Insight server uses the Elasticsearch search engine and the Logstash data-collection engine to gather data from the ElectricFlow server for use in the DevOps Insight dashboards.

4. If you want to specify a non-default port number, enter that number, or accept the default port number by pressing `Enter`.

   The following prompt appears:

   ```
   Choose the port which will be used by the Elasticsearch service for
   communication between nodes within the Elasticsearch cluster [9300]
   ```

   This port is used for internal communication between nodes within the Elasticsearch cluster.

5. If you want to specify a non-default port number, enter that number, or accept the default port number by pressing `Enter`.

   The following prompt appears:

   ```
   Choose the port which will be used by Logstash [9500]
   ```

6. If you want to specify a non-default port number, enter that number, or accept the default port number by pressing `Enter`.

   The following prompt appears:

   ```
   Choose the port which will be used by the Logstash service for the Logstash
   monitoring APIs [9600]
   ```

   This port is used by the Logstash monitoring APIs that provide runtime metrics about Logstash.

7. If you want to specify a non-default port number, enter that number, or accept the default port number by pressing `Enter`.

   The following prompt appears:

   ```
   Do you want to specify additional Elasticsearch cluster mode settings? [y/N] y
   ```

8. (Optional) Enter `y` if you want to add this system to a DevOps Insight server cluster. Otherwise, enter `n`.

   If you enter `y`, the following prompt appears:

   ```
   Please ensure that all nodes in the cluster are configured with the same
   cluster name and the minimum number of master eligible nodes.

   Specify the name of the Elasticsearch cluster [elasticsearch]
   ```

   > **Note:** The following prompts related to the cluster are skipped if you declined to configure it automatically.

9. Enter the name of the cluster.

   The following prompt appears:

   ```
   Specify comma-delimited list of other nodes in the Elasticsearch cluster that
   are likely to be live and reachable [127.0.0.1,[::1]]
   ```

10. Enter any additional nodes that are running DevOps Insight and can become part of the cluster.

    These can be any nodes (whether they are master-eligible or not). You can enter any combination of IP addresses or host names.

    The following prompt appears:

    ```
    Specify minimum number of master-eligible nodes that must be visible in order
    to form an Elasticsearch cluster [1]
    ```

11. Enter the minimum number of master-eligible nodes that must be visible in order to form a cluster.

    For details about how to determine how many master-eligible nodes you need for your cluster, see Creating a DevOps Insight Server Cluster on page 4-36. The master node will be elected from the list of master-eligible nodes.

    For details about master-eligible nodes, see the Node module in the *Elasticsearch Reference*. For details about master elections, see the Zen Discovery module in the *Elasticsearch Reference*.

    Important:
    If you specify `1`, you are asked to confirm this number in the following warning:

    ```
    The minimum number of master eligible nodes is set to 1. This can result
    in data loss in case of network failure in a cluster with two or more
    master eligible nodes.

    Please refer to the ElectricFlow Installation Guide for more details.

    Please confirm if you would like to proceed. [N/y] n
    ```

    To prevent data loss in case of network failure, the minimum number of master-eligible nodes that must be visible in the cluster must be set to a quorum of master-eligible nodes:

    (Number of master-eligible nodes in the cluster / 2) + 1

    For example, in a cluster with three master-eligible nodes, the minimum number of master-eligible nodes should be set to 2.

    The minimum number of master-eligible nodes should be set to 1 only if you intend to run a single-node cluster. For a multi-node cluster, the minimum number of master-eligible nodes must be set to a quorum as described above.

    The following prompt appears:

    ```
    Specify the name of this node in the Elasticsearch cluster [loc-10-lin-ub1604-
    64]
    ```

12. Enter the name of this node in the cluster.

    This serves as a unique identifier and therefore must be a unique name in the cluster.

    The following prompt appears:

    ```
    Is this node eligible to be elected as the master node, which controls the
    Elasticsearch cluster? [n/Y]
    ```

13. Enter `Y` if this node is master-eligible. (If this is the only node, it must be master-eligible.) Otherwise, enter `n`.

    The following prompt appears:

    ```
    Does this node holds data and performs data related operations such as CRUD,
    search, and aggregations? [n/Y]
    ```

14. Enter `Y` if this node holds data and performs data-related operations such as CRUD, search, and aggregations. Otherwise, enter `n`.

    The following prompt appears:

    ```
    Installer will automatically create a user with user name "reportuser" to
    connect to Elasticsearch.

    Specify a password for this user []
    ```

    > **Note:** If you are currently using the default Elasticsearch password for the `reportuser` user for regular access to the DevOps Insight server services, Electric Cloud recommends that you enter `y` so that you can change that password.

15. Enter the password that will be used to access the server. The installer will automatically create a user named `reportuser` with the password that you provide. If you do not specify a password, the installer generates a default password `changeme`.

    The following prompt appears:

    ```
    Confirm password []
    ```

16. Enter the same password as before.

    The following prompt appears:

    ```
    Do you want to regenerate the certificates used for secured access to
    ElectricFlow DevOps Insight Server? [y/N]
    ```

17. Enter `y` if you want you regenerate the certificates that are used by the DevOps Insight services. Otherwise, enter `N`.

    The following prompt appears:

    ```
    Do you want to provide the certificate file containing a CA-signed certificate
    for the ElectricFlow DevOps Insight Server, any intermediate CA certificates
    and a private key [y/N]
    ```

18. Enter `y` if you want to provide the path to a new PKCS#12 certificate file of the signing certification authority used for TLS/SSL certificates. Otherwise, enter `N`.

    Any certificate regeneration will occur with the new certificate if you specify one.

    The following prompt appears:

    ```
    Do you want to move the Elasticsearch data directory? [y/N]
    ```

19. Enter `N` to continue, or enter `y` to specify different directory locations.

    The following prompt appears:

    ```
    Do you want to specify the remote ElectricFlow server which will be configured
    to interact with the services being installed?
    ```

20. Enter `y` if you want to automatically configure the remote ElectricFlow server to interact with the services being installed.

    > **Note:** The following prompts related to the configuration of the remote ElectricFlow server are skipped if you declined to configure it automatically.

    The following prompt appears:

    ```
    Specify the host[:port] of the remote ElectricFlow server []
    ```

21. Enter the name of the ElectricFlow server that will communicate with this DevOps Insight server. If the remote server is using a non-default HTTPS port, you must specify the host name as *<host>*:*<port>*. If you do not specify a port, HTTPS port 8443 is assumed (the same as the ElectricFlow server default port).

    The following prompt appears:

    ```
    Specify the user name with which to login to "<remote host>" [admin]
    ```

22. Enter the name of an ElectricFlow user on the ElectricFlow server who has sufficient privileges to edit server settings. This field defaults to the ElectricFlow-supplied `admin` user.

    The following prompt appears:

    ```
    Specify the password for "<remote user>" on "<remote host>" []
    ```

23. Enter the password for the ElectricFlow user. The default password for the `admin` user is `changeme`.

    The following prompt appears:

    ```
    The ElectricFlow DevOps Insight Server will be configured on ElectricFlow
    server version <version> on <remote host>
    ```

    The following information appears:

    ```
    ================================================================

    For DevOps Insight to work, please update the following settings on the
    ElectricFlow server

    Go to the "Administration->DevOps Insight Server" tab and update the following:

    * Enable DevOps Insight
    * URL for Logstash service on the DevOps Insight Server: https://ip-10-0-0-
    84.us-west-1.compute.internal:9500
    * URL for Elasticsearch service on the DevOps Insight Server: https://ip-10-0-
    0-84.us-west-1.compute.internal:9200
    * Username: reportuser
    * Password: Password for authenticating with the DevOps Insight Server

    ================================================================
    Installing ElectricFlow DevOps Insight Server...
    Installing elasticsearch...
    Installing logstash...
    Installing jre-64...
    Copied log file to "/opt/electriccloud/electriccommander/logs/reporting"
    Installation complete.
    ```

    The DevOps Insight server uses the Elasticsearch search engine and the Logstash data-collection and log-parsing engine to gather data from the ElectricFlow server for use in the various DevOps Insight dashboards.

    ElectricFlow is installed on the machine.

Silent (Unattended) Upgrade Method

You can run the ElectricFlow DevOps Insight server installer in unattended (silent) mode with no user interface on either Windows or Linux. Enter one of the following commands from a command line.

- Linux: `sudo ./ElectricFlowDevOpsInsightServer-x64-<version> --mode silent <arguments>`
- Windows: `ElectricFlowDevOpsInsightServer-x64-<version>.exe --mode silent <arguments>`

where:

- *<version>* is your ElectricFlow DevOps Insight server version number.
- *<arguments>* represents any additional silent install arguments for upgrading the server.

  For a list of the available arguments, see Silent Install Arguments on page 3-113.

Reconfiguring the DevOps Insight Server After the Upgrade

The installers (GUI, interactive console, and silent mode) for the DevOps Insight server do not preserve the configuration setting for the DevOps Insight server host name (`--hostName`) or the setting for the Elasticsearch number of shards (`--elasticsearchNumberOfShards`) during the upgrade from 7.3 to 9.0. If you specified nondefault values during the 7.3 Reporting server installation, you must re-specify these settings during the upgrade. (All other settings are preserved.)

Configuring the DevOps Insight Server on the ElectricFlow Server

If you chose to skip the option to configure the remote ElectricFlow server during the installation or upgrade of the DevOps Insight server, you must do so afterward to ensure connectivity and authentication between the DevOps Insight server and the ElectricFlow server. To do this, you use the **Administration** > **DevOps Insight Server** tab in the Automation Platform. For details, see the "DevOps Insight Server Configuration" section in the *ElectricFlow User Guide* at http://docs.electric-cloud.com/eflow_doc/FlowIndex.html.

Checking the DevOps Insight Server Configuration on the ElectricFlow Server

You can confirm the correct DevOps Insight Server settings by entering the following ectool command on the ElectricFlow server:

```
ectool getDevOpsInsightServerConfiguration
```

Following is sample output:

```
<response requestId="1" nodeId="192.168.5.138">
    <devOpsInsightServerConfiguration>
      <devOpsInsightServerConfigurationId>12642169-71c4-11e7-8a08-
0050568f29b0</devOpsInsightServerConfigurationId>
      <createTime>2017-07-26T05:34:19.404Z</createTime>
      <elasticSearchUrl>https://192.168.5.54:9200</elasticSearchUrl>
      <enabled>1</enabled>
      <lastModifiedBy>admin</lastModifiedBy>
      <logStashUrl>https://192.168.5.54:9500</logStashUrl>
      <modifyTime>2017-07-26T05:40:13.458Z</modifyTime>
      <owner>admin</owner>
      <userName>reportuser</userName>
    </devOpsInsightServerConfiguration>
  </response>
```

For details about the `getDevOpsInsightServerConfiguration` options, enter

```
ectool getDevOpsInsightServerConfiguration --help
```

Testing Connectivity and Authentication Between the DevOps Insight Server and the ElectricFlow Server

After you enable connectivity and authentication between the DevOps Insight server and the ElectricFlow server, you can perform a test by using one of the following methods:

- Check the **Test Connection** checkbox in the **Administration** > **DevOps Insight Server** subtab of the Administration Platform web UI on the ElectricFlow server and click **OK**.
- Enter the following ectool command on the ElectricFlow server:

  ```
  ectool setDevOpsInsightServerConfiguration --testConnection 1
  ```

  For details about the `setDevOpsInsightServerConfiguration` options, enter

  ```
  ectool setDevOpsInsightServerConfiguration --help
  ```

  For example, the following response appears if the user name or password is incorrect:

  ```
  ectool error [InvalidCredentials]: HTTP/1.1 401 Unauthorized: Access to
    'https://192.168.5.54:9500' is denied due to invalid credentials.
  ```

  Also, for example, the following response appears if you specify an invalid `elasticSearchUrl` or `logstashUrl`:

  ```
  ectool error [InvalidUrl]: The url 'https://192.168.5.54:9500' is invalid
  ```

  The following example shows the response when a valid `elasticSearchUrl` is used:

  ```
  /opt/electriccloud/electriccommander/bin$ ./ectool
  setDevOpsInsightServerConfiguration
    --elasticSearchUrl https://192.168.5.54:9200 --testConnection 1
  ```

To do so, you use the **Administration** > **DevOps Insight Server** tab in the Automation Platform. For details, see "DevOps Insight Server Configuration" in the "DevOps Insight" chapter of the *ElectricFlow User Guide* at http://docs.electric-cloud.com/eflow_doc/FlowIndex.html.

Chapter 10: Uninstalling ElectricFlow

This section contains instructions for uninstalling ElectricFlow or the DevOps Insight server from various types of platforms.

Uninstalling ElectricFlow on Linux, UNIX, or macOS

Use the following procedures to uninstall ElectricFlow completely from a Linux, UNIX, or macOS server, web server, agent, or developer machine.

You can uninstall agents using any of the following accounts:

- root
- Any account with `sudo` privileges
- (UNIX or macOS only) Any non-root account without `sudo` privileges

Uninstalling ElectricFlow Using root or an Account with sudo Privileges

1. Log in as root or using an account with `sudo` privileges.
2. Run one of the following commands:
   - Linux: `/opt/electriccloud/electriccommander/uninstall`
   - UNIX or macOS: `/opt/electriccloud/electriccommander/uninstaller/uninstall`

   The following prompt appears:

   ```
   This will completely remove ElectricFlow from your system. Are you sure you want to do this? [y/N]
   ```

   If you did not invoke the uninstaller as root, the following prompt appears:

   1. Linux: `/Root Required: You must be root to run uninstall this application.`
   2. UNIX or macOS: `/opt/electriccloud/electriccommander/uninstaller/uninstall must be run as root or use --nonRoot switch for non-root user.`
3. Enter `y` to start the uninstallation.

   The following prompts appear:

   ```
   Uninstalling ElectricFlow...

   Uninstall complete.
   ```
4. If you will be reinstalling ElectricFlow, remove any remaining files from the `/opt/electriccloud/electriccommander` directory.

   The uninstaller does not remove the files in this directory that ElectricFlow created or modified after the initial installation. You should relocate the files if you need them for troubleshooting.

Uninstalling UNIX or macOS ElectricFlow Agents Using a Non-root Account

A non-root account is one that is not root and also does not have `sudo` privileges.

Important: Running the uninstaller without root or `sudo` privileges is not recommended. When run without root or `sudo` privileges, the installer cannot remove the files that provide automatic start for the agent services as well as other files as described below.

To use a non-root account to uninstall UNIX or macOS ElectricFlow agents:

1. Log in as the agent service user.

   This is the user that owns the installed agent files and runs the agent processes. This user was specified during installation with root or `sudo` privileges or is the user that performed the non-root, non-`sudo` installation. You can find this user by entering `ls -l <install_dir>`.

2. Enter `/opt/electriccloud/electriccommander/uninstall --nonRoot`.

   The following prompt appears:

   ```
   This will completely remove ElectricFlow from your system. Are you sure you want
   to do this? [y/N]
   ```

3. Enter `y` to start the uninstallation.

   If you lack sufficient privileges on the installation directory, the following error appears, and you must obtain sufficient privileges before continuing:

   ```
   Error: insufficient privileges to uninstall the ElectricFlow agent which is
   located in directory:
    /opt/electriccloud/electriccommander
   ```

   If you have sufficient privileges on the installation directory, the following prompts appear:

   ```
   Found /opt/electriccloud/electriccommander/MANIFEST, please wait while files are
   deleted

   Uninstall completed.  Files backed up to /opt/electriccloud/electriccommander

   Uninstall was performed in the non-root mode.
   Please manually remove the ElectricFlow services from system auto-start.
   ```

   The uninstaller displays the following errors if control files for the agent service exist in `/Library/LaunchDaemons/` on macOS or in `/etc/` on other platforms:

   ```
   rm: cannot remove '/etc/rc.d/init.d/ecmdrAgent': Permission denied
   rm: cannot remove '/etc/rc.d/rc2.d/S99ecmdrAgent': Permission denied
   rm: cannot remove '/etc/rc.d/rc2.d/S99ecmdrAgent': Permission denied
   ```

   If the uninstallation is successful, the uninstaller exits with the following prompts:

   ```
   Uninstall completed.  Files backed up to /opt/electriccloud/electriccommander

   Uninstall was performed in the non-root mode.
   Please manually remove the ElectricFlow services from system auto-start.
   ```

4. If you will be reinstalling ElectricFlow agents, remove any remaining files from the `/opt/electriccloud/electriccommander` directory.

   The uninstaller does not remove the files in this directory that ElectricFlow created or modified after the initial installation. You should relocate the files if you need them for troubleshooting.

5. If you will be reinstalling ElectricFlow agents, delete the control files for the agent service manually to avoid errors.

   A non-root uninstallation does not delete these files. The files are in `/Library/LaunchDaemons/` on macOS and in `/etc/` on other platforms.

Uninstalling ElectricFlow on Windows

Choose one of the following procedures to completely uninstall ElectricFlow from a Windows server, web server, agent, or developer machine.

Uninstalling on Windows 2008 or Windows 7

Use this procedure to completely uninstall ElectricFlow from a Windows 2008 or Windows 7 machine.

1. Go to **Control Panel** > **Uninstall a program**.
2. Select **ElectricCommander**.
3. Click **Uninstall**.

   The system displays an "uninstall complete" prompt when ElectricFlow is removed.
4. Check the `<install_location>\electriccloud\electriccommander` directory and `C:\ProgramData\electriccloud\electriccommander` directory for any files that might remain. The uninstaller does not remove files that have been created or modified by ElectricFlow after the initial installation is complete.
5. Remove the files if you will reinstall ElectricFlow. You might want to move the files to a new location if you need to retain the files for troubleshooting.

Uninstalling the ElectricFlow DevOps Insight Server on Linux

Use the following procedure to uninstall the ElectricFlow DevOps Insight server completely from a Linux machine.

1. Log in as root or using an account with sudo privileges.
2. Disable the DevOps Insight server.

   To do so, use one of the following methods:

   - Uncheck the **Enable DevOps Insight** checkbox in the **Administration** > **DevOps Insight Server** subtab in the Automation Platform UI and click **Save**.
   - Enter the following API command:

     ```
     ectool setDevOpsInsightServerConfiguration --enabled 0
     ```
3. Enter the following command:

   ```
   /opt/electriccloud/electriccommander/uninstall-reporting
   ```

   The following prompt appears:

   ```
   This will completely remove ElectricFlow DevOps Insight Server from your
   system.  Are you sure you want to do this? [n/Y]
   ```

4. Enter `y` to start the uninstallation.

   The following prompts appear when the software is uninstalled:

   ```
   Uninstalling ElectricFlow DevOps Insight Server...

   Uninstall complete.
   ```

Uninstalling the ElectricFlow DevOps Insight Server on Windows

Use the following procedure to completely uninstall the ElectricFlow DevOps Insight server from a Windows machine.

1. Disable the DevOps Insight server.

   To do so, use one of the following methods:

   - Uncheck the **Enable DevOps Insight** checkbox in the **Administration** > **DevOps Insight Server** subtab in the Automation Platform UI and click **Save**.
   - Enter the following API command:

     ```
     ectool setDevOpsInsightServerConfiguration --enabled 0
     ```

2. Go to **Control Panel > Uninstall a program**.
3. Select **ElectricFlow DevOps Insight Server**.
4. Click **Uninstall**.

   The system displays an `uninstall complete` prompt when the ElectricFlow DevOps Insight server is removed.

Chapter 11: Configuring Disaster Recovery and Recovering from a Disaster

This topic has the steps to configure a Disaster Recovery (DR) setup. It explains what to expect after a successful failover.

Disaster Recovery Environment Setup

As part of a DR environment setup, you need to set up a secondary ElectricFlow site. This site must include a complete setup of all ElectricFlow components including:

- Database (Oracle, SQL Server, or MySQL)
- ElectricFlow server
- Web server
- DevOps Insight Server
- Repository server
- Zookeeper
- Gateway agent
- Agent

ElectricFlow components that can be load balanced include the ElectricFlow server, web server, DevOps Insight server, repository server, and gateway agent. Go to Installing and Configuring a Load Balancer on page 4-7 for details.

Along with replicating the component setup, data stores also need to be replicated and made available for ElectricFlow to operate the following:

- Database (using the vendor-recommended replication)
- DevOps Insight server (by restoring snapshots of Elasticsearch indices)
- Repository Server Data Store (replication of shared file locations where artifacts are stored)
- Plugins (typically copied to a shared file location)
- Certificates signed by an ElectricFlow CA (Certificate Authority)

Configurations and Settings

Follow these guidelines on the primary and secondary sites:

- Both the primary and secondary sites should be running the same version of ElectricFlow.
- Under normal operation, it should not be possible to do active transactions directly on the secondary (or replicated) database. That is, the secondary database should be in replication mode.
- The ElectricFlow server in the secondary site must be configured and must point to the secondary (or replicated) database.

- These servers must not automatically restart after a reboot. That is, all ElectricFlow servers in the secondary site should be set to start in Manual mode. This prevents inadvertent write operations into the replicated database. Details on recommended steps for setting up a secondary site appear below.
  - ElectricFlow server
  - Web server
  - DevOps Insight server services (Elasticsearch service and Logstash service)
- We recommend using DNS Failover to minimize the downtime when moving from the primary site to the secondary site. This allows end users accessing the web servers to continue using the same URL. Agents that are running jobs can send their finish job notification to the secondary server, allowing the jobs to succeed.

To configure Disaster Recovery, follow these steps:

1. Add following lines to the `wrapper.conf` file for the server nodes:

   ```
   wrapper.java.additional.1600=DCOMMANDER_IGNORE_SERVER_MISMATCH=1

   wrapper.java.additional.1601=DCOMMANDER_PRESERVE_SESSIONS=1
   ```

   This avoids the following errors during the failover to the secondary site:

   ```
   20160815T22:10:52.406 | 10.0.2.206 | DEBUG | bootstrap
   |                                    | schemaMaintenance
   | OperationInvoker |           Exception: InvalidServer: The ZooKeeper/Exhibitor
   setting of the last cluster
   ('ip100179155:8080,ip100145251:8080,ip100133239:8080') to connect to the
   database is different from the ZooKeeper/Exhibitor setting of the current
   cluster
   ('ip1008088:8080,ip100196103:8080,ip100515:8080'). Check that the cluster
   is configured for the correct database. To allow this server cluster to become
   the new
   owner for this data, update the database configuration with the
   ignoreServerMismatch
   flag set.

   20160815T22:10:52.406 | 10.0.2.206 | WARN | bootstrap |
   |                                    |
   | ServerStatus                 | InvalidServer: The ZooKeeper/Exhibitor setting
   of
   the last cluster ('ip100179155:8080,ip100145251:8080,ip100133239:8080') to
   connect to the database is different from the ZooKeeper/Exhibitor setting of the
   current cluster ('ip1008088:8080,ip100196103:8080,ip100515:8080'). Check
   that the cluster is configured for the correct database. To allow this server
   cluster
   to become the new owner for this data, update the database configuration with
   the
   ignoreServerMismatch flag set.
   ```

2. Make sure that the following recommended standard setup steps are performed:

    1. `commander.properties` in ZooKeeper should have the `COMMANDER_SERVER_NAME` set to the FQDN (Fully Qualified Domain Name) of the load balancer of the ElectricFlow server cluster. This can be set using the following command:

        ```
        ecconfigure --serverName <FLOW_SERVER_LOAD_BALANCER_FQDN>
        ```

    2. Both the primary and secondary sites should have the same files (including content):

        - `keystoreFile`
        - `passkeyFile`
        - `commander.properties`

        In the cluster setup, these files are stored in their respective ZooKeeper instances (the primary and secondary instances).

        > **Note:** The ZooKeeper connection is configured using:
        > `ecconfigure --serverZooKeeperConnection <ZooKeeper_servers_comma_seperated_list>`
        >
        > For example:
        > `ecconfigure --serverZooKeeperConnection ip1008088:2181,ip100196103:2181,ip100515:2181`

    3. Run the following command on the web servers to ensure that the `COMMANDER_SERVER` property in the `httpd.conf` file is set to ElectricFlow server's load balancer FQDN:

        ```
        ecconfigure --webTargetHostName <FLOW_SERVER_LOAD_BALANCER_FQDN>
        ```

        > **Note:** The `httpd.conf` file is usually in `apache/conf` on a Linux machine and `ProgramData\Electric Cloud\ElectricCommander\apache\conf` on a Windows machine.

        The `--webTargetHostName` argument modifies the ElectricFlow web server configuration and therefore also attempts to restart the ElectricFlow web server. If you used the `ecconfigure` command without `sudo` as recommended, the `commanderApache` service will not start and produces an error. Therefore, you must restart it manually afterward using `sudo`. You can also use the `--skipServiceRestart` argument to avoid the `ecconfigure` command's restart attempt and the error message.

4. Similarly, on each repository server, run the following command to set `COMMANDER_HOST` in the `server.properties` file:

   `ecconfigure --repositoryTargetHostName <FLOW_SERVER_LOAD_BALANCER_FQDN>`

   > **Note:** These are the default locations for the `server.properties` file:
   > - In Linux, `/opt/electriccloud/electriccommander/conf/repository/server.properties`
   > - In Windows, `C:\ProgramData\Electric Cloud\ElectricCommander\conf\repository\server.properties`

5. Set the "Server IP address" in the ElectricFlow server property to the Flow Server Load Balancer FQDN (the same value as *<FLOW_SERVER_LOAD_BALANCER_FQDN>*).

6. Set the "Stomp Client URI" server property to `stomp+ssl://<FLOW_SERVER_LOAD_BALANCER_FQDN>:61613`.

   > **Note:** If the load balancer does SSL termination, uncheck the **Use SSL for Stomp** option. All the ElectricFlow server nodes must be restarted to affect this change, because by default, the check box is selected.

7. In the **Cloud** > **Resource** page, register all the ElectricFlow local agents (the ones that run on same machine as ElectricFlow server) from both primary and secondary sites. Make sure that for the cluster setup, you do not have *local* resources.

8. Create resource pools named "*local*" and "*default*". For both, add ElectricFlow local agents.

   Both the local and default pools are used by the ElectricFlow standard job processing. For example, sentry jobs run on local pool resources.

9. (Optional) If trusted agents are used, in addition to copying the keystore file, along with keystore file, the `conf/security` folder should be copied to the secondary site's ZooKeeper. This folder stores the ElectricFlow Certificate Authority information along with the certificates that are signed by ElectricFlow.

   Perform the following steps to copy this folder from the primary to secondary site:

   1. Log into the primary site's ElectricFlow server, and run the following command to get the `conf/security` folder from the primary ZooKeeper into the local `/tmp/electriccommander/conf/security` folder:

      - Linux:

        ```
        COMMANDER_ZK_CONNECTION=<ZooKeeper_Primary_Server_
        IP>:2181 <install_dir>/jre/bin/java -cp <install_
        dir>/server/bin/zk-config-tool-jar-with-dependencies.jar
        com.electriccloud.commander.zkconfig.ZKConfigTool --
        readFolder /commander/conf/security
        /tmp/electriccommander/conf/security
        ```

      - Windows:

        ```
        "C:\Program Files\Electric
        Cloud\ElectricCommander\jre\bin\java.exe" -DCOMMANDER_ZK_
        CONNECTION=<ZooKeeper_Primary_Server_IP>:2181 -jar
        "C:\Program Files\Electric
        Cloud\ElectricCommander\server\bin\zk-config-tool-jar-
        with-dependencies.jar"
        com.electriccloud.commander.cluster.ZKConfigTool --
        readFolder /commander/conf/security
        c:\<path>\electriccommander\conf\security
        ```

2. Log into the secondary site's ElectricFlow server, and run the following command to upload the `conf/security` folder from the local folder to ZooKeeper:

   - Linux:

     ```
     COMMANDER_ZK_CONNECTION=<ZooKeeper_Secondary_Server_IP>:2181 <install_dir>/jre/bin/java -cp <install_dir>/server/bin/zk-config-tool-jar-with-dependencies.jar com.electriccloud.commander.zkconfig.ZKConfigTool --writeFolder /commander/conf/security /tmp/electriccommander/conf/security
     ```

   - Windows:

     ```
     "C:\Program Files\Electric Cloud\ElectricCommander\jre\bin\java.exe" -DCOMMANDER_ZK_CONNECTION=<ZooKeeper_Secondary_Server_IP>:2181 -jar "C:\Program Files\Electric Cloud\ElectricCommander\server\bin\zk-config-tool-jar-with-dependencies.jar" com.electriccloud.commander.cluster.ZKConfigTool --writeFolder /commander/conf/security c:\<path>\electriccommander\conf\security
     ```

3. Set the repository data store on the primary and secondary sites.

   Repository servers set up on primary and secondary sites can share the same repository data store.

   The repository data can be replicated, and the repository servers can point to the respective data store locations. For each repository server, set the `REPOSITORY_BACKING_STORE` in the `server.properties` file to a UNC path on a network share on the file server.

   For example:

   ```
   REPOSITORY_BACKING_STORE=//10.0.109.72/repo_data/repositorydata
   ```

4. Register the repository server in the ElectricFlow UI. If the repository server cluster is set up, use the load balancer URL (for example, `https://<REPOSITORY_SERVER_LOAD_BALANCER_FQDN>:8200`).

   During the failover to the secondary site, FQDN should point to the repository servers in the secondary site.

5. It is recommended that the plugins folder in the network share must be accessible from the remote web servers as mentioned in Universal Access to the Plugins Directory on page 5-21.

6. For the initial installation and setup of the secondary site, perform the following recommended steps:
    1. Set up the secondary database in normal or nonreplicated mode.
    2. Follow instructions as described in this installation guide to set up all the servers, including the ElectricFlow, web, repository, and Zookeeper servers.
    3. Make sure that the `database.properties` file is set up to point to the correct secondary site's database server. This file will be stored in ZooKeeper when the cluster setup is used. For the primary site, it should point to the primary database. For the secondary site, it should point to the secondary database.
    4. Ensure that all the servers are running properly, including the connection to the database. At this time, the secondary site is not set up for replication, and operates as a separate installation of ElectricFlow.
    5. Before setting up the secondary site's database in replication mode, shut down all secondary ElectricFlow servers, web servers, repository servers, and others. After these steps are performed, there should not be any write transactions to the secondary database.
    6. The first time that the secondary database is set up, a schema for database tables is created. Before proceeding to set this database with replication, this schema may have to be deleted. This avoids a schema name conflict when replication is enabled.
    7. Based on the disaster recovery option chosen for the database, set up the secondary database in replication mode.

Disaster Recovery Environment Setup for DevOps Insight Server

Follow the steps below to include the DevOps Insight server in your disaster recovery environment setup.

1. Setup identical DevOps Insight server installations on the primary site and the secondary sites.
2. Ensure that the ElectricFlow server on the primary site is configured to point to the FQDN (Fully Qualified Domain Name) of the load balancer of the DevOps Insight server cluster in the primary site.
   Similarly, the ElectricFlow server on the secondary site must be configured to point to the FQDN (Fully Qualified Domain Name) of the load balancer of the DevOps Insight server cluster in the secondary site.
3. Use snapshots to create backups of indices from the primary site at regular intervals, for example, daily. The backups can be stored in a shared file system or on AWS S3 storage. See the section Maintaining DevOps Insight Server Data on page 12-11 for details on creating snapshots.
4. Restore the snapshots to the DevOps Insight server cluster running on the secondary site at regular intervals, for example, daily or weekly. You Sneed to start the Elasticsearch service for the DevOps Insight server on the secondary site for restoring the snapshots. You may choose to move the snapshot files to a different location for backup and archiving purposes once they have been restored on the secondary site.

Steps to Perform During a Disaster Recovery Failover

When a disaster event happens that interrupts the operations on the primary site, follow these steps to move the operations to the secondary site.

1. Shut down any services that might still be running on the primary site. With the exception of the DevOps Insight server, all other components including the database, ElectricFlow server, and repository server can be shut down. Doing this ensures that no more transactions happen on the primary site.
   For the DevOps Insight server see Disaster Recovery Failover Steps for a DevOps Insight Server on page 11-8 below.
2. Begin switching operations to the secondary site by restoring and updating the secondary site's database with the latest data. The steps to do this may vary based on the disaster recovery method used for database.
3. Delete the `brokerdata` folder on the ElectricFlow server nodes. For example, in Windows, delete the `C:\ProgramData\Electric Cloud\ElectricCommander\brokerdata` folder.
4. Follow the DNS failover procedure, and update the DNS entries to point to servers in the secondary site. This includes updating the entries for servers including web server, ElectricFlow server, and repository server. There may also be other servers based on configuration.
5. Bring up all the servers, with the exception of the DevOps Insight server, and infrastructure in the secondary site. Start the ElectricFlow services on different machines, including the ElectricFlow server, web server, repository server, and gateway agents. For DevOps Insight server see Disaster Recovery Failover Steps for a DevOps Insight Server on page 11-8 below.
6. Based on the nature of the disaster event, certain active operations running on the primary site may be interrupted, and need to be restarted. For example, a build or deploy application process may fail and error out. Use the ElectricFlow UI to review such failures, and take the appropriate corrective actions, usually by executing those failed processes again.
7. The secondary's site database now acts as the master. We recommend that you set up a database that will act as a slave.

Disaster Recovery Failover Steps for a DevOps Insight Server

1. If the Elasticsearch service for the DevOps Insight server is still running on the primary site, then create a final snapshot before shutting down the service.
2. Restore any snapshots created since the last scheduled restoration on the secondary site.

The secondary site's DevOps Insight server now acts as the primary cluster. We recommend that you setup a schedule for creating snapshots from this cluster and restoring into another cluster.

Server Maintenance

See Maintaining ElectricFlow on page 12-1 for details on ElectricFlow server maintenance.

Chapter 12: Maintaining ElectricFlow

This section contains common maintenance procedures.

- Switching from an Alternate Database to the Built-In Database on page 12-11

ElectricFlow Server Backups

You should back up your existing ElectricFlow data frequently. We recommend full regular (nightly) database backups and database backups before an upgrade.

Data Backup Methods

There are two ways to back up your data. You can use a database-specific backup tool to create a database dump, or you can use the ElectricFlow (`ectool export`) tool to create a complete XML database backup. This section describes the differences between the two types of backups.

> **Important:** We recommend that you do not use the `ectool export` tool with jobs on an active system to create a trusted database backup.

Database Dumps

You must use a database-specific backup tool to create a database dump. Database dumps have the following characteristics:

- The backup process takes much less time to complete than full XML exports
- Database dumps (for example MySQL) must be performed while the database is live, up, and running.
- You can quickly restore a database from a database dump.

> **Note:** A database dump can only be restored to the same type of database. If you are planning to switch your database type when you restore from the backup, you must create an XML backup.

Complete XML Database Backup

You must use the `ectool export` tool to create a complete XML database backup. Complete XML database backups have the following characteristics:

- The tool must be used while the ElectricFlow server is running.
- The database backup process can take considerably longer than simply creating a database dump, but this method is necessary in the following situations:
  - Backing up the database is not an available option.
  - You need to migrate from one type of database to another. For example, MySQL to Oracle.
  - You want a full export in a text form you can search with an editor.

> **Note:** It might not be feasible to run a full XML export regularly (such as nightly). So if jobs are running, in order to speed up the full export (and to help prevent issues with importing the data later), you should use the `--excludeJobs` option. For more information about the `ectool export` command, see the ElectricFlow API Guide at http://docs.electric-cloud.com/eflow_doc/FlowIndex.html.

Preparing for a Backup

Before you backup your ElectricFlow server:

- Make sure you have plenty of free space available because full database dumps and XML export files can be extremely large.
  - Compress database dumps if they are not compressed by default.
- Regularly perform maintenance such as, but not limited to:
  - Pruning job workspaces
  - Deleting or compressing ElectricFlow log files
  - Deleting unused projects and/or procedures

> **Note:** You can use ElectricFlow to perform backups by creating a procedure that runs the database dump or export command.

Backing Up an ElectricFlow Server

Use the following procedure to back up your ElectricFlow server data. Review Preparing for a Backup on page 12-2 before performing this procedure.

1. Choose one of the following methods to back up your data:
   - Use a database-specific backup tool to create a database dump.
   - Use `ectool export` to create a complete XML database backup.

   > **Note:** For more information about database backup methods, see Data Backup Methods on page 12-1.

2. Save the `passkey` file. The full path name of this file is `/opt/electriccloud/electriccommander/conf/passkey` in Linux or `C:\ProgramData\Electric Cloud\ElectricCommander\conf\passkey` in Windows . When you restore your server, this passkey must be in place so ElectricFlow can decrypt passwords for user impersonation, LDAP, and the database connection.
3. Back up the plugins directory.
   - The plugins directory is stored in a server setting property (`/server/settings/pluginsDirectory`).
   - If the property does not exist, the server uses the default location, which is the `plugins` subdirectory in the data directory.

4. Back up the files containing your custom configurations and settings to ensure all important settings are saved.
    - The default location for ElectricFlow server and agent configuration files is the `conf` subdirectory in the data directory.
    - The default location for the Apache web server configuration files is the `apache/conf` subdirectory in the data directory.
5. Verify that your backup contains the following items:
    - Database dump and/or XML export
    - The `passkey` file
    - The contents of the plugins directory
    - Configuration files
    - Keystores

ElectricFlow Server Restores

This section describes common restore related procedures for recovering ElectricFlow data.

Preparing for a Restore

Before you attempt to restore an ElectricFlow server:

- You *must* have a backup of your source ElectricFlow server.
    - If you are restoring your data to the exact same database or the same database type (for example, from one MySQL database to another MySQL database on a different system), a database backup is sufficient.
    - If you are switching to a different database type, you will need an XML export.

    > **Note:** Any activity on the source server *after the backup was created* will not exist on the destination server.

- The destination system must have an ElectricFlow server already installed and running, and this server must be running the same version or newer version than the source server.

Restoring Your ElectricFlow Server

The following section contains various procedures for restoring ElectricFlow data. Review Preparing for a Restore on page 12-3 section before performing any of the following procedures.

> **Note:** All ectool commands used in the following scenarios are fully documented in the ElectricFlow online help system. See the "Using ectool and the ElectricFlow API" help topic.

Restore the Same ElectricFlow Server and Database

Use the following procedure to restore your ElectricFlow server because of a catastrophic failure or unsuccessful upgrade.

1. Obtain a backup of the source system.
2. Stop the destination ElectricFlow server. For more information, see Starting and Stopping Servers and Agents Manually on page 12-18 for platform-specific commands.
3. If you are using a database dump (where the source and destination systems must both be using the same type of database), load the backup into the destination database.

   This will be done with a command specific to the database you are using.
4. Start the destination ElectricFlow server.
5. If you are using an XML export file, use the `ectool import` command to import the data into the destination ElectricFlow server.
6. Use the `ectool shutdownServer --restart 1` command to restart the destination server.

Keep the Same ElectricFlow Server but Switch the Database

Use the following procedure to restore your ElectricFlow server if you are doing one of these tasks:

- Switching from the built-in database installation to an external database.
- Upgrading to a higher performance system for the database.

1. Obtain a backup of the source system.
2. Stop the destination ElectricFlow server. For more information, see Starting and Stopping Servers and Agents Manually on page 12-18 for platform-specific commands.
3. Stop and disable the original database.
4. If you are using a database backup (where the source and destination systems must both be using the same type of database), load the database dump into the destination database.

   This will be done with a command specific to the database you are using.
5. Start the destination ElectricFlow server.
6. Set the server database configuration to point to the new database. Point to the new database one of these ways:

   See the "Database Configuration" help topic in the ElectricFlow web interface.

   Use the `ectool setDatabaseConfiguration`command.
7. If you are using an XML export file, use the `ectool import` command to import the data into the destination ElectricFlow server.
8. Use the `ectool shutdownServer --restart 1` command to restart the destination server.

Switch the ElectricFlow Server but Keep the Same Database

Before switching the server, be aware of the following:

- All files and directories copied to the Destination ElectricFlow Server should be owned by the user configured to run the ElectricFlow server daemon.

- Make sure that the host name of local agent is set to *127.0.0.1* using **Cloud** > **Resources** > **Local**> **Resource Details**.
- When you install ElectricFlow without a built-in database, you can configure the database only by using ectool.

Use the following procedure to restore ElectricFlow if you are upgrading to a higher performance ElectricFlow server system.

- Make sure you have a backup of the source system.
- Check the IP Address System property by selecting **Administration** > **Server** > **Settings** on the old (source) ElectricFlow system.

  This field is empty by default to enable dynamic connections between the ElectricFlow server and agents.

  If the field is not empty, you must enter the IP address for the Destination ElectricFlow Server.
- Stop the destination ElectricFlow server.

  For more information, see Starting and Stopping Servers and Agents Manually on page 12-18 for platform-specific commands.
- Stop and disable the source ElectricFlow server.
- Copy the `passkey` and `keystore` files from the source ElectricFlow backup to the destination system. These files are in `/opt/electriccloud/electriccommander/conf/` in Linux and in `C:\ProgramData\Electric Cloud\ElectricCommander\conf\` in Windows.
- Copy the backed-up plugins to the destination system.

  You may encounter one of these scenarios:

  - If the `/server/settings/pluginsDirectory` property does not exist, the server uses the default location (the `plugins` subdirectory in the data directory).

    Copy the backed-up plugins to that directory on the destination system.
  - The plugins are stored in a local directory valid on both systems.

    Copy the backed-up plugins to the same directory on the destination system.
  - The plugins are stored in a shared directory valid on both systems.

    You do not need to do anything.
  - The plugins are stored in a directory not accessible on the destination system.

    This can happen

    - If the source and destination systems have different operating systems (such as Windows to Linux).
    - If the plugins directory on the source system is on a drive that does not exist on the destination system.

    Copy the backed-up plugins to a new directory accessible to the destination system. When the server starts, set the `/server/settings/pluginsDirectory` property to the new directory and restart the ElectricFlow server.

- If you use a MySQL database, do these steps on destination system:
  - Install the MySQL JDBC driver. For details, see Installing the MySQL JDBC Driver on page 3-181.
  - Configure access to the ElectricFlow database user from the IP address or FQDN on the destination system.
- Start the destination ElectricFlow server.
- Because the ElectricFlow host changed, connect the ElectricFlow database to the new host:

  On the command-line:

  1. Use `ectool setDatabaseConfiguration` to specify a database configuration and set the `--ignoreServerMismatch` option.
  2. Use the following command to restart the destination server: `ectool shutdownServer --restart 1`.

  In the web interface, you should automatically be redirected to the **Database Configuration** page.

  1. Enter the appropriate database configuration.
  2. Select the **Ignore server hostname mismatch** check box.
  3. Select **Same instance on a new host**.
  4. Click **Save and Restart**.
- If you copied the plugins directory to a directory that does not match the plugins directory on the source system:

  1. Set the `/server/settings/pluginsDirectory` property to this new directory.

     You can use the `ectool setProperty` command to set this value.
  2. Restart the ElectricFlow server.

Switch Both the ElectricFlow Server and Database

Use the following procedure to restore ElectricFlow if you are upgrading to higher performance systems for both the ElectricFlow server and the database.

1. Make sure you have a backup of the source system.
2. Check the IP Address System property by selecting **Administration > Server > Settings** on the old ElectricFlow system.

   This field is empty by default to enable dynamic connections between the ElectricFlow server and agents.

   If the field is not empty, you must enter the IP address for the new ElectricFlow server.
3. Stop the destination ElectricFlow server.

   For more information, see Starting and Stopping Servers and Agents Manually on page 12-18 for platform-specific commands.
4. Stop and disable the source ElectricFlow server.
5. Stop and disable the original database.

6. Copy the `passkey` file from the backup to the destination system. The full path name of this file is `/opt/electriccloud/electriccommander/conf/passkey` in Linux or `C:\ProgramData\Electric Cloud\ElectricCommander\conf\passkey` in Windows.
7. Copy the backed-up plugins to the destination system.

   You may encounter one of these scenarios:

   - If the `/server/settings/pluginsDirectory` property does not exist, the server uses the default location (the `plugins` subdirectory in the data directory).

     Copy the backed-up plugins to that directory on the destination system.
   - The plugins are stored in a local directory valid on both systems.

     Copy the backed-up plugins to the same directory on the destination system.
   - The plugins are stored in a shared directory valid on both systems.

     You do not need to do anything.
   - The plugins are stored in a directory not accessible on the destination system.

     This can happen:

     - If the source and destination systems have different operating systems (such as Windows to Linux).
     - If the plugins directory on the source system is on a drive that does not exist on the destination system.

     Copy the backed-up plugins to a new directory accessible to the destination system. When the server starts, set the `/server/settings/pluginsDirectory` property to the new directory and restart the ElectricFlow server.
8. If you are using a database backup (where the source and destination systems must both be using the same type of database), load the database dump into the destination database.

   This operation is completed with a command specific to the database you are using.
9. Start the destination ElectricFlow server.
10. Because we have replaced the `passkey`, the database password is no longer valid. You need to reset the database password (default: `commander`) and ignore the `passkey` mismatch either from the command-line or the web interface.

    - On the command-line, use `ectool setDatabaseConfiguration` to specify the password and set the `--ignoreServerMismatch` and `--ignorePasskeyMismatch` options.
    - In the web interface, you should automatically be redirected to the **Database Configuration** page. Enter the database password and select the **ignore invalid passkey** check box.
11. If you are using an XML export file, use the `ectool import` command to import the data into the destination ElectricFlow server.
12. Use the `ectool shutdownServer --restart 1` to restart the destination server.

13. If you copied the plugins directory to a directory that does not match the plugins directory on the source system, set the `/server/settings/pluginsDirectory` property to this new directory and restart the ElectricFlow server.

    You can use the `ectool setProperty` command to set this value.

Create a Clone of the ElectricFlow Server and the Database

Use the following procedure to restore your ElectricFlow server if you are setting up a production-like environment for testing.

1. Make sure you have a backup of the source system.
2. Check the IP Address System property by selecting **Administration > Server > Settings** on the old ElectricFlow system.

   This field is empty by default to enable dynamic connections between the ElectricFlow server and agents.

   If the field is not empty, you must enter the IP address for the new ElectricFlow server.
3. Stop the destination ElectricFlow server.

   For more information, see Starting and Stopping Servers and Agents Manually on page 12-18 for platform-specific commands.
4. Copy the `passkey` file from the backup to the destination system. The full path name of this file is `/opt/electriccloud/electriccommander/conf/passkey` in Linux or `C:\ProgramData\Electric Cloud\ElectricCommander\conf\passkey` in Windows.
5. Copy the backed-up plugins to the destination system.

   You may encounter one of these scenarios:

   - If the `/server/settings/pluginsDirectory` property does not exist, the server uses the default location (the `plugins` subdirectory in the data directory).

     Copy the backed-up plugins to that directory on the destination system.
   - The plugins are stored in a local directory valid on both systems.

     Copy the backed-up plugins to the same directory on the destination system.
   - The plugins are stored in a shared directory valid on both systems.

     You do not need to do anything.
   - The plugins are stored in a directory not accessible on the destination system.

     This can happen:

     - If the source and destination systems have different operating systems (such as Windows to Linux).
     - If the plugins directory on the source system is on a drive that does not exist on the destination system.

     Copy the backed-up plugins to a new directory accessible to the destination system. When the server starts, set the `/server/settings/pluginsDirectory` property to the new directory and restart the ElectricFlow server.

6. If you are using a database backup (the source and destination systems must be using the same type of database), create the destination database, give the appropriate database user permissions to the schema (as mentioned in External Database Configuration on page 5-2), and load the database dump into the destination database.

   This operation is completed with a command specific to the database you are using.

7. If you are using a database backup, disable schedules, resources, or both on both servers.

   - Two servers should never *talk* to the same agent. The two servers share the same identity because they share exact copies of the database.
   - Disabling schedules prevents jobs from launching unexpectedly.
   - Disabling resources prevents scheduled or manually launched jobs from running on production agents. This operation is completed with a command specific to the database you are using.

8. Start the destination ElectricFlow server.

9. Because we have replaced the `passkey`, the database password is no longer valid. You need to reset the database password (default: `commander`) and ignore the `passkey` mismatch either from the command-line or the web interface.

   - On the command-line, use `ectool setDatabaseConfiguration` to specify the password and set the `--ignoreServerMismatch` and `--ignorePasskeyMismatch` options.
   - In the web interface, you should automatically be redirected to the **Database Configuration** page. Enter the database password and select the **ignore invalid passkey** check box.

11. If you are using an XML export file, disable schedules, resources, or both on both servers.

    - Two servers should never "talk" to the same agent. The two servers share the same identity because they share exact copies of the database.
    - Disabling schedules prevents jobs from being launched unexpectedly.
    - Disabling resources prevents scheduled or manually launched from running on production agents.

    Disable the schedules and resources one of these ways:

    - Modify the import file by replacing `<resourceDisabled>0</resourceDisabled>` with `<resourceDisabled>1</resourceDisabled>`.
    - Use the `ectool import` command with the `--disableSchedules` flag turned on to disable schedules.

12. Use the `ectool shutdownServer --restart 1` command to restart the destination server.

13. If you copied the plugins directory to a directory that does not match the plugins directory from the source system,set the `/server/settings/pluginsDirectory` property to the new directory and restart the ElectricFlow server.

    You can use the `ectool setProperty` command to set this value.

Switching to an Alternate Database from the Built-In Database

If you did not deselect the "database" check box during installation, you can switch to another database at any time. You can use this procedure to switch from the built-in database or to switch from the current alternate database to a different alternate database.

> **Note:** If you are using two different ElectricFlow servers in a non-HA configuration, they cannot point to the same database.

The export operation is run by the server process, not through ectool. The command is not run by the agent, but by the server itself. Therefore, it has some impact if the server agent service user and the server service user are different. For example, the following commands in the same step are executed by two different users:

```
mkdir ("/path/foo");
$ec->export("/path/foo/project.xml",
{path=>"/projects/MYPROJ"}
);
```

The /path/foo directory creation is executed by the agent service, which means that the agent user needs permission to create the directory. The export is executed by the ElectricFlow service user.

Use these procedures to configure a new database and migrate the existing data.

Preventing Database Changes During the Export

Before you perform an export, ensure that the ElectricFlow server is inactive (meaning that it cannot accept jobs) by completing the following steps on the server:

1. Disable ElectricSentries.
2. Disable project schedules.
3. Check that no jobs are running on any resources.
4. Disable all resources so that no new job steps can run.

This ensures a complete XML file by preventing changes to the ElectricFlow database during the export.

Exporting and Importing Your Data

1. Export your data by entering the following command:

   ```
   ectool export <filename> --compress 1
   ```

2. Set the database configuration using the web interface or ectool. For more information, see Configuring ElectricFlow to Use an Alternate Database on page 5-4.
3. Restart the ElectricFlow server by entering the following command:

   ```
   ectool shutdownServer --restart 1
   ```

4. Import your data by entering the following command:

   ```
   ectool import <filename> --force 1
   ```

Switching from an Alternate Database to the Built-In Database

You can switch to the built-in (default) ElectricFlow database at any time. The following procedure shows how to switch to the built-in database from any ElectricFlow-supported alternate database.

Using the built-in database is possible only if during the installation it was activated via the `--installDatabase` installer parameter or by the corresponding options in the GUI installer.

> **Note:** If you are using two different ElectricFlow servers in a non-HA configuration, they cannot point to the same database.

1. If the built-in database service is disabled, enable and start it.

   To do so, enter the following ecconfigure command:

   ```
   ecconfigure --databaseEnableService=1
   ```

2. Point the ElectricFlow server to use the built-in database.

   You can use the UI or the ectool `setDatabaseConfiguration` command. If you did *not* change the default values for the password (`changeme`) and port (`8900`), enter

   ```
   ectool setDatabaseConfiguration --databaseType builtin --hostName localhost --
   userName root --databaseName eflow
   ```

   If you *have* changed the default values (either during or after installation) for the password and port, enter

   ```
   ectool setDatabaseConfiguration --databaseType builtin --hostName localhost --
   userName root --password <password> --port <port_number> --databaseName eflow
   ```

Maintaining DevOps Insight Server Data

The DevOps Insight server uses the Elasticsearch search engine and the Logstash data-collection and log-parsing engine to gather data from the ElectricFlow server for use in the Deployments, Releases, and Release Command Center dashboards. The DevOps Insight server also receives predictive analytics data (based on raw Elasticsearch data) from the DevOps Foresight server. For information about the DevOps Foresight server (packaged and licensed separately), see the *DevOps Foresight Installation and User Guide* at http://documentation.electric-cloud.com/html/eflow_doc/FlowIndex.html.

Backing Up DevOps Insight Server Elasticsearch Data

You should back up your existing DevOps Insight server data frequently. We recommend full regular (nightly) backups and a backup before an upgrade. For further details on archiving and restoring Elasticsearch indices, see the Elasticsearch 5.3 documentation at https://www.elastic.co/guide/en/elasticsearch/reference/5.3/modules-snapshots.html.

You should consider the following points for the DevOps Insight server when you set up the Elasticsearch snapshot repository:

- When you register the location of the shared file system repository in the `path.repo` setting in the `elasticsearch.yml` file, you must specify the setting in the `Custom Settings` section to

ensure that it is preserved during upgrades.

Following is an example for Linux platforms:

```
path.repo: ["/home/ecloud/bb", "/mount/backups", "/mount/longterm_backups"]
```

Following is an example for a remote shared folder location on Windows platforms using a Windows UNC path:

```
path.repo: ["\\\\<MY_SERVER>\\Snapshots"]
```

- Because the DevOps Insight server is configured with SSL authentication, the `curl` command format must be as follows:

  ```
  curl -k -X <POST|PUT> -E <data_
  dir>/conf/reporting/elasticsearch/admin.crtfull.pem --key <data_
  dir>/conf/reporting/elasticsearch/admin.key.pem https://<DevOps_Insight server_
  host_name>:<Elasticsearch port>/<request_URI>
  ```

  For example:

  ```
  curl -k -X POST -E /opt/ef/conf/reporting/elasticsearch/admin.crtfull.pem --key
  /opt/ef/conf/reporting/elasticsearch/admin.key.pem
  https://localhost:Elasticsearch_port/_snapshot/my_backup/snapshot_1/_restore
  ```

- The Elasticsearch indices created by ElectricFlow through the DevOps Insight server begin with `ef-` so they can be selected using the `ef-*` index pattern.
- Most Elasticsearch indices follow a time-based index naming scheme and use `-yyyy` as the suffix for the index name, where `yyyy` is the year associated with the document.

  For example, all deployments for the year 2018 will be stored in the index named `ef-deployment-2018`. This time-based naming scheme can be used in your archiving strategy for the DevOps Insight server.

Removing Old DevOps Insight Elasticsearch Data

DevOps Insight provides insight and visibility into not just your ongoing releases and deployments, but also historic releases. So you must retain old data in the DevOps Insight server.

You can provide sufficient disk space for the DevOps Insight server based on its the usage requirements in Disk Usage on page 2-12. However, if you must remove very old data from the DevOps Insight server to reclaim disk space, follow the recommendations explained below.

Ensuring Sufficient Disk Space for Storing DevOps Insight Data

Make sure that enough disk space is provided for storing DevOps Insight data for the last *n* years based on your data retention requirements. For details about calculating disk usage requirements for the DevOps Insight server based on your data-generation patterns, see Disk Usage on page 2-12.

Removing the Old Data

Elasticsearch is the underlying analytics store for the DevOps Insight server. The DevOps Insight server data is stored as indices in Elasticsearch. If you must remove old data, you should use Elasticsearch Curator to delete old indices. For more information about Elasticsearch Curator, see https://www.elastic.co/guide/en/elasticsearch/client/curator/5.3/index.html.

1. Install Elasticsearch Curator on the system where the DevOps Insight server is installed.

   The curator CLIs `curator_cli` and `curator` use a configuration file that contains Elasticsearch connection settings.

   Following is a sample YAML configuration file that you can use for connecting to an Elasticsearch cluster or instance that is backing the DevOps Insight server:

   ```
   client:
     hosts:
       - 127.0.0.1
     port: Elasticsearch_port
     use_ssl: True
     certificate: data_dir/conf/reporting/elasticsearch/chain-ca.pem
     client_cert: data_dir/conf/reporting/elasticsearch/admin.crtfull.pem
     client_key: data_dir/conf/reporting/elasticsearch/admin.key.pem
     ssl_no_validate: False
     http_auth:
     timeout: 30
     master_only: False
   ```

   where *`Elasticsearch port`* is the Elasticsearch port number and *`data_dir`* is the DevOps Insight server data directory path.

2. Run the following command to verify that you can connect to Elasticsearch using the configuration file:

   ```
   curator_cli --config curator-config.yml show_indices
   ```

   The Elasticsearch indices created by ElectricFlow begin with `ef-`. Most of the ElectricFlow indices follow a time-based index naming scheme and use `-yyyy` as the suffix for the index name, where *yyyy* is the year associated with the record. For example, all deployments for the year 2018 are stored in the index named `ef-deployment-2018`.

   Following is a sample YAML action file to delete ElectricFlow indices older than seven years. You can increase the number of years for which to retain the old indices based on your data retention policies.

   ```
   actions:
     1:
       action: delete_indices
       description: >-
         Delete ElectricFlow DevOps Insight indices older than 7 years
       options:
         ignore_empty_list: True
         timeout_override:
         continue_if_exception: False
         disable_action: False
       filters:
       - filtertype: pattern
         kind: prefix
         value: ef-
       - filtertype: period
         period_type: relative
         source: name
         range_from: -8
         range_to: -7
         timestring: '-%Y'
         unit: years
   ```

3. Run the following command to do a dry run using the configuration file and the action file:

   ```
   curator --config curator-config.yml --dry-run curator-action.yml
   ```

   This shows you the indices that will be deleted but will not actually delete them.

4. Verify the dry run output.

5. Schedule the following curator command to run periodically to delete the old indices based on your YAML action file by entering:

   ```
   curator --config curator-config.yml curator-action.yml
   ```

Removing Incorrect DevOps Insight Elasticsearch Data

If incorrect data is loaded into DevOps Insight server, for example, during building or testing of a script meant to send reporting data to the DevOps Insight server, you can delete this data using these steps:

1. Identify the Elasticsearch index from which incorrect data needs to be deleted.

   DevOps Insight server indices are named using the pattern `ef-report-object-name-yyyy`. So assuming that you used the `sendReportingData` API to send the data to the DevOps Insight server, and the report object name was `test`, then the corresponding index name would be `ef-test-2019`.

2. Back up the index before deleting any data in case something goes wrong and you need to restore the data.

   a. Log in to the system running the DevOps Insight server.

   b. Open a terminal window and change directories to the DevOps Insight server `conf/` directory.

      On Linux, the default path is

      ```
      /opt/electriccloud/electriccommander/conf/reporting
      ```

   c. Run the following commands:

      ```
      # Create backup index
      curl -vk -XPUT 'https://127.0.0.1:Elasticsearch_port/backup-test' -E
      elasticsearch/admin.crtfull.pem --key elasticsearch/admin.key.pem

      # Copy the data from the original index to the backup index
      curl -XPOST 'https://127.0.0.1:Elasticsearch_port/_reindex?pretty' -E
      elasticsearch/admin.crtfull.pem --key elasticsearch/admin.key.pem -H
      'Content-Type: application/json' -d'
      {
        "source": {
          "index": "ef-test-2019"
        },
        "dest": {
          "index": "backup-test"
        }
      }'
      ```

3. Use the Elasticsearch `_delete_by_query to` API delete the data from the original index based on criteria that uniquely identify the data to be deleted.

   For example, if the data with a field named `projectName` and value of `motorbike` needs to be deleted, the following command deletes documents matching the criteria in the index `ef-test-2019`:

```
curl -vk -XPOST "https://127.0.0.1:Elasticsearch_port/ef-test-2019/_delete_by_
query?pretty"
-H 'Content-Type: application/json' -E elasticsearch/admin.crtfull.pem --key
elasticsearch/admin.key.pem -d'

{

  "query": {

    "term": {

      "projectName": "motorbike-backend"

    }

  }

}

'
```

Apache Web Server or Agent Certificates

By default, ElectricFlow generates a temporary self-signed certificate during web server installation. This certificate is used whenever a browser makes an HTTPS connection to the Apache server. Because the certificate is self-signed, browsers will generate untrusted certificate prompts. To prevent these types of warnings, you must generate a new Apache web server or agent certificate signed by a recognized certificate authority (CA).

> **Important:** Before performing any of the following procedures, back up the `$DATA_DIRECTORY/-conf` and `$DATA_DIRECTORY/apache/conf` directories.

Generating a CA Request

Use the following procedure to generate a CA request.

1. Locate the `DATA_DIRECTORY` directory for your platform. The default directory locations are:
   - Linux – `/opt/electriccloud/electriccommander`
   - Windows 2008 or Windows 7 – `C:\ProgramData\Electric Cloud\ElectricCommander`
2. Locate the appropriate certificate signing request file generated during installation:
   - Agent – `$DATA_DIRECTORY/conf/agent.csr`
   - Web Server – `$DATA_DIRECTORY/apache/conf/server.csr`

3. (Optional) Update `server.csr` with custom SSL configuration data.

   - Edit the file `$DATA_DIRECTORY/apache/conf/serverssl.cnf` to add your custom configuration data.
   - Then, if you are on Linux:

     From `<DATA_DIRECTORY>/apache/conf`, enter:

     ```
     OPENSSL_CONF="<DATA_DIRECTORY>/apache/conf/serverssl.cnf" openssl req -new
     -key server.key -out server.csr
     ```

     For example:

     ```
     OPENSSL_
     CONF="/opt/electriccloud/electriccommander/apache/conf/serverssl.cnf"
     openssl req -new -key server.key -out server.csr
     ```

   - Or, if you are on Windows:

     Set the value of the environment variable `OPENSSL_CONF` to the full path to the file `serverssl.cnf`.

     Then, from `<DATA_DIRECTORY>/apache/conf`, enter:

     ```
     set "OPENSSL_CONF=<DATA_DIRECTORY>\apache\conf\serverssl.cnf"
     ```

     For example:

     ```
     set "OPENSSL_CONF=c:\ProgramData\Electric
     Cloud\ElectricCommander\apache\conf\serverssl.cnf"
     ```

     Finally, generate a certificate signing request by entering:

     ```
     openssl req -new -key server.key -out server.csr
     ```

Sending the CA Request

Send the `server.csr` (or `agent.csr`) file to a certificate authority to sign the certificate. The CA verifies the information inside and sends you a signed certificate in response. The signed certificate includes the original certificate and the CA signature.

Installing the Signed Certificate

Installing a New Certificate

To install a signed certificate:

1. Replace the existing certificate in the `DATA_DIRECTORY` directory with the new signed certificate you received from the CA. The signed certificate file should be placed in one of the following locations:
   - Agent – `$DATA_DIRECTORY/conf/agent.crt`
   - Web Server – `$DATA_DIRECTORY/apache/conf/server.crt`
2. Restart the agent and/or Apache services.

Replacing an Expired Certificate

The `$DATA_DIRECTORY/apache/conf/ssl.conf` file contains the following relevant lines for the web server certificate and key:

```
SSLCertificateFile conf/server.crt

SSLCertificateKeyFile conf/server.key
```

To replace an expired certificate with a new certificate:

1. Generate a new server key.
2. Generate a CA request.
3. Get the certificate signed by your CA.
4. Replace the above files in the `$DATA_DIRECTORY/apache/conf` folder.
5. Restart the agent and/or Apache services.

> **Note:** The ElectricFlow web server does not use a keystore.

Using chkconfig

`chkconfig` is a simple command-line tool for maintaining the `/etc/rc[0-6].d` directory hierarchy. This tool relieves system administrators from the task of directly manipulating numerous symbolic links in those directories. The Linux `chkconfig` command can be used to manipulate ElectricFlow services running on UNIX platforms.

`chkconfig`—updates and queries runlevel information for system services

```
chkconfig --list [name]
chkconfig --add name
chkconfig --del name
chkconfig [--level levels] name <on|off|reset>
chkconfig [--level levels] name
```

Examples

```
(list current settings for the local ElectricFlow repository service)
/sbin/chkconfig commanderRepository --list
commanderRepository 0:off 1:off 2:off 3:on 4:off 5:on 6:off

(disable autostart on reboot)
/sbin/chkconfig commanderRepository off
/sbin/chkconfig commanderRepository --list
commanderRepository 0:off 1:off 2:off 3:off 4:off 5:off 6:off
```

> **Note:** For every service, each runlevel has either a "start" script or a "stop" script. When switching runlevels, `init` will not restart an already-started service and will not re-stop a non-running service.

Starting and Stopping Servers and Agents Manually

ElectricFlow servers and agents must be manually stopped and started for administrative maintenance, upgrades, third-party software installations, or system maintenance.

Stopping the ElectricFlow Agent Service

To stop the ElectricFlow agent service, choose one of the following options.

Windows

1. Go to **Control Panel** > **Administrative Tools** > **Services**.
2. Right-click **ElectricFlow Agent** and click **Stop**.

Windows Command Line

Open a command window as Administrator and enter:

```
sc stop CommanderAgent
```

Linux

Log into a shell as `root` and enter one of the following commands:

- Linux: `/etc/init.d/commanderAgent stop`
- Solaris: `/etc/init.d/ecmdrAgent stop`
- AIX: `/etc/rc.d/init.d/ecmdrAgent stop`
- HP-UX: `/sbin/init.d/ecmdrAgent stop`
- macOS: `launchctl unload /Library/LaunchDaemons/ecmdrAgent.plist`

Stopping All ElectricFlow Server Services

To stop all ElectricFlow server services, choose one of the following options.

Windows

1. Go to **Control Panel** > **Administrative Tools** > **Services**.
2. Right-click **ElectricFlow Server** and click **Stop**.
3. Right-click **ElectricFlow Web Server** and click **Stop**.
4. Right-click **ElectricFlow Database** (if it exists) and click **Stop**.
5. Right-click **ElectricFlow Repository Server** and click **Stop**.

Windows Command Line

Open a command window as Administrator and enter:

1. `sc stop CommanderServer`
2. `sc stop CommanderApache`
3. `sc stop CommanderDatabase`
4. `sc stop CommanderRepository`

Linux

Log into a shell as `root` and enter:

1. `/etc/init.d/commanderServer stop`
2. `/etc/init.d/commanderApache stop`

3. `/etc/init.d/CommanderDatabase stop`
4. `/etc/init.d/commanderRepository stop`

Stopping All DevOps Insight Services

DevOps Insight uses services for Elasticsearch and Logstash. To stop these services, log into the DevOps Insight server and choose one of the following options. Because Logstash sends data to Elasticsearch, you stop Logstash first to prevent Logstash errors.

Windows

1. Go to **Control Panel** > **Administrative Tools** > **Services**.
2. Right-click **ElectricFlow Logstash Service** and click **Stop**.
3. Right-click **ElectricFlow Elasticsearch Service** and click **Stop**.

Windows Command Line

Open a command window as Administrator and enter:

1. `sc stop CommanderLogstash`
2. `sc stop CommanderElasticsearch`

Linux

Log into a shell as `root` and enter:

1. `/etc/init.d/commanderLogstash stop`
2. `/etc/init.d/commanderElasticsearch stop`

Starting the ElectricFlow Agent Service

To start the ElectricFlow agent service, choose one of the following options.

Windows

1. Go to **Control Panel** > **Administrative Tools** > **Services**.
2. Right-click **ElectricFlow Agent** and click **Start**.

Windows Command Line

Open a command window as Administrator and enter:

```
sc start CommanderAgent
```

Linux

Log into a shell as `root` and enter one of the following commands:

- Linux: `/etc/init.d/commanderAgent stop`
- Solaris: `/etc/init.d/ecmdrAgent stop`
- AIX: `/etc/rc.d/init.d/ecmdrAgent stop`
- HP-UX: `/sbin/init.d/ecmdrAgent stop`
- macOS: `launchctl load /Library/LaunchDaemons/ecmdrAgent.plist`

Starting All ElectricFlow Server Services

To start all ElectricFlow server services, choose one of the following options.

Windows

1. Go to **Control Panel** > **Administrative Tools** > **Services**.
2. Right-click **ElectricFlow Database** (if it exists) and click **Start**.
3. Right-click **ElectricFlow Server** and click **Start**.
4. Right-click **ElectricFlow Web Server** and click **Start**.
5. Right-click **ElectricFlow Repository Server** and click **Start**.

Windows Command Line

Open a command window as Administrator and enter:

1. `sc start CommanderDatabase`
2. `sc start CommanderServer`
3. `sc start CommanderApache`
4. `sc start CommanderRepository`

Linux

Log into a shell as `root` and enter:

1. `/etc/init.d/CommanderDatabase start`
2. `/etc/init.d/commanderServer start`
3. `/etc/init.d/commanderApache start`
4. `/etc/init.d/commanderRepository start`

Starting All DevOps Insight Services

To start the DevOps Insight services (Elasticsearch and Logstash), log into the DevOps Insight server and choose one of the following options. Because Logstash sends data to Elasticsearch, you start Elasticsearch first to prevent Logstash errors.

Windows

1. Go to **Control Panel** > **Administrative Tools** > **Services**.
2. Right-click **ElectricFlow Elasticsearch Service** and click **Start**.
3. Right-click **ElectricFlow Logstash Service** and click **Start**.

Windows Command Line

Open a command window as Administrator and enter:

1. `sc start CommanderElasticsearch`
2. `sc start CommanderLogstash`

Linux

Log into a shell as `root` and enter:

1. `/etc/init.d/commanderElasticsearch start`
2. `/etc/init.d/commanderLogstash start`

Collecting ElectricFlow Logs

You can collect ElectricFlow logs as well as user-defined logs (such as for Apache or Oracle WebLogic) for all components in an ElectricFlow standalone server and its agents or in an ElectricFlow cluster. These logs are as follows:

- ElectricFlow server logs
- ElectricFlow agent logs
- ElectricFlow repository server logs
- ElectricFlow job logs
- ElectricFlow installer logs
- Apache (web server) logs
- User-defined logs

ElectricFlow technical support might ask you for one or more these logs to troubleshoot issues. You can use one of three methods to collect logs:

- Collecting Logs by Using the Logs Collection Self-Service Catalog Item on page 12-22
- Collecting Logs by Running the EC-FlowLogCollector Plugin Procedure Directly on page 12-28
- Collecting Logs Manually on page 12-31

The first two methods let you collect logs automatically from a standalone ElectricFlow server and one or more of its agents or a cluster of two or more ElectricFlow servers and one or more agents on each server. In the third method, you collect the log files individually from each server or agent system.

Prerequisites and Limitations for ElectricFlow Log Collection

For ElectricFlow server prerequisites, prerequisites for collecting logs from all cluster nodes, as well as limitations, see the online Help file for the underlying ElectricFlow plugin by clicking **Administration** > **Plugins** > **EC-FlowLogCollector** > **Help**.

Collecting Logs by Using the Logs Collection Self-Service Catalog Item

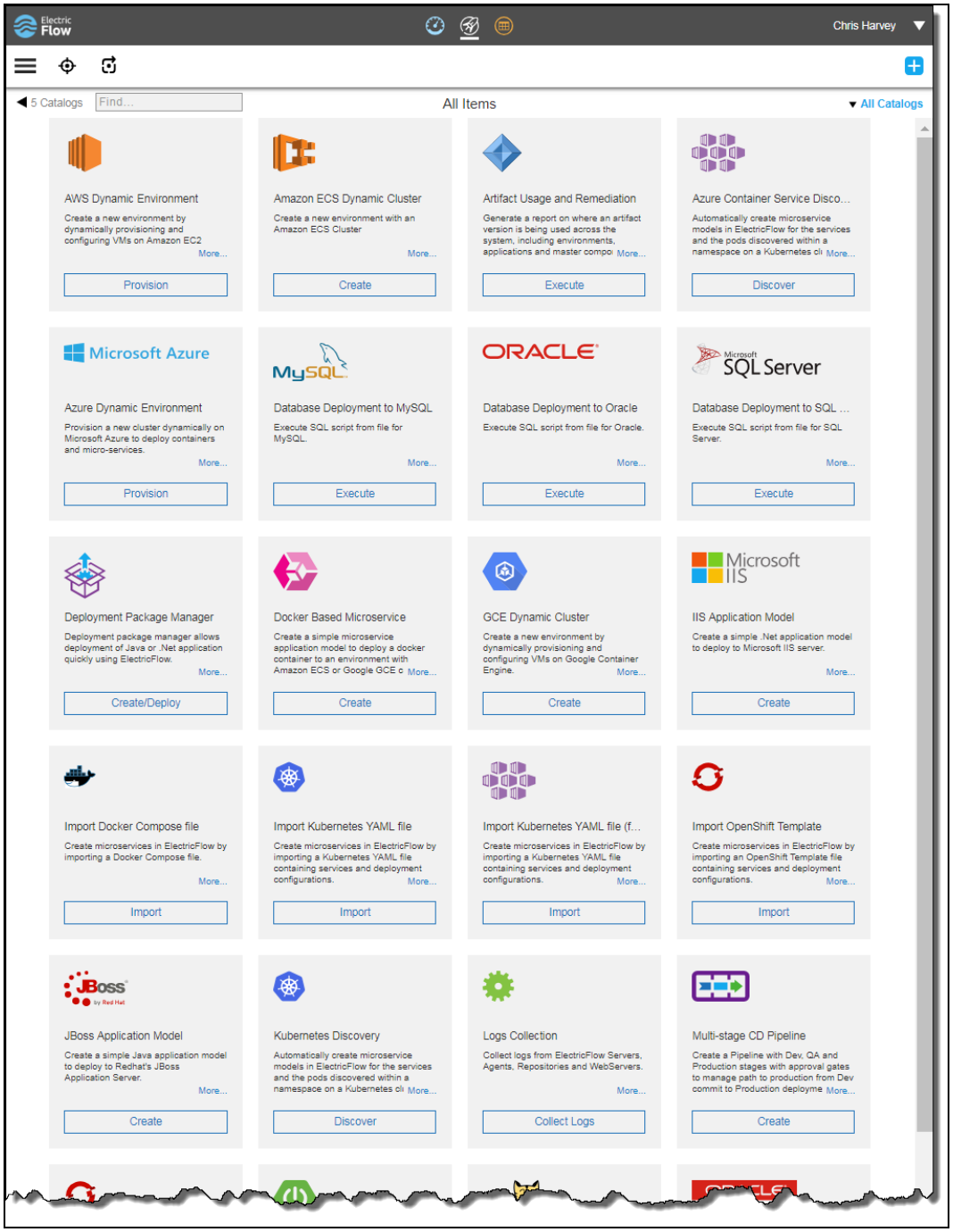
Collecting the Logs via the Self-Service Catalog

To collect logs via the Self-Service Catalog:

1. Open the home page of the ElectricFlow web UI by browsing to https://<*ElectricFlow_server*>/flow/.

2. Click the (Self-Service Catalogs) button.

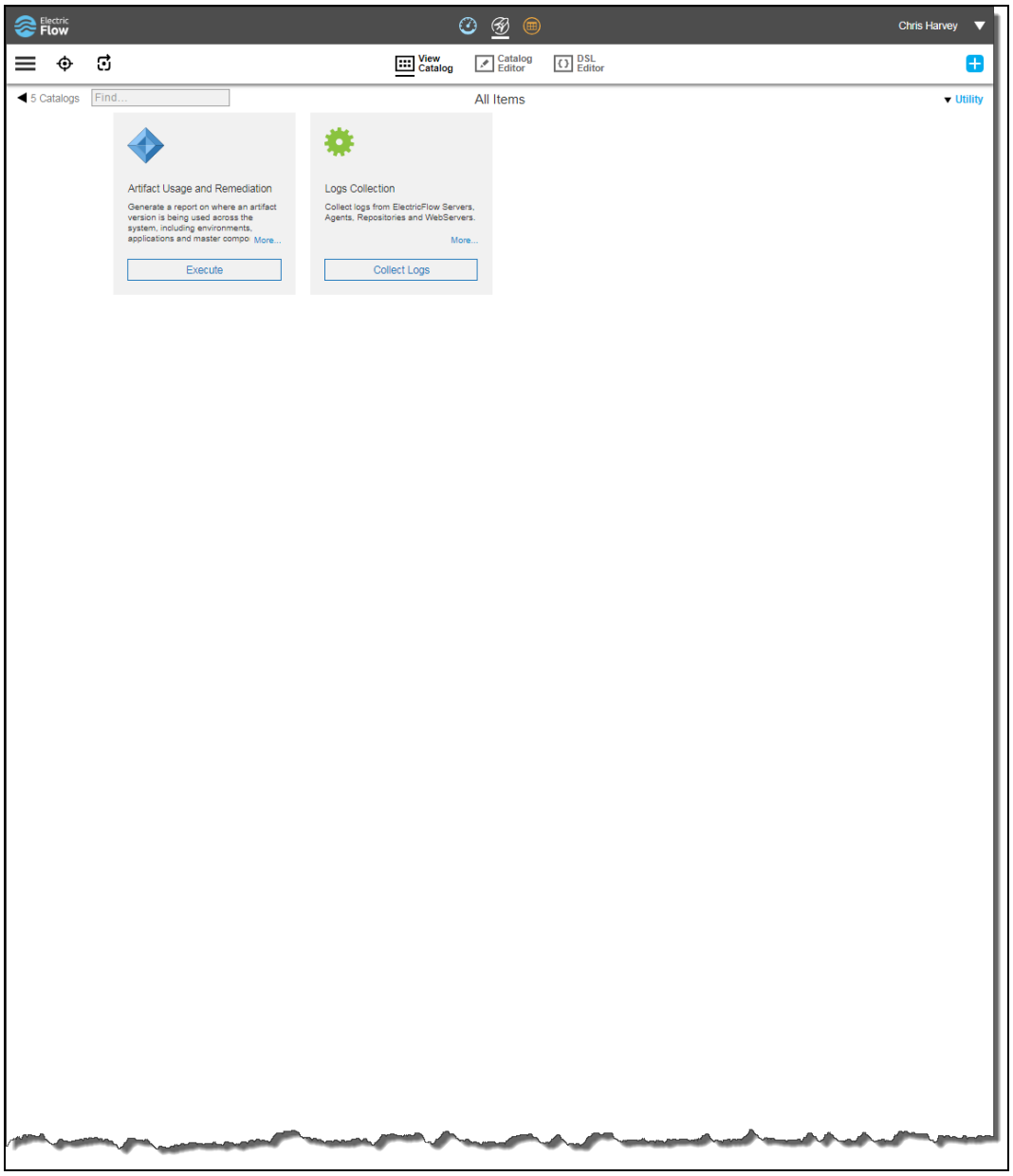
   The Self Service Catalog appears:



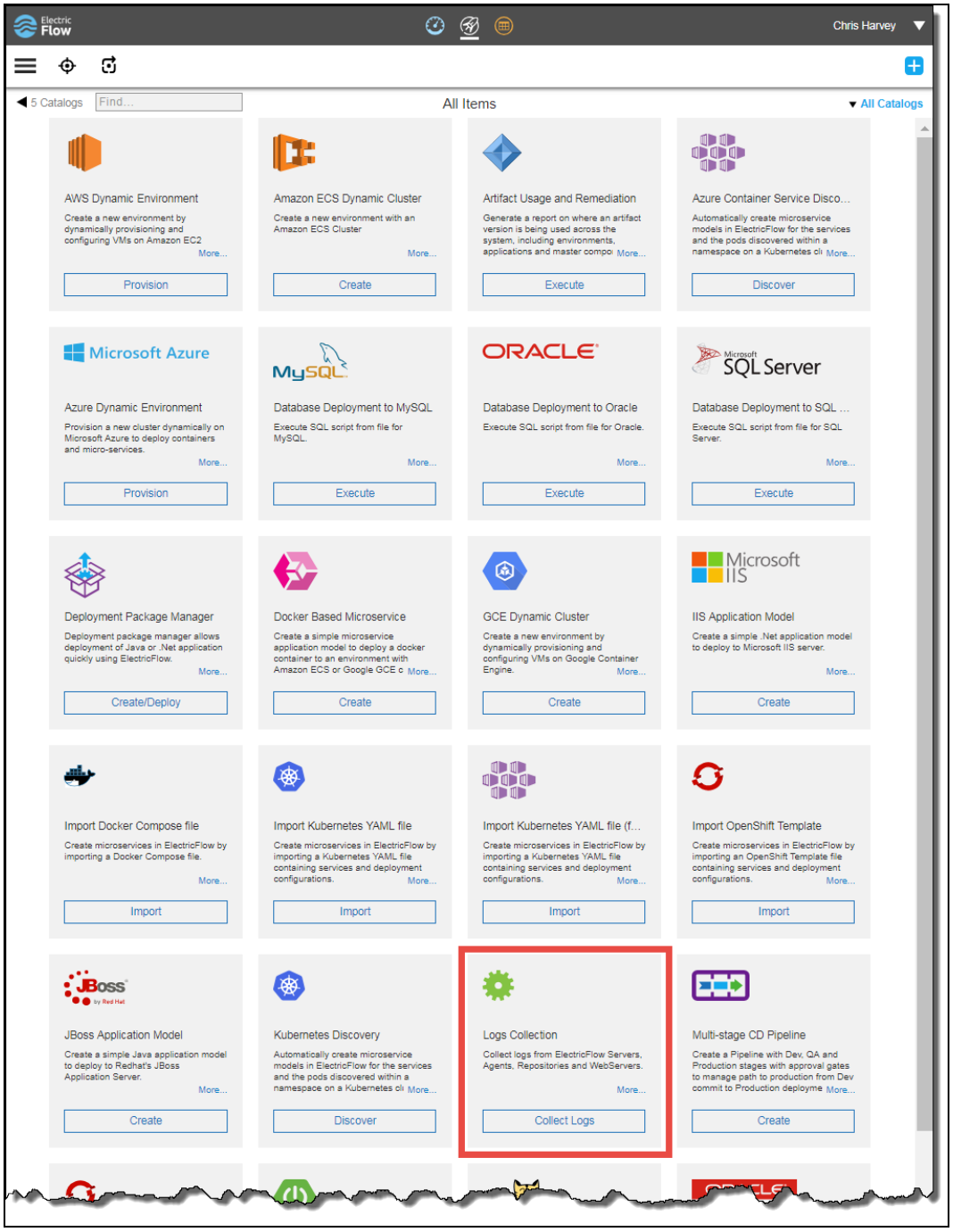
For more information about using the Self Service Catalog, see the "Self-Service Catalogs" chapter in the *ElectricFlow User Guide* at http://docs.electric-cloud.com/eflow_doc/FlowIndex.html.

3. Click the **All Catalogs** pull-down menu and choose **Utility** to filter the selection to items in the utility category.

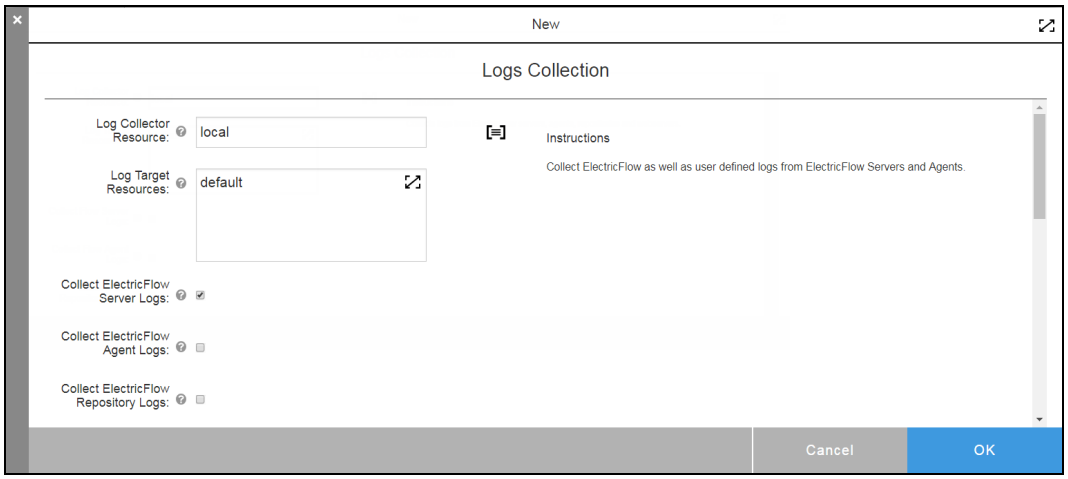
   The list is filtered to display only the Utility catalog:



4. Go to the **Logs Collection** catalog item, and click **Collect Logs**:



The **Logs Collection** dialog box appears:



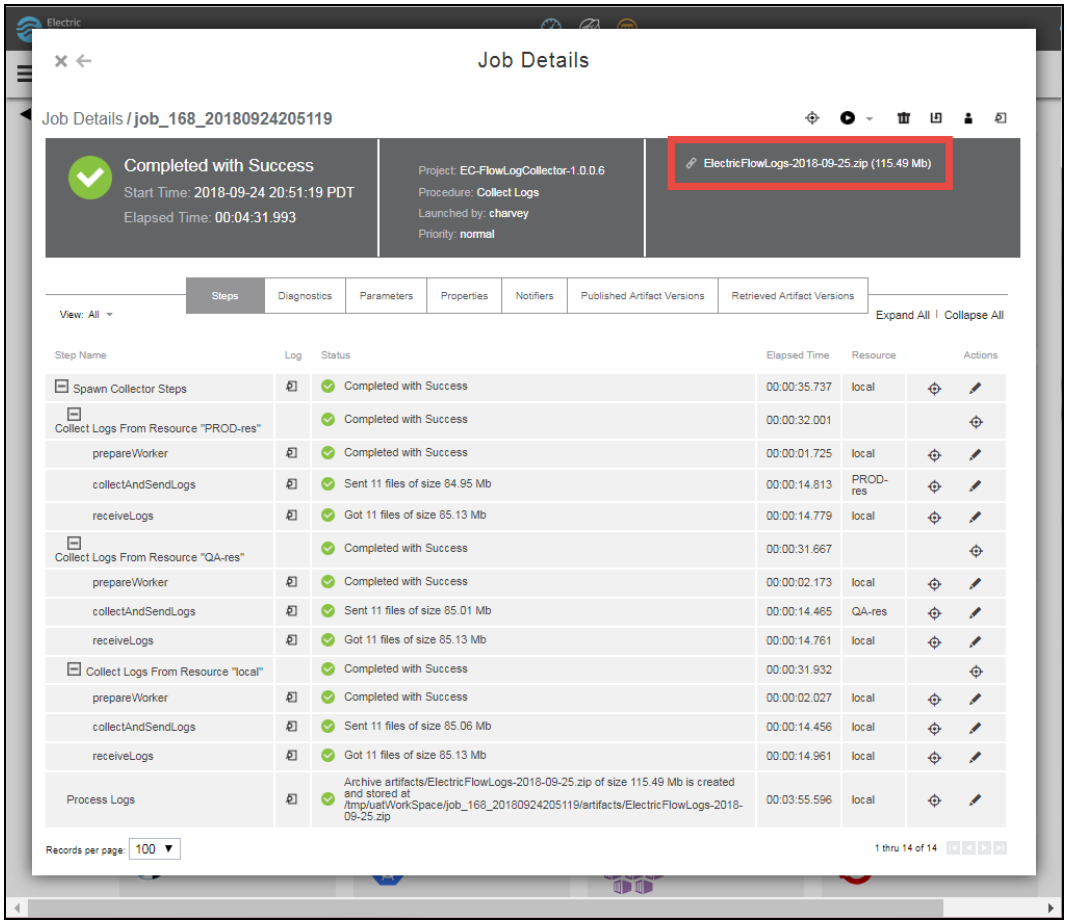
5. (Optional) Enter any additional parameters into the **Logs Collection** dialog box.

   The required parameters are **Log Collector Resource** and **Log Target Resources**. These parameters are defaulted to `local` and `default` respectively. All other parameters are optional and do not require values to produce a basic set of logs.

   For descriptions of all parameters (such as how they are chosen, limitations, and pre-requisites) for this procedure, see the online Help file for the underlying ElectricFlow plugin by clicking **Administration** > **Plugins** > **EC-FlowLogCollector** > **Help**.

6. Click **OK.**

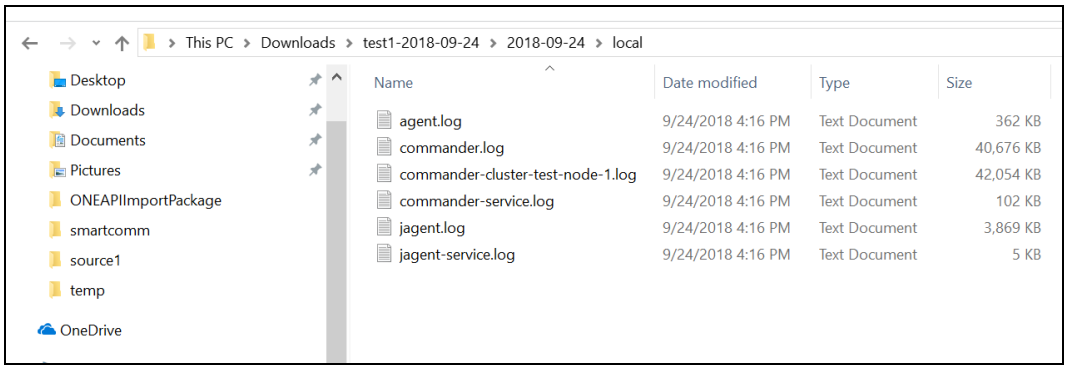
   The catalog item produces a .zip file of the individual logs. This file appears on the resulting **Job Details** page. For example:



7. Right-click to save the .zip file to your system.
8. Either unzip the file to see the individual logs or send it to Electric Cloud technical support for analysis via an existing support ticket.

Log File Contents

Following is an example of the contents of a log file after unzipping:



The `local` folder in the example above contains the logs for the `local` resource, which is the default resource. Each resource that you specify in the **Log Target Resources** field as described above will have its own folder, which will be named after that resource.

Collecting Logs by Running the EC-FlowLogCollector Plugin Procedure Directly

The log collection functionality is based on the underlying EC-FlowLogCollector plugin. This plugin is bundled with ElectricFlow and performs the actual collection of logs from ElectricFlow servers and agents.

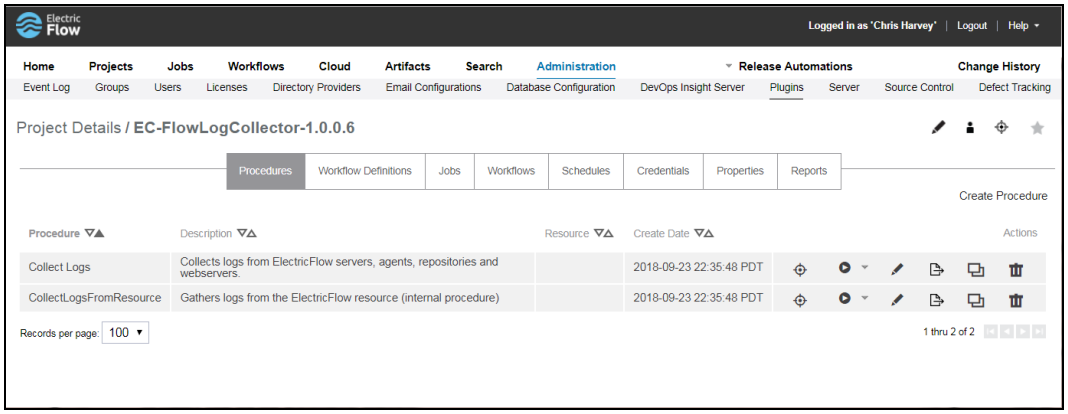
> **Tip:** The EC-FlowLogCollector plugin is also available for downloading at https://-github.com/electric-cloud.

An alternative to collecting logs via the **Logs Collection** Self-Service Catalog item is to run the plugin's **Collect Logs** procedure directly in the Automation Platform.

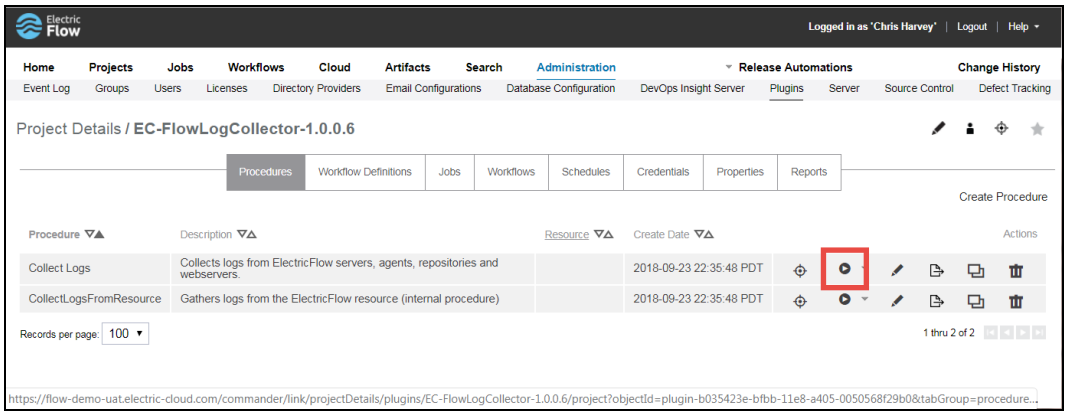
To collect logs via the Automation Platform:

1. Go to https://<*ElectricFlow_server*>/commander/.
2. Click **Administration** > **Plugins** > **EC-FlowLogCollector**.

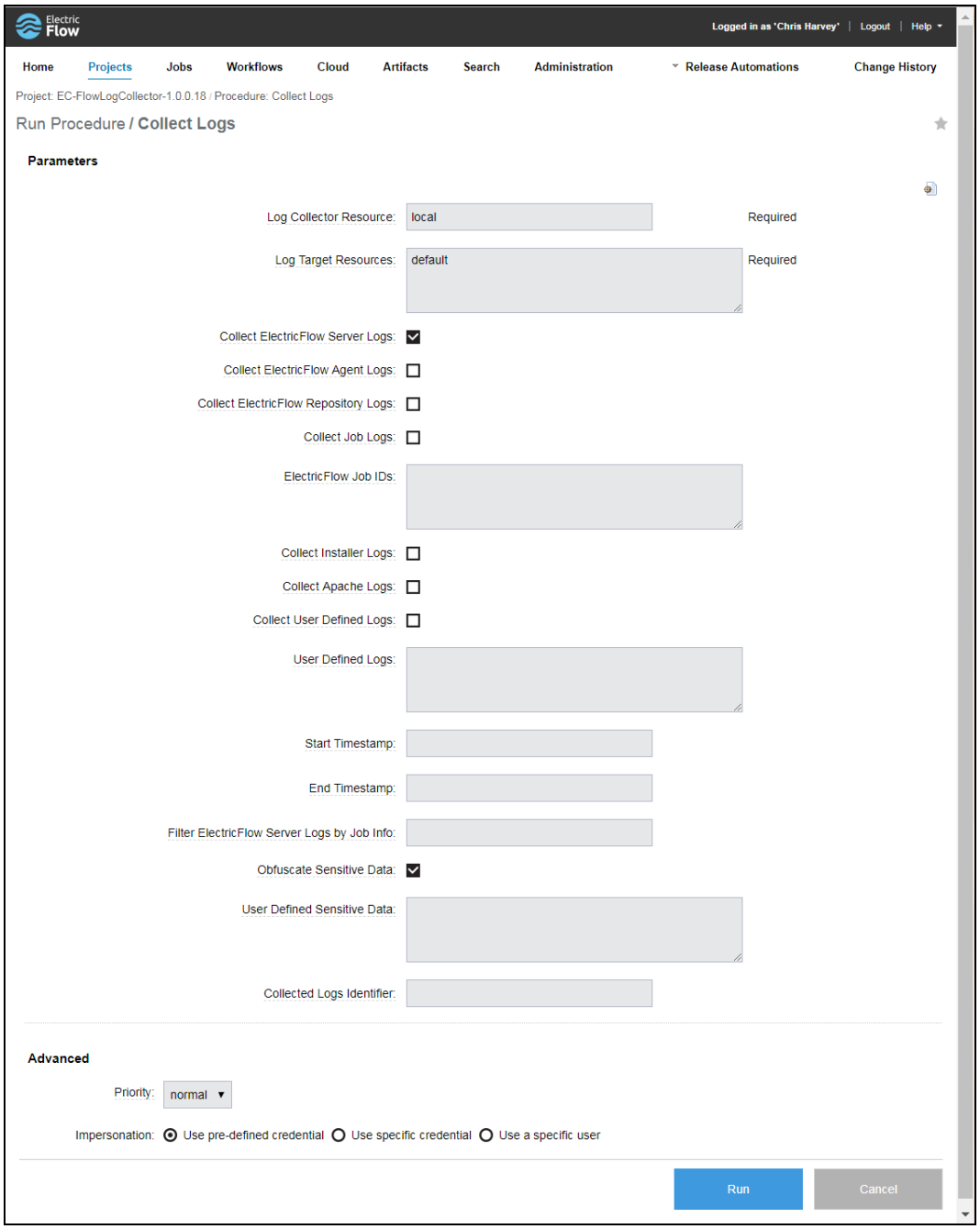
   The **Project Details** page for the **EC-FlowLogCollector** plugin appears:



3. Click the (Run Immediately) button for the **Collect Logs** procedure in the plugin:



The **Collect Logs** page appears:



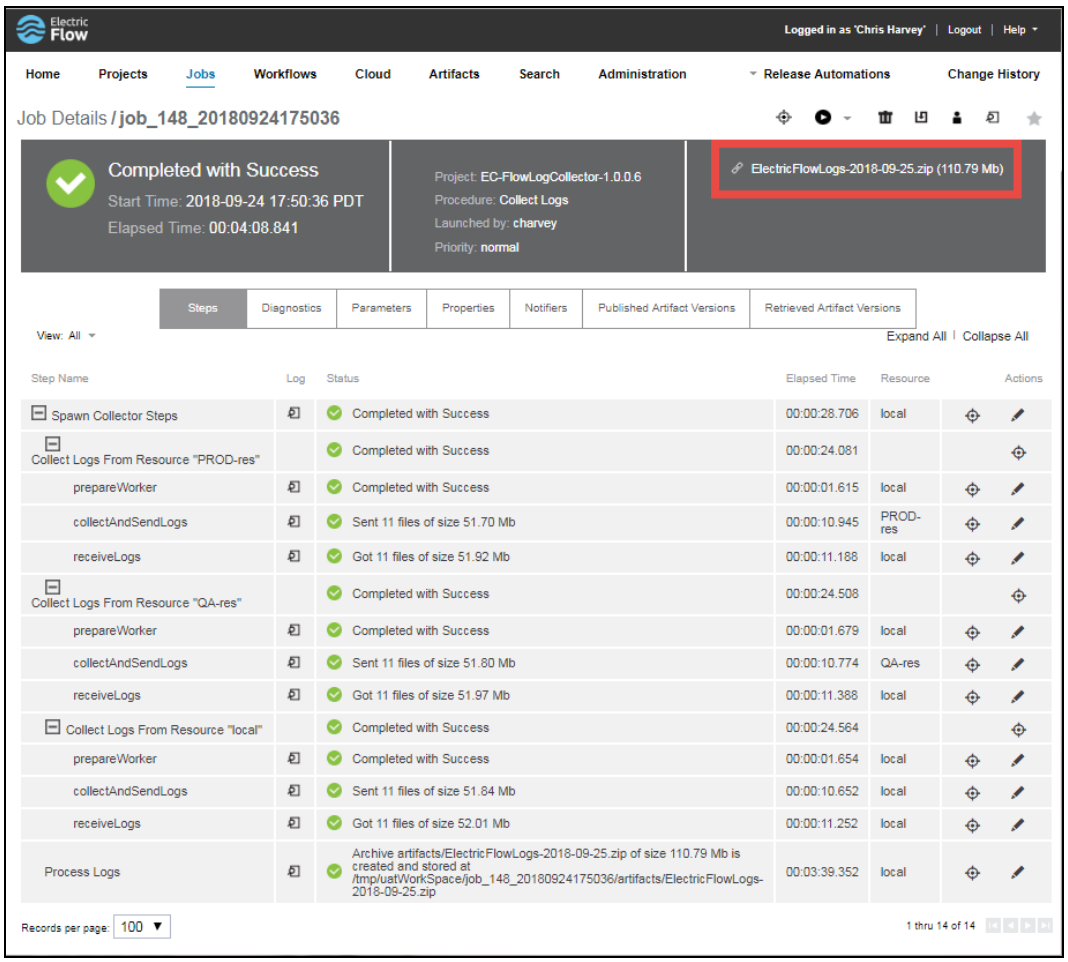
4. (Optional) Enter any additional parameters into the **Logs Collection** dialog box.

   The required parameters are **Log Collector Resource** and **Log Target Resources** and are defaulted to `local` and `default` respectively. All other parameters are optional and do not require values to produce a basic set of logs.

   For descriptions of all parameters (such as how they are chosen, limitations, and pre-requisites) for this procedure, see the online Help file for the underlying ElectricFlow plugin by clicking **Administration** > **Plugins** > **EC-FlowLogCollector** > **Help**.

5. Click **Run**.

   The procedure generates a .zip file on the resulting **Job Details** page. For example:



6. Right-click to save the .zip file to your system.
7. Either unzip the file to see the individual logs or send it to Electric Cloud technical support for analysis via an existing support ticket.

   For an explanation of these logs, see Log File Contents on page 12-27.

Collecting Logs Manually

You collect logs manually from individual systems from the locations listed below. The following information is for default "run time" log locations.

Agent Logs

Platform	Default Path
Windows	C:\ProgramData\Electric Cloud\ElectricCommander\logs\agent
Linux or UNIX	/opt/electriccloud/electriccommander/logs

Agent logs "roll over" periodically so individual logs do not grow too large, and older logs are deleted. Roll-over parameters are configurable in `conf/logback.xml` and `conf/agent.conf`.

Server Logs

Platform	Default Path
Windows	C:\ProgramData\Electric Cloud\ElectricCommander\logs
Linux or UNIX	/opt/electriccloud/electriccommander/logs

Server logs "roll over" periodically so individual logs do not grow too large and older logs are deleted. Roll-over parameters are configurable in `conf/logback.xml` and `conf/agent.conf`.

Web Server Logs

Platform	Default Path
Windows	C:\ProgramData\Electric Cloud\ElectricCommander\apache\logs
Linux or UNIX	/opt/electriccloud/electriccommander/apache/logs

Repository Server Logs

Platform	Default Path
Windows	C:\ProgramData\Electric Cloud\ElectricCommander\logs\repository
Linux or UNIX	/opt/electriccloud/electriccommander/logs/repository

Installer Logs

Platform	Default Path
Windows	C:\ProgramData\Electric Cloud\ElectricCommander\logs
Linux or UNIX	/opt/electriccloud/electriccommander/logs

Web Interface Online Help System

Open the ElectricFlow online help system for more information. Click the **Help** link in the top-right corner of any product web page to see a help topic for that page.

When the help system opens, We recommend reviewing the Help table of contents. All Help folders above the Web Interface Help folder are user-guide style help topics that provide more detailed information on each of their subjects.

If you generally prefer to use a command-line tool rather than the ElectricFlow web interface, you will find complete ectool (the ElectricFlow command-line tool) and API (perl script) commands and options within the online help system too.

Chapter 13: Troubleshooting an ElectricFlow Installation

This chapter contains troubleshooting procedures for some of the more common issues you might experience during the ElectricFlow installation process. More troubleshooting information can be found in ElectricFlow Knowledge Base articles located at https://helpcenter.electric-cloud.com/hc/en-us/sections/200516863-Commander-KB.

ElectricFlow Server is Unresponsive and Displays an OutOfMemory Error

Description

The ElectricFlow server becomes unresponsive and displays an `OutOfMemory` error prompt that states that the server is out of `PermGen` space. This prompt occurs when the server is not configured properly for a 64-bit JVM.

Workaround

Change a setting in `wrapper.conf`.

1. Locate the `wrapper.conf` file.
   - The file to edit on Linux is:

     ```
     /opt/electriccloud/electriccommander/conf/wrapper.conf
     ```

   - The file to edit on Windows is:

     ```
     <ElectricCommander data dir>/conf/wrapper.conf
     ```

   - On Windows 7 or 2008, the `DATA_DIRECTORY` is typically:

     ```
     C:\ProgramData\Electric Cloud\ElectricCommander
     ```

2. Add the following line to the end of the existing "`wrapper.java.additional`" section, being careful to use the next consecutive number.

   In the following example, 7 was the last pre-existing sequence number in this section.

   ```
   wrapper.java.additional.8=-XX:MaxPermSize=128m
   ```

   > **Note:** You should set the Java `MaxPermSize` to a number greater than the default (which is 84m) on an 64-bit ElectricFlow server (java) installation.

3. After the file has been edited, restart the server to utilize the new value:
   - On Linux: `/etc/init.d/commanderServer restart`
   - On Windows: use "Services"

Windows PHP Does Not Handle Time Zones Correctly

Description

PHP does not handle certain operating system time zones correctly on a Windows system. If the web server is running on a machine set for one of these time zones, users connected to that web server will see all times displayed as UTC times, instead of the web server time zone.

Workaround

In the `config.php` file, you must explicitly set the PHP "timezone_identifier".

To set the timezone:

1. Edit the following file.

   ```
   C:\Program Files\Electric
   Cloud\ElectricCommander\apache\htdocs\commander\config.php
   ```

2. Add the following line anywhere between the opening and closing PHP tags:

   ```
   date_default_timezone_set("<timezone_identifier>");
   ```

   For example:

   To set the timezone for Taipei, you would add: `date_default_timezone_set("Asia/Taipei");`

   For a complete list of supported time zones, see http://us2.php.net/manual/en/timezones.php.

ElectricFlow Self-Signed Server Certificate Fails Security Scan

Description

You might need to replace the self-signed ElectricFlow server certificate if it fails the security scan.

> **Note:** If you are using a certificate authority (CA) certificate or an intermediate CA certificate instead and it has expired, see ElectricFlow CA or Intermediate CA Server Certificate Expires on page 13-4 for details about updating it.

There are three relevant configuration entries in the `server/conf/commander.properties` file:

```
COMMANDER_HTTPS_PORT=8443

COMMANDER_KEYSTORE=file:conf/keystore

COMMANDER_KEYSTORE_PASSWORD=abcdef
```

Where:

- `COMMANDER_HTTPS_PORT` configures the SSL port
- `COMMANDER_KEYSTORE` is the location of the java keystore where the ElectricFlow HTTP server finds its host certificate
- `COMMANDER_KEYSTORE_PASSWORD` is the password to the keystore

Workaround

Follow these steps to generate and inject a self-signed certificate for 1 year.

1. Back up the keystore file.
2. Delete the original key.

```
user@USER /cygdrive/c/ProgramData/Electric Cloud/ElectricCommander/conf

$ "c:/Program Files/Electric Cloud/ElectricCommander/jre/bin/keytool" -delete -
alias jetty -keystore keystore -keypass passkey

Enter keystore password: abcdef
```

3. Generate and inject a new certificate.

```
user@USER /cygdrive/c/ProgramData/Electric Cloud/ElectricCommander/conf

$ "c:/Program Files/Electric Cloud/ElectricCommander/jre/bin/keytool" -keystore
keystore -alias jetty -genkey -keyalg RSA -sigalg MD5withRSA -validity 365

Enter keystore password: abcdef

What is your first and last name?
  [Unknown]:  localhost

What is the name of your organizational unit?
  [Unknown]: <Enter>

What is the name of your organization?
  [Unknown]: <Enter>

What is the name of your City or Locality?
  [Unknown]: <Enter>

What is the name of your State or Province?
  [Unknown]: <Enter>

What is the two-letter country code for this unit?
  [Unknown]: <Enter>

Is CN=localhost, OU=Unknown, O=Unknown, L=Unknown, ST=Unknown, C=Unknown
correct?
  [no]:  yes

Enter key password for <jetty>
  (RETURN if same as keystore password): <Enter>
```

4. Restart the server.

   Your new certificate will look similar to this:

```
user@USER /cygdrive/c/ProgramData/Electric Cloud/ElectricCommander/conf

$ "c:/Program Files/Electric Cloud/ElectricCommander/jre/bin/keytool" -list -v -
keystore keystore_orig -keypass passkey

Enter keystore password: abcdef

Keystore type: JKS

Keystore provider: SUN

Your keystore contains 1 entry
```

```
Alias name: jetty

Creation date: Jan 31, 2012

Entry type: PrivateKeyEntry

Certificate chain length: 1

Certificate[1]:

Owner: CN=localhost, OU=Unknown, O=Unknown, L=Unknown, ST=Unknown, C=Unknown

Issuer: CN=localhost, OU=Unknown, O=Unknown, L=Unknown, ST=Unknown, C=Unknown

Serial number: 4f28603f

Valid from: Tue Jan 31 13:42:23 PST 2012 until: Wed Jan 30 13:42:23 PST 2013

Certificate fingerprints:

MD5:  38:50:CD:29:8C:16:3A:78:29:0F:45:56:E0:CA:42:D9

SHA1: 9B:A3:E4:EA:A7:C0:3A:ED:BF:63:24:18:F0:08:78:22:59:85:BC:8A

Signature algorithm name: MD5withRSA

Version: 3

*******************************************

*******************************************
```

ElectricFlow CA or Intermediate CA Server Certificate Expires

Description:

When using a certificate authority (CA) certificate or an intermediate CA certificate, the certificate expires and causes certificate-related errors.

> **Note:** ElectricFlow uses a self-signed certificate by default. This section describes how to update a CA or intermediate CA certificate if you have used one to replace the self-signed certificate. If you are using the self-signed certificate instead and it has expired, see ElectricFlow Self-Signed Server Certificate Fails Security Scan on page 13-2 for details about updating it.

Workaround:

ElectricFlow certificates use Jetty. Follow these steps to update the existing certificate in the keystore and then publish it to Zookeeper:

1. Shut down all nodes on the ElectricFlow cluster except for one node.
2. Go to the ElectricFlow *<install_dir>* directory on the node.
3. Delete the existing certificate from the keystore by entering:

   ```
   jre/bin/keytool -delete -alias jetty -keystore keystore -keypass passkey
   ```

4. Generate a new key pair.

   Specify a validity (in days) and a key size of either 1024 or 2048 by entering:

   ```
   jre/bin/keytool -keystore keystore -alias jetty -genkey -keyalg RSA -sigalg
   MD5withRSA -validity 3650 -keysize 2048
   ```

5. Generate a certificate signing request (CSR) from the keystore by entering:

   ```
   jre/bin/keytool -certreq -alias jetty -keystore keystore -file certreq.csr
   ```

6. Sign the CSR using your CA.

7. Import the signed certificate into the keystore by entering:

   ```
   jre/bin/keytool -importcert -file <certificate> -keystore keystore -alias jetty
   ```

8. If ElectricFlow is clustered, publish the keystore to Zookeeper.

   Go to the *<install_dir>*/conf directory and use the steps in Uploading Configuration Files to ZooKeeper on page 4-25. For example, enter the following command.

   - Linux:

     ```
     COMMANDER_ZK_CONNECTION=<ZooKeeper_Server_IP>:2181 ../jre/bin/java -jar
     ../server/bin/zk-config-tool-jar-with-dependencies.jar
     com.electriccloud.commander.cluster.ZKConfigTool --keystoreFile keystore
     ```

   - Windows:

     ```
     "C:\Program Files\Electric Cloud\ElectricCommander\jre\bin\java.exe" -
     DCOMMANDER_ZK_CONNECTION=<ZooKeeper_Server_IP>:2181 -jar "C:\Program
     Files\Electric Cloud\ElectricCommander\server\bin\zk-config-tool-jar-with-
     dependencies.jar" com.electriccloud.commander.cluster.ZKConfigTool --
     databasePropertiesFile database.properties --keystoreFile keystore
     ```

Linux Upgrade Breaks Symbolic Links

Description

When using the Linux installer to perform an upgrade, you might encounter problems moving broken symbolic links. You might see errors that begin with a line similar to, "`could not read "/opt/electriccloud/electriccommander/workspace/FileOperationsLinux-LocalMove-7689/fileSymLink": no such file or directory`".

Workaround

Manually remove the file and rerun the installer if you encounter these types of errors.

Chapter 14: Performing Agent-Only Installations

This chapter describes how to install the ElectricFlow agent:

- in "pseudo" 64-bit and "pure" 64-bit versions on Linux.
- in 32-bit and "pseudo" 64-bit versions on Windows.

An agent is an ElectricFlow component that runs on a machine resource. It executes ElectricFlow job steps, monitors step progress, and records job completion information.

Graphical User Interface Installation Methods

The graphical user interface installation methods are supported by Windows platforms and Linux platforms running the X Window System.

Running an Express Agent Graphical User Interface Installation (Agent-Only Installer)

The agent software must be installed on each machine you intend to use with ElectricFlow. An agent is an ElectricFlow component that runs on a machine resource. The agent executes ElectricFlow job steps, monitors step progress, and records job completion information.

Certain ElectricFlow installers allow you to perform installations as a non-root user or a user without `sudo` privileges. To determine whether a particular installer has an option to run in this mode, see Availability of Installers with a Non-Root/Non-sudo or Non-Administrator Mode on page 3-2.

Review Before You Install ElectricFlow on page 3-7 before performing these steps.

> **Note:** You install ElectricFlow agent software on Windows or Linux with this installation method. For Solaris, HP-UX, macOS, AIX, or other supported UNIX agent-only machines, see Non-Server Platform Installation Method for UNIX Agents on page 14-30.

1. Download the appropriate agent-only installer file.

   For details, see ElectricFlow Installer Files on page 3-1.

2. (Linux only) Enter the following command to make the installer file executable:

   ```
   chmod +x <agent_installer_file>
   ```

   For example, enter:

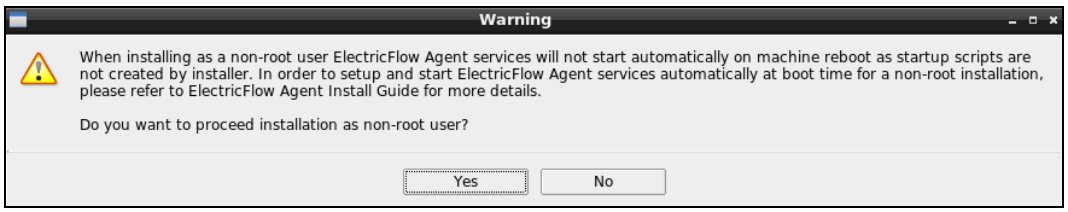
   ```
   chmod +x ElectricFlowAgent-x64-8.4.0.129860-new-with-64bit-perl
   ```

3. Do one of the following to start the installation:

   - For Linux with root or `sudo` privileges or for Windows installations, double-click the installer file.
   - For non-root/non-`sudo` installations, enter:

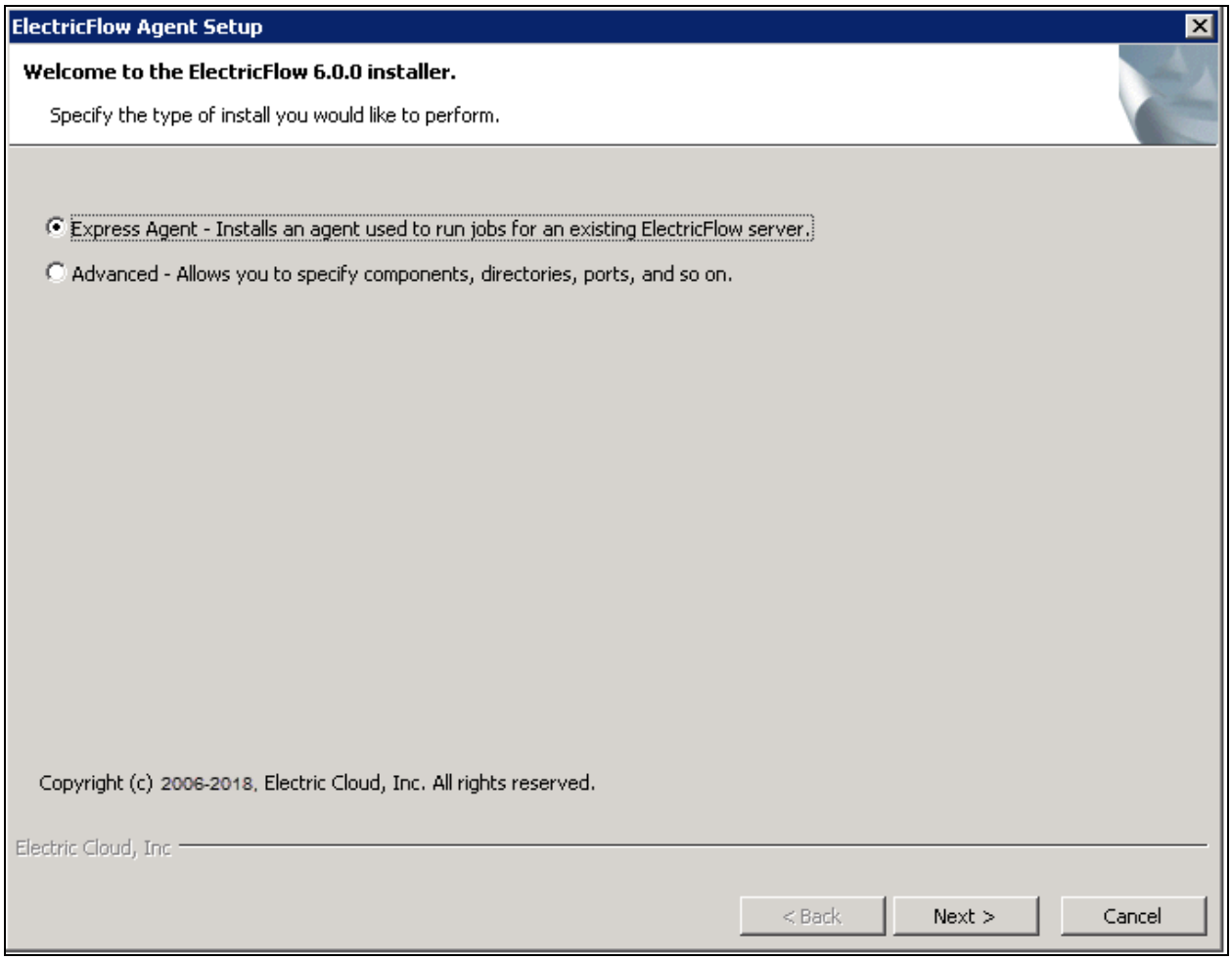
     `./<agent_installer_file> --nonRoot`

     For this installation type, the following warning appears:



4. For non-root/non-`sudo` installations, click **Yes** to dismiss the warning.

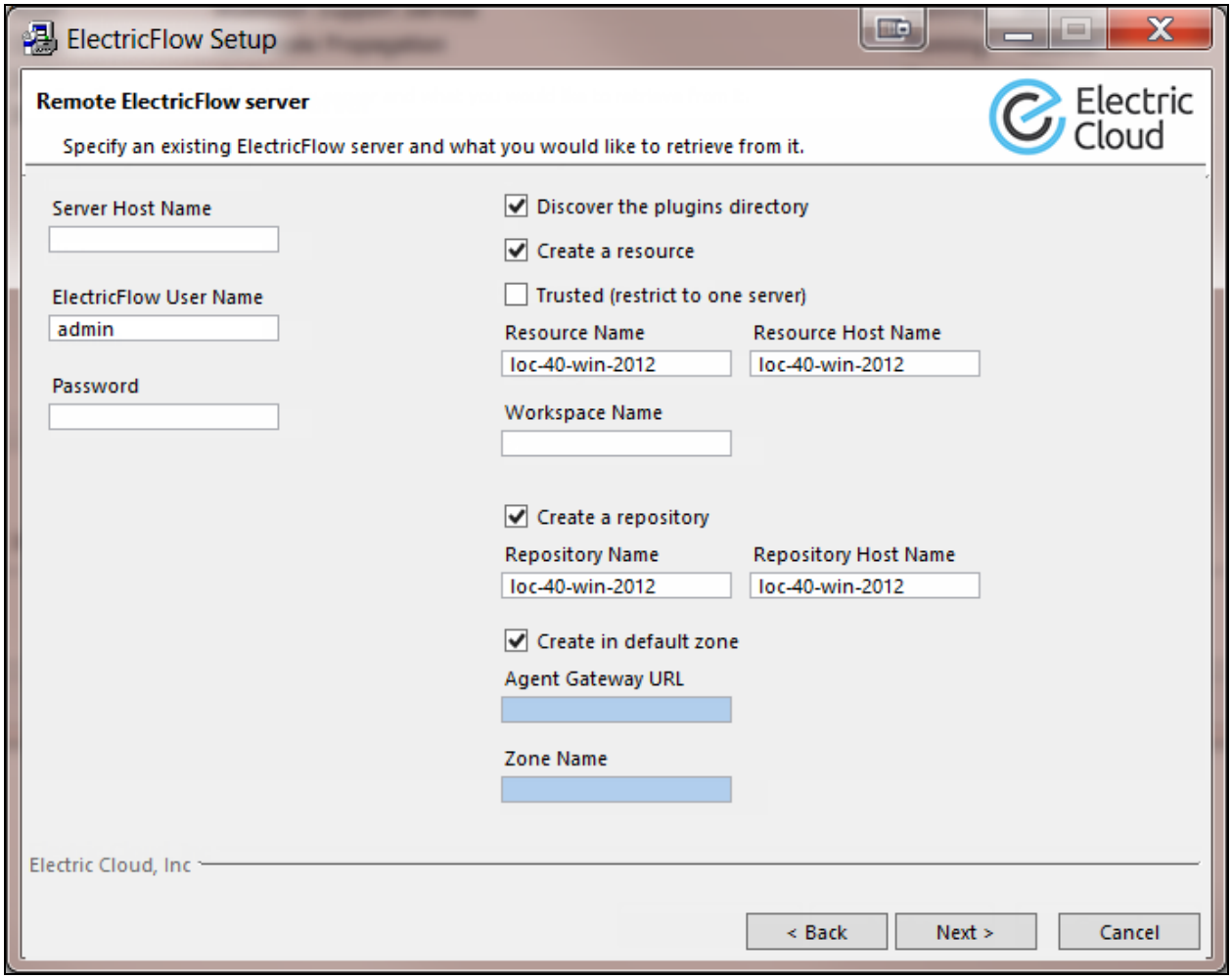
   The **Welcome to the ElectricFlow Installer** screen appears:



Note: Different options might appear depending on the operating system.

5. Select the **Express Agent** installation option, and then click **Next** to continue.

   The Remote ElectricFlow Server screen appears:



6. Complete the following information on the **Remote ElectricFlow Server** screen:

   - **Server Host Name**—Use this field to enter the name of the ElectricFlow server that will communicate with this agent. If the remote server is using a non-default HTTPS "port, you must specify the Server Host Name as *<host>*:*<port>*. If you do not specify a port, HTTPS port 8443 is assumed (the same as the ElectricFlow server default port).
   - **ElectricFlow User Name**—Use this field to enter the name of an ElectricFlow user on the ElectricFlow server who has sufficient privileges to create a resource. This field defaults to the ElectricFlow-supplied `admin` user.
   - **Password**—Use this field to enter the password for the ElectricFlow user. The default password for the `admin` user is `changeme`.

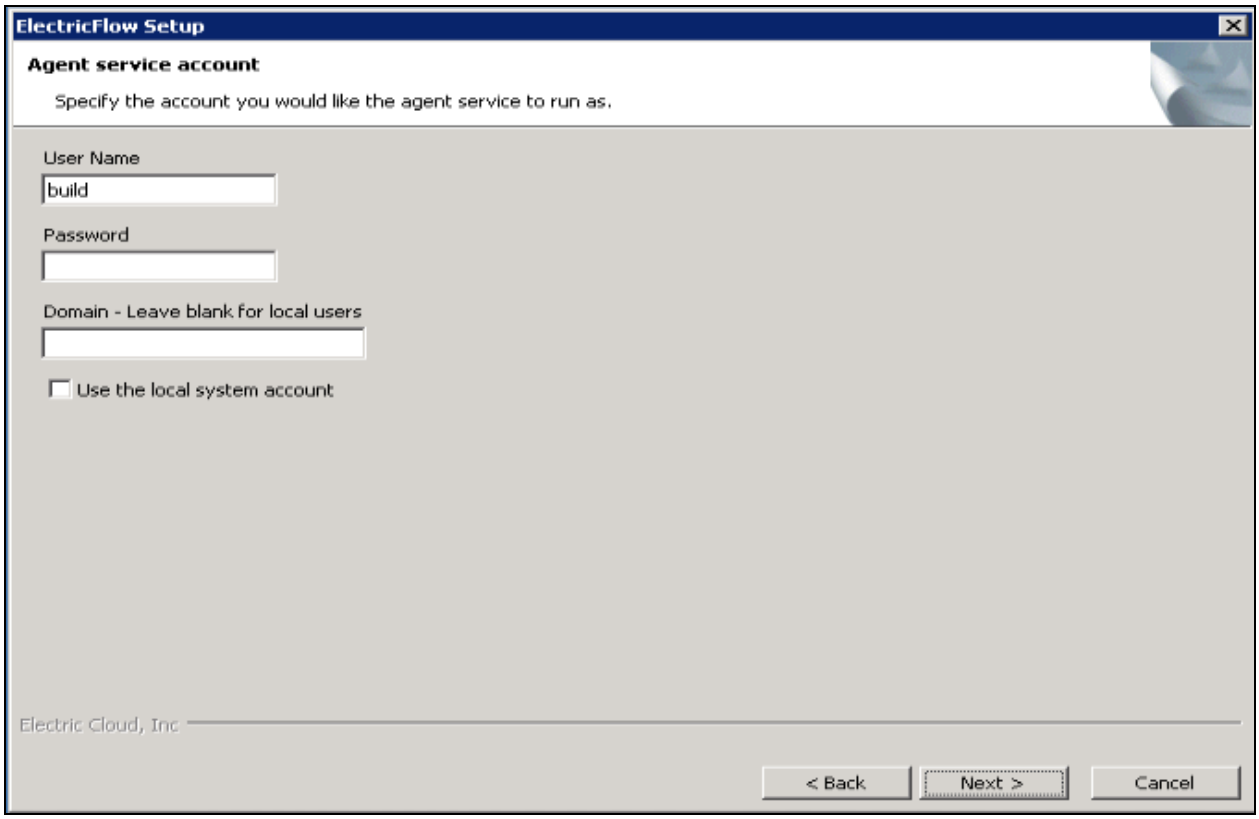
- **Discover the plugins directory**—Select this check box if you want the agent machine to have access to the plugins directory. You should allow access to the plugins directory so agents have access to collections of features, third-party integrations, or third-party tools.

> **Note:** The plugins directory on the ElectricFlow server must be "shared" before the agent machine can use "discover" to find the directory. For more information, see Universal Access to the Plugins Directory on page 5-21

- **Create a resource**—Select this check box if you want to create a resource on the remote ElectricFlow server for the agent you are installing.
- **Trusted**—Select this check box to restrict this agent to one ElectricFlow server. The agent will not respond to incoming communication from any other ElectricFlow server. This is useful when you want to create a secure production environment, but generally not needed for test or development systems.
- **Resource Name**—Use this field to enter the name of the resource you would like to use for the agent. This field is available for use when the Create a resource check box is selected.
- **Create in default zone**—Select this check box if you want to create the agent in the default zone.
- **Agent Gateway URL**—Use this field to enter the URL of the gateway used to communicate with the ElectricFlow server. This field is available for use when the Create in default zone check box is cleared.
- **Zone Name**—Use this field to enter the name of the zone used during remote agent or remote repository creation. This field is available for use when the Create in default zone check box is cleared.

7. Click **Next** to continue.

   The Agent Service Account screen appears:

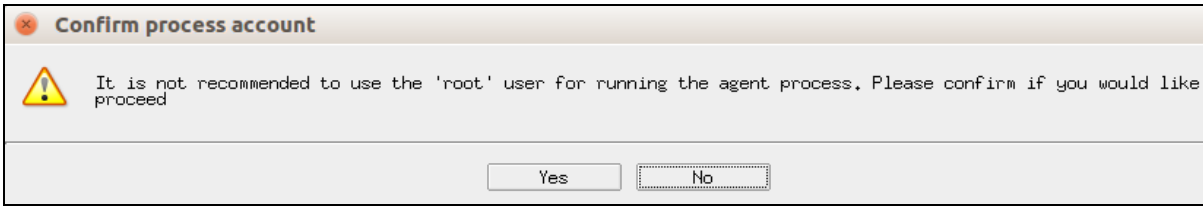


8. Select the appropriate steps for your platform and complete the following information on the screen.

   - On Linux root or `sudo` installations:

     - **User Name**—Use this field to enter the name of the user who owns the ElectricFlow agent process.

       The user/group that the agent runs as must have permission to write to the `$INSTALL_DIRECTORY/log` directory. If you specify `root`, click **Yes** when the following confirmation appears:



     - **Group Name**—Use this field to enter the name of the group who owns the ElectricFlow agent process.

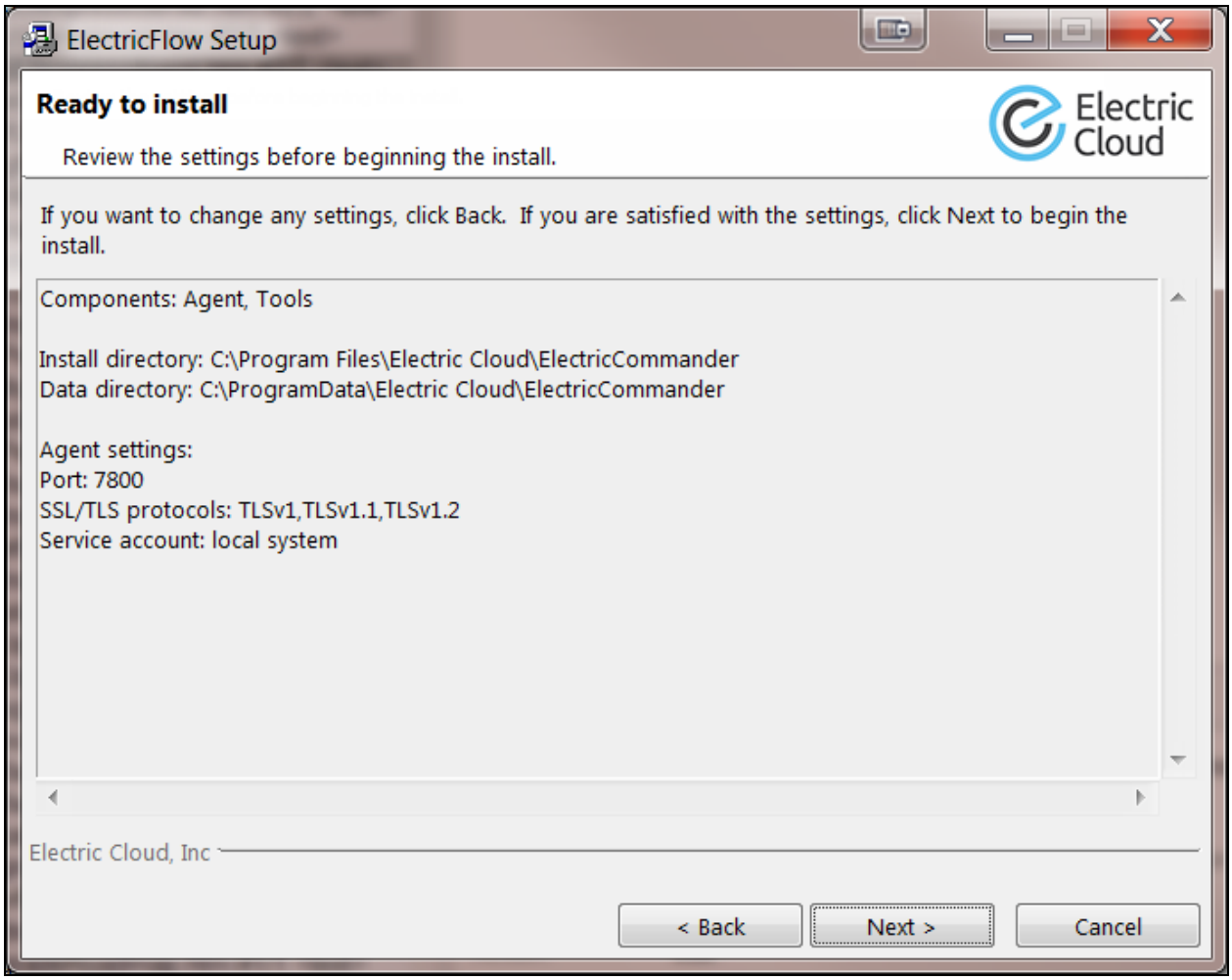
- On Windows:
  - **User Name**—Use this field to enter the name of the user who will run the ElectricFlow agent service.

    The user that the agent runs as must have permission to write to the `$INSTALL_DIRECTORY/log` directory.
  - **Password**— Use this field to enter the password of the user who will run the ElectricFlow agent service.
  - **Domain**—Use this field to enter the domain name information for the user. For example, electric-cloud.com. Leave this field blank if this is a local user.
  - **Use the local system account**—Select this check box if you want the ElectricFlow agent service to run as the Windows local system account.

    > **Note:**
    > The Windows local system account cannot access network resources such as shared file systems used for plugins or workspaces. Therefore, do not use this option for a clustered server deployment, which requires a shared file system for plugins. This option is typically used only for installing agents on numerous machines, which would otherwise require that you create a new account on each of those machines.

9. Click **Next** to continue.

   The **Ready to Install** screen appears:

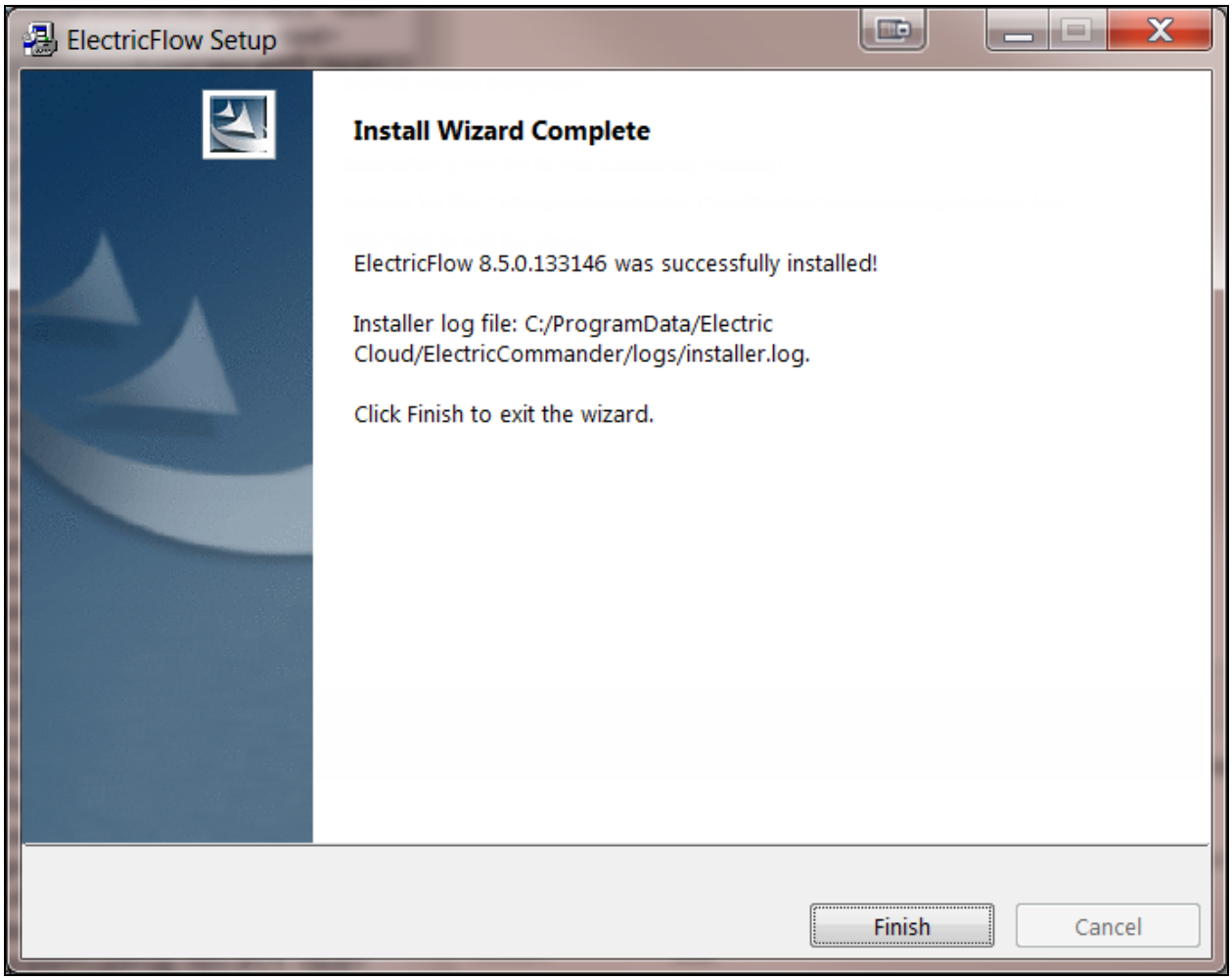


10. Review your selections.

    Use the **Back** button to change settings if necessary.

11. Click **Next** to continue.

    ElectricFlow installs the agent and tools components. This process can take a few minutes. **The Installation Wizard Complete** screen appears:



12. Click **Finish** to complete the installation.
13. For non-root/non-`sudo` Linux installations, configure autostart for the ElectricFlow agent service.

    For instructions, see Configuring Services Autostart for Non-Root/Non-sudo Linux Installations on page 5-11.

Running an Advanced Agent Graphical User Interface Installation (Agent-Only Installer)

The agent software must be installed on each machine you intend to use with ElectricFlow. An agent is an ElectricFlow component that runs on a machine resource. The agent executes ElectricFlow job steps, monitors step progress, and records job completion information.

Certain ElectricFlow installers allow you to perform installations as a non-root user or a user without `sudo` privileges. To determine whether a particular installer has an option to run in this mode, see Availability of Installers with a Non-Root/Non-sudo or Non-Administrator Mode on page 3-2.

Review Before You Install ElectricFlow on page 3-7 before performing this procedure.

Note: You install ElectricFlow agent software on Windows or Linux with this installation method. For Solaris, HP-UX, macOS, AIX, or other supported UNIX agent-only machines, see Non-Server Platform Installation Method for UNIX Agents on page 14-30.

1. Download the appropriate agent-only installer file.

   For details, see ElectricFlow Installer Files on page 3-1.

2. (Linux only) Enter the following command to make the installer file executable:

   ```
   chmod +x <agent_installer_file>
   ```

   For example, enter:

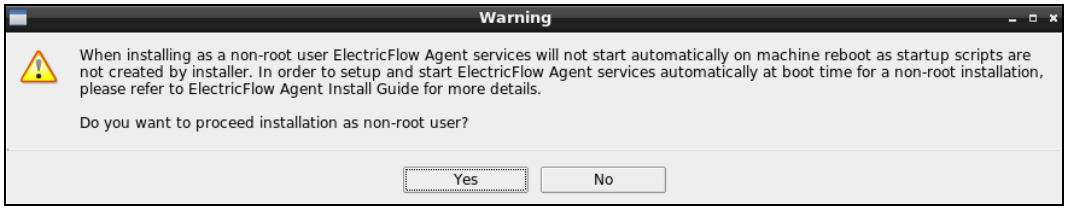
   ```
   chmod +x ElectricFlowAgent-x64-8.4.0.129860-new-with-64bit-perl
   ```

3. Do one of the following to start the installation:

   - For Linux with root or `sudo` privileges or for Windows installations, double-click the installer file.
   - For non-root/non-`sudo` installations, enter:

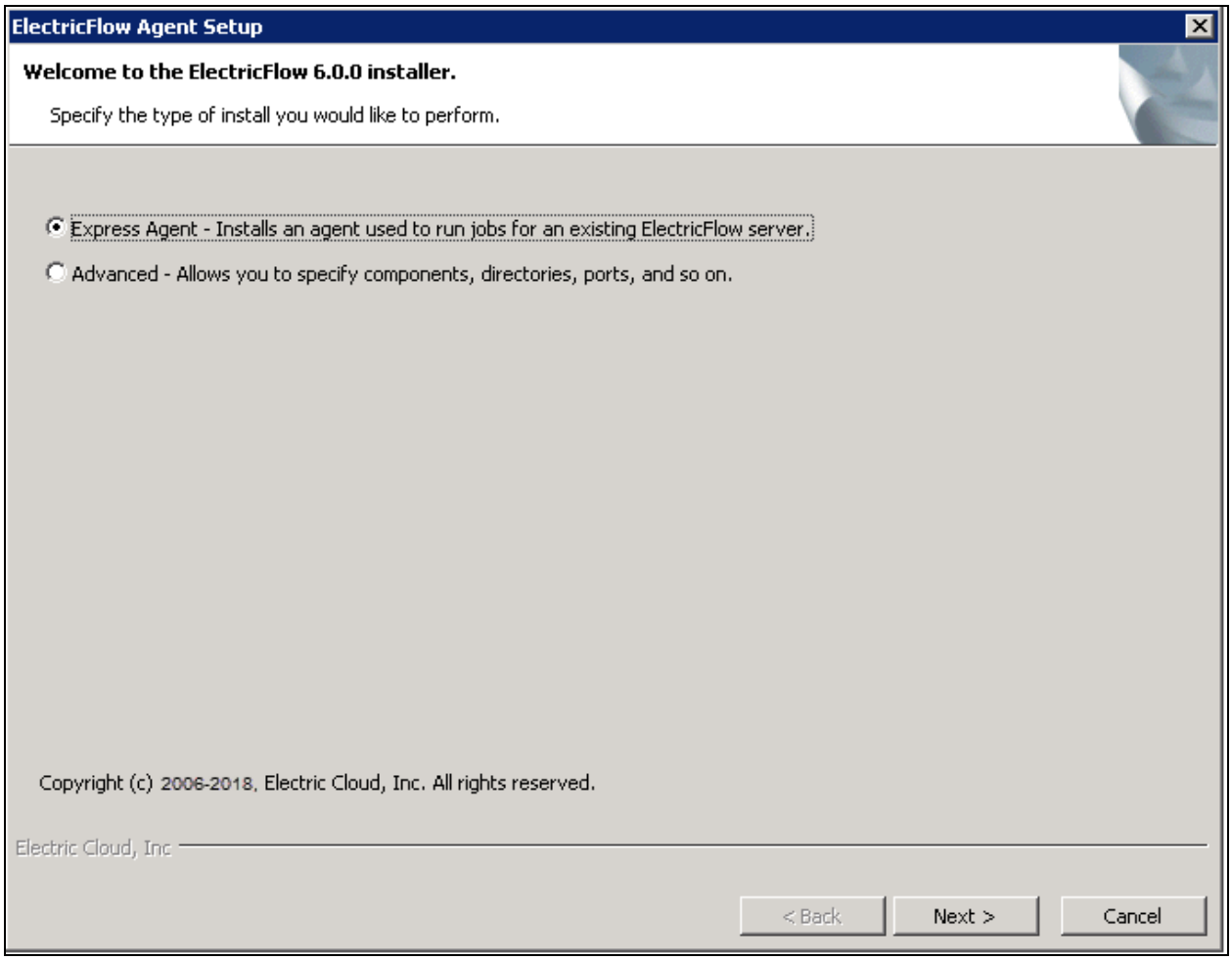
     ```
     ./<agent_installer_file> --nonRoot
     ```

     For this installation type, the following warning appears:

4. For non-root/non-`sudo` installations, click **Yes** to dismiss the warning.

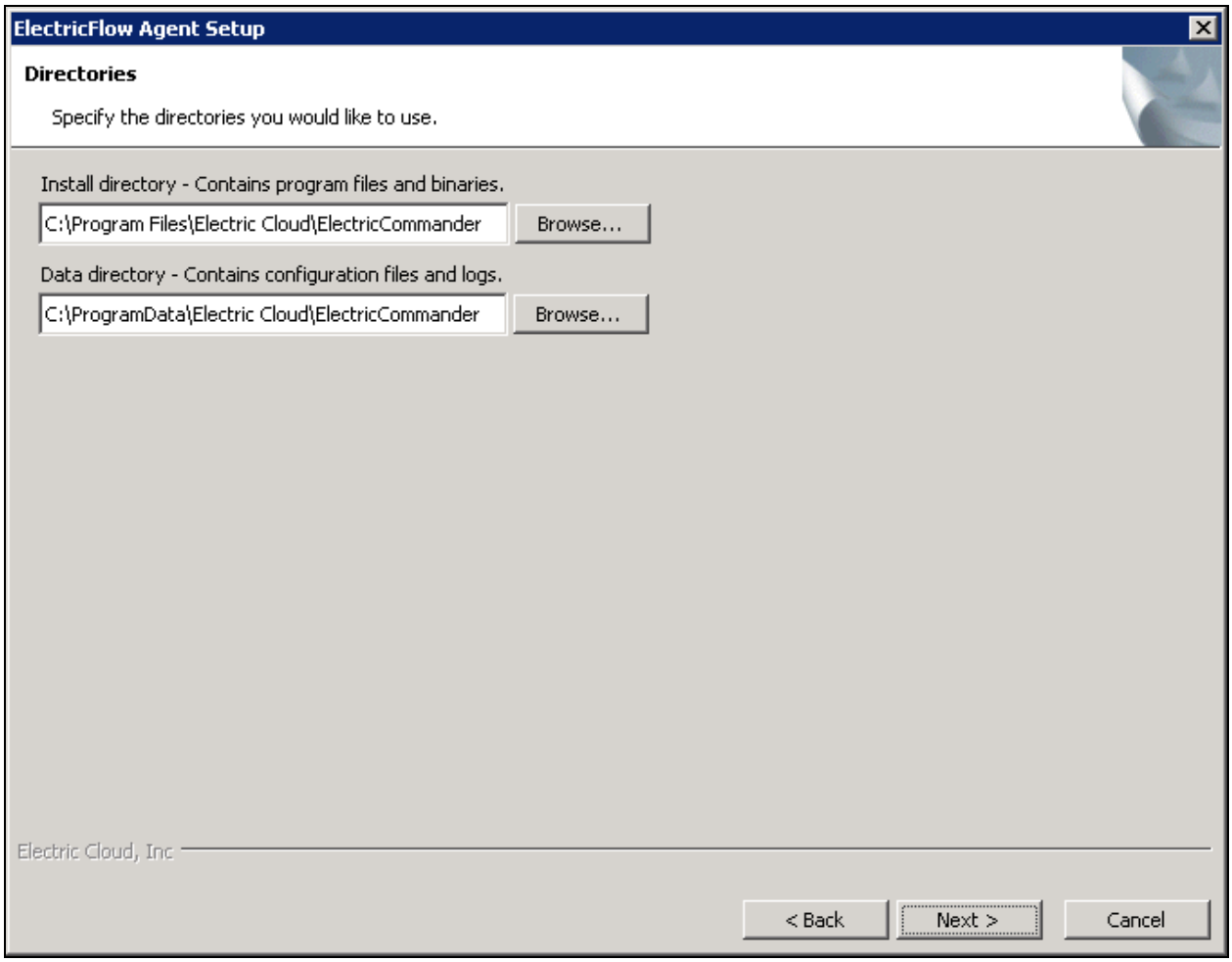
   The **Welcome to the ElectricFlow Installer** screen appears:



Note: Different options might appear depending on the operating system.

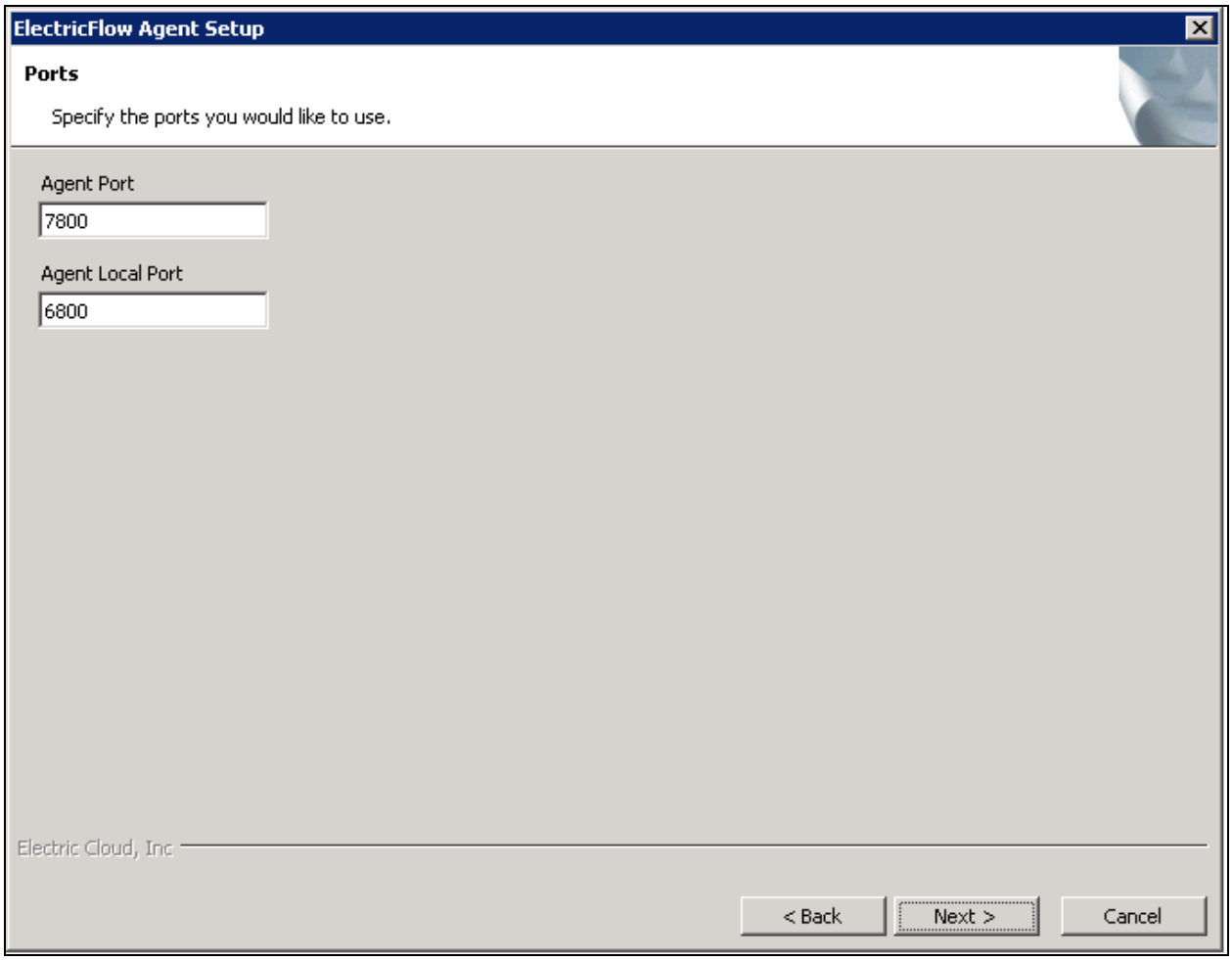
5. Select the **Advanced Agent**installation option, and then click **Next** to continue.

   The **Directories** screen appears:



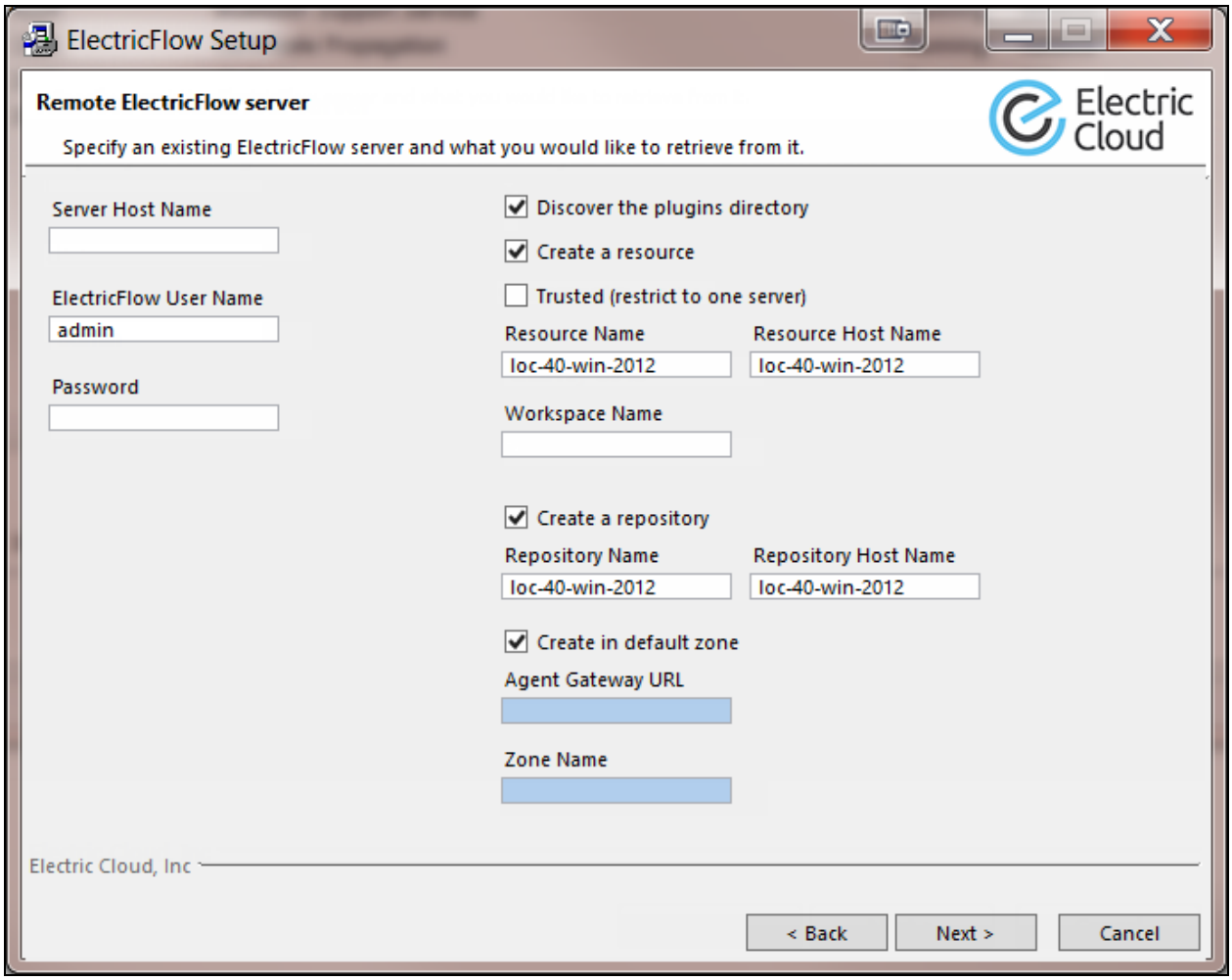
6. Complete the following information on the **Directories** screen:
   - **Install directory**—Use this field to enter a new installation directory path for program files and binaries.
   - **Data directory**—Use this field to enter a new installation directory path for configuration files and logs.
7. Click **Next** to continue.

   The **Ports** screen appears:



8. Complete the following information on the **Ports** screen:
   - **Agent port**—Use this field to specify a different port to eliminate any conflicts with your existing system configuration.
   - **Agent local port**—Use this field to specify a different port to be used by the agent for HTTP communication on the localhost network interface.
9. Click **Next** to continue.

   The Remote ElectricFlow Server screen appears:



10. Complete the following information on the Remote ElectricFlow Server screen:

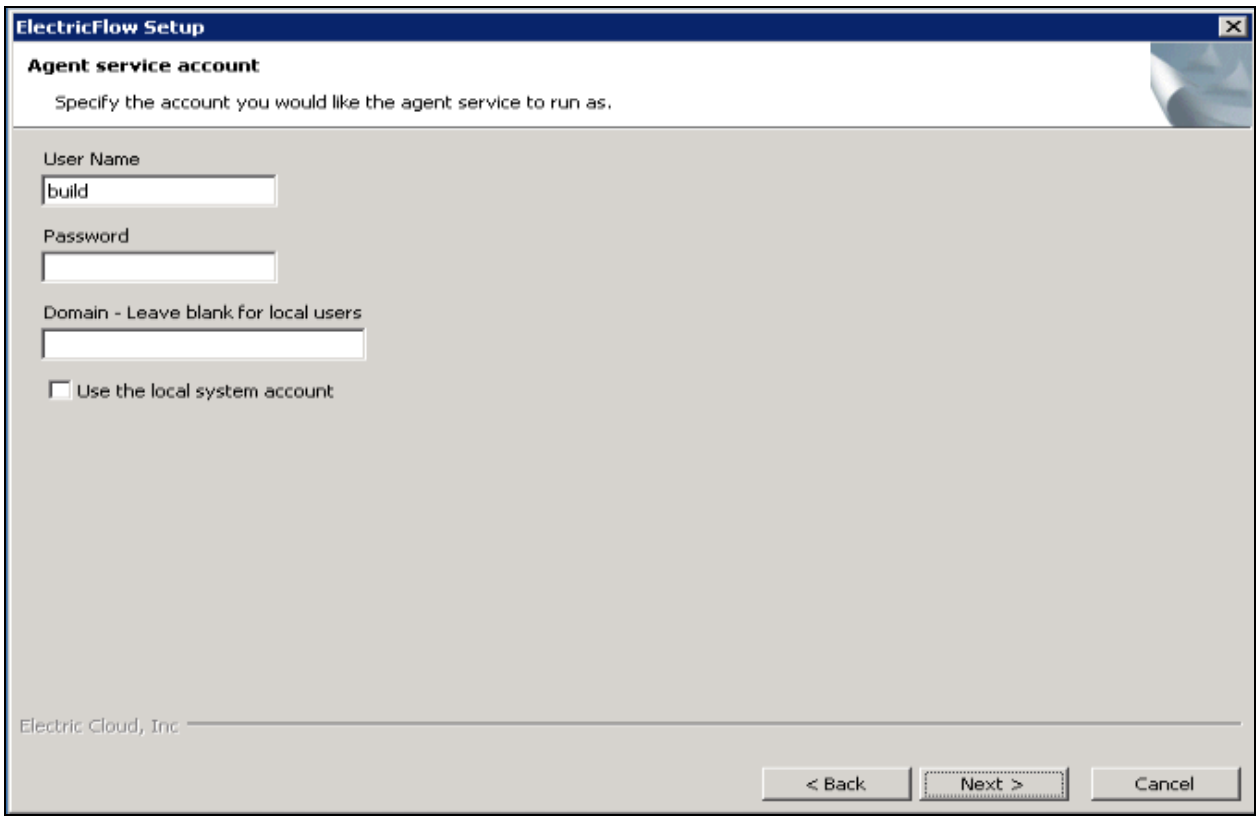
    - **Server Host Name**—Use this field to enter the name of the ElectricFlow server that will communicate with this agent. If the remote server is using a non-default HTTPS port, you must specify the Server Host Name as *<host>*:*<port>*. If you do not specify a port, HTTPS port 8443 is assumed (the same as the ElectricFlow server default port).
    - **ElectricFlow User Name**—Use this field to enter the name of an ElectricFlow user on the ElectricFlow server who has sufficient privileges to create a resource. This field defaults to the ElectricFlow-supplied `admin` user.
    - **Password**—Use this field to enter the password for the ElectricFlow user. The default password for the `admin` user is `changeme`.
    - **Discover the plugins directory**—Select this check box if you want the agent machine to have access to the plugins directory. You should allow access to the plugins directory so agents have access to collections of features, third-party integrations, or third-party tools.

> **Note:** The plugins directory on the ElectricFlow server must be "shared" before the agent machine can use "discover" to find the directory. For more information, see Universal Access to the Plugins Directory on page 5-21

- **Create a resource**—Select this check box if you want to create a resource on the remote ElectricFlow server for the agent you are installing.
- **Trusted**—Select this check box to restrict this agent to one ElectricFlow server. The agent will not respond to incoming communication from any other ElectricFlow server. This is useful when you want to create a secure production environment, but generally not needed for test or development systems.
- **Resource Name**—Use this field to enter the name of the resource you would like to use for the agent. This field is available for use when the Create a resource check box is selected.
- **Create in default zone**—Select this check box if you want to create the agent in the default zone.
- **Agent Gateway URL**—Use this field to enter the URL of the gateway used to communicate with the ElectricFlow server. This field is available for use when the Create in default zone check box is cleared.
- **Zone Name**—Use this field to enter the name of the zone used during remote agent and or remote repository creation. This field is available for use when the Create in default zone check box is cleared.

11. Click **Next** to continue.

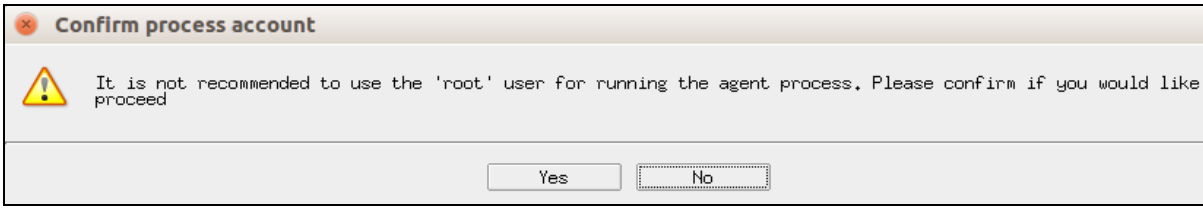
    The **Agent Service Account** screen appears:



12. Select the appropriate steps for your platform and complete the following information on the screen:

    - On Linux root or `sudo` installations:
      - **User Name**—Use this field to enter the name of the user who owns the ElectricFlow agent process.

        The user/group that the agent runs as must have permission to write to the `$INSTALL_DIRECTORY/log` directory. If you specify `root`, click **Yes** when the following confirmation appears:



- **Group Name**—Use this field to enter the name of the group who owns the ElectricFlow agent process.

- Windows systems:
  - **User Name**—Use this field to enter the name of the user who will run the ElectricFlow agent service.

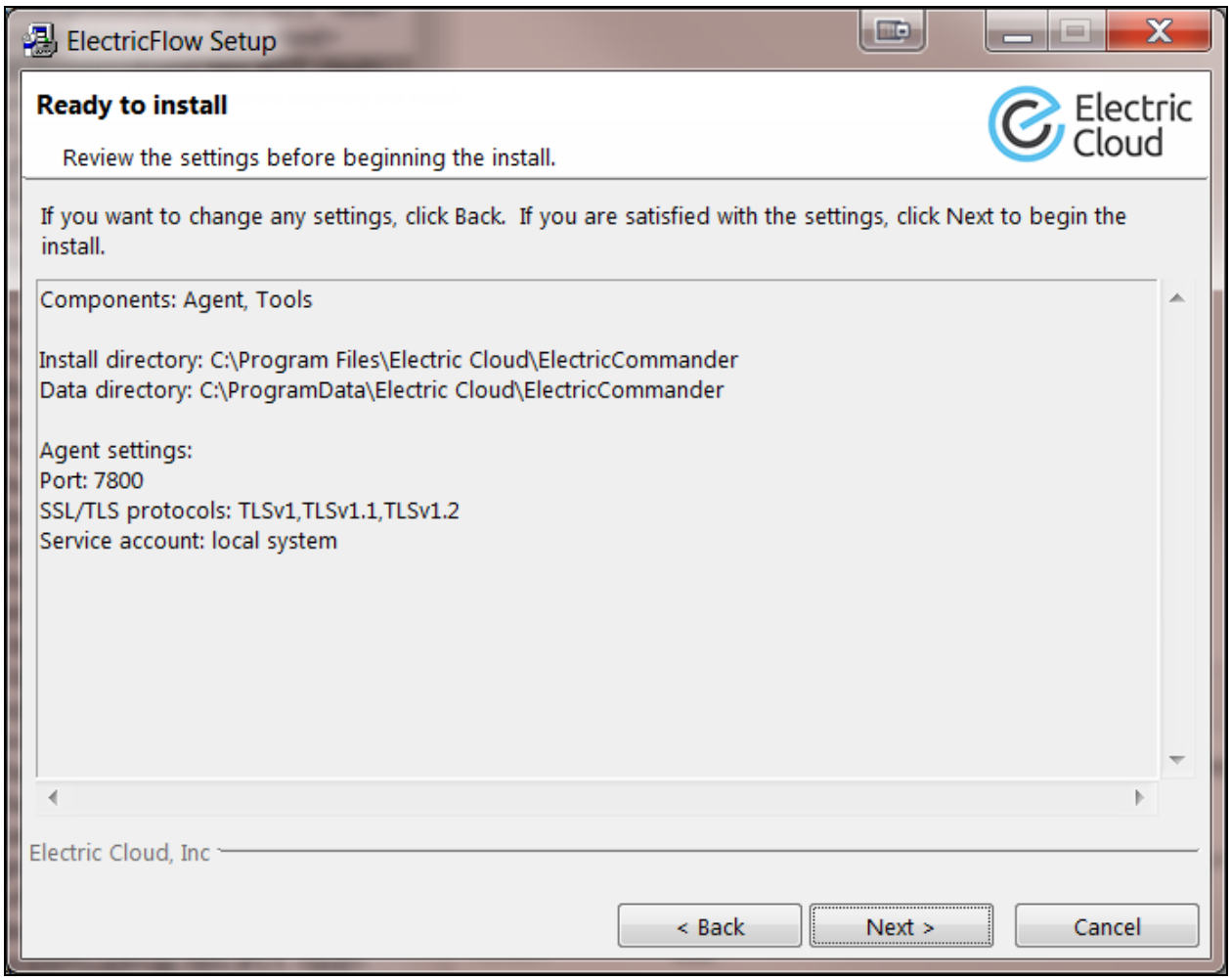
    The user that the agent runs as must have permission to write to the `$INSTALL_DIRECTORY/log` directory.
  - **Password**— Use this field to enter the password of the user who will run the ElectricFlow agent service.
  - **Domain**—Use this field to enter the domain name information for the user. For example, electric-cloud.com. Leave this field blank if this is a local user.
  - **Use the local system account**—Select this check box if you want the ElectricFlow agent service to run as the Windows local system account.

    > **Note:**
    > The Windows local system account cannot access network resources such as shared file systems used for plugins or workspaces. Therefore, do not use this option for a clustered server deployment, which requires a shared file system for plugins. This option is typically used only for installing agents on numerous machines, which would otherwise require that you create a new account on each of those machines.

13. Select the appropriate steps for your platform and complete the information on the screen.

14. Click **Next** to continue.

    The **Ready to Install Screen** appears:

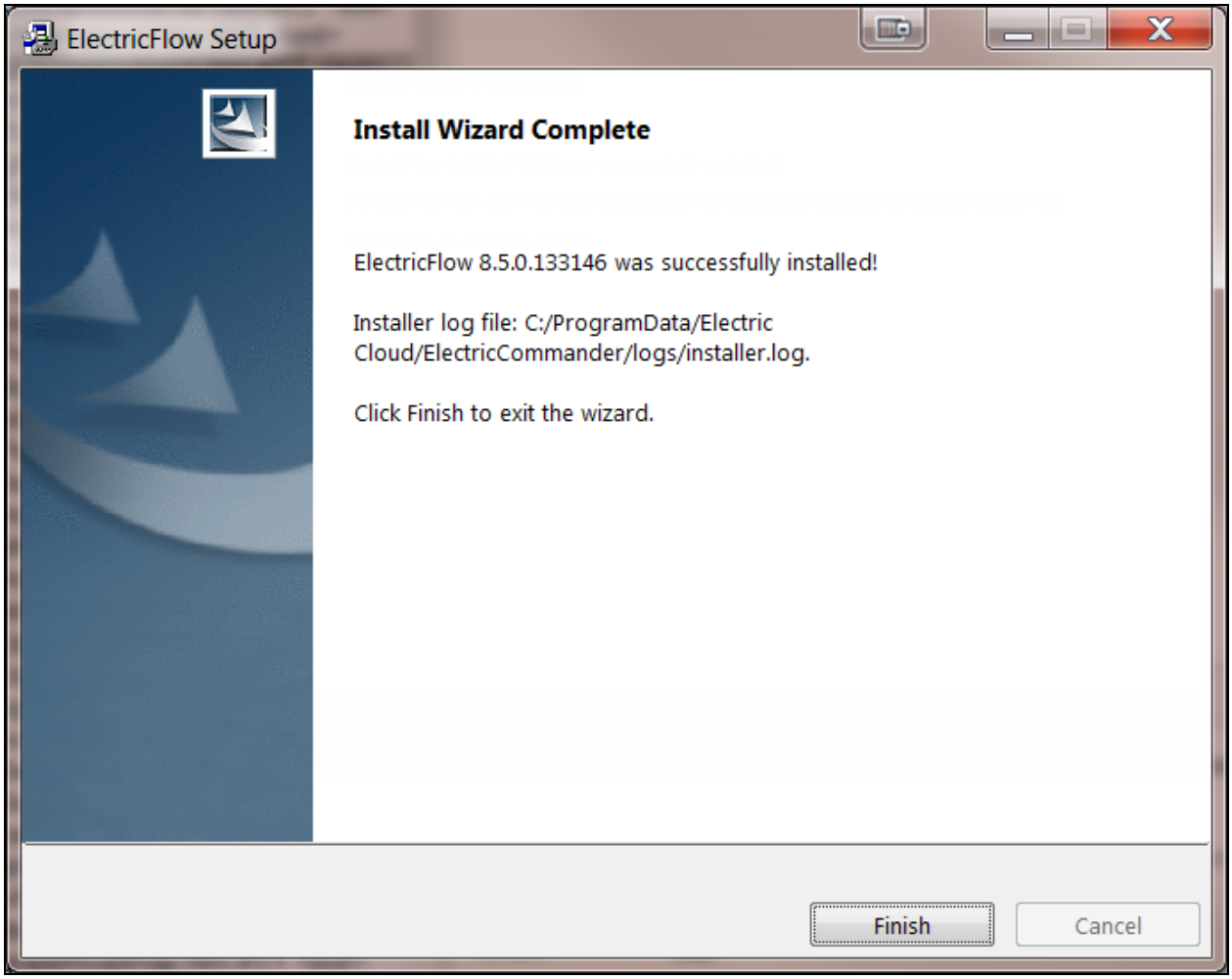


15. Verify your selections.

    Use the **Back** button to change settings if needed.

16. Click **Next** to continue.

    ElectricFlow installs the agent and tools components. This process can take a few minutes. **The Installation Wizard Complete** screen appears:



17. Click **Finish** to complete the installation.
18. For non-root/non-`sudo` Linux installations, configure autostart for the ElectricFlow agent service.

    For instructions, see Configuring Services Autostart for Non-Root/Non-sudo Linux Installations on page 5-11.

Interactive Command-Line Installation Methods

The interactive command-line installation methods are supported only for Linux-only installations on a local Linux volume. Electric Cloud does not support installing the ElectricFlow server on a network volume.

Note: You install ElectricFlow agent software on Linux with this installation method. For Solaris, HP-UX, macOS, AIX, or other supported UNIX agent machines, see Non-Server Platform Installation Method for UNIX Agents on page 14-30.

Running an Express Agent Command-Line Installation

The agent software must be installed on each machine you intend to use with ElectricFlow. An agent is an ElectricFlow component that runs on a machine resource. The agent executes ElectricFlow job steps, monitors step progress, and records job completion information.

Review Before You Install ElectricFlow on page 3-7 before performing this procedure.

1. Enter the following command to make the installer file executable:

   ```
   chmod +x ./ElectricFlow-<version>
   ```

2. Choose one of the following commands to begin the upgrade:

   - If you have a Linux platform, enter `./ElectricFlow-<version>`.
   - For installations with root or `sudo` privileges and the X Window System, override the installer GUI by entering:

     ```
     ./<agent_installer_file> --mode console
     ```

   The following prompt appears:

   ```
   Copyright (c) 2006-2018, Electric Cloud, Inc. All rights reserved.

   This will install ElectricFlow on your computer. Continue? [n/Y]
   ```

3. Continue the installation by entering `y`.

   The following prompt appears:

   ```
   Specify the type of setup you would like to perform: expressServer,
   expressAgent, or advanced. [expressServer]
   ```

4. Enter: `expressAgent`.

   The following prompt appears:

   ```
   Discover the plugins directory from a remote ElectricFlow server? [n/Y]
   ```

5. Enter `y` if you want the agent machine to have access to the plugins directory. You should allow access to the plugins directory so agents have access to collections of features, third-party integrations, or third-party tools.

   Important: The plugins directory on the ElectricFlow server must be "shared" before the agent machine can use "discover" to find the directory. For more information, see Universal Access to the Plugins Directory on page 5-21.

   The following prompt appears:

   ```
   Create a resource for the installed agent on a remote ElectricFlow server? [n/Y]
   ```

6. Enter `y` to automatically create a resource object for the agent on a remote ElectricFlow server. This option is recommended to save time configuring new ElectricFlow resources for **pre-existing** ElectricFlow servers.

   The following prompt appears:

   ```
   Register as trusted agent (required for gateway)? [y/N]
   ```

   > **Note:** Making an agent trusted restricts the agent to one ElectricFlow server. The agent will not respond to incoming communication from any other ElectricFlow server. This is useful when you want to create a secure production environment, but generally not needed for test or development systems.

   > **Important:**
   > You can run gateways without trusted agents. However, you should use gateways with trusted agents to prevent security issues in the firewall between zones connected by a gateway.
   >
   > There are exceptions to using gateways without trusted agents:
   >
   > - The firewall between two zones is not required in your environment or is needed only to protect the ElectricFlow server.
   > - There is a specific reason to use gateways without trusted agents, such as a requirement to prevent unauthorized users from accessing your network. All incoming traffic from the internet is routed to a data center through a load balancer, and the load balancer routes the traffic to the appropriate machine in your network.

7. Choose one of the following options:

   - If a gateway used to communicate with the ElectricFlow server, you must select `y`. This option allows you to create a trusted network connection between the agent and server under the same certificate authority. This will allow the agent and the ElectricFlow server to communicate across the network.
   - If there is no gateway between the agent and ElectricFlow server, enter `n`.

   > **Important:** If you deviated from the recommended agent options, you will see variations in the installation options that appear on your system.

   The following prompt appears:

   ```
   Create repository and/or agent in the default zone? [n/Y]
   ```

8. Enter `y` to create the agent in the default zone.

   The following prompt appears:

   ```
   Specify the hostName:port of a remote ElectricFlow server the agent, repository
   server and/or web server being installed can link to. The port is only required
   if it is not the default. [] <hostName:port>
   ```

9. Enter the Server Host Name of the ElectricFlow server that will communicate with this agent. You must specify the Server Host Name as `<hostName>:>port>` if the remote server is using a non-default HTTPS port. If you do not specify a port, HTTPS port 8443 is assumed (the same as the ElectricFlow server default port).

   The following prompt appears:

   ```
   Specify the user name with which to login to <hostName>:<port>. [admin]
   ```

10. Enter the user name of a user on the ElectricFlow server who has sufficient privileges to create a resource. The default is the ElectricFlow-supplied `admin` user.

    The following prompt appears:

    ```
    Specify the password for "<electricflow_user>" on <hostName>:<port>. []
    ```

11. Enter the password for the ElectricFlow user. The default password for the `admin` user is `changeme`.

    The following prompt appears:

    ```
    Specify the name of the resource to create on <<hostName>:<port>. [<resource_
    name>]
    ```

12. Enter the following information if the agent must be registered as a trusted agent. These options only appear if you entered `y` for `Register as trusted agent (required for gateway)? [y/N]`.

    1. Enter a resource name to use on the ElectricFlow server.

       The following prompt appears:

       ```
       Specify the agent gateway URL in the form of 'ipOrHostname:port' []
       ```

    2. Enter an agent gateway URL. This is the URL of the gateway used to communicate with the ElectricFlow server.

       The following prompt appears:

       ```
       Specify the zone name for the agent and/or repository []
       ```

    3. Enter the Zone Name. This is the zone used during remote agent and or remote repository creation.

       The following prompt appears:

       ```
       Specify the user the agent will run as. []
       ```

    4. Enter a user name. This is the user who owns the ElectricFlow agent process. For example, you might enter `build`.

13. The user/group that the agent runs as must have permission to write to the `$INSTALL_DIRECTORY/log` directory. If you specify `root`, enter `y` when the following confirmation appears:

    ```
    It is not recommended to use the 'root' user for running the agent process.
    Please confirm if you would like to proceed [y/N]
    ```

    The following prompt appears:

    ```
    Specify the group the agent will run as. []
    ```

14. Enter a Group Name. This is the group that owns the ElectricFlow agent process. For example, you might enter `build`.

    ElectricFlow is installed on the machine. When the installation completes successfully, a prompt that contains the line "`ElectricFlow <version> was successfully installed!`" appears.

Running an Express Agent Command-Line Installation (Agent-Only Installer)

The agent software must be installed on each machine you intend to use with ElectricFlow. An agent is an ElectricFlow component that runs on a machine resource. The agent executes ElectricFlow job steps, monitors step progress, and records job completion information.

Certain ElectricFlow installers allow you to perform installations as a non-root user or a user without `sudo` privileges. To determine whether a particular installer has an option to run in this mode, see Availability of Installers with a Non-Root/Non-sudo or Non-Administrator Mode on page 3-2.

Review Before You Install ElectricFlow on page 3-7 before performing this procedure.

1. Download the appropriate agent-only installer file.

   For details, see ElectricFlow Installer Files on page 3-1.

2. Enter the following command to make the installer file executable:

   ```
   chmod +x <agent_installer_file>
   ```

   For example, enter:

   ```
   chmod +x ElectricFlowAgent-x64-8.4.0.129860-new-with-64bit-perl
   ```

3. Choose one of the following commands to begin the installation:

   - For installations with root or `sudo` privileges, enter:

     ```
     ./<agent_installer_file>
     ```

   - For installations with root or `sudo` privileges and the X Window System, override the installer GUI by entering:

     ```
     ./<agent_installer_file> --mode console
     ```

   - For non-root/non-`sudo` installations, enter:

     ```
     ./<agent_installer_file> --mode console --nonRoot
     ```

4. After the confirmation prompt, continue the installation by entering `y`.

   The following prompt appears:

   ```
   Specify the type of setup you would like to perform: expressAgent or advanced.
   [expressAgent]
   ```

5. Press `Enter` to accept `expressAgent`.

   The following prompt appears:

   ```
   Discover the plugins directory from a remote ElectricFlow server? [n/Y]
   ```

6. Enter `y` if you want the agent machine to have access to the plugins directory. You should allow access to the plugins directory so agents have access to collections of features, third-party

integrations, or third-party tools.

> **Important:** The plugins directory on the ElectricFlow server must be "shared" before the agent machine can use "discover" to find the directory. For more information, see Universal Access to the Plugins Directory on page 5-21.

The following prompt appears:

```
Create a resource for the installed agent on a remote ElectricFlow server? [n/Y]
```

7. Enter `y` to automatically create a resource object for the agent on a remote ElectricFlow server. This option is recommended to save time configuring new ElectricFlow resources for existing ElectricFlow servers.

   The following prompt appears:

   ```
   Register as trusted agent? [y/N]
   ```

   Making an agent trusted restricts the agent to one ElectricFlow server. The agent will not respond to incoming communication from any other ElectricFlow server. This is useful when you want to create a secure production environment, but generally not needed for test or development systems.

   > **Important:**
   > You can run gateways without trusted agents. However, you should use gateways with trusted agents to prevent security issues in the firewall between zones connected by a gateway.
   >
   > There are exceptions to using gateways without trusted agents:
   >
   > - The firewall between two zones is not required in your environment or is needed only to protect the ElectricFlow server.
   > - There is a specific reason to use gateways without trusted agents, such as a requirement to prevent unauthorized users from accessing your network. All incoming traffic from the internet is routed to a data center through a load balancer, and the load balancer routes the traffic to the appropriate machine in your network.

8. Choose one of the following options:

   - If a gateway is used to communicate with the ElectricFlow server, you must select `y`. This option allows you to create a trusted network connection between the agent and server under the same certificate authority. This will allow the agent and the ElectricFlow server to communicate across the network.
   - If there is no gateway between the agent and ElectricFlow server, enter `n`.

     > **Note:** If you deviated from the recommended agent options, you will see variations in the installation options that appear on your system.

     For root or `sudo` installations, The following prompt appears:

     ```
     Specify the user the agent will run as. []
     ```

9. (Root or `sudo` installations) Enter a user name. This is the user who owns the ElectricFlow agent process. For example, you might enter `build`.

   The user/group that the agent runs as must have permission to write to the `$INSTALL_DIRECTORY/log` directory. If you specify `root`, enter `y` when the following confirmation appears:

   ```
   It is not recommended to use the 'root' user for running the agent process.
   Please confirm if you would like to proceed [y/N]
   ```

   The following prompt appears:

   ```
   Specify the group the agent will run as. []
   ```

10. (Root or `sudo` installations) Enter a Group Name. This is the group that owns the ElectricFlow agent process. For example, you might enter `build`.

    ElectricFlow is installed on the machine. When the installation completes successfully, a prompt that contains the line "`ElectricFlow <version> was successfully installed!`" appears.

11. For non-root/non-`sudo` Linux installations, configure autostart for the ElectricFlow agent service.

    For instructions, see Configuring Services Autostart for Non-Root/Non-sudo Linux Installations on page 5-11.

Running an Advanced Agent Command-Line Installation (Agent-Only Installer)

The agent software must be installed on each machine you intend to use with ElectricFlow. An agent is an ElectricFlow component that runs on a machine resource. The agent executes ElectricFlow job steps, monitors step progress, and records job completion information.

Certain ElectricFlow installers allow you to perform installations as a non-root user or a user without `sudo` privileges. To determine whether a particular installer has an option to run in this mode, see Availability of Installers with a Non-Root/Non-sudo or Non-Administrator Mode on page 3-2.

Review Before You Install ElectricFlow on page 3-7 before performing this procedure.

1. Download the appropriate agent-only installer file.

   For details, see ElectricFlow Installer Files on page 3-1.

2. Enter the following command to make the installer file executable:

   ```
   chmod +x <agent_installer_file>
   ```

   For example, enter:

   ```
   chmod +x ElectricFlowAgent-x64-8.4.0.129860-new-with-64bit-perl
   ```

3. Choose one of the following commands to begin the upgrade:
   - For installations with root or `sudo` privileges, enter:

     `./<agent_installer_file>`
   - For installations with root or `sudo` privileges and the X Window System, override the installer GUI by entering:

     `./<agent_installer_file> --mode console`
   - For non-root/non-`sudo` installations, enter:

     `./<agent_installer_file> --mode console --nonRoot`
4. After the confirmation prompt, continue the installation by entering `y`.

   The following prompt appears:

   ```
   Specify the type of setup you would like to perform: expressAgent or advanced.
   [expressAgent]
   ```
5. Enter `advanced`.

   The following prompt appears:

   ```
   Specify the install directory (for program files and binaries).
   [/opt/electriccloud/electriccommander]
   ```
6. Enter a new installation directory path for program files and binaries.

   The following prompt appears:

   ```
   Specify the data directory (for configuration files and logs).
   [/opt/electriccloud/electriccommander]
   ```
7. Enter a new installation directory path for configuration files and logs.

   The following prompt appears:

   ```
   Specify the agent port. [7800]
   ```
8. Enter a different port to eliminate any conflicts with your existing system configuration.

   The following prompt appears:

   ```
   Specify the agent local port. [6800]
   ```
9. Enter a different port to be used by the agent for HTTP communication on the localhost network interface.

   The following prompt appears:

   ```
   Discover the plugins directory from a remote ElectricFlow server? [n/Y]
   ```
10. Enter `y` if you want the agent machine to have access to the plugins directory. You should allow access to the plugins directory so agents have access to collections of features, third-party integrations, or third-party tools.

    Note: The plugins directory on the ElectricFlow server must be "shared" before the agent machine can use "discover" to find the directory. For more information, see Universal Access to the Plugins Directory on page 5-21.

The following prompt appears:

```
Create a resource for the installed agent on a remote ElectricFlow server? [n/Y]
```

11. Enter `y` to automatically create a resource object for the agent on a remote ElectricFlow server. This option is recommended to save time configuring new ElectricFlow resources for existing ElectricFlow servers.

    The following prompt appears:

    ```
    Register as trusted agent? [y/N]
    ```

    Making an agent trusted restricts the agent to one ElectricFlow server. The agent will not respond to incoming communication from any other ElectricFlow server. This is useful when you want to create a secure production environment, but generally not needed for test or development environments.

    > **Important:**
    > You can run gateways without trusted agents. However, you should use gateways with trusted agents to prevent security issues in the firewall between zones connected by a gateway.
    >
    > There are exceptions to using gateways without trusted agents:
    >
    > - The firewall between two zones is not required in your environment or is needed only to protect the ElectricFlow server.
    > - There is a specific reason to use gateways without trusted agents, such as a requirement to prevent unauthorized users from accessing your network. All incoming traffic from the internet is routed to a data center through a load balancer, and the load balancer routes the traffic to the appropriate machine in your network.

12. Choose one of the following options:

    - If a gateway is used to communicate with the ElectricFlow server, you must select `y`. This option allows you to create a trusted network connection between the agent and server under the same certificate authority. This will allow the agent and the ElectricFlow server to communicate across the network.
    - If there is no gateway between the agent and ElectricFlow server, enter `n`.

    > **Important:** If you deviated from the recommended agent options, you will see variations in the installation options that appear on your system.

    For root or `sudo` installations, The following prompt appears:

    ```
    Specify the user the agent will run as. []
    ```

13. (Root or `sudo` installations) Enter a user name. This is the user who owns the ElectricFlow agent process. For example, you might enter `build`.

    The user/group that the agent runs as must have permission to write to the `$INSTALL_DIRECTORY/log` directory. If you specify `root`, enter `y` when the following confirmation appears:

    ```
    It is not recommended to use the 'root' user for running the agent process.
    Please confirm if you would like to proceed [y/N]
    ```

The following prompt appears:

```
Specify the group the agent will run as. []
```

14. (Root or `sudo` installations) Enter a Group Name. This is the group that owns the ElectricFlow agent process. For example, you might enter `build`.

    ElectricFlow is installed on the machine. When the installation completes successfully, a prompt that contains the line "`ElectricFlow <version> was successfully installed!`" appears.

15. For non-root/non-`sudo` Linux installations, configure autostart for the ElectricFlow agent service.

    For instructions, see Configuring Services Autostart for Non-Root/Non-sudo Linux Installations on page 5-11.

Running an Express Agent Command-Line Installation (Agent-Only Installer) When the Server Uses Registered and Concurrent Licenses

Use this procedure when the ElectricFlow server uses a mix of registered and concurrent licenses.

The agent software must be installed on each machine you intend to use with ElectricFlow. An agent is an ElectricFlow component that runs on a machine resource. The agent executes ElectricFlow job steps, monitors step progress, and records job completion information.

Certain ElectricFlow installers allow you to perform installations as a non-root user or a user without `sudo` privileges. To determine whether a particular installer has an option to run in this mode, see Availability of Installers with a Non-Root/Non-sudo or Non-Administrator Mode on page 3-2.

Review Before You Install ElectricFlow on page 3-7 before performing this procedure.

1. Download the appropriate agent-only installer file.

   For details, see ElectricFlow Installer Files on page 3-1.

2. Enter the following command to make the installer file executable:

   ```
   chmod +x <agent_installer_file>
   ```

   For example, enter:

   ```
   chmod +x ElectricFlowAgent-x64-8.4.0.129860-new-with-64bit-perl
   ```

3. Choose one of the following commands to begin the upgrade:

   - For installations with root or `sudo` privileges, enter:

     ```
     ./<agent_installer_file>
     ```

   - For installations with root or `sudo` privileges and the X Window System, override the installer GUI by entering:

     ```
     ./<agent_installer_file> --mode console
     ```

   - For non-root/non-`sudo` installations, enter:

     ```
     ./<agent_installer_file> --mode console --nonRoot
     ```

4. After the confirmation prompt, enter `y` to continue the installation.

   The following prompt appears:

```
Specify the type of setup you would like to perform: expressAgent or advanced.
[expressAgent]
```

5. Press `Enter` to accept `expressAgent`.

   The following prompt appears:

   ```
   Discover the plugins directory from a remote ElectricFlow server? [n/Y]
   ```

6. Enter `y` if you want the agent machine to have access to the plugins directory. You should allow access to the plugins directory so agents have access to collections of features, third-party integrations, or third-party tools.

   > **Note:** The plugins directory on the ElectricFlow server must be "shared" before the agent machine can use "discover" to find the directory. For more information, see Universal Access to the Plugins Directory on page 5-21.

   The following prompt appears:

   ```
   Create a resource for the installed agent on a remote ElectricFlow server? [n/Y]
   ```

7. Enter `y` to automatically create a resource object for the agent on a remote ElectricFlow server. This option is recommended to save time configuring new ElectricFlow resources for *pre-existing* ElectricFlow servers.

   The following prompt appears:

   ```
   Register as trusted agent? [y/N]
   ```

   Making an agent trusted restricts the agent to one ElectricFlow server. The agent will not respond to incoming communication from any other ElectricFlow server. This is useful when you want to create a secure production environment, but generally not needed for test or development environments.

   > **Important:**
   > You can run gateways without trusted agents. However, you should use gateways with trusted agents to prevent security issues in the firewall between zones connected by a gateway.
   >
   > There are exceptions to using gateways without trusted agents:
   >
   > - The firewall between two zones is not required in your environment or is needed only to protect the ElectricFlow server.
   > - There is a specific reason to use gateways without trusted agents, such as a requirement to prevent unauthorized users from accessing your network. All incoming traffic from the internet is routed to a data center through a load balancer, and the load balancer routes the traffic to the appropriate machine in your network.

8. Enter `n` if you are installing the ElectricFlow Community Edition.

   The following prompt appears:

   ```
   Create repository and/or agent in the default zone? [y/n]
   ```

9. Enter `y`.

   The following prompt appears:

   ```
   Specify the host:port of a remote ElectricFlow server that the agent,
   repository server and/or web server being installed can link to. The
   port is only required if it is not the default. []
   ```

10. Enter the `<host:port>`.

    The following prompt appears:

    ```
    Specify the user name with which to login to "<host:port>". [admin]
    ```

11. Enter `admin`.

    The following prompt appears:

    ```
    Specify the password for "admin" on "<host:port>". []
    ```

12. Enter a password.

    The following prompt appears:

    ```
    Specify the name of the resource to create on "<host:port>". []
    ```

13. Enter a resource name.

    The following prompt appears:

    ```
    Specify resource type for remote server: Registered or Concurrent. []
    ```

14. Enter `Registered`.

    For root or `sudo` installations, The following prompt appears:

    ```
    Specify the user the agent will run as. []
    ```

15. (Root or `sudo` installations) Enter a user name.

    This is the user who owns the ElectricFlow agent process. For example, you can enter `deploy`.

    The user/group that the agent runs as must have permission to write to the `$INSTALL_DIRECTORY/log` directory. If you specify `root`, enter `y` when the following confirmation appears:

    ```
    It is not recommended to use the 'root' user for running the agent process.
    Please confirm if you would like to proceed [y/N]
    ```

    The following prompt appears:

    ```
    Specify the group the agent will run as. []
    ```

16. (Root or `sudo` installations) Enter a group name.

    This is the group that owns the ElectricFlow agent process. For example, you can enter `deploy`.

    ElectricFlow is installed on the machine.

    When the installation completes successfully, a prompt that contains the line "`ElectricFlow <version> was successfully installed!`" appears.

17. For non-root/non-`sudo` Linux installations, configure autostart for the ElectricFlow agent service.

    For instructions, see Configuring Services Autostart for Non-Root/Non-sudo Linux Installations on page 5-11.

Non-Server Platform Installation Method for UNIX Agents

To install agents and tools on UNIX machines that are not supported ElectricFlow server platforms, you must use a UNIX installer file instead of the `./ElectricFlow-<version>` installer file (which works only for server installation). This file is named `commander_<OStype>.bin` and is available on the Electric Cloud FTP site. For more information about supported agent platforms, see Supported Agent Platforms on page 2-2.

Interactive Command-Line Installation Method for UNIX or macOS Agents

The agent software must be installed on each machine you intend to use with ElectricFlow. An agent is an ElectricFlow component that runs on a machine resource. The agent executes ElectricFlow job steps, monitors step progress, and records job completion information.

This section describes how to install agents and tools on UNIX (not Linux or Windows) machines. These include Solaris, HP-UX, macOS, and AIX machines. Agent upgrades are not supported on these platforms.

You can install agents using any of the following accounts:

- root
- Any account with sudo privileges
- (UNIX or macOS only) Any non-root account without sudo privileges

Installing Agents Using root or an Account with sudo Privileges

To install agents and tools on UNIX or macOS machines using root or an account with sudo privileges:

1. Obtain the UNIX or macOS installer file for your agent platform as described in Non-Server Platform Installation Method for UNIX Agents on page 14-30.
2. Log in as root.
3. Enter `chmod +x ./commander_<OStype>.bin` to ensure that the installer is executable.

   where `<OStype>` is the agent platform. For example:

   ```
   chmod +x ./commander_powerpc_AIX71.bin
   ```

4. Run `./commander_<OStype>.bin`.

   The following prompts appear:

   ```
   Checking installer integrity, please wait...
   ElectricFlow 7.2.0.116649 for AIX Installer
   Copyright 2006-2018 Electric Cloud, Inc. All rights reserved.


   Press CTRL-C to exit at any time.
   ```

```
Press Enter to accept default settings.


log file: /tmp/commander_install_20170321_115947.log
This suite installer can install several different product options.
Note: The default is to install everything.
Which products would you like to install (agent, tools):
```

5. Enter `agent` or press `Enter`.

   (You can also install the tools only by entering `Tools`.) The agent and tools will be installed. The following prompts appear:

```
Installing agent and tools.


Where would you like the software to be installed?
NOTE: The destination should NOT be an nfs filesystem.


Enter destination directory (default is /opt):
```

6. Enter the destination directory path.

   The following prompt appears:

```
Enter an existing user to own installed agent files and run agent processes:
```

7. Enter the name of the user to own the ElectricFlow agent files and run the agent processes.

   The following prompt appears:

```
Enter an existing user group to own installed agent files and run agent
processes.
Or hit Enter to choose the primary group (default is '<primary group>'):
```

8. Enter the group name of the user to own the ElectricFlow agent files and run the agent processes or press `Enter` to use the user's primary group.

   The following prompt appears:

```
Enter the agent port (default is 7800):
```

9. Accept the default port or specify a different port if needed to eliminate conflicts with your existing system configuration, and then press `Enter`.

   The installer extracts and installs the software. When the installation is complete, the following prompt appears:

```
OK: Installation successful!
```

Installing Agents Using a Non-root Account or an Account Without sudo Privileges

In this type of installation, the installer starts the agent service and runs it as the user that performed the installation.

> **Important:** Running the installer without root or sudo privileges is not recommended. When run without root or sudo privileges, the installer cannot install the files that provide automatic start for the agent services, so you must configure automatic restart manually.

To install agents and tools on UNIX or macOS machines using a non-root account without sudo privileges:

1. Log in as the user to own the installed agent files and run the agent processes.
2. Obtain the UNIX or macOS installer file for your agent platform as described in Non-Server Platform Installation Method for UNIX Agents on page 14-30.
3. Run `chmod +x ./commander_<OStype>.bin` to ensure that the installer is executable.

   <OStype> is the agent platform. For example:

   ```
   chmod +x ./commander_powerpc_AIX71.bin
   ```

4. Enter `./commander_<OStype>.bin --nonRoot` to start the installation.

   The following prompts appear:

   ```
   Checking installer integrity, please wait...
   ElectricFlow 7.2.0.116649 for AIX Installer
   Copyright 2006-2018 Electric Cloud, Inc. All rights reserved.


   Press CTRL-C to exit at any time.


   Press Enter to accept default settings.


   log file: /tmp/commander_install_20170321_115947.log
   This suite installer can install several different product options.
   Note: The default is to install everything.
   Which products would you like to install (agent, tools):
   ```

   > **Note:**
   > Failure to include the `--nonRoot` argument causes the following error:
   >
   > ```
   > This installer must be invoked in a root context.
   >
   > ERROR: Install failed. Exiting installer.
   > ```

5. Enter `agent` or press `Enter`.

   (You can also install the tools only by entering `Tools`.) The agent and tools will be installed. The following prompts appear:

```
Installing agent and tools.


Where would you like the software to be installed?
NOTE: The destination should NOT be an nfs filesystem.


Enter destination directory (default is /opt):
```

6. Enter the destination directory path.

   Note:
   If you lack sufficient privileges on the destination directory, the following error appears, and you must obtain sufficient privileges before continuing:

```
Could not create "/bin/electriccloud/electriccommander".
```

   If the directory that you entered already exists, the following prompts appear:

```
Directory "/opt/electriccloud/electriccommander" already exists.


Do you want to delete and overwrite it (Y/n)?
```

7. If the directory already exists, enter `Y` to overwrite it.

   The following prompts appear:

```
Non-root install mode. Current user 'build' will be used as owner for installed
agent files and run agent processes.

Enter an existing user group to own installed agent files and run agent
processes.
Or hit Enter to choose the primary group (default is '<primary group>'):
```

8. Enter the group name of the user to own the ElectricFlow agent files and run the agent processes or press `Enter` to use the user's primary group.

   The group that the agent runs as must have permission to write to the `$INSTALL_DIRECTORY/log` directory.

   > **Note:**
   > If you are not a member of the group, the following prompt appears, and you must enter a different group:
   >
   > ```
   > The combination of agent user 'build' and agent group 'foo' is invalid.
   > Please try again.
   >
   > Enter an existing user group to own installed agent files and run agent
   > processes.
   > Or hit Enter to choose the primary group (default is '<primary group>'):
   > ```

   After you successfully enter the group name, the following prompt appears:

   ```
   Enter the agent port (default is 7800):
   ```

9. Accept the default port or specify a different port if needed to eliminate conflicts with your existing system configuration, and then press `Enter`.

   The installer extracts and installs the software. Then the following prompts appear. Note that the directory to contain the agent services varies by platform:

   ```
   Please wait while the services are configured and started...

   Services are started automatically during configuration.

   To manually start services use following command(s):
   /opt/electriccloud/electriccommander/startup/ecmdrAgent start

   To start services at system startup,
   copy files at /opt/electriccloud/electriccommander/startup
   to the init.d directory '/etc/rc.d/init.d'
   and make corresponding links in /etc/rcX.d directories.
   ```

   When the installation is complete, the following prompt appears:

   ```
   OK: Installation successful!
   ```

Unattended (Silent) Installation Method for UNIX or macOS Agents

The agent software must be installed on each machine you intend to use with ElectricFlow. An agent is an ElectricFlow component that runs on a machine resource. The agent executes ElectricFlow job steps, monitors step progress, and records job completion information.

This section describes how to install agents and tools silently on UNIX (not Linux or Windows) machines. These include Solaris, HP-UX, macOS, and AIX machines. Agent upgrades are not supported on these platforms.

You can install agents using any of the following accounts:

- root
- Any account with sudo privileges
- (UNIX or macOS only) Any non-root account without sudo privileges

Silent Installation Command Arguments

The following table lists the available arguments.

Argument	Description
`-q`	Runs the installer in silent mode. The default installation options are used unless you override them on the command line or in an installation configuration file.
`--nonRoot`	(UNIX or macOS only) Runs the installer using a non-root account without sudo privileges. The agent service will run as the user that performed the installation.
`-f`	Removes and replaces any existing files in the destination directory. This argument completely removes the directory but does *not* uninstall the previous version. For details about upgrades, see Roadmap for Upgrading ElectricFlow on page 6-1.
`--config`	Specifies a file containing installation parameters and values.

Running a Silent Installation

Important: Running the installer without root or sudo privileges is not recommended. When run without root or sudo privileges, the installer cannot install the files that provide automatic start for the agent services, so you must configure automatic restart manually.

To run a silent UNIX or macOS agent installation:

1. Obtain the UNIX or macOS installer file for your agent platform as described in Non-Server Platform Installation Method for UNIX Agents on page 14-30.
2. If you are *not* installing as a non-root user without sudo privileges, log in as root or as a user with sudo privileges.
3. Run `chmod +x ./commander_<OStype>.bin` to ensure that the installer is executable.
4. Run `commander_<OStype>.bin -q <arguments>`.

   where <*OStype*> is the agent platform. For example:

   ```
   commander_powerpc_AIX71.bin -q -f --config myconfig
   ```

   For installation using a non-root account without sudo privileges, you must include the `--nonRoot` argument. Failure to do so causes the following error:

   ```
   This installer must be invoked in a root context.

   ERROR: Install failed. Exiting installer.
   ```

Example Parameters in an Installation Configuration File

Following is an example of parameters in a configuration file for silent installation of agents using root or an account with sudo privileges:

```
EC_INSTALL_TYPE=agent
DESTINATION_DIR="/opt"
AGENT_USER_TO_RUN_AS="bill jones"
AGENT_GROUP_TO_RUN_AS=engineering
EC_AGENT_PORT=7800
EC_AGENT_LOCAL_PORT=6800
```

Following is an example of parameters in a configuration file for silent installation of tools using root or an account with sudo privileges:

```
EC_INSTALL_TYPE=tools
DESTINATION_DIR="/opt"
USER_TO_RUN_AS=sally
GROUP_TO_RUN_AS=engineering
```