Installation and Configuration Guide

Software version: 3.2

Documentation date: January, 2018

info@bluetalon.com

BlueTalon Inc., 541 Jefferson Ave, Redwood City, CA 94063

Table of Contents

Introduction

This document is for system administrators responsible for installing and maintaining the BlueTalon® software.

- Software is available under a commercial license.
- Contact sales@bluetalon.com to obtain the software.

This document contains some security recommendations that should be followed and implemented.

New Software Features

New BlueTalon 3.2.4 software features:

- Policy Engine supports an address parameter named listening_ip, the IP address for monitoring requests. Required for installing multiple Policy Engines on Docker.
- Script bt-ambari-plugin-install.sh accepts command line arguments for silent non-interactive installation.
- Policy and Audit hosts can be dynamically discovered during the Cloudera Manager installation of BlueTalon software.
- Optimized Activity Monitor database write operations.
- Support separate PostgreSQL instances for Audit and Policy databases in Docker.
- During a new deployment, skip the requests in the Scheduler thread. This results in less network traffic.
- GetStatus call provides additional information for skipped status (referral snapshot name). New column was added to DeployTrack database table to retain the additional name of the referral snapshot name (rolling name) for skipped requests.
- Addition of deployment ID (a unique number) to the log file lines for tracking each deployment. Also, addition of the total deploy time and total snapshot time to the log for other processes.
- Extended the attribute filter handling in the Cassandra Enforcement Point for prepared statements.
- REST API data masking function updates. The API functions have updated expressions, descriptions, input types, and return types.
- Resolved checkmarx issue reported in the Enforcement Points job.
- Resolved AccessDenied exception for the drop database command if the database already exists.
- FSEP supports new Boolean true/false configuration parameter named "btwebhdfs.remote.cluster.support.enabled" to allow cross cluster traffic. Can be enabled or disabled using the Ambari Web Manager console. Enabled by default.
- Added end character (;) for running SQL statements in VARC files. Previously, the termination was <ARCEOS>, which is now removed.
- Ambari plug in accepts custom Java home setting.
- Query caching to improve performance of long queries using Cassandra Enforcement Point.

- Addition of XML file configuration parameter named start_after_install to enable or disable software service start up after installation.
- HAWQ Enforcement Point no longer stops when invalid schema name is specified in a set search_path command.
- Retrieve JCE files from the same location as the repository host.
- REST API appends to an existing data domain white list instead of overwriting the white list.
- REST API allows removal of a data domain when the data domain has a white list.
- Configure the static list of Policy Engines using the query modification API instead of discovering the list using the policy configuration service.
- Removed bt-jre included in Audit and Policy packages.
- Redirect deployment log to log4j log files.
- Can specify user defined Java home in XML file, environment variable, or detect from path.
- FSEP resets Hadoop.security.token.service.use_ip property.

New BlueTalon 3.2.3 software features:

- Specify a Policy Engine tag for FSEP. Only the Policy Engine that matches the tag is returned and used by the Enforcement Point.
- Cloudera Manager monitoring extension for BlueTalon services.
- Support for silent install of Policy package using XML file.
- Fixed masking in non-default database (keyspace) for Cassandra data domain.
- FSEP shared secret in Hue file browser.
- Policy Console improvements for Boolean filters.
- Delete Snapshot option in Policy Console.
- Refactor of SecureSecretAuthenticator to remove dependency on hadoop-common jar and commons-collections jar.
- Scripts to enable Oozie shared secret authentication on CDH.
- Compiled BlueTalon XML parser with JDK 1.6.
- Refactor of BlueTalon custom driver to optimize the Policy Engine requests for each query.
- Rules bootstrap JSON file for Cassandra data domain.
- Docker scripts allow setting of sys logger verbosity and location for all components.
- Installer allows customer specified JRE.
- FSEP log includes the Policy Engine tag from which the FSEP service was started.

New BlueTalon 3.2.2 software features:

- Enforcement Point set up script handles Policy Engine HA (High Availability) for FSEP.
- FSEP HA has a configuration property to connect to the configuration service instead of the Policy Engine. In 3.2.2, this is also updated in the Ambari plug in.
- New Enforcement Point for Hive Metastore service.
- Test Connection action for the User Domain to validate the user domain connection.

- Configurable query for user search in Active Directory for authorization. Previosuly, the query for OpenLDAP and Active Directory was fixed and based on the UID and SAMAccountName.
- Multiple Policy Engine instances in HA mode to support continuity if one Policy Engine has a problem.
- Impala "no login" mode operates with password authentication.
- Deployment optimization (policy snapshot feature). Allows background deployment of changes to improve performance.
- Client RPM for secure access to WebHDFS. Enables computers outside of a cluster to send shared secrets.
- Audit Console refactoring. Improves user experience.
- Dynamic multi-component routing for Audit Console.
- Packaging and set up scripts for Oozie shared secret authentication support.
- Refactor of Audit REST API and implementation of REST API 2.0.
- Optimization of deploy process to handle duplicate rules.
- Examine status of BlueTalon services, and allow start, stop, and restart of services from Cloudera Manager.
- Hue support for HiveServer2 and Impala using token ID and session ID.
- Support 0.0.0.0/32 192.0.0.0/28 address formats to compare multiple IP addresses in a cluster for a trusted user.
- Instructions to configure and validate YARN REST EP (reverse proxy) for CDH 5.X.
- LDAP users can access home folder on HDFS when LDAP groups are imported into LDAP user domain.
- Packaging support for Hive Metastore Enforcement Point.
- Boolean filters added to Policy Console.
- Audit Console integrated with installer.
- Enforcement Point set up script requests user entry of Policy Engine end points and data domain name.
- Integration of deployment REST API changes into Policy Console.
- Default value for priority in the log4j.xml templates is set to "error" instead of "all".
- Added spark-hdp-assembly.jar to Spark classpath in bluetalon-post-install-conf-ambari script.
- Improvements for Policy Engine HA.
- GET version API call for Audit Console.
- Deployment log output is sent to the bt-policy-engine.log file instead of the .out file. (Only server start and stop activities send the log to the .out file.)
- Removed RPM dependency on open-jdk in EP RPM.

New Documentation Items

New documentation items:

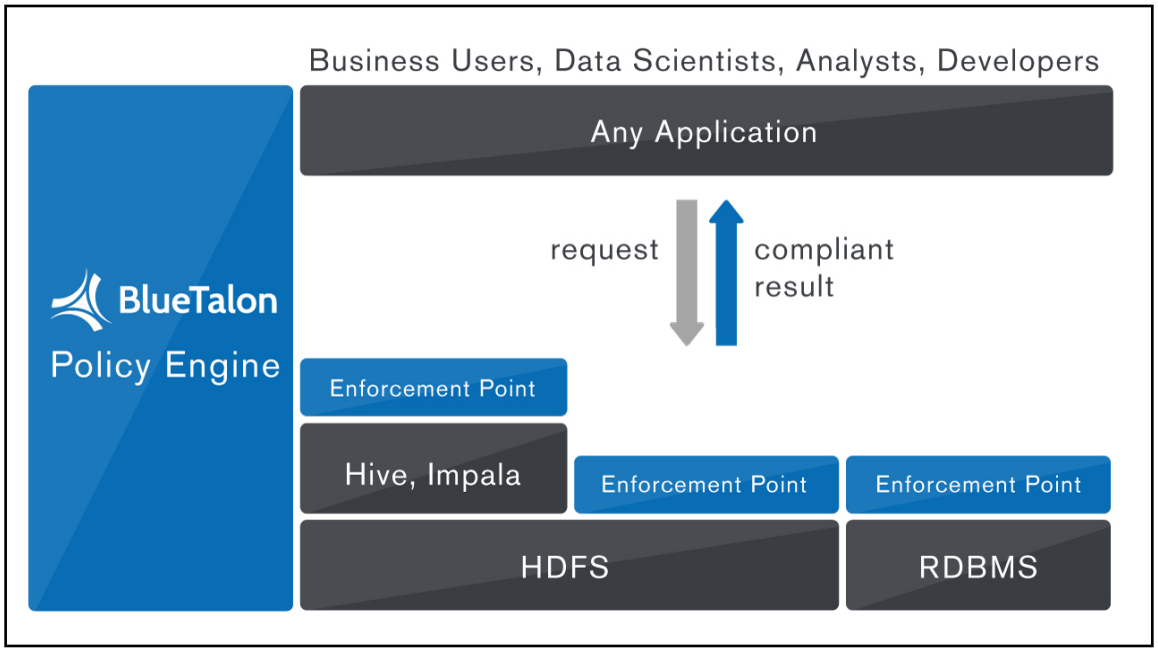
- Quick start for Hadoop. Includes sample data and commands for using Hadoop and BlueTalon software.
- How to enable Policy Engine and Enforcement Point logs for system troubleshooting.
- Coverage for silent mode installation of Enforcement Points. Includes XML parameters and examples for PostgreSQL, FSEP, HDP, CDH, Hive, Impala, and Spark.
- Description of XML file format and parameters for silent install of Policy and Audit packages.
- How to configure Impala for Isilon.
- Updates to HDFS data domain examples.
- How to set a custom Java home location.
- Enable or disable software service start up after installation using the XML parameter start_after_install.
- Coverage of Policy Engine tag for FSEP. Only the Policy Engine that matches the tag is returned and used by the Enforcement Point.
- How to configure Hue with Hive and Impala Enforcement Points.
- Set HDFS permission fallback. Parameter bt.hdfs.plugin.union.hdfs.native.perms in the file bt-hdfs-site.xml is set to true by default.
- How to resolve Enforcement Point concurrency issue.

Software Overview

BlueTalon delivers precise, consistent, and dynamic data-centric security across multiple repositories and enables security administrators to deploy security policies and control user access to enterprise data.

BlueTalon enables enterprises to:

- **Reduce the risk related to data breaches** by enforcing precise user access policies.
- **Maximize analytic possibilities** without restricting users or compromising security.
- **Manage regulatory or industry compliance** with proactive protection and auditing.



The BlueTalon Policy Engine serves as a lever of control between users, applications, and enterprise data repositories. BlueTalon protects data with fine-grained access control and dynamic data masking. BlueTalon controls and audits the data that can be seen and modified across various on-premises and cloud data sources.

BlueTalon is transparent to queries and applications. The BlueTalon Enforcement Points intercept queries from applications and sends them to the BlueTalon Policy Engine. The Policy Engine parses the queries according to defined policies and rules. A policy-compliant query is returned to the Enforcement Points and a policy-compliant dataset result is returned to the user.

The BlueTalon Audit Engine records the query made and the modified policy-compliant query that defines the data consumed. The Audit Engine monitors who attempted to access data, when, and whether they were allowed or denied access. Requests made by users and the modified policy-compliant queries can be viewed in the BlueTalon Audit Console.

Software Components

The following table summarizes the BlueTalon software components.

Component	Description
Policy Engine	Background service for providing policy decisions used in data access control. The Policy Engine enforces data access rules by parsing all data requests and modifying them based on user identity for policy compliance.
Audit Engine	Background service for aggregating audit of data request events from across different enforcement points. The Audit Engine monitors data access and provides visibility over who has accessed which data.
Enforcement Point	Background service that proxies the database or file system. Enforcement Points: 1. Intercept queries and data requests. 2. Forward data requests to the Policy Engine for policy compliance enforcement.

	3. Sends the modified policy compliant request received back from the Policy Engine to the database for execution. Examples of Enforcement Points include PostgreSQL, Hive, Impala, Oracle, DB2, and HDFS.
Policy Console	Web user interface for creating security policies and managing other Policy Engine settings.
Audit Console	Web user interface for viewing audit and monitoring reports created by the Audit Engine.
Policy and Log Stores	PostgreSQL databases that store: • Policies created in the Policy Console. • Enforceable policies deployed on the Policy Engine. • Audit logs captured by the Audit Engine.

Hadoop Quick Start

To get started with Hadoop:

1. Use a computer with at least 16 GB of memory and 4 CPUs.
2. Set up a Hadoop virtual machine or use an existing computer with Hadoop database instance. Download the Hadoop virtual machine software from:
   a. Cloudera at www.cloudera.com/downloads.html
   b. Hortonworks at www.hortonworks.com/products/sandbox/#downloads
3. Create users and test data.

   Perform these commands:

```
useradd alice
passwd alice
useradd bob
passwd bob
useradd charlie
passwd charlie

nano accounts.csv
```

   Paste this text into the file:

```
147274739-9,Frances Harrison,9-(192)357-8851,10/27/56,993-
941527,37726,6393451850134970,52.90
457389751-8,Gregory Patterson,3-(684)454-2444,11/22/69,233-57-
5483,21278,5602245983499780,44.87
360272064-0,Earl James,5-(177)394-3277,5/23/83,303-64-
0766,26595,3534227478434860,3.36
551058833-0,Christine Robertson,5-(437)964-8463,5/11/88,318-79-
4141,50716,5602223026663730,5.22
713553396-8,Daniel Crawford,1-(875)657-9518,6/1/99,214-89-
4670,28693,3571338359613280,22.63
359936857-0,Shirley Holmes,6-(362)871-3036,4/1/80,964-49-
0690,13189,3534219304003200,30.17
887693755-2,Kelly Adams,9-(887)292-8810,2/12/88,304-47-
5814,84901,5602248578416630,35.09
091619799-9,Bonnie Freeman,2-(163)102-1214,1/26/71,940-51-
8723,73248,3547967170434520,69.74
151872601-1,Joyce Mcdonald,3-(504)572-0648,9/18/64,919-59-
```

```
1100,82573,5401856433043540,88.15
956961211-8,Karen Burton,1-(533)256-2922,3/5/54,479-48-
8766,56134,3536534268148670,7.08
```

Save the file and exit nano.

Perform these commands:

```
hdfs hdfs dfs -mkdir /user/shared
hdfs dfs -put accounts.csv /user/shared
hdfs dfs -ls /user/shared/accounts.csv
hdfs dfs -cat /user/shared/accounts.csv
hdfs dfs -chmod -R ugo+wr /user/shared
hive
!connect jdbc:hive2://localhost:10000/default alice
create table accounts(id string, name string, phone string,
birthdate string, soc_sec_no string, zip bigint, credit_card bigint,
balance decimal(4,2)) row format delimited fields terminated by ','
lines terminated by '\n' stored as textfile;
load data inpath '/user/shared/accounts.csv' into table accounts;
select * from accounts;
!close
!quit
```

4. Download and install the BlueTalon software on the computer running Hadoop:
    a. To perform a manual installation:
        i. Download BlueTalon software packages. See Download Software Packages on page 30.
        ii. Set up BlueTalon Audit and Policy packages. See Set Up Audit and Policy Packages on page 33.
        iii. Configure BlueTalon data domain and Enforcement Points for HDFS and Hive:
            A. For File System EP on HDFS, see Configure FSEP for Standard Hadoop HDFS on page 89.
            B. For Hive EP, see Hive Enforcement Point on page 149.
    b. To perform HDP installation, see Configure FSEP for HDP on page 91.
    c. To perform CDH installation, see Configure FSEP for CDH on page 126.
5. Create rules, test rules, and examine audit logs. To create HDFS rules, see the section *HDFS Rules* in the *BlueTalon Security Administration Guide*.
6. Add users to BlueTalon user domains. See User Domains on page 38.

To request help, email BlueTalon at sales@bluetalon.com.

System Requirements

This section describes the minimum system requirements.

Policy Engine and Audit Engine

For the computers running the BlueTalon Policy Engine and Audit Engine, the specification for each computer is:

- CPU: 4 cores minimum.
- Memory: 8 GB minimum.
- Storage: 100 GB minimum. Storage is determined by the size of your BlueTalon logs.
- Operating system: CentOS or RHEL 6.5, 6.6, or 6.7.

Enforcement Points

BlueTalon Enforcement Points typically run on the same computer as the data sources.

Example exceptions:

- For HDFS on a NameNode with clustered databases, you can use a dedicated host for an Enforcement Point.
- If alternative HDFS security solutions are installed (example: Sentry), the Policy Engine runs on a proxy host.

The computer specification for an Enforcement Point is:

- CPU: 2 cores minimum.
- Memory: 2 GB minimum.
- Storage: 5 GB minimum. Storage is determined by the size of your BlueTalon logs.
- Operating system: CentOS or RHEL 6.5, 6.6, or 6.7.

Ports

The following table shows the port requirements.

- EP is an abbreviation for Enforcement Point. This is the computer on which you install the BlueTalon Enforcement Point package.
- Mgmt is an abbreviation for Management. BlueTalon mgmt represents the computer(s) on which you install the BlueTalon Audit and Policy packages.
- Ensure the ports required for BlueTalon to operate are open, but do not open any unnecessary ports.

Where	Permission	From	To	Port	Required For
VPC Security Group	allow	any	VPC	22	SSH to BlueTalon EC2 instances. (Required for cloud deployment only.)
	allow	any	VPC	80	BlueTalon Policy Console.

	allow	any	VPC	443	BlueTalon Audit Console.
	allow	VPC	Windows AD or OpenLDAP	389	LDAP (Not required if LDAPS is used.)
	allow	VPC	Windows AD or OpenLDAP	636	LDAP over SSL (LDAPS).
	allow	VPC	download.cloud.bluetalon.com	80	Downloading BlueTalon software.
	deny	any	VPC	any	(Optional) If no other ports are required by anything else.
FSEP instance	allow	any	VPC	22	SSH to BlueTalon EC2 instances.
	allow	BlueTalon EP	download.cloud.bluetalon.com	80	Downloading BlueTalon software.
	allow	any	VPC	40070	Client to connect to FSEP.
BlueTalon mgmt instance	allow	any	BlueTalon mgmt	22	SSH to BlueTalon mgmt.
	allow	any	BlueTalon mgmt	8111	BlueTalon Policy Console.
	allow	any	BlueTalon mgmt	8112	BlueTalon Audit Console.
	allow	BlueTalon EP	BlueTalon mgmt	1555	Policy decision communications.
	allow	BlueTalon EP	BlueTalon mgmt	1600	Live configuration of EP.
	allow	BlueTalon EP	BlueTalon mgmt	9093	Sending audit entries. Used by bt-audit-kafka service.
	allow	BlueTalon EP	BlueTalon mgmt	2182	Load balancing for Audit Engine. Used by bt-audit-zookeeper service.
	allow	BlueTalon mgmt	Windows AD	389	End user authentication.
	allow	BlueTalon mgmt	BlueTalon mgmt	80 --> to 8111	Port forwarding for BlueTalon Policy Console.
	allow	BlueTalon mgmt	BlueTalon mgmt	443 --> to 8112	Port forwarding for BlueTalon Audit Console.
	allow	BlueTalon mgmt	download.cloud.bluetalon.com	80	Downloading BlueTalon software.
	deny	any	BlueTalon mgmt	any	(Optional) If no other ports are required by anything else.
BlueTalon EP EC2 instance	allow	any	BlueTalon EP	22	SSH to BlueTalon EP.

	allow	any	BlueTalon EP	5432	Client applications to query through BlueTalon EP; alternatively use the following rule.
	allow	app VM	BlueTalon EP	5432	Client applications to query through BlueTalon EP.
	allow	BlueTalon EP	download.cloud.bluetalon.com	80	Downloading BlueTalon software.
PostgreSQL RDS	allow	BlueTalon EP	PostgreSQL RDS	5432	Allowing BlueTalon EP to proxy RDS. (Required for PostgreSQL only.)
	deny	any	PostgreSQL RDS	5432	Securing PostgreSQL RDS from direct access. (Required for PostgreSQL only.)
Windows AD	allow	BlueTalon mgmt	Windows AD	389	End user authentication.
	allow	BlueTalon mgmt	Windows AD or OpenLDAP	636	End user authentication and user to group lookup over LDAPS.
AWS Route 53 services	forward	any	PostgreSQL RDS:5432	BlueTalon EP:5432	Forwarding traffic from PostgreSQL RDS to BlueTalon EP using DNS translation.

Policy Console and Audit Console

For the BlueTalon Policy Console and Audit Console, use Google Chrome version 42 or higher on:

- Windows
- MacOS X
- Linux

The supported versions of the operating systems are those that support Google Chrome version 42 or higher.

Java SE Runtime Environment

For all computers that will run any BlueTalon component, the Java SE Runtime Environment must be installed before installing the BlueTalon software:

- Java SE Runtime Environment 7u79 or 7u80
- Java SE Runtime Environment is available for download from http://www.oracle.com/technetwork/java/javase/downloads/jre7-downloads-1880261.html

Data Platforms and Clients

The following table shows the supported versions for the data platforms and client applications.

Data Platform	Resource	Enforcement Point	Client Application	Version
HDFS	Folder File	NameNode plugin (one per name node)	Java driver (for R) Pig Hue 3.7 Pig R 2.1 MapReduce Spark	Hortonworks HDP 2.3 - Hadoop 2.7.1 Hortonworks HDP 2.2 - Hadoop 2.6.0 Cloudera CDH 5.3 - Hadoop 2.6.0 HDInsight v 3.1 - HDFS 2.4.0 HDInsight v 3.2 - HDFS 2.6.0
	Folder File	Filesystem Proxy (one per data node)	FsShell client MapReduce Pig Hue 3.7 R 2.1 Spark	Hortonworks HDP 2.3 - Hadoop 2.7.1 Hortonworks HDP 2.2 - Hadoop 2.6.0 Cloudera CDH 5.3 - Hadoop 2.6.0 Cloudera CDH 5.4 – Hadoop 2.6.0 Cloudera CDH 5.5 – Hadoop 2.6.0 HDInsight v 3.1 - HDFS 2.4.0 HDInsight v 3.2 - HDFS 2.6.0 Isilon OneFS 8.0.0.1
Hive	Database Table Column Cell	HiveServer2 Proxy (one per HiveServer2)	ODBC driver JDBC driver Python driver Beeline	Hortonworks HDP 2.3 - Hive 1.2.1 HDInsight 3.1 - Hive 0.13.1 HDInsight 3.2 - Hive 0.14.0 Cloudera CDH 5.3 - Hive 0.13.1 Cloudera CDH 5.4 - Hive 1.1 Cloudera CDH 5.5 – Hive 1.1
Impala	Database Table Column Cell	Impala Proxy (one per impalad process)	ODBC driver JDBC driver Python driver Impala shell	Cloudera 5.3 - Impala version 2.1.1 Cloudera 5.4 – Impala version 2.2.0
Oracle	Database Schema Table Column Cell	TNS Listener Proxy (one per TNS listener)	ODBC driver JDBC driver Python driver SQL Plus	Oracle Database 10g Oracle Database 11g
PostgreSQL	Database Schema Table Column	Server Proxy (one per postmaster process)	ODBC driver JDBC driver Python driver Psql client	PostgreSQL 8.4 PostgreSQL 9.1

	Cell			
Greenplum	Database Schema Table Column Cell	Master Listener Proxy (one per postmaster process)	ODBC driver JDBC driver Python driver	EMC/Pivotal v. 4.3.8.3
Amazon Redshift	Database Schema Table Column Cell	JDBC Endpoint Proxy (one or more per Redshift instance)	ODBC driver JDBC driver Psql client	Amazon Redshift 1.0.1059
Cassandra	Keyspace Table Field Subfield	Node Plugin (one per Cassandra node)	Cqlsh client	Cassandra 3.0.6
Spark	Database Schema Table Column Cell	Thrift Server Proxy (one per Spark master)	ODBC driver JDBC driver	Spark 1.5.1
DB2	Table Column	Server Proxy (one per JDBC endpoint)	ODBC driver JDBC driver DB2 client	IBM DB2 9.7.2
MySQL	Database Schema Table Column Cell	Server Proxy	ODBC driver JDBC driver MySQL native driver	MySQL 5.6.23
Microsoft SQL Server	Database Schema Table Column Cell	Server Proxy	JDBC driver	Microsoft SQL Server 2012

If your platform is not listed in the previous table, contact support@bluetalon.com to request a certification for your version of the database platform.

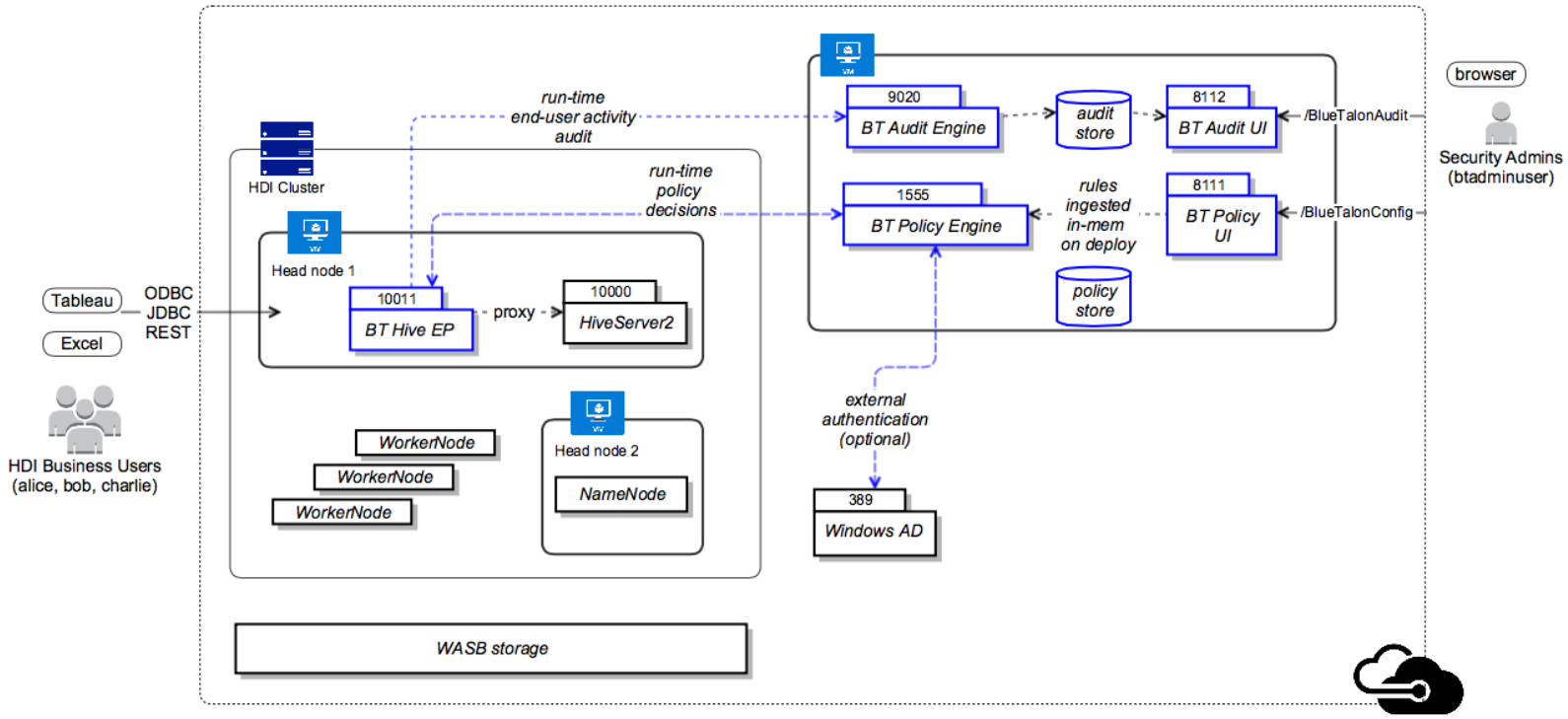
Example Implementations

This section shows example BlueTalon implementations.

Azure HDInsight

HDInsight is a cloud distribution on Microsoft Azure for Hadoop. HDInsight includes implementations of Apache Spark, HBase, Kafka, Storm, Pig, Hive, Interactive Hive, Sqoop, Oozie, and Ambari. BlueTalon integrates with HDInsight to provide security.

The following diagram shows a Microsoft Azure HDInsight (HDI) implementation.



BlueTalon software is available in the Azure Marketplace as a Virtual Machine or a Solution Template. Use the:

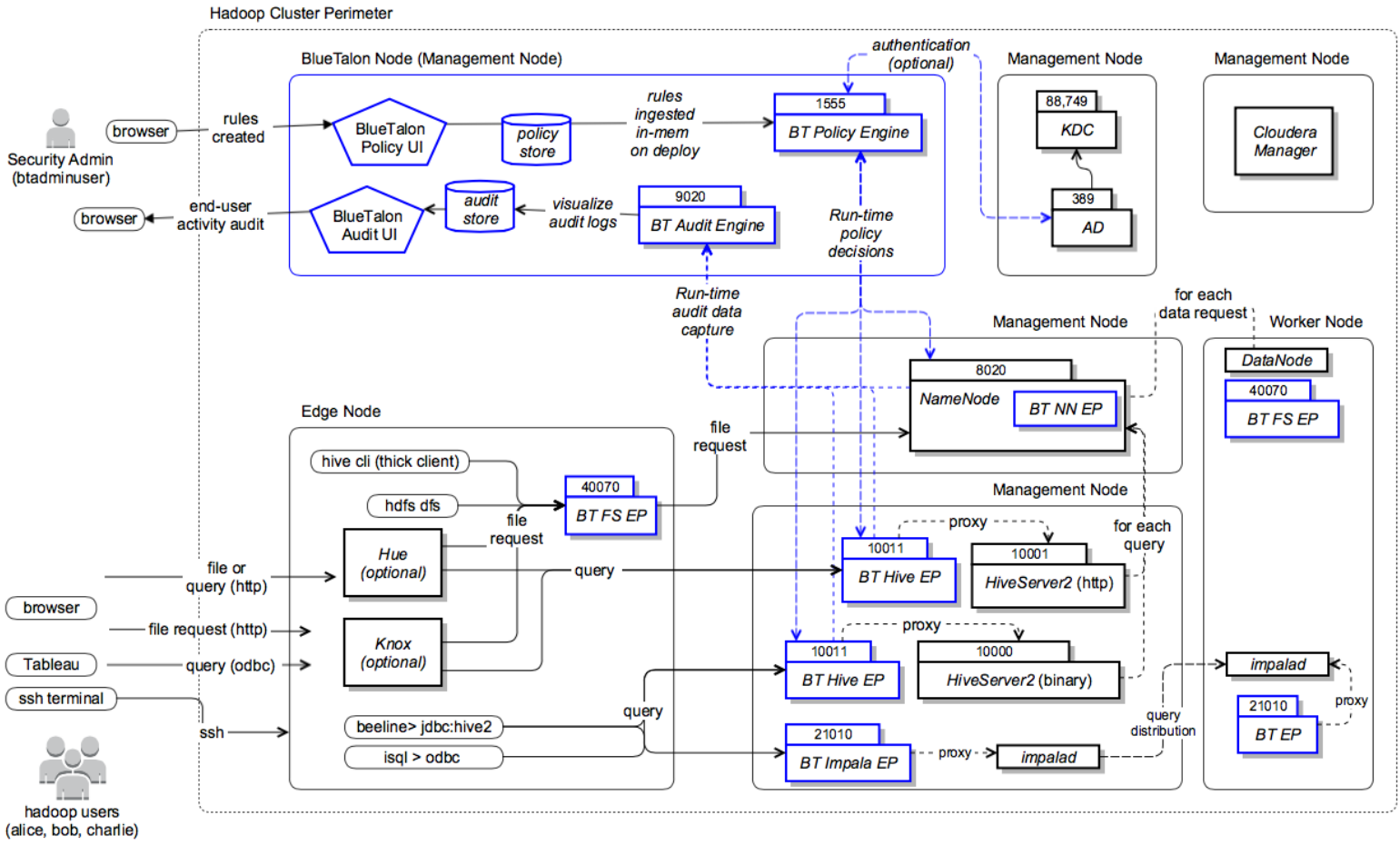
- Solution Template for a single persistent HDI cluster.
- Virtual Machine for enforcing consistent security across multiple persistent or transient HDI clusters and relational databases.

The Solution Template with the default parameters deploys BlueTalon on an application VM along with the HDI cluster in the configuration shown in the previous diagram.

Cloudera Hadoop Distribution

CDH is Cloudera's open-source platform distribution for Apache Hadoop. BlueTalon is secure certified by Cloudera. Security administrators create rules and view audit records through the user interface (or the API) that drive the run-time Policy and Audit Engines on a management node of the cluster.

The following diagram shows a Cloudera Hadoop Distribution implementation.

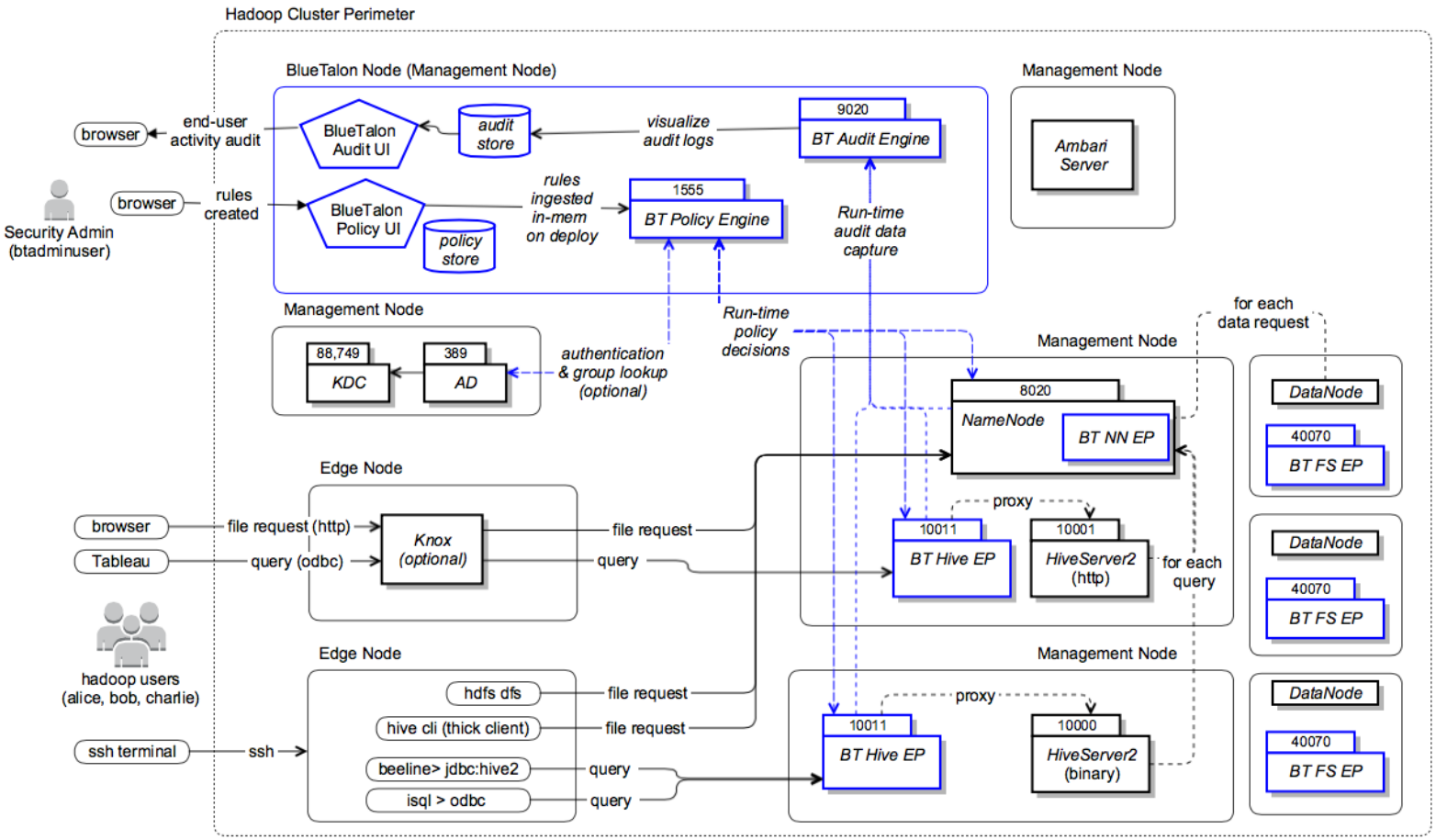


Hive, Impala, and HDFS file requests go through the BlueTalon Hive EP, Impala EP, and either the BlueTalon NameNode EP or the File System EP. FSEP and Impala EP are installed on each compute node.

Hortonworks Hadoop Distribution

HDP is Hortonwork's open-source Apache Hadoop distribution based on a centralized architecture (YARN). Hortonworks HDP powers customer applications and delivers robust analytics for decision making and innovation. BlueTalon provides security for HDP.

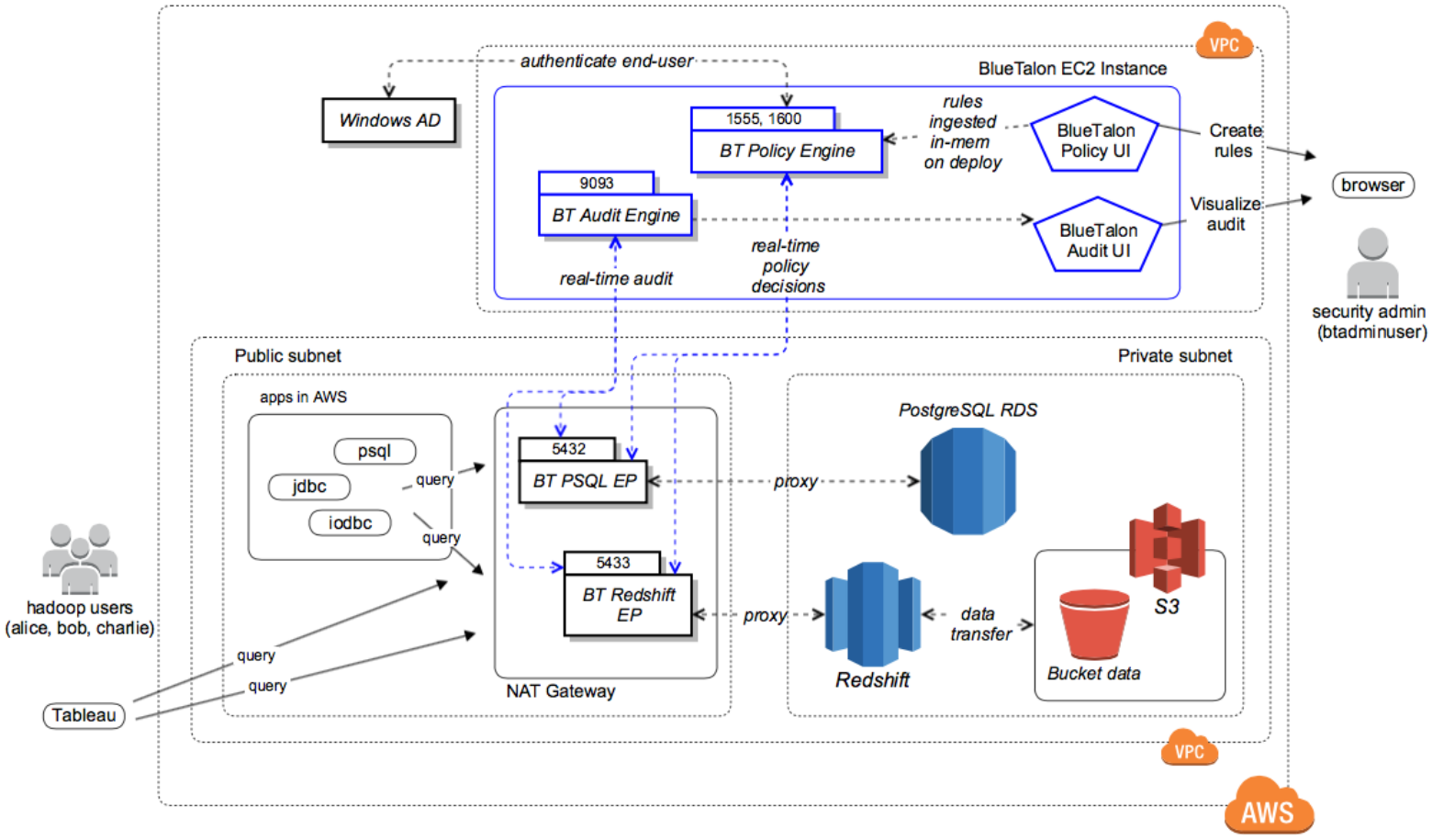
The following diagram shows a Hortonworks Hadoop Distribution implementation.



AWS Redshift

AWS Redshift is a petabyte-scale data warehouse for analyzing data using existing business intelligence tools. BlueTalon provides security for AWS Redshift.

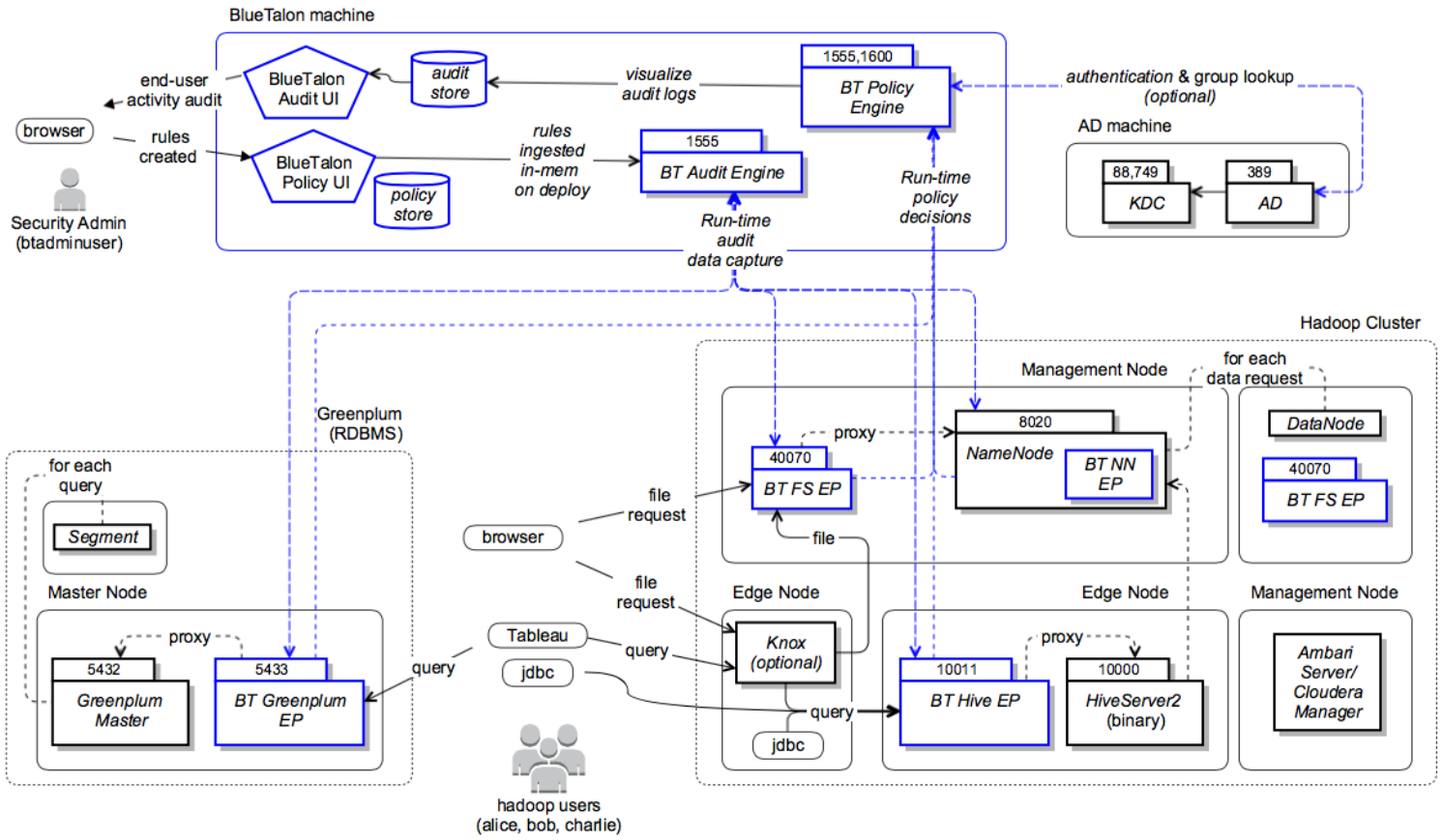
The following diagram shows an AWS Redshift implementation.



On-Premises Implementation with Hadoop and RDBMS

Apache Hadoop is an open-source software framework used for distributed storage and processing of very large data sets. BlueTalon provides security for Apache Hadoop.

The following diagram shows an on-premises implementation with Hadoop and any RDBMS.



Obtain Software

This section describes how to obtain the BlueTalon software.

Prerequisites

- Super user privilege for the computers on which you are installing the BlueTalon software. Perform all commands in this document as the super user or use sudo.
- If you do not have super user or sudo access for the computers on which you are installing BlueTalon, contact your sales representative or sales@bluetalon.com for alternative packages.
- The BlueTalon software download location might require a password. If you are unable to access the download location shown later, obtain the password from your sales representative or sales@bluetalon.com.

Software Packages

The following table shows the BlueTalon software packages.

Package Name	Content
bluetalon-audit	Audit Engine, Audit Repository, and Audit Console.
bluetalon-ep	Enforcement Points (EPs).
bluetalon-policy	Policy Engine, Policy Repository, and Policy Console.

HDP and CDH

For installation of the BlueTalon software on HDP and CDH, you use the instructions in the section links shown in the following table.

Installation Type	Section
FSEP for HDP	Configure FSEP for HDP on page 91
FSEP for CDH	Configure FSEP for CDH on page 126

Download Software Packages

This section describes how to download the BlueTalon software packages.

Installation for Computer Connected to Internet

If your BlueTalon installation computer is connected to the Internet:

1. Log in as a super user on the computer where you are installing BlueTalon.
2. Change directories:
   ```
   cd /etc/yum.repos.d
   ```
3. Download the BlueTalon software packages. Example:
   ```
   wget
   ```

```
https://repo.cloud.bluetalon.com/BT/centos6/updates/3.2.4.0/BlueTalo
nLatest.repo
```

4. Ensure the packages are ready for installation:
```
yum search bluetalon
```

   For the package list, see Software Packages on page 30.

Installation for Computer Not Connected to Internet

If your BlueTalon installation computer is not directly connected to the Internet:

1. Download the tar ball files for your platform using any computer connected to the Internet.

Platform	Type	Download URL
Centos6	RPM tar ball	https://repo.cloud.bluetalon.com/BT/centos6/updates/3.2.4.0/BT-Centos6-3.2.4.0-rpm-tarball.tar.gz https://repo.cloud.bluetalon.com/BT/centos6/updates/3.2.4.0/BT-Centos6-3.2.4.0-rpm-tarball.tar.gz.asc https://repo.cloud.bluetalon.com/BT/centos6/updates/3.2.4.0/BT-Centos6-3.2.4.0-rpm-tarball.tar.gz.md5
Centos6	Tars tar ball	https://repo.cloud.bluetalon.com/BT/centos6/updates/3.2.4.0/BT-Centos6-3.2.4.0-tars-tarball.tar.gz https://repo.cloud.bluetalon.com/BT/centos6/updates/3.2.4.0/BT-Centos6-3.2.4.0-tars-tarball.tar.gz.asc https://repo.cloud.bluetalon.com/BT/centos6/updates/3.2.4.0/BT-Centos6-3.2.4.0-tars-tarball.tar.gz.md5
Ubuntu12.04	Tars tar ball	https://repo.cloud.bluetalon.com/BT/ubuntu12.04/updates/3.2.4.0/BT-ubuntu12.04-3.2.4.0-tars-tarball.tar.gz https://repo.cloud.bluetalon.com/BT/ubuntu12.04/updates/3.2.4.0/BT-ubuntu12.04-3.2.4.0-tars-tarball.tar.gz.asc https://repo.cloud.bluetalon.com/BT/ubuntu12.04/updates/3.2.4.0/BT-ubuntu12.04-3.2.4.0-tars-tarball.tar.gz.md5

2. Extract the contents of the tar ball file.
3. Copy the files to the computer where you are installing BlueTalon.
4. As a privileged user, set up the file repository using the following command from the directory containing the copied RPM files. The command creates a directory ./repodata, which allows yum to install the RPM files.
```
createrepo ./
```
5. If the createrepo command is not found, perform these commands as a privileged user:
```
yum search createrepo
yum install createrepo –y
createrepo ./
```
6. Create a file named LocalBlueTalon.repo in the directory /etc/yum.repos.d to point to the repo file in the directory containing the RPM files. Add the following text to the file:
```
[bluetalon]
name=bluetalon
baseurl=file:///var/repo/BlueTalon
enabled=1
gpgcheck=0
```
7. Ensure the packages are ready for installation:
```
yum search bluetalon
```

For the package list, see Software Packages on page 30.

Set Up Audit and Policy Packages

This section describes how to:

- Install the BlueTalon Audit and Policy packages.
- Perform subsequent system configuration.

You install the Audit and Policy packages on the computer(s) designated as your BlueTalon Management computer(s). You can install the Audit package on a different computer from the Policy package.

Change Default Passwords

Passwords:

- The installation scripts prompt you to set passwords.
- Do not use the default passwords.
- Set your own strong passwords in accordance with your organization's security requirements.
- You can set the passwords after installation. See Set Strong Passwords on page 194.

Install Packages

You use yum to install the packages in the following order:

1. Audit
2. Policy
3. Enforcement Points

Install Audit Package

Install the Audit package:

```
yum install bluetalon-audit-3.2.4
```

If you are installing a software version other than 3.2.4, you replace the digits with the version you are installing. Also applies to the rest of the commands in this document.

Install Policy Package

Install the Policy package:

```
yum install bluetalon-policy-3.2.4
```

Run Configuration Scripts

Run the configuration scripts on your BlueTalon Management computer.

Default Script Values

When you run the scripts shown in this document, you will see prompts with default values shown in square brackets. Example:

```
I understand the terms and accept BlueTalon Online Evaluation License.
(yes/no) [no] :
```

In the example, the default is no. To accept a default, you press Enter. To change it, you enter a new value (example: yes).

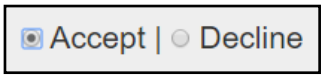
Run Audit Setup Script

Run the Audit setup script:

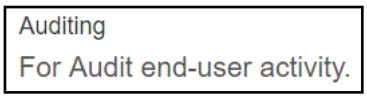
1. Run the script as a privileged user:
   `/opt/bluetalon/3.2.4/audit/scripts/bluetalon-audit-setup`

   You can also run the script in silent mode, which does not prompt you to enter details. See the following section for details.
2. Follow the prompts.
3. When the installation is complete, note the URL to access the BlueTalon Audit Console. You will need it later. URL format:
   `http://<fully qualified domain name or IP address:port>/BlueTalonAudit`

   Examples:
   `http://TestServer.TestCompany.com:8112/BlueTalonAudit`
   `http://127.0.0.1:8112/BlueTalonAudit`
4. Open a compatible Web browser. See Policy Console and Audit Console Requirements on page 22.
5. Ensure the Audit Console is accessible. If you cannot access the Audit Console, ensure:
   a. Hostname for the computer is accessible.
   b. Firewall rules are correct.
6. Examine the license agreement.
7. If you accept the license agreement, select Accept.

8. Click Auditing.

9. Log in with the user name and the password.

   Defaults:
   btadminuser
   P@ssw0rd

Run Audit Setup Script in Silent Mode

(Optional) You can also run the Audit setup script in silent mode, which does not prompt you to enter details. Run:

```
/opt/bluetalon/3.2.4/audit/scripts/bluetalon-audit-setup -accept -silent
```

To change the default parameters for the silent install, edit this file:

`/opt/bluetalon/3.2.4/audit/conf/bluetalon-audit.xml`

The file format and parameters are described in the following sections.

Silent Mode XML File Format

Format for the silent mode XML file:

```
<bluetalon-audit xmlns="">
  <log_directory_path></log_directory_path>
  <conf_directory_path></conf_directory_path>
  <db_pwd></db_pwd>
  <db_port></db_port>
  <use_old_data></use_old_data>
  <old_db_pwd></old_db_pwd>
  <zookeeper_port></zookeeper_port>
  <kafka_port></kafka_port>
  <kafka_host></kafka_host>
  <login_pwd></login_pwd>
  <visualize_port></visualize_port>
  <passphrase></passphrase>
  <polling_interval></polling_interval>
  <batch_size></batch_size>
  <start_after_install></start_after_install>
</bluetalon-audit>
```

The parameters are described in the following section.

Silent Mode XML File Parameters

The following table shows the parameters for the silent mode XML file.

Parameter	Optional or Required	Example	Description
log_directory_path	Optional	/opt/bluetalon/current/audit/logs	Path for log files.
conf_directory_path	Optional	/opt/bluetalon/current/audit/conf	Path for configuration files.
db_pwd	Required	bt#123	Password for audit database.
db_port	Required	5434	Port for audit database.
use_old_data	Optional	no	Allow use of old audit records.
old_db_pwd	Optional	MyPassword	Password of old audit database.
zookeeper_port	Required	2182	Port for ZooKeeper.
kafka_port	Required	9093	Port for Kafka.
kafka_host	Required	localhost	Host name where Kafka is running.
login_pwd	Required	P@ssw0rd	Password for Audit Console.

visualize_port	Required	8112	Port for visualize service.
passphrase	Required	MyPassphrase	Passphrase to encrypt and decrypt Audit Console and database password.
polling_interval	Required	1000	Time period in seconds for examining the Policy Engine for policy changes made after the last polling interval.
batch_size	Required	2000	Batch size in bytes.
java_home	Optional	/usr/lib/jvm/jre-1.6.0-openjdk.x86_64	Custom java home path.
start_after_install	Optional	true	Enable or disable audit service execution after installation.

Run Policy Setup Script

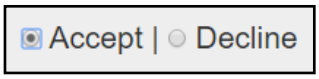
Run the Policy setup script:

1. Run the script as a privileged user:
   `/opt/bluetalon/3.2.4/policy/scripts/bluetalon-policy-setup`

   You can also run the script in silent mode, which does not prompt you to enter answers. See the following section for details.
2. Follow the prompts:
   a. Examine and accept the license agreement.
   b. Set your own passwords.
3. When you are prompted to enter the hostname where the Kafka service is running (bt-audit-kafka):
   a. Enter the hostname or IP address where you ran the Audit package setup script in the previous section. The Kafka service runs on that same computer. Example IP address: 10.0.1.7.
   b. Ensure port 9093 is open. See Ports on page 20.
   c. If the Kafka server is not accessible, the script will prompt you to proceed anyway. **You must enter the Kafka hostname before the BlueTalon software will operate correctly.**
4. When the installation is complete, note the URL to access the BlueTalon Policy Console. You will need it later. URL format:
   `http://<fully qualified domain name or IP address:port>/BlueTalonConfig`

   Examples:
   `http://TestServer.TestCompany.com:8111/BlueTalonConfig`
   `http://127.0.0.1:8111/BlueTalonConfig`
5. Open Web browser and ensure the Policy Console is accessible. If you cannot access the Policy Console, ensure:
   a. Hostname for the computer is accessible.
   b. Firewall rules are correct.
6. Examine the license agreement.

7. If you accept the license agreement, select Accept.



8. Click Policies and Auditing.

9. Log in with the user name and the password.

   Defaults:
   btadminuser
   P@ssw0rd

Run Policy Setup Script in Silent Mode

(Optional) You can also run the Policy setup script in silent mode, which does not prompt you to enter details. Run:

```
/opt/bluetalon/3.2.4/policy/scripts/bluetalon-policy-setup -accept -silent
```

To change the default parameters for the silent install, edit this file:

```
/opt/bluetalon/3.2.4/policy/conf/bt-policy.xml
```

The file format and parameters are described in the following sections.

Silent Mode XML File Format

Format for the silent mode XML file:

```
<bluetalon-policy xmlns="">
  <log_directory_path></log_directory_path>
  <conf_directory_path></conf_directory_path>
  <db_pwd></db_pwd>
  <login_pwd></login_pwd>
  <use_old_data></use_old_data>
  <old_db_pwd></old_db_pwd>
  <kafka_host></kafka_host>
  <kafka_port></kafka_port>
  <db_port></db_port>
  <ui_port></ui_port>
  <pdp_port></pdp_port>
  <config_service_port></config_service_port>
  <java_home></java_home>
  <start_after_install></start_after_install>
  <listening_ip></listening_ip>
```

```
</bluetalon-policy>
```

The parameters are described in the following section.

Silent Mode XML File Parameters

The following table shows the parameters for the silent mode XML file.

Parameter	Optional or Required	Example	Description
log_directory_path	Optional	/opt/bluetalon/current/policy/logs	Path for log files.
conf_directory_path	Optional	/opt/bluetalon/current/policy/conf	Path for configuration files.
db_pwd	Required	bt#123	Password for policy database.
db_port	Required	5433	Port for policy database.
use_old_data	Optional	no	Allow use of old policy records.
old_db_pwd	Optional	MyPassword	Password of old policy database.
kafka_host	Required	localhost	Host name where Kafka is running.
kafka_port	Required	9093	Port for Kafka.
login_pwd	Required	P@ssw0rd	Password for Policy Console.
ui_port	Required	8111	Port for Policy Console.
pdp_port	Required	1555	Port for PDP service.
config_service_port	Required	1600	Port for configuration service.
java_home	Optional	/usr/lib/jvm/jre-1.6.0-openjdk.x86_64	Custom java home path.
start_after_install	Optional	true	Enable or disable policy service execution after installation.
listening_ip	Optional	127.0.0.1	IP Address for Policy Engine to listen on

User Domains

A user domain is a set of user accounts.

Users are stored in the user authentication system already implemented in your organization.

You:

- Can import user metadata from an external user authentication system into BlueTalon.
- Cannot add, modify, or delete users in an external system from BlueTalon.

The following table shows the user domain types.

User Domain Type	Description
Internal to BlueTalon	User IDs and passwords are managed in the database. New user accounts can be

	created using the Policy Console or the Command Line Shell. Typically used for small clusters when Windows Active Directory or OpenLDAP are not available.
Windows Active Directory	User IDs and passwords are managed by Windows Active Directory.
OpenLDAP	User IDs and passwords are managed by OpenLDAP.
Kerberos	User IDs and passwords are managed by Kerberos. The database server and clients are configured to use Kerberos.
OSUser	Used when there is only one node computer where users are created. Typically used for demonstration purposes when Windows Active Directory or OpenLDAP are not available.

If you use more than one user domain:

- You can use the user domains for user to group mappings.
- The password can only be managed in one user domain.

Proxy Authentication

For PostgreSQL and Cassandra Enforcement Points, BlueTalon provides an optional feature called proxy authentication to simplify user management.

Proxy authentication allows a different user, known as the proxy authentication user, to be sent to the database instead of the actual end user performing the database operation:

- A proxy authentication user is the user name and password that is set for the data domain in the Policy Console.
- The proxy authentication user is used by the Enforcement Point to connect to the database.
- The Enforcement Point must be configured to use proxy authentication.
- Proxy authentication works with PostgreSQL and Hive.

If an Enforcement Point is configured to use proxy authentication:

- The end user name and password are sent from the client program to the Enforcement Point.
- The end user name and password are authenticated by the Policy Engine against the user domain.
- You cannot use a Kerberos user domain with proxy authentication. You can use the other user domain types.
- The Enforcement Point sends the proxy authentication user name and password to the database.
- The database operation is performed as the proxy authentication user.
- The end user does not have to be in the database.

To configure proxy authentication, see Configure Proxy Authentication on page 205.

OpenLDAP or Windows Active Directory

If you use OpenLDAP or Windows Active Directory with:

- Hadoop: Nodes should be configured to use OpenLDAP or Windows Active Directory for user authentication. Otherwise, user IDs should be presented as local accounts on all cluster nodes.

- PostgreSQL, Hive, Impala, Redshift, Greenplum: You do not configure the databases to work with OpenLDAP or Windows Active Directory.

  If you want to use the proxy authentication feature, then you configure the Enforcement Point to use proxy authentication. The proxy authentication feature causes all connections to the database to use the same database user account, which is specified in the data domain.

- All other databases (examples: Oracle, SQL Server, MySQL, DB2): You must configure the databases to use Windows Active Directory and OpenLDAP.

If you use OpenLDAP, then you must specify the fully qualified user name for all connections to the data sources.

Connect to Windows Active Directory

If you are using Windows Active Directory, you perform the steps in this section:

- Run the LDAP searches in this section to test the connection to your LDAP server.
- Ensure the LDAP searches run successfully before you add a BlueTalon user domain.

Install LDAP Search Utility

Install the LDAP search utility (ldapsearch) from a yum package.

LDAP Search to Examine All User Groups

Perform an LDAP search, which:

- Connects to a Windows Active Directory server using a non-SSL connection that binds to a user name and password.
- Returns the groups that a specified user belongs to.

```
ldapsearch -LLL -D "<bind user name>" -b "<bind DN>" -H ldap://<LDAP
server FQDN>:389 "cn=<common user name>*" dn memberOf -W
```

The parameters are described in the following table.

Parameter	Description
bind user name	User name of the Windows Active Directory account that can perform a group look up.
bind DN	Bind identity for the LDAP search.
LDAP server FQDN	Fully qualified domain name of the LDAP server.
common user name	Common name of the user to examine group membership for.
-W	Prompts for the LDAP password of the user account specified in <bind user name>.

Example execution of an LDAP search that returns the groups that the Administrator user belongs to:

```
ldapsearch -LLL -D "Administrator@example.com" -b "dc=example,dc=com" -
H ldap://domain.example.com:389 "cn=Administrator*" dn memberOf -W

Enter LDAP Password: XXX

dn: CN=Administrator,CN=Users,DC=example,DC=com

memberOf: CN=BT-All,CN=Users,DC=example,DC=com
```

The returned result shows that the Administrator user belongs to the BT-All group.

- Use your own parameters. The example is for illustrative purposes only.
- Ensure your parameters are correct and that you can connect to your LDAP server.
- You will need your own bind user name, bind DN, LDAP server FQDN, and password when you add a BlueTalon user domain.
- Record your parameters and ensure they are correct before you add a BlueTalon user domain.

LDAP Search to Ensure User Belongs to One Group

Perform an LDAP search to ensure a specified user is a member of one of the groups in the filter list:

```
ldapsearch -LLL -D "<bind user name>" -b "<bind DN>" -H ldap://<LDAP
server FQDN>:389 "(&(member:1.2.840.113556.1.4.1941:=<user
DN>)(|(distinguishedName=<group 1 DN>)(distinguishedName=<group 2
DN>))" dn memberOf -W
```

The parameters are described in the following table.

Parameter	Description
bind user name	User name of the Windows Active Directory account that can perform a group look up.
bind DN	Bind identity for the LDAP search.
LDAP server FQDN	Fully qualified domain name of the LDAP server.
user DN	Full distinguished name of an account you can use for BlueTalon.
group 1 DN	Full distinguished name of a group you can use for BlueTalon.
group 2 DN	Full distinguished name of any other group. You can use any name.
-W	Prompts for the LDAP password of the user account specified in <bind user name>.

Example execution of an LDAP search that examines the Administrator user:

```
ldapsearch -LLL -D "Administrator@example.com" -b "dc=example,dc=com" -
H ldap://fqdn:389
"(&(member:1.2.840.113556.1.4.1941:=CN=Administrator,CN=Users,DC=exampl
e,DC=com)(|(distinguishedName=CN=Administrator,CN=Users,DC=example,DC=c
om)(distinguishedName=CN=BT-
All,CN=Users,DC=example,DC=com)(distinguishedName=CN=User239
SUser239,CN=Users,DC=example,DC=com)(distinguishedName=CN=User239
SUser239,CN=Users,DC=example,DC=com)))" dn memberOf -W

Enter LDAP Password: XXX

filter:
(&(member:1.2.840.113556.1.4.1941:=CN=Administrator,CN=Users,DC=example
,DC=com)(|(distinguishedName=CN=Administrator,CN=Users,DC=example,DC=co
m)(distinguishedName=CN=BT-
All,CN=Users,DC=example,DC=com)(distinguishedName=CN=User239
SUser239,CN=Users,DC=example,DC=com)(distinguishedName=CN=User239
SUser239,CN=Users,DC=example,DC=com)))

requesting: dn memberOf
```

```
dn: CN=BT-All,CN=Users,DC=example,DC=com
```

The result shows that the Administrator user belongs to the BT-All group.

- You use your own parameters. The example is for illustrative purposes only.
- Ensure your parameters are correct and that you can connect to your LDAP server.

Change Windows Active Directory Realm for BlueTalon Policy Server

Add the realm to the file /opt/bluetalon/3.2.4/policy/pap/conf/server.xml:

```
<Realm className="org.apache.catalina.realm.JNDIRealm"
connectionURL="ldaps://<AD_HOST>:<AD_PORT>" debug="10"
connectionName="<AD_USER@FQDN>" connectionPassword="<PWD_OF_AD_USER>"
referrals="follow" userBase="<BASE_DN>"
userSearch="(sAMAccountName={0})"
userSubtree="true" roleBase="<BASE_DN>" roleName="name"
roleSubtree="true"
roleSearch="(member={0})" allRolesMode="authOnly"/>
```

Use your own settings for the parameters shown in bold in the previous file extract.

Example (use your own IP address and port, which are shown as X characters):

```
<Realm className="org.apache.catalina.realm.JNDIRealm"
connectionURL="ldaps://XXX.XXX.XXX.XXX:XXX" debug="10"
connectionName="Administrator@corp.bluetalon.com"
connectionPassword="agetak#123" referrals="follow"
userBase="dc=corp,dc=bluetalon,dc=com"
userSearch="(sAMAccountName={0})"
userSubtree="true" roleBase="dc=corp,dc=bluetalon,dc=com"
roleName="name"
roleSubtree="true" roleSearch="(member={0})" allRolesMode="authOnly"/>
```

Restart the service bt-policy-server.

```
service bt-policy-server restart
```

You can now log in with SSL users to the BlueTalon Policy Console.

Change Windows Active Directory Realm for BlueTalon Audit Server

Add the realm to the file /opt/bluetalon/3.2.4/audit/bluetalon-audit-visualize-basic/conf/server.xml:

```
<Realm className="org.apache.catalina.realm.JNDIRealm"
connectionURL="ldaps://<AD_HOST>:<AD_PORT>" debug="10"
connectionName="<AD_USER@FQDN>" connectionPassword="<PWD_OF_AD_USER>"
referrals="follow" userBase="<BASE_DN>"
userSearch="(sAMAccountName={0})"
userSubtree="true" roleBase="<BASE_DN>" roleName="name"
roleSubtree="true"
roleSearch="(member={0})" allRolesMode="authOnly"/>
```

Use your own settings for the parameters shown in bold in the previous file extract.

Example (use your own IP address and port, which are shown as X characters):

```
<Realm className="org.apache.catalina.realm.JNDIRealm"
connectionURL="ldaps://XXX.XXX.XXX.XXX:XXX" debug="10"
connectionName="Administrator@corp.bluetalon.com"
connectionPassword="agetak#123" referrals="follow"
userBase="dc=corp,dc=bluetalon,dc=com"
userSearch="(sAMAccountName={0})"
```

```
userSubtree="true" roleBase="dc=corp,dc=bluetalon,dc=com"
roleName="name"
roleSubtree="true" roleSearch="(member={0})" allRolesMode="authOnly"/>
```

Restart the service bt-audit-server:

```
service bt-audit-server restart
```

You can now log in with SSL users to the BlueTalon Audit Console.

Kerberos

If you use Kerberos with Hadoop:

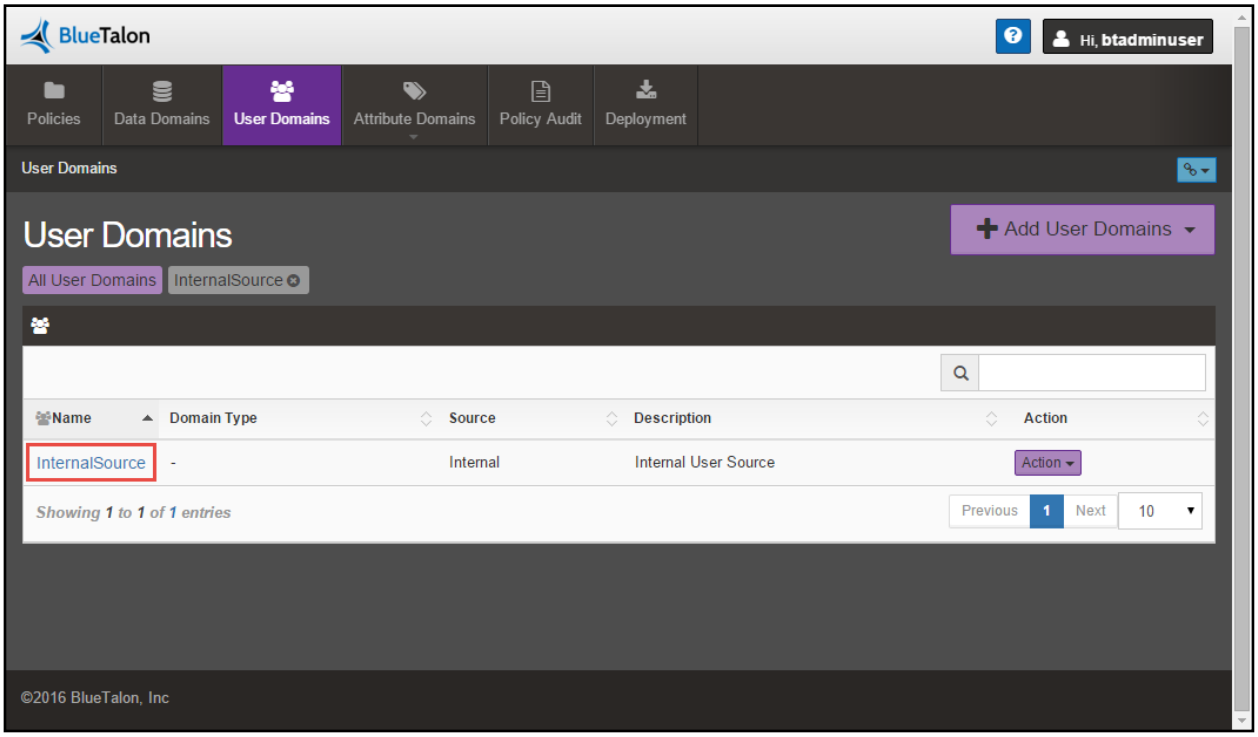
- You typically add a Kerberos user domain in addition to Windows Active Directory. If the user IDs and passwords are managed by Kerberos, then you only add a Kerberos user domain.
- You configure Kerberos and Windows Active Directory to operate with each other outside of BlueTalon.
- When you use Kerberos and Windows Active Directory, the user IDs and passwords are managed by Windows Active Directory but the Kerberos single sign on tickets are managed by Kerberos.
- You can also use MIT Kerberos without Windows Active Directory or OpenLDAP if the user accounts and passwords are managed directly in the MIT Kerberos KDC.

Add User Domain

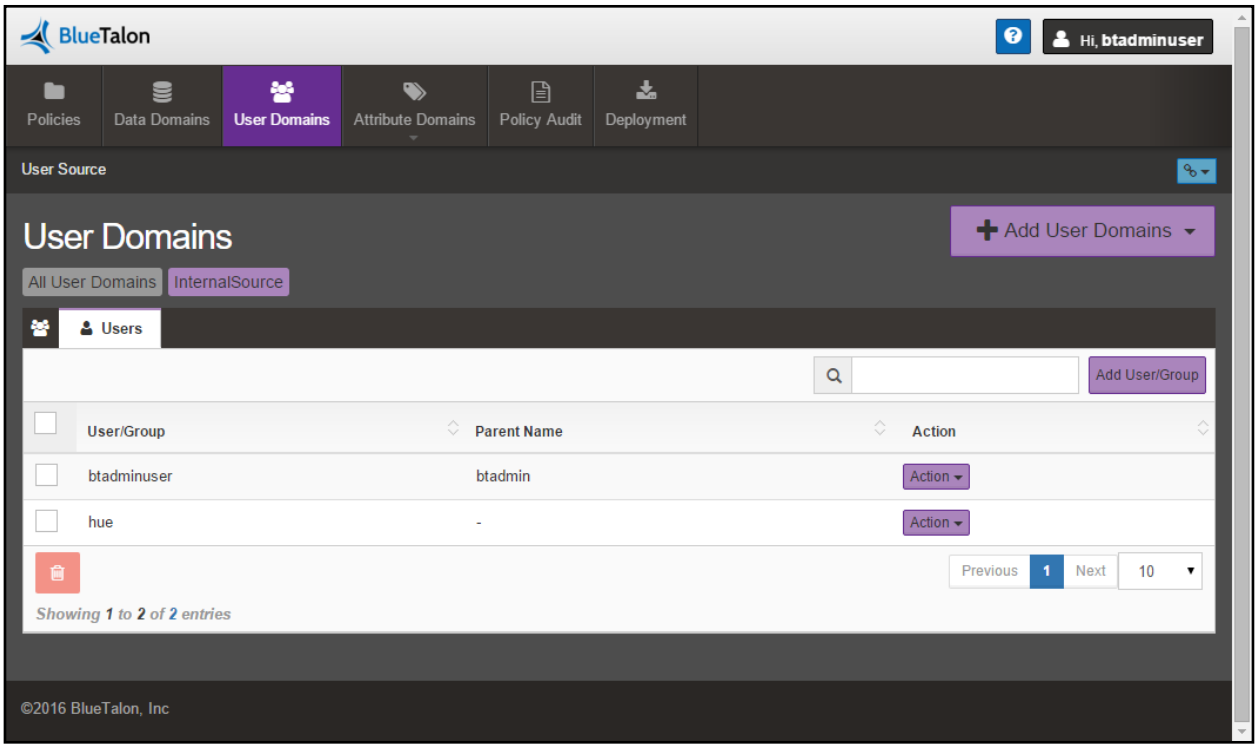
For testing, you can use the InternalSource user domain already included in the default installation.

Examine the InternalSource user domain:

1. In the Policy Console, select the User Domains tab. The InternalSource user domain is shown in the list of user domains.



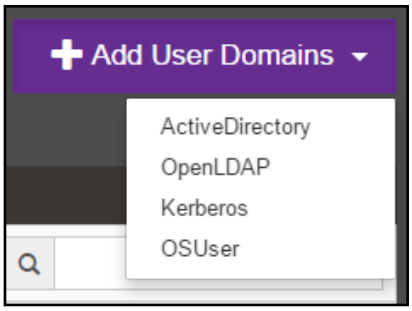
2. Click InternalSource. The users in the domain are shown.



If your organization is required to use an external user authentication system, you can add an additional user domain for that system using the Policy Console.

(Optional) To add a user domain:

1. Click the User Domains tab.
2. Click Add User Domains and select a user domain type.



3. Enter the user domain details.

Field	Description	Example
User Source Name	Name for the user domain.	TestUserDomain
Host Address	Fully qualified domain name for the user authentication system.	TestServer.TestCompany.com 127.0.0.1
Port	Port number of the user authentication system. Not required for all user	389

	authentication systems.	
Base DN	Base Distinguished Name (DN). The location in the user authentication system directory hierarchy where the user search starts from. Use the Base DN to restrict the searches on group membership to a: • Department • Location • Business function • Organizational unit The Policy Engine does not search in directories above the Base DN, but can search at one of these levels: • Base level • One level down • Down the entire hierarchy	OU=people,O=TestCompany.com • The LDAP search examines the OU=people subtree in the O=TestCompany.com directory tree. • OU is an organizational unit. • O is an organization.
Description	Description for the user domain.	Test user domain
Apply Advanced Filter (Click to show Filter field.)	Filter is a search constraint for users in the external user authentication system. Not available for all user authentication systems. Use a filter to restrict users on: • When they log in • Where they log in from • What their location is • What their last names are • Other details The Policy Engine can use any valid LDAP filter, but it does not validate the syntax of the query. For user domains, the Policy Engine only imports entries that are of type objectClass=user or objectClass=group. For attributes, the Policy Engine does not add any additional filters.	(&(objectCategory=person)(objectClass=contact)(|(sn=Jones)(sn=Hawthorne))) Returns the contacts with a last name of "Jones" or "Hawthorne".
User Name	Administration user account.	testUser
Password	Password for the administration user account.	testPassword

4. Click the Deployment tab and deploy.

Password Authentication with PostgreSQL or Hive

If you are implementing password authentication with either of these Enforcement Points:

- PostgreSQL
- Hive

You turn on password verification in the user domain.

For the InternalSource user domain, password verification is on by default.

For these user domains:

- Windows Active Directory
- OpenLDAP

You turn on password verification when using a PostgreSQL or Hive Enforcement Point. Otherwise, password verification is not used.

This does not apply to a Kerberos user domain.

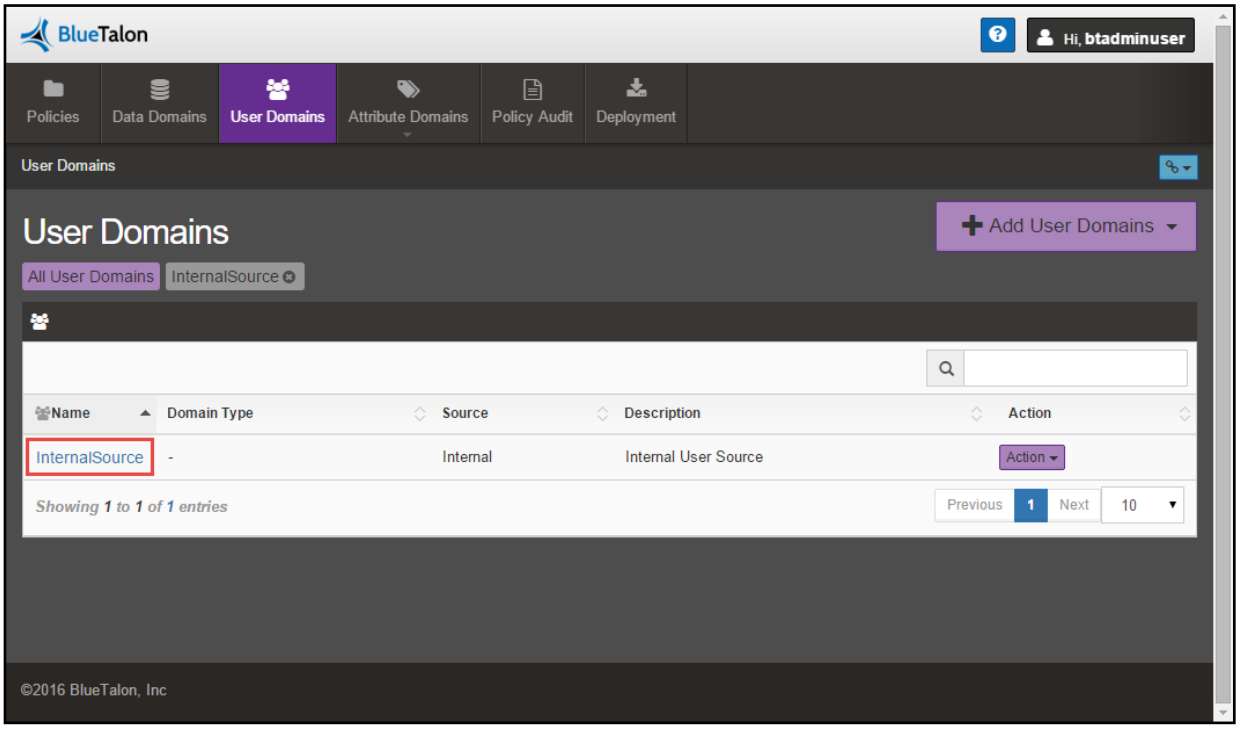
Add Group to User Domain

(Optional) You can add groups of users to a domain. **You do not have to perform these steps for initial installation and testing.**

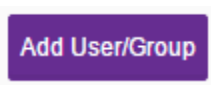
For all user domains except InternalSource, groups are added outside of BlueTalon.

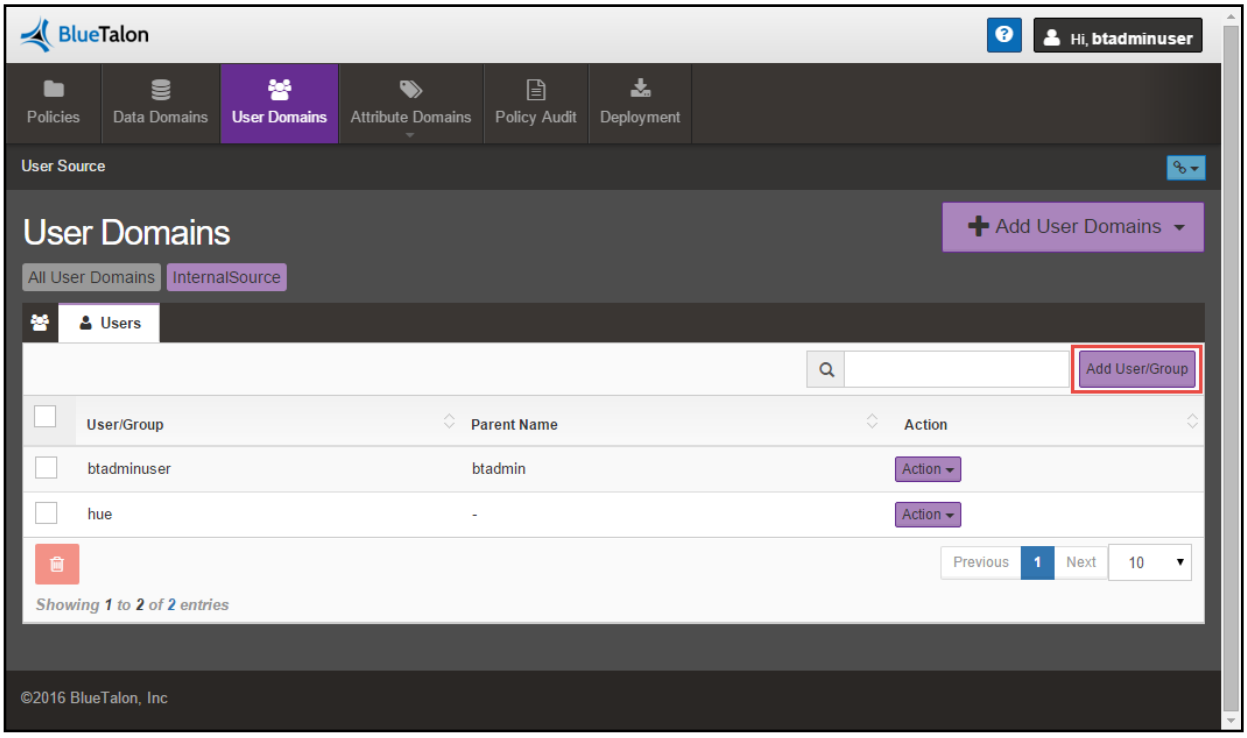
To add a group to the InternalSource user domain:

1. Select the User Domains tab.
2. Click the user domain. Example: InternalSource.



3. Click Add User/Group on the right of the screen.

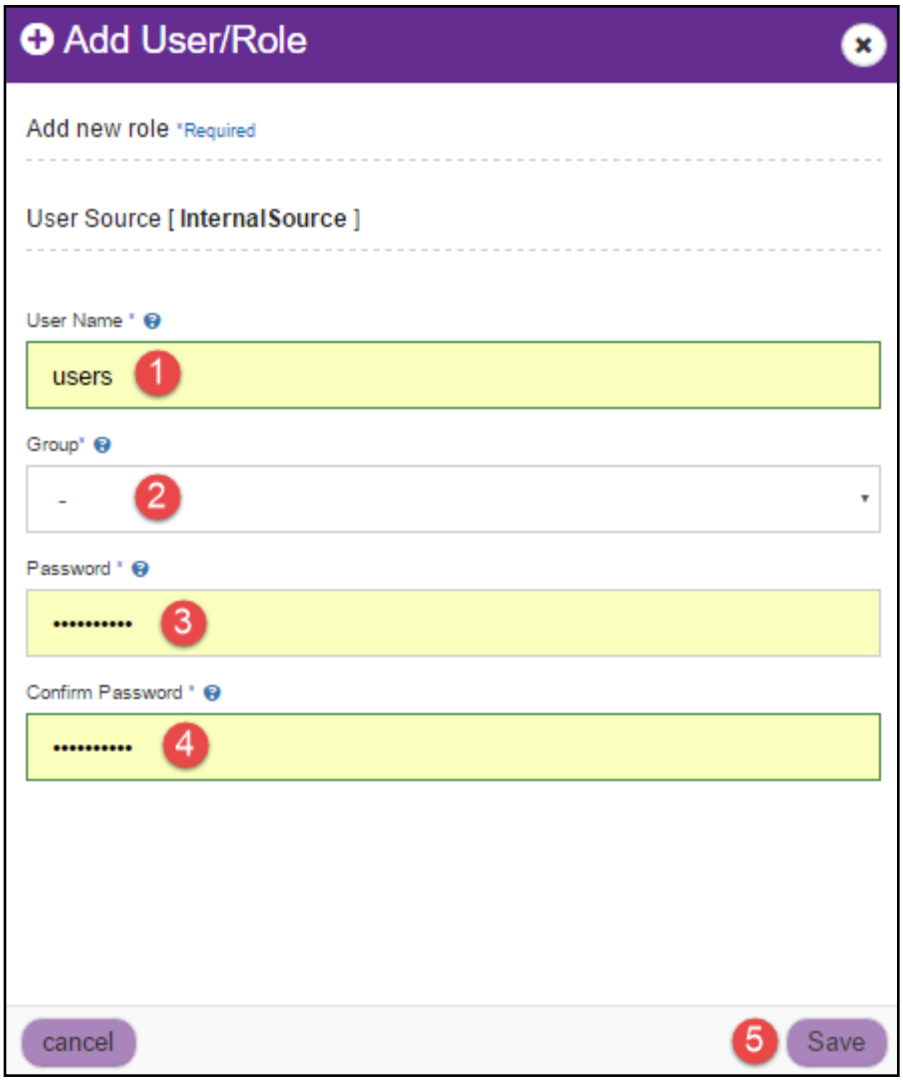




4. Add a group called users. Group names must be alphanumeric strings without spaces. To add a group:

    a. Specify the group name in the User Name field. See item 1 in the following screenshot.

b. Leave the Group field set to "-". See item 2 in the following screenshot.



c. Set a password. See items 3 and 4 in the previous screenshot.

d. Click Save. See item 5 in the previous screenshot.

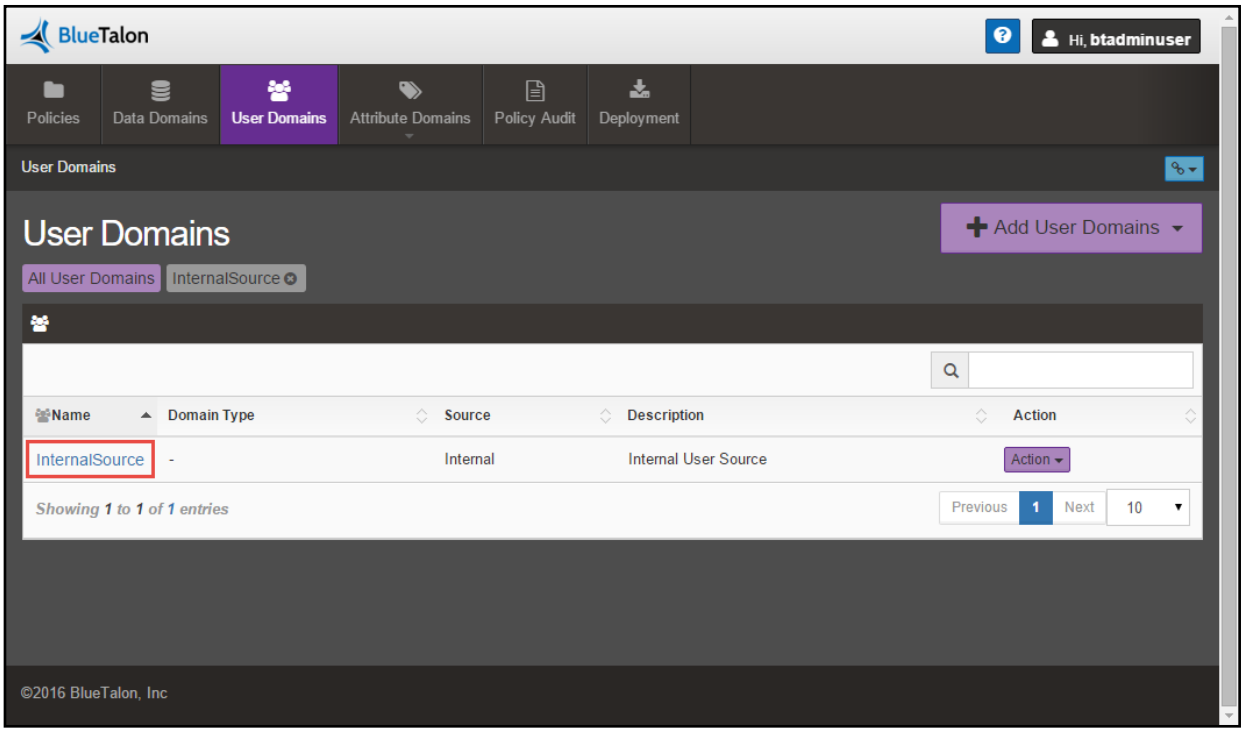
5. Click the Deployment tab and deploy.

Add User to Internal User Domain

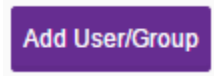
For testing, add a user named alice to the Internal user domain:

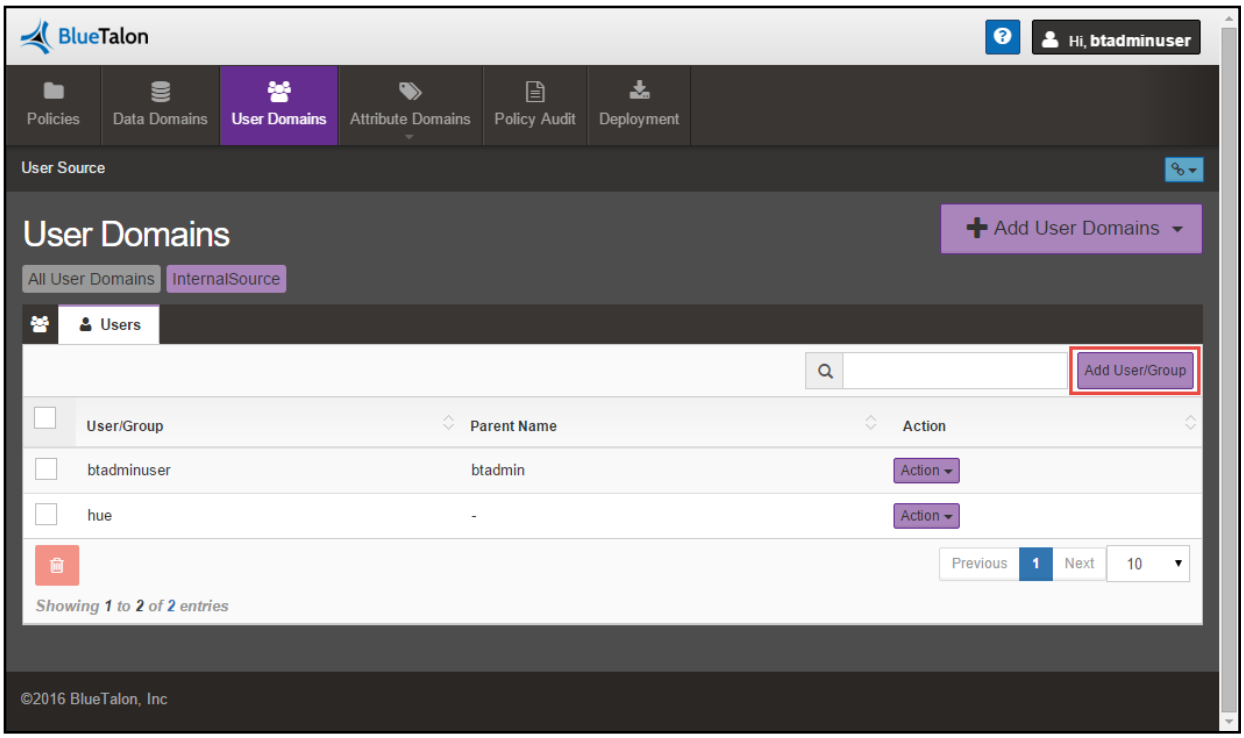
1. Click the User Domains tab.

2. Click the user domain InternalSource.



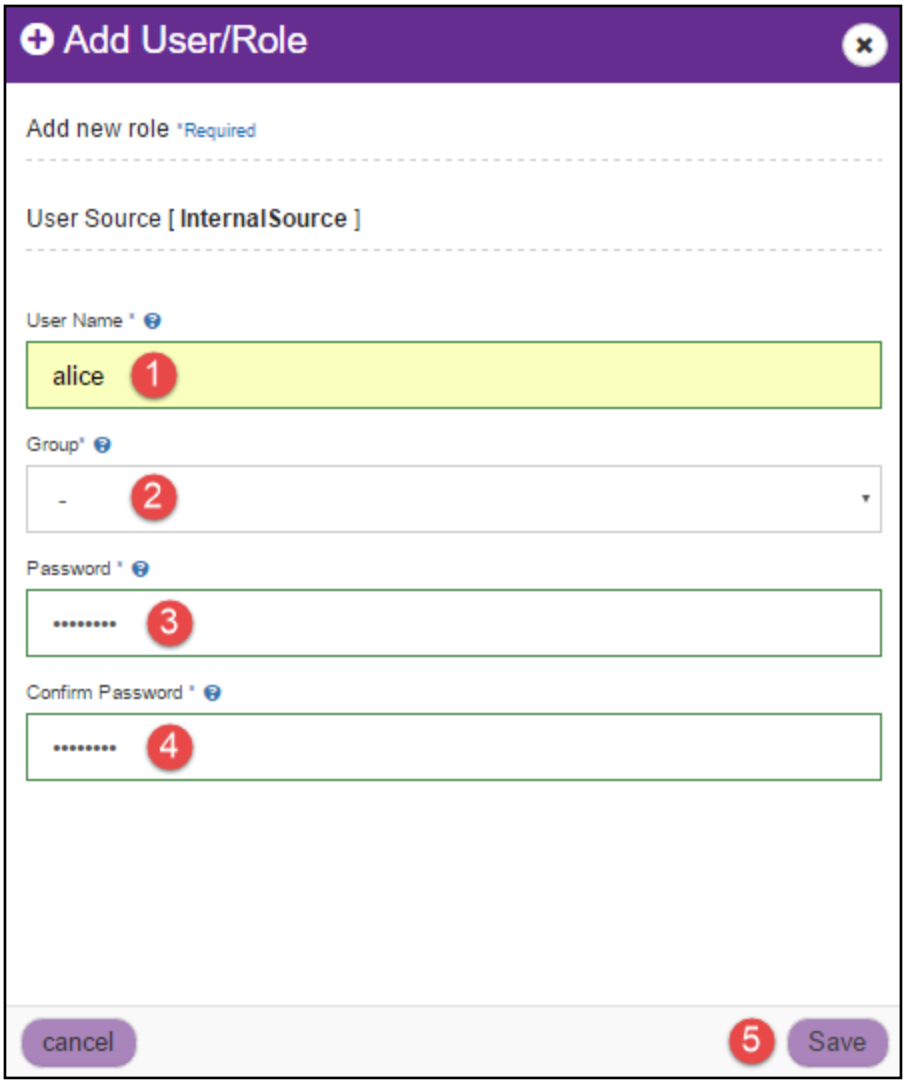
3. Click Add User/Group on the right of the screen.





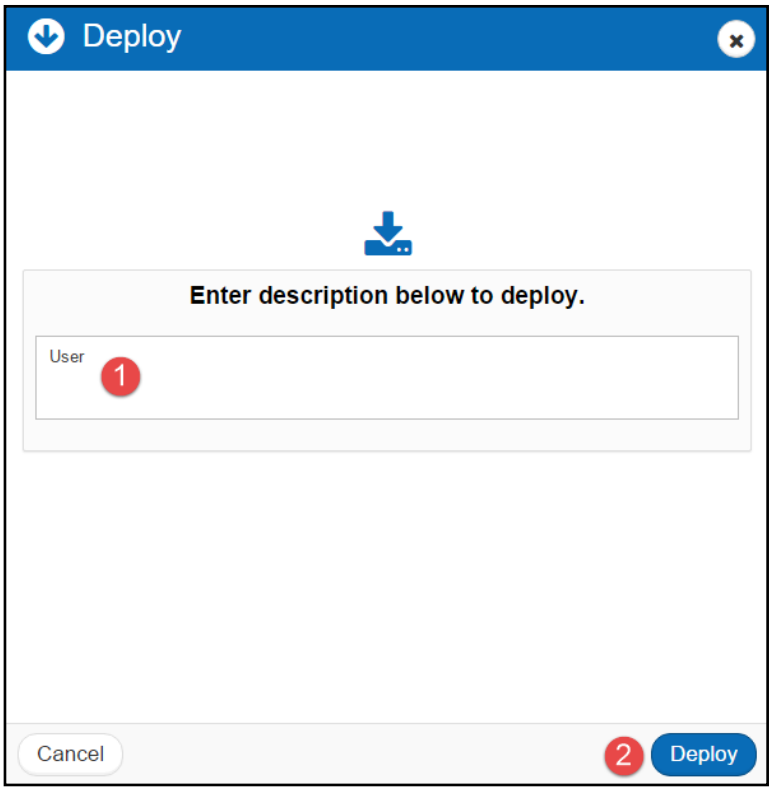
4. Set the user name to alice. See item 1 in the following screenshot.

5. (Optional) Select a previously created group or select "-" from the Group selection. See item 2 in the following screenshot.



6. Set a password. See items 3 and 4 in the previous screenshot.
7. Click Save. See item 5 in the previous screenshot.

8. Click the Deployment tab and deploy.



Set Up Enforcement Point for On-Premises Relational Databases

BlueTalon Enforcement Points intercept queries from applications and sends them to the BlueTalon Policy Engine. The Policy Engine parses the queries according to defined policies and rules. A policy-compliant query is returned to the Enforcement Points and a policy-compliant dataset result is returned to the user.

This section describes how to set up a BlueTalon Enforcement Point for on-premises relational databases.

- You run the commands on the computer designated as your BlueTalon Enforcement Point computer.
- Many of the steps in the PostgreSQL Enforcement Point section are common to the other relational database Enforcement Points.
- You should read the PostgreSQL Enforcement Point section first before setting up other types of Enforcement Points.
- Other Enforcement Point sections refer to steps in the PostgreSQL Enforcement Point section.

Database Table for Testing Enforcement Point

As you will see later when you perform the steps for installing an Enforcement Point that protects a relational database, this document instructs you to create a database table named accounts for testing your Enforcement Point. You can also use your own database table.

Example accounts table for PostgreSQL:

```
CREATE TABLE accounts (id VARCHAR(15), name VARCHAR(25), phone
VARCHAR(20), birthdate VARCHAR(10), soc_sec_no VARCHAR(15), zip BIGINT,
credit_card BIGINT, balance DECIMAL(4,2));
```

As part of the Enforcement Point testing described later, you will add a rule to mask the soc_sec_no column.

If you use your own database table, and do not want to create the accounts table, you will select a column from your own database table to mask.

PostgreSQL Enforcement Point

This section describes how to set up an Enforcement Point for PostgreSQL.

Collect Information

The following table shows the information you must collect before installing the Enforcement Point.

Item	Description
Database	Database name. Example: TestDatabase
Host address	Fully qualified domain name or IP address of computer running the database.

	Examples: • TestServer.TestCompany.com • 127.0.0.1
Port	Port the database receives JDBC connections. Example: 5432
User ID	User ID for a database account with privileges to obtain metadata. If the Enforcement Point is configured with the proxy authentication feature, then this account is used to connect to the database for all users. You will learn more about proxy authentication later.
Password	Password for the database account.

Create Test Data

This section describes how to create a:

- Database user named alice.
- Database table named accounts.

If your database already contains these items, you can skip this section.

Connect to the database using the appropriate database client as a user with privileges to create other users. Example format for PostgreSQL:

```
psql -h <fully qualified domain name or computer IP address> -p <port>
-U <privileged user> -d <database name>
```

Examples for PostgreSQL:

```
psql -h TestServer.TestCompany.com -p 5432 -U postgres -d postgres
```

```
psql -h 127.0.0.1 -p 5432 -U postgres -d postgres
```

Create a database user named alice. Ensure alice has privileges to log in and create tables. Example for PostgreSQL:

```
CREATE USER alice WITH PASSWORD 'mypassword';
```

Use your own strong password.

Exit the database client and connect to the database as alice. Example for PostgreSQL:

```
\q
```

```
psql -h 127.0.0.1 -p 5432 -U alice -d postgres
```

Create a database table named accounts and add example data. Example for PostgreSQL:

```
CREATE TABLE accounts (id VARCHAR(15), name VARCHAR(25), phone
VARCHAR(20), birthdate VARCHAR(10), soc_sec_no VARCHAR(15), zip BIGINT,
credit_card BIGINT, balance DECIMAL(4,2));
```

```
INSERT INTO accounts (id, name, phone, birthdate, soc_sec_no, zip,
credit_card, balance) VALUES ('147274739-9', 'Frances Harrison', '9-
(192)357-8851', '10/27/1956', '993941527', 37726, 6393451850134970,
52.90);
```

```
INSERT INTO accounts (id, name, phone, birthdate, soc_sec_no, zip,
credit_card, balance) VALUES ('457389751-8', 'Gregory Patterson', '3-
(684)454-2444', '11/22/1969', '233575483', 21278, 5602245983499780,
44.87);
```

```
INSERT INTO accounts (id, name, phone, birthdate, soc_sec_no, zip,
credit_card, balance) VALUES ('360272064-0', 'Earl James', '5-(177)394-
3277', '05/23/1983', '303640766', 26595, 3534227478434860, 3.36);

INSERT INTO accounts (id, name, phone, birthdate, soc_sec_no, zip,
credit_card, balance) VALUES ('551058833-0', 'Christine Robertson', '5-
(437)964-8463', '05/11/1988', '318794141', 50716, 5602223026663730,
5.22);

INSERT INTO accounts (id, name, phone, birthdate, soc_sec_no, zip,
credit_card, balance) VALUES ('713553396-8', 'Daniel Crawford', '1-
(875)657-9518', '06/11/1999', '214894670', 28693, 3571338359613280,
22.63);

INSERT INTO accounts (id, name, phone, birthdate, soc_sec_no, zip,
credit_card, balance) VALUES ('359936857-0', 'Shirley Holmes', '6-
(362)871-3036', '04/01/1980', '964490690', 13189, 3534219304003200,
30.17);

INSERT INTO accounts (id, name, phone, birthdate, soc_sec_no, zip,
credit_card, balance) VALUES ('887693755-2', 'Kelly Adams', '9-
(887)292-8810', '02/12/1988', '304475814', 84901, 5602248578416630,
35.09);

INSERT INTO accounts (id, name, phone, birthdate, soc_sec_no, zip,
credit_card, balance) VALUES ('091619799-9', 'Bonnie Freeman', '2-
(163)102-1214', '01/26/1971', '940518723', 73248, 3547967170434520,
69.74);

INSERT INTO accounts (id, name, phone, birthdate, soc_sec_no, zip,
credit_card, balance) VALUES ('151872601-1', 'Joyce Mcdonald', '3-
(504)572-0648', '09/18/1964', '919591100', 82573, 5401856433043540,
88.15);

INSERT INTO accounts (id, name, phone, birthdate, soc_sec_no, zip,
credit_card, balance) VALUES ('956961211-8', 'Karen Burton', '1-
(533)256-2922', '03/05/1954', '479488766', 56134, 3536534268148670,
7.08);
```

Examine the id, name, and soc_sec_no columns in the accounts table. Example for PostgreSQL:

```
SELECT id, name, soc_sec_no FROM accounts;
     id      |        name         | soc_sec_no
-------------+---------------------+-------------
 147274739-9 | Frances Harrison    | 993941527
 457389751-8 | Gregory Patterson   | 233575483
 360272064-0 | Earl James          | 303640766
 551058833-0 | Christine Robertson | 318794141
 713553396-8 | Daniel Crawford     | 214894670
 359936857-0 | Shirley Holmes      | 964490690
 887693755-2 | Kelly Adams         | 304475814
 091619799-9 | Bonnie Freeman      | 940518723
 151872601-1 | Joyce Mcdonald      | 919591100
 956961211-8 | Karen Burton        | 479488766
```

```
(10 rows)
```

You will create a rule to mask the social security number stored in the soc_sec_no column later.

Add Data Domain

A data domain stores information about an external data source.

You import the external data source information into a data domain: For relational database sources like PostgreSQL, you can import table and column information.

You can add a data domain from the BlueTalon Policy Console or a command line shell using the BlueTalon REST API.

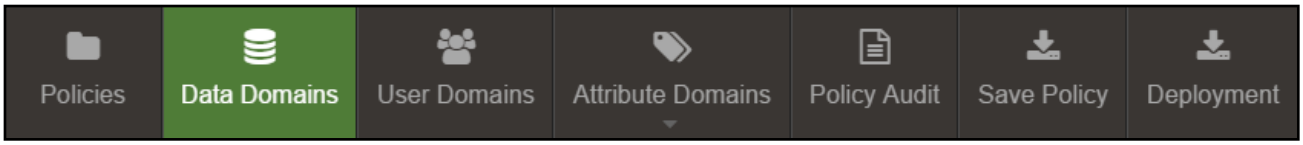
Add Data Domain from Policy Console

Add a data domain from the Policy Console:

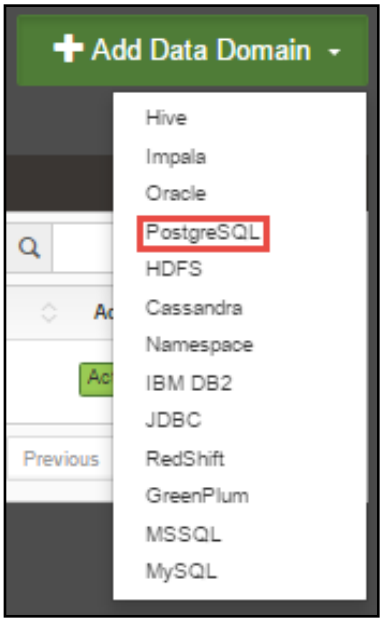
1. Open Web browser.
2. Go to the Policy Console.
3. Log in with the user name and the password.

   Defaults:
   btadminuser
   P@ssw0rd

   You can also configure the Policy Console to use OpenLDAP or Windows Active Directory for authentication.
4. Click the Data Domains tab.

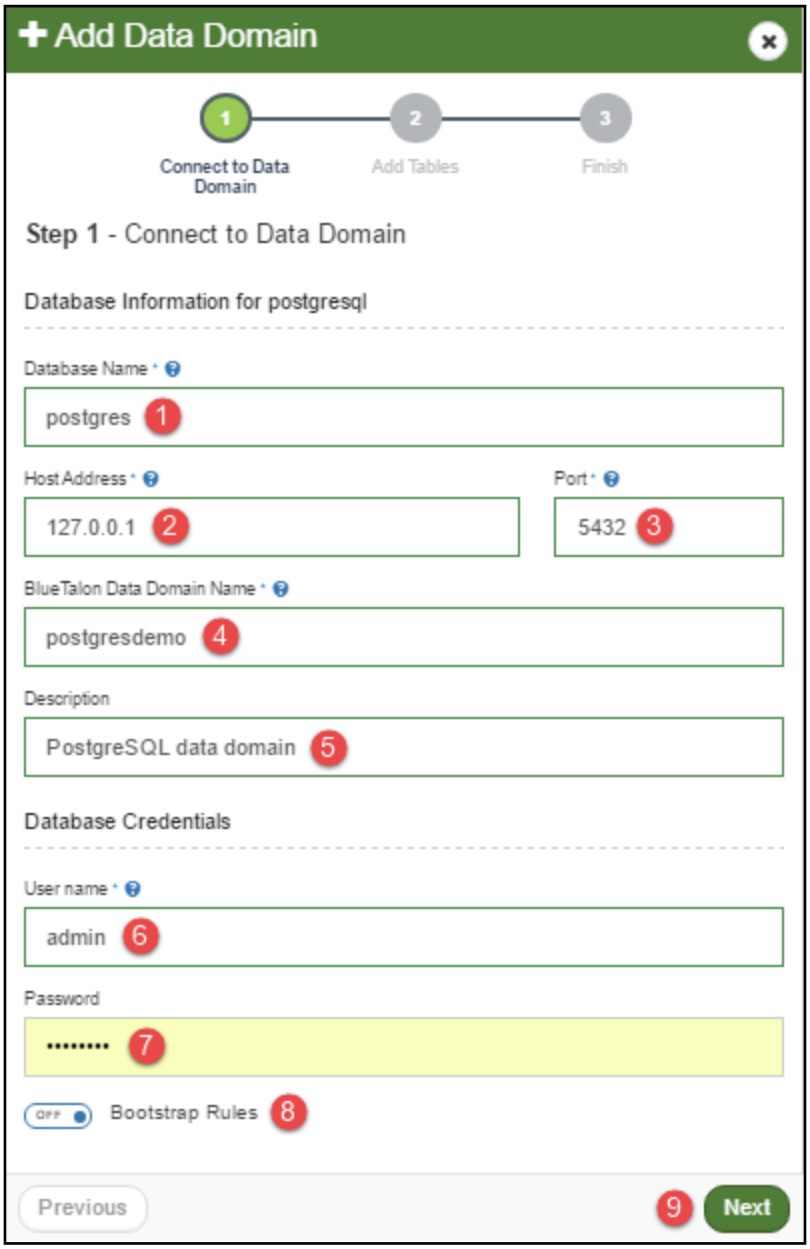


5. Select Add Data Domains on the right of the window. Select your data domain type. Example: PostgreSQL.



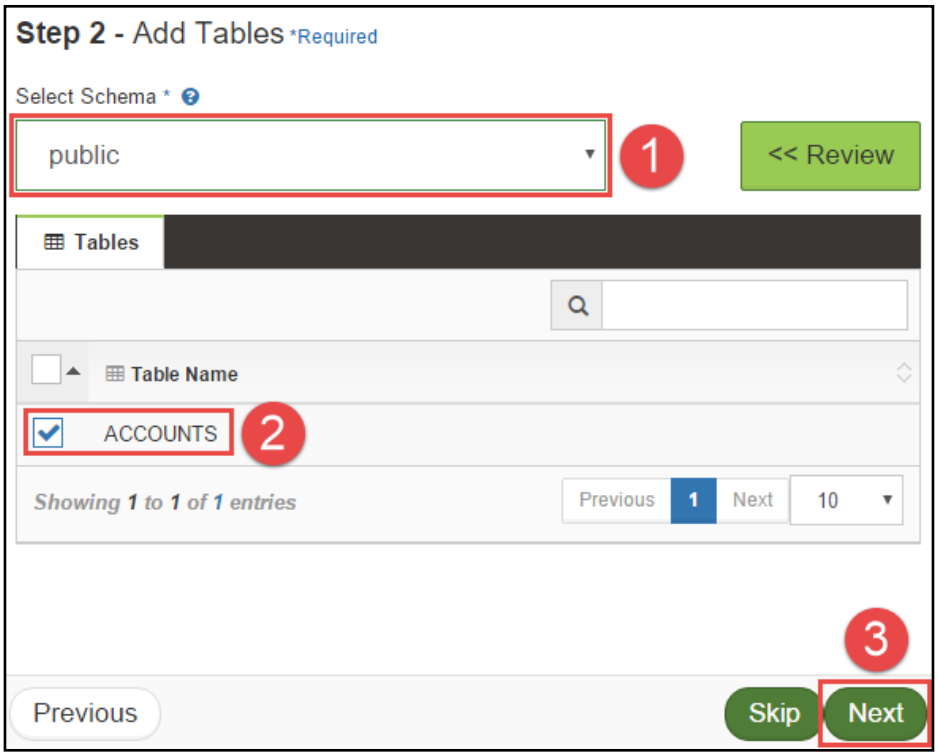
6. Enter the data domain details. See Collect Information on page 52, which varies for each database type. If you are adding a data domain for a database other than PostgreSQL, use the details for your database type.
    a. Database name. See item 1 in the following screenshot.
    b. Host address. See item 2 in the screenshot.
    c. Port. See item 3 in the screenshot.
    d. Data domain name. You will need the name later when you configure the Enforcement Point. See item 4 in the screenshot.

       Examples:
       postgresdemo
       PostgreSQLDataDomain
    e. Data domain description. See item 5 in the screenshot.
    f. Database user ID. See item 6 in the screenshot.
    g. Password for the database user ID. See item 7 in the screenshot.
    h. Bootstrap Rules. Options: On or off. When set to on, you provide the location of a JSON file with predefined security rules. See item 8 in the screenshot. For initial testing, you can leave the option off.



7. Click Next. See item 9 in the previous screenshot.
8. If the Policy Console cannot connect to the database, then you will see an error message. You must:
    a. Ensure the database is running.
    b. Ensure you entered the correct database connection information.
    c. Exit the data domain set up, resolve the database connection problem, and then restart the data domain set up steps from the beginning.
9. In the next screen:
    a. Select the schema. Example: Public. See item 1 in the following screenshot.

b. Select the accounts table. See item 2 in the following screenshot.



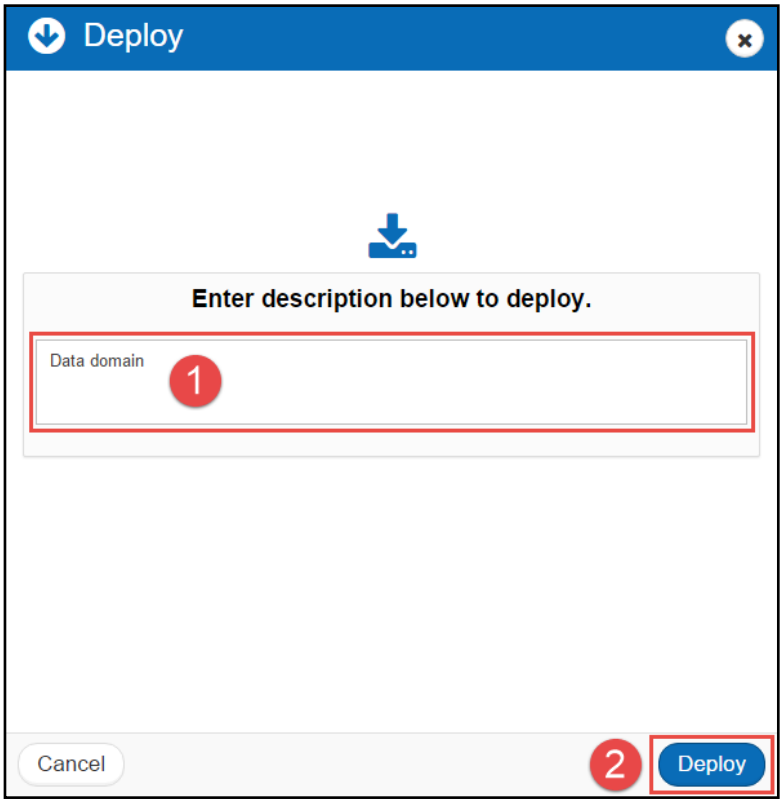
c. Click Next. See item 3 in the previous screenshot. The metadata information is imported.

10. Click the Deployment tab.

11. In the deployment screen:

a. Enter an optional description. See item 1 in the following screenshot.

b. Click Deploy. See item 2 in the screenshot. This implements your changes in the run-time system.



12. If this is your first deployment, an Enforcement Point with a new configuration is created. You must turn on the new enforcement point using an SSH connection to the computer where the Enforcement Point package is installed.

Notes:

- You might have to authenticate multiple times as you proceed with the installation.
- Enter your user name and password. Example: btadminuser with a password of P@ssw0rd.

Add Data Domain Using REST API

(Optional) You can use the REST API to create data domains. To view the parameters, see https://policy.readme.io/docs/resource_domains.

The following REST API example creates a PostgreSQL data domain:

```
curl -u btadminuser:P@ssw0rd --header "Content-type: application/json"
--request POST
http://localhost:8111/PolicyManagement/1.0/resource_domains --data '{
    "db_type": "postgresql",
    "hostname": "127.0.0.1",
    "port": "5432",
    "db_name": "postgres",
    "resource_domain_name": "PostgreSQLDataDomain",
    "description": "PostgreSQL data domain",
    "login_auth": "true",
    "username": "postgres",
    "password": "datalake",
    "schema": "public"
}'
```

Example return result from the previous command:

```
{
  "status":"success",
  "message":"Resource domain added successfully",
  "timestamp":"04/21/2016-04:57:14"
}
```

Example REST API deploy command:

```
curl -u btadminuser:P@ssw0rd  --header "Content-type: application/json"
--request PUT http://localhost:8111/PolicyManagement/1.0/deploy --data
'{"user":"btadminuser","description":"Deployed change"}'
```

Always ensure you deploy the new data domain.

Install Enforcement Point Package

Install the Enforcement Point package:

1. Enter:
   `yum install bluetalon-ep-3.2.4`
2. Follow the prompts.

Configure Enforcement Point Using Setup Script

Configure the Enforcement Point:

1. Run:
   `/opt/bluetalon/3.2.4/pep/scripts/bluetalon-ep-setup`

   You can also run the script in silent mode, which does not prompt you to enter details. See the following section for details.
2. Accept the license agreement.
3. Enter the information.

Item	Description	Examples
Enforcement Point type	Type of Enforcement Point. Options: • PostgreSQL • Hive • Impala • Oracle • IBM DB2 • Fsshell • HDFS • FSEP • MySQL • MSSQL • Cassandra • Spark SQL	To install an Enforcement Point for PostgreSQL, select the option for PostgreSQL.

	Note: For these data source types, you pick PostgreSQL as the Enforcement Point type: • RedShift • GreenPlum • Amazon RDS for PostgreSQL	
Enforcement Point port	JDBC end point port that client applications use to obtain secure data.	1557
Database Hostname	Fully qualified domain name or IP address of the computer running the database. Use the value set in the Policy Console for the data domain you created earlier.	localhost 127.0.0.1 TestServer.TestCompany.com
Database Port	Port for the database. Use the value set in the Policy Console for the data domain you created earlier.	5432
Enforcement Point mode	Options: • Forward Only Install the Enforcement Point but do not audit or enforce rules. • Audit Only Push audit messages to the bt-audit-kafka service. • Enforce Only Enforce rules provided by the bt-policy-engine service. • Audit and Enforce Push audit messages to the bt-audit-kafka service and enforce rules provided by the bt-policy-engine service.	Audit and Enforce
Policy package hostname	Fully qualified domain name or IP address of the computer on which the BlueTalon Policy Engine is running.	10.0.1.7
Port for bt-policy-configuration service	Port for the BlueTalon Policy Engine.	1600
Data domain name	Name of the data domain. Use the value set in the Policy Console for the data domain you created earlier.	PostgreSQLDataDomain postgresdemo
Custom data domain name to use in the service and configuration file name	Typically set this domain name to the same value as the previous data domain value.	PostgreSQLDataDomain postgresdemo
Validate password against a user domain or delegate authentication to the database server. This question asks if you want the Enforcement Point to take the password provided to the database client application and check	Options: • Yes • No	Yes (To use proxy authentication, set to yes. Proxy authentication is described in the secton "Configure Enforcement Point for Proxy Authentication".)

the password with the user domain.		
Validate password by the enforcement point, the user ID and password can be the values entered by the end user or can be provided at the time of defining the Data Domain as a proxy user. This user account must be present on the database also. This question asks if you want the Enforcement Point to take the user ID and password set in the data domain. A matching user account must be created in the database.	Options: • Yes • No	Yes (To use proxy authentication, set to yes. Proxy authentication is described in the secton "Configure Enforcement Point for Proxy Authentication".)
Hostname where the bt-audit-kafka service is running	Fully qualified domain name or IP address of the computer on which the bt-audit-kafka service is running.	TestSever.dx.internal.cloudapp.net

Example Run of Enforcement Point Setup Script

Example run of Enforcement Point setup script for PostgreSQL:

```
/opt/bluetalon/3.2.4/pep/scripts/bluetalon-ep-setup

<Omitted license agreement for brevity>

I understand the terms and accept BlueTalon Online Evaluation License.
(yes/no) [no] : yes


Checking prerequisites for bluetalon ep package

Check for required commands...

Check for rm ............ PASS

Check for mv ............ PASS

Check for tr ............ PASS

Check for su ............ PASS

Check for ln ............ PASS

Check for cp ............ PASS

Check for sed ........... PASS

Check for cat ........... PASS

Check for awk ........... PASS

Check for echo .......... PASS

Check for read .......... PASS

Check for grep .......... PASS

Check for unset ......... PASS

Check for mkdir ......... PASS
```

```
Check for chown ......... PASS
Check for chmod ......... PASS
Check for source ........ PASS
Check for export ........ PASS
Check for command ....... PASS
Check for chkconfig ..... PASS
Check for getent ........ PASS


Select the enforcement point type:
(1)  - PostgreSQL
(2)  - Hive
(3)  - Impala
(4)  - Oracle
(5)  - IBM DB2
(6)  - Fsshell
(7)  - HDFS
(8)  - FSEP
(9)  - RedShift
(10) - GreenPlum
(11) - MySQL
(12) - MSSQL
(13) - Cassandra
(14) - Spark SQL

Enter choice: 1

Enter the port that the enforcement point will listen on [1557] :

Provide the PostgreSQL database information:

Hostname [localhost] :

Port [5432] :

Select the configuration of audit and enforcement behavior of the
enforcement point:
```

```
(1) - Forward Only.      This will install enforcement point but not
audit or enforce rules.

(2) - Audit Only.        This will push audit to bt-audit-kafka
service.

(3) - Enforce Only.      This will enforce rules provided by bt-policy-
engine service.

(4) - Audit and Enforce. This will push audit to bt-audit-kafka service
and enforce rules provided by bt-policy-engine service.


Enter choice [4]:

Please enter the policy configuration service information in form of
HOST:PORT,HOST:PORT,....

Value [127.0.0.1:1600]:

Verify the connectivity of policy configuration service(s)...

Successfully connected with all policy configuration services.


Provide the name of the Data Domain to use with the enforcement point.
This is required.

This can be configured using the BlueTalon Policy UI either before or
after configuring the enforcement point.

Data Domain name : postgresdemo

Enter custom data domain name to use in service and conf file
[postgresdemo] :

Do you want to proceed with default name postgresdemo in service and
conf file name (yes/no) [yes] :

Default name postgresdemo is used as service and conf file name.


Enforcement point can validate a password against a User Domain or
delegate authentication to the database server.

Do you want the enforcement point to also validate the password
(yes/no)? [no] : yes


For validation of password by the enforcement point, the userid and
password can be the values entered by end-user or can be provided at
the time of defining the Data Domain as a proxy user. This user account
must be present on the database also.

Do you want to configure enforcement point to use the proxy account
(yes/no)? [no] :yes


Do you want to use specific endpoint for this enforcement point
(yes/no) [no] :


Enter the hostname where bt-audit-kafka service is running [XXX] :
```

```
Please confirm the hostname is XXX (yes/no)? [yes] :
Configured bt-audit-kafka service for use with the enforcement point.
Starting bt-postgresql-ep-postgresdemo service:            [  OK  ]
```

Configure Enforcement Point Using Setup Script in Silent Mode

Ensure you have deployed your data domain before deploying an Enforcement Point.

(Optional) You can also run the script in silent mode, which does not prompt you to enter details.

To configure the Enforcement Point with a preconfigured XML file, run:

```
/opt/bluetalon/3.2.4/pep/scripts/bluetalon-ep-setup –accept –f
/tmp/postgresqlep.xml
```

The following sections show the XML file format, the parameters, and an example XML file.

Enforcement Point Silent Mode XML File Format

Format for the Enforcement Point silent mode XML file:

```
<EnforcementPoint xmlns="">
  <Parameter>Value</Parameter>
  <Parameter>Value</Parameter>
  ...
</EnforcementPoint>
```

Enforcement Point Silent Mode XML File Parameters

The following table shows the parameters for the Enforcement Point silent mode XML file.

Parameter	Optional or Required	Example	Description
hostname	Optional	localhost	Hostname of data source (default is localhost).
port	Optional	5432 (for PostgreSQL)	Port on which the data source is listening.
impala_shell_port	Optional	21000	Impala Shell port. For Impala data domain only.
audit_ip	Optional	localhost	IP address of Kafka broker.
audit_port	Optional	9093	Port on which Kafka is listening.
resource_domain_name	Required	MyDataDomain	Name of data domain.
custom_name	Optional	MyCustomName (Default is value for resource_domain_name)	Data domain custom name. Creates a service file with the name specified in custom_name instead of the data domain name.
db_type	Required	1	Database type. Examples: 1. PostgreSQL 2. Hive 3. Impala

			4. Oracle 5. IBM DB2 6. Fsshell 7. HDFS 8. FSEP 9. RedShift 10. GreenPlum 11. MySQL 12. MSSQL 13. Cassandra 14. Spark
pdp_hostname	Optional	localhost	Host name where Policy Engine is running. For DB2 only.
pdp_port	Optional	1555	Port on which Policy Engine is listening. For DB2 only.
ep_port	Optional	1557	Port on which Enforcement Point is listening.
impala_shell_ep_port	Optional	1558	Port on which Impala Shell Enforcement Point is listening. For Impala data domain only.
mode	Optional	4	Enforcement Point mode: 1. Forward Only. Installs Enforcement Point, but does not audit or enforce rules. 2. Audit Only. Push audit to bt-audit-kafka service. 3. Enforce Only. Enforce rules provided by bt-policy-engine service. 4. Audit and Enforce. Push audit to bt-audit-kafka service and enforce rules provided by bt-policy-engine service.
db_name	Required	MyDatabase	Name of database. Required for DB2.
schema	Optional	db2admin	Schema name. Required for DB2.
dbtcp_ip (deprecated)	Optional	localhost	Host address where policy configuration service is running.
dbtcp_port (deprecated)	Optional	1600	Port on which policy configuration service listens.
pwd_auth	Optional	yes	Enables password authentication for PostgreSQL, Hive, and Impala data domain. If parameter is enabled, then the password of the end user is also verified. If parameter is disabled, only the user name is verified.
proxy_auth	Optional	yes	Enables account delegation. Only used if pwd_auth is enabled. If parameter is enabled, then instead of passing the end user credentials to the back end database, the credentials used at the time of defining the data domain are passed.
namenode_lock_down	Optional	no	Used for HDFS plug in only. If parameter is disabled, then the HDFS plug in allows

			queries. If parameter is enabled, then queries are denied.
kerberos	Optional	yes	Enables Enforcement Point for Kerberos. Required for Hive or Impala data domain.
bt_keytab_path	Required	/home/ec2-user/bluetalon.keytab	Path of BlueTalon service key tab file. Used to impersonate Kerberos account. Required if Kerberos is enabled.
user_keytab_path	Required	/home/ec2-user/MyUser.keytab	Path of user key tab file. Used to authenticate with Kerberos server. Required if Kerberos is enabled.
user_principal	Required	MyUser	Name of user principal, whose key tab file is supplied. Required if Kerberos is enabled.
traffic_divert	Optional	yes	When enabled, diverts HDFS commands to FSEP.
hadoop_conf_path	Optional	/etc/hadoop/conf/	Directory where Hadoop configuration files are located. Files are used to configure FSEP.
hdfs_principal	Required	hdfs/sandbox.bluetalon.com@EXAMPLE.COM	HDFS service principal. Required if Kerberos is enabled for FSEP.
hdfs_keytab	Required	/home/ec2-user/hdfs.keytab	Path of key tab file for HDFS service principal. Required if Kerberos is enabled for FSEP.
http_principal	Required	HTTP/sandbox.bluetalon.com@EXAMPLE.COM	HTTP service principal. Required if Kerberos is enabled for FSEP.
http_keytab	Required	/home/ec2-user/http.keytab	Path of key tab file for HDFS service principal. Required if Kerberos is enabled for FSEP.
enable_zookeeper	Optional	yes	If enabled, adds ZooKeeper information in fsep-site.xml.
zookeeper_connection	Required	127.0.0.1:2123	ZooKeeper server information (host:port,host:port,host:port,...). Required if enable_zookeeper is enabled.
check_audit_service	Optional	true	If enabled, ensures audit service is running.
check_policy_service	Optional	true	If enabled, ensures PDP service is running.
ambari_login	Optional	admin	Ambari user name. Required to run bluetalon-post-install-conf-ambari script on Ambari server.
ambari_passwd	Optional	password	Ambari password. Required to run bluetalon-post-install-conf-ambari script on Ambari server.
endpoint_tag	Optional	DEFAULT	End point tag. Enforcement Point only retrieves the list of Policy Engine servers associated with the specified tag.
policy_conf_list	Optional	172.30.0.44:1600	List of policy configuration services (IP:PORT,IP:PORT,....). If services are not found, then read the deprecated parameters dctcp_ip and dbtcp_port.
start_after_install	Optional	true	If disabled, then Enforcement Point service will not be

			started at the end of the Enforcement Point installation.
fsep_authentication_sharedsecret_enabled	Optional	true	Enables the shared secret feature.
fsep_authentication_sharedsecret_file	Optional	/home/${USER}/.fsepSharedSecret	File system path where shared secret files are stored. Can use ${USER} macro.
provision_shared_secrets	Optional	bluetalon:bluetalon123,hdfs:hdfs123,yarn:yarn123,mapred:mapred123,hive:hive123,ambari-qa:ambari-qa123	List of shared secrets to provision (USERNAME1:SECRET1,USERNAME2:SECRET2,…).
java_home	Optional	/usr/lib/jvm/jre-1.6.0-openjdk.x86_64	Customer specified java_home. See Java SE Runtime Environment on page 22 for supported versions. Only used for FSEP.

Example PostgreSQL Enforcement Point Silent Mode XML File

Example PostgreSQL Enforcement Point silent mode XML file:

```
<EnforcementPoint xmlns="">
  <hostname>64.13.142.164</hostname>
  <port>5432</port>
  <db_type>1</db_type>
  <resource_domain_name>POSTGRESDEMO</resource_domain_name>
  <audit_ip>192.168.0.126</audit_ip>
  <ep_port>1234</ep_port>
  <dbtcp_ip>192.168.0.126</dbtcp_ip>
  <java_home>/usr/lib/jvm/jre-1.6.0-openjdk.x86_64</java_home>
</EnforcementPoint>
```

Silent mode configuration is complete.

Configure Enforcement Point Using REST API

Ensure you have deployed your data domain before deploying an Enforcement Point.

(Optional) You can configure an Enforcement Point using the REST API.

The following table shows the command line shell parameters for adding a PostgreSQL Enforcement Point using the REST API.

Parameter Name	Data Type	Required	Description	Example
ep_port	String	Yes	Port that Enforcement Point service listens on.	5433
audit_ip	String	Yes	IP address that Audit Engine is listening on.	127.0.0.1
dbtcp_ip	String	Yes	IP address that Policy Engine is listening on.	127.0.0.1
proxy_auth	Boolean	No	Connect to data source using proxy user ID and password	

			instead of end user ID and password. Options: • true • false	
pwd_auth	Boolean	No	Validate password against a user domain or delegate authentication to the database server. Options: • true • false	

REST API example that creates a PostgreSQL enforcement point for an existing data domain named mypostgres:

```
curl -u btadminuser:P@ssw0rd --header "Content-type: application/json"
--request POST
http://localhost:8111/PolicyManagement/1.0/resource_domains/mypostgres/
enforcement_points --data '{
  "ep_port": "1557",
  "audit_ip": "127.0.0.1",
  "dbtcp_ip": "127.0.0.1",
  "proxy_auth": "yes",
  "pwd_auth": "yes"
}'
```

Example return result from the previous command:

```
{
  "status":"success",
  "message":"Enforcement point setup complete",
  "timestamp":"04/21/2016-04:57:14"
}
```

Example REST API deploy command:

```
curl -u btadminuser:P@ssw0rd  --header "Content-type: application/json"
--request PUT http://localhost:8111/PolicyManagement/1.0/deploy --data
'{"user":"btadminuser","description":"Deployed change"}'
```

Enforcement Point Files and Service

The following table shows the Enforcement Point files and service.

Item	Detail
Binary files path	/opt/bluetalon/3.2.4/pep/psql-ep/bin
Configuration file	/etc/bluetalon/pep/psql-ep/conf/bt-postgres-ep-<data domain name>.conf
Logs	/var/log/bluetalon/pep/psql-ep/logs/bt-postgresql-ep-<data domain name>.log
Service name	bt–postgresql-ep-<data domain name>

Test Enforcement Point

This section describes how to test the Enforcement Point.

Verify Enforcement Point Service is Running

Verify the Enforcement Point service is running:

1. Open a command line shell and log in as a privileged user.
2. Enter:
   ```
   service --status-all | grep bt
   ```
3. Examine the status for your Enforcement Point.

   The service name for the PostgreSQL Enforcement Point is:
   ```
   bt-postgresql-ep-<data domain name>
   ```

   Example service status result:
   ```
   bt-postgresql-ep-postgresdemo (pid  3960) is running...
   ```
4. Note the process ID shown as "pid" in the previous step. Example pid: 3960.
5. If the service is not running, enter:
   ```
   service bt-postgresql-ep-<data domain name> start
   ```
6. Examine the network statistics and search for the process ID pid, enter:
   ```
   netstat -plnt | grep 3960
   ```
7. View the IP address and port in the results from the previous step. Example:
   ```
   tcp   0  0 10.0.1.7:1557  0.0.0.0:*  LISTEN  3960/arcdspsql
   ```
8. Note the IP address and port in the previous step.

   Examples:
   IP address: 10.0.1.7
   Port: 1557

   You will need the IP address and port later when you set up the Enforcement Point to connect to the database.

If you discover dead processes in step 2, see these troubleshooting sections:

- Audit Services on page 232.
- Policy Services on page 235.
- Enforcement Point Services on page 237.

Add Policy

A policy is a set of security enforcement rules that restrict data access.

Example: Only reveal the last four digits of a social security number. Mask the other digits.

A policy:

- Allows a limited set of users to access the information they need.
- Denies all other access by default.

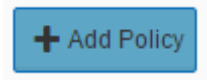
You:

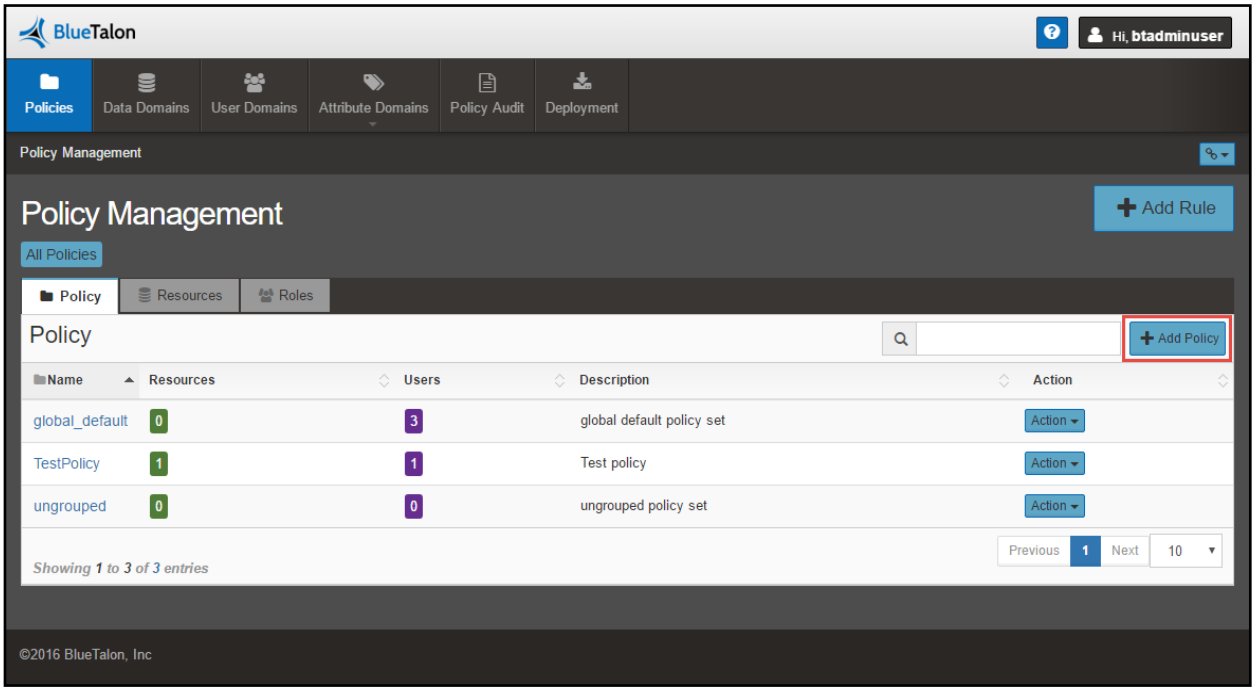
- Create a policy.

- Add users to the policy.
- Add rules to the policy.
    - Set the resource for each rule.
    - For a relational database source, a resource is a table or column.

Add a policy:

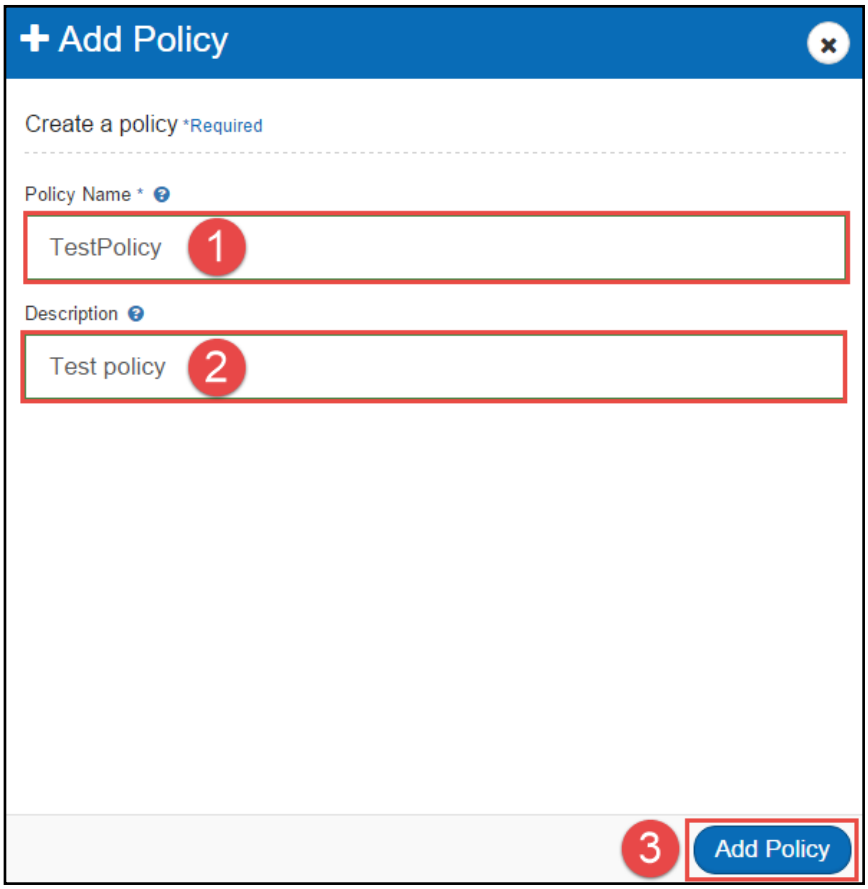
1. Click the Policies tab.
2. Click Add Policy on the right of the screen.





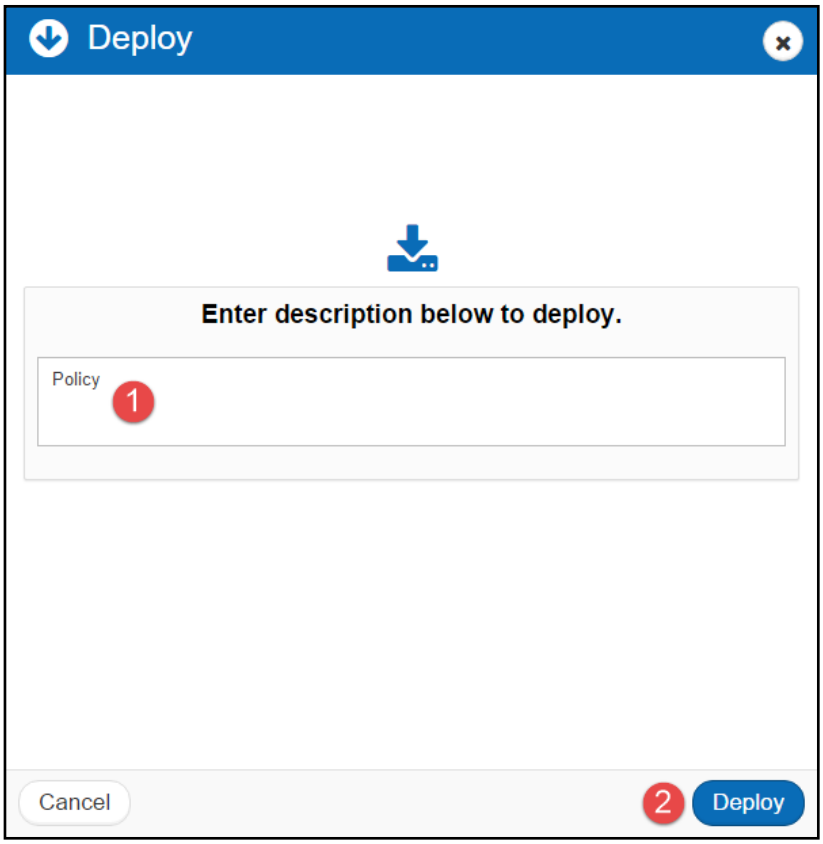
3. Enter a policy name. Example: TestPolicy. See item 1 in the following screenshot.

4. Enter a policy description. Example: Test policy. See item 2 in the following screenshot.



5. Click Add Policy. See item 3 in the previous screenshot.

6. Click the Deployment tab and deploy.

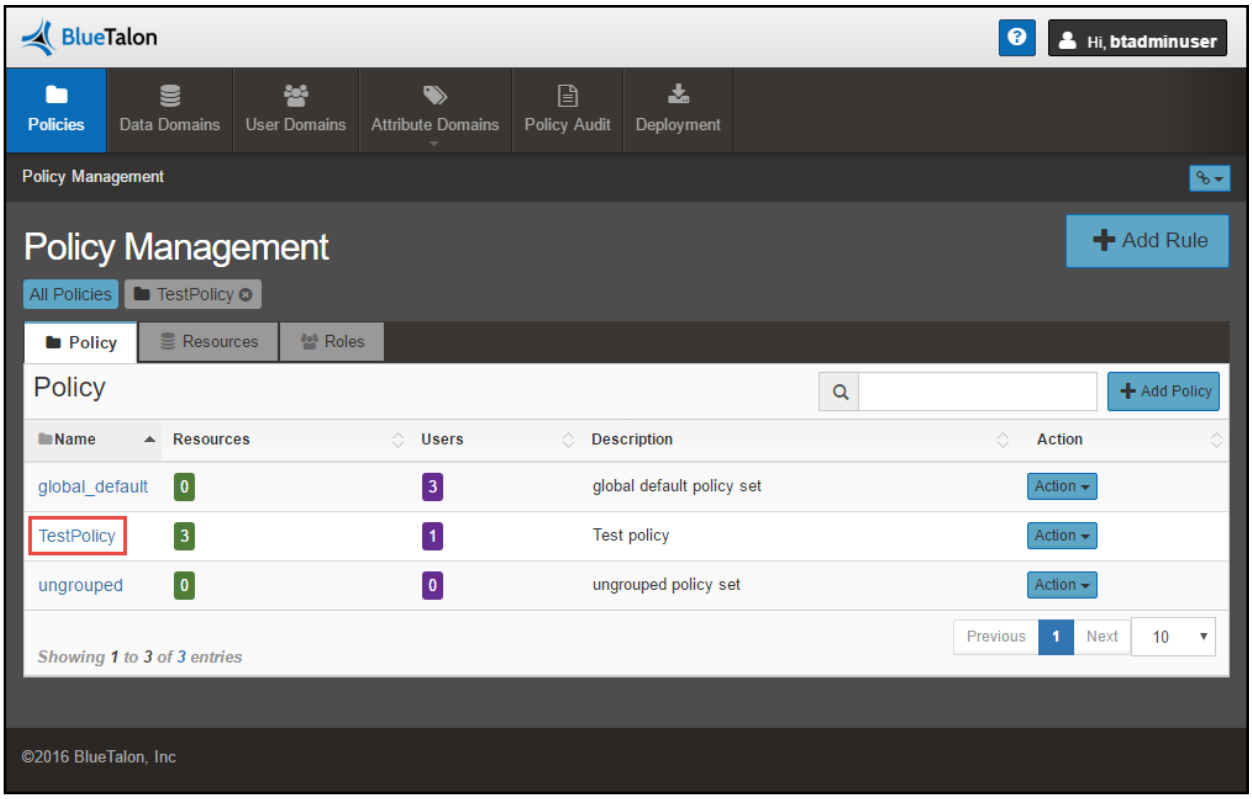


Add User to Policy

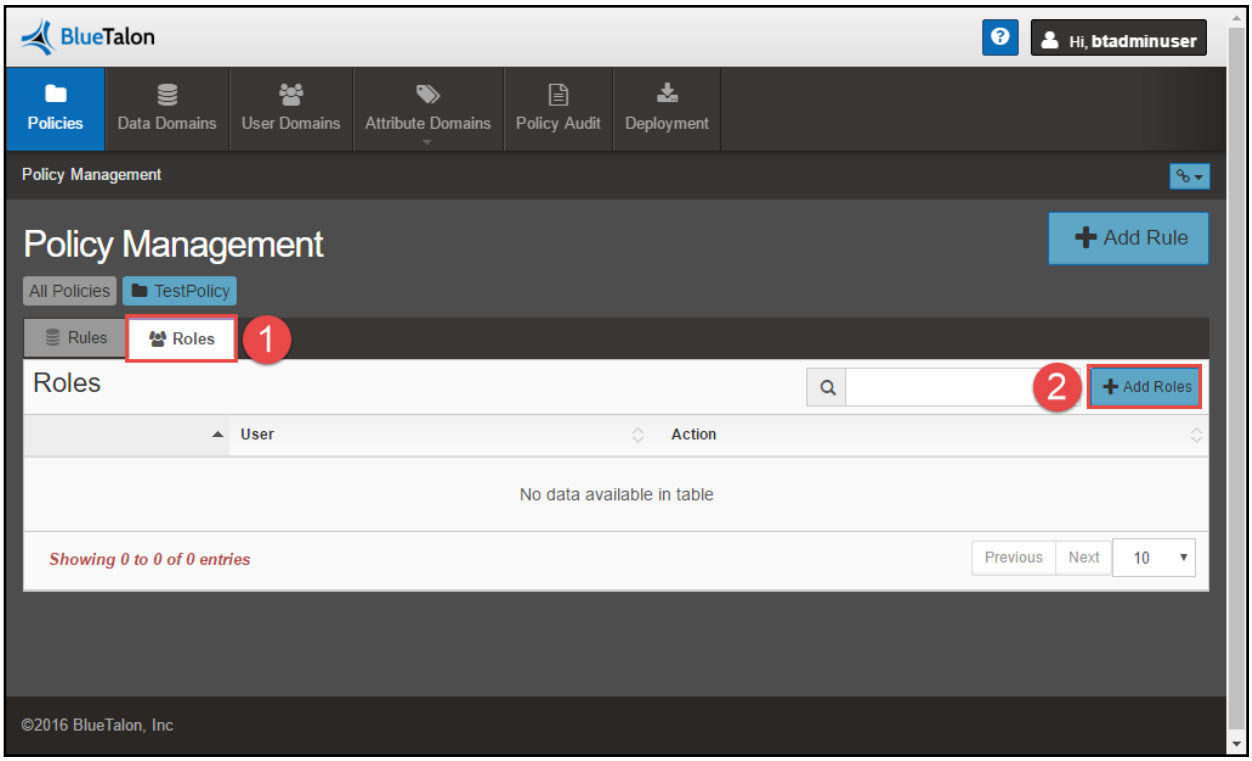
Add the alice user to the policy:

1. Click the Policies tab.

2. Click the policy you created earlier. Example: TestPolicy.

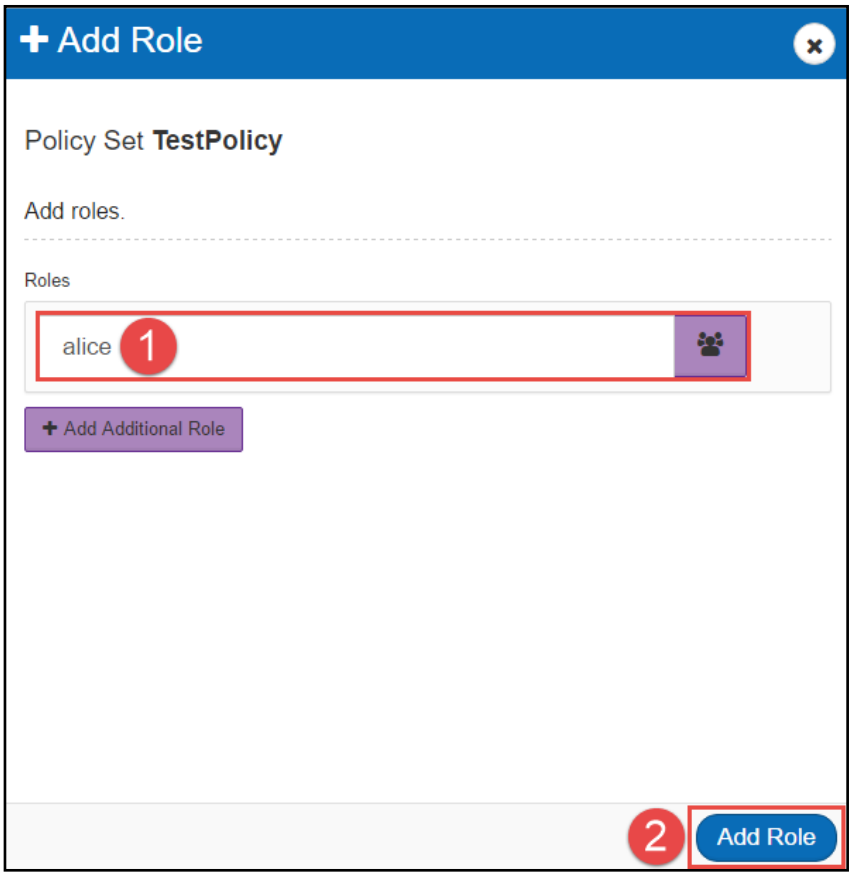


3. Click the Roles subtab. See item 1 in the following screenshot.



4. Click Add Roles. See item 2 in the previous screenshot.

5. Select the alice user. See item 1 in the following screenshot.



6. Click Add Role. See item 2 in the previous screenshot.
7. Click the Deployment tab and deploy.

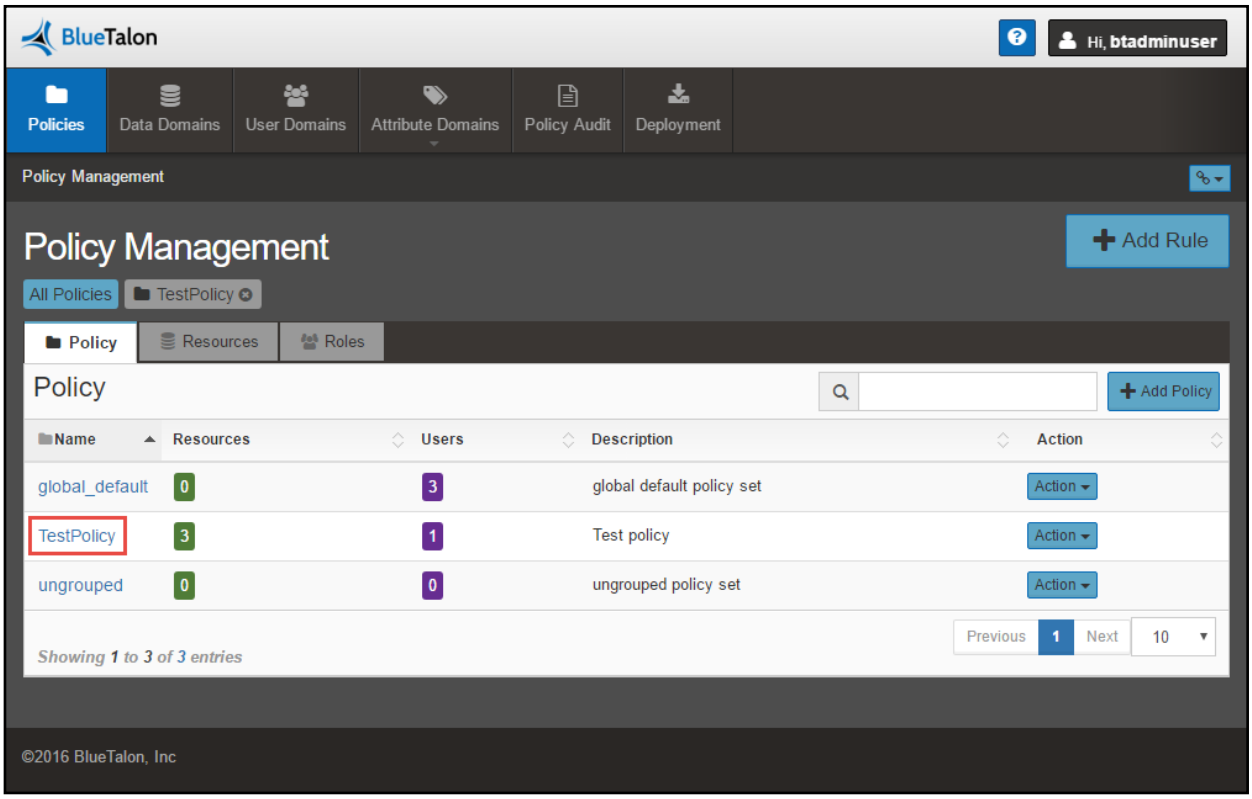
Add Rule to Policy

You add a rule to the policy to allow access to a resource. For a relational database source, a resource is a table or column.

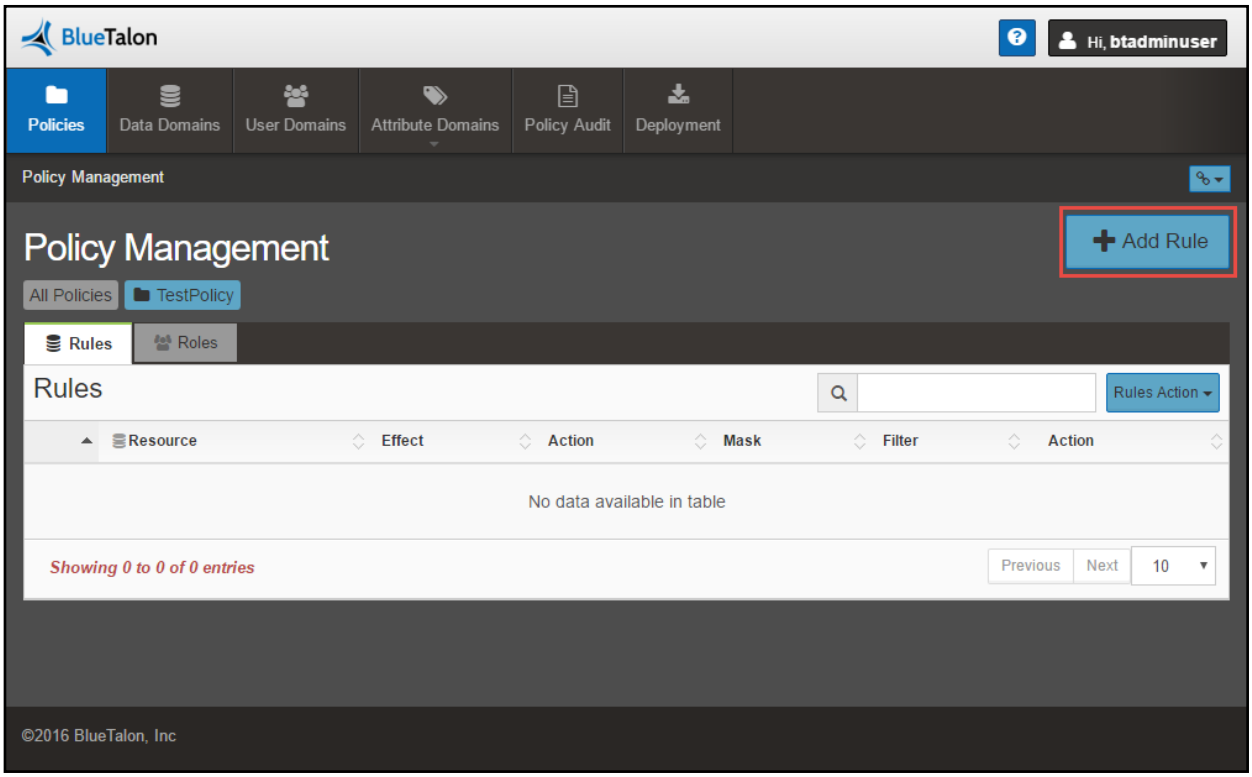
Add a rule to the policy:

1. Click the Policies tab.

2. Click the policy you created earlier. Example: TestPolicy.



3. Click Add Rule on the right of the screen.



4. Add a rule with a mask (example: hide everything except the last four digits of the social security number).
    a. Set the domain to the data domain you created earlier.

       Example for PostgreSQL: postgresdemo. See item 1 in the following screenshot.

    b. Set the resource to the soc_sec_no column of the accounts table you created earlier.

       The resource has this format:
       Database.Schema.Table.Column

       Example for PostgresSQL:
       postgres.public.accounts.soc_sec_no

       See item 2 in the following screenshot.

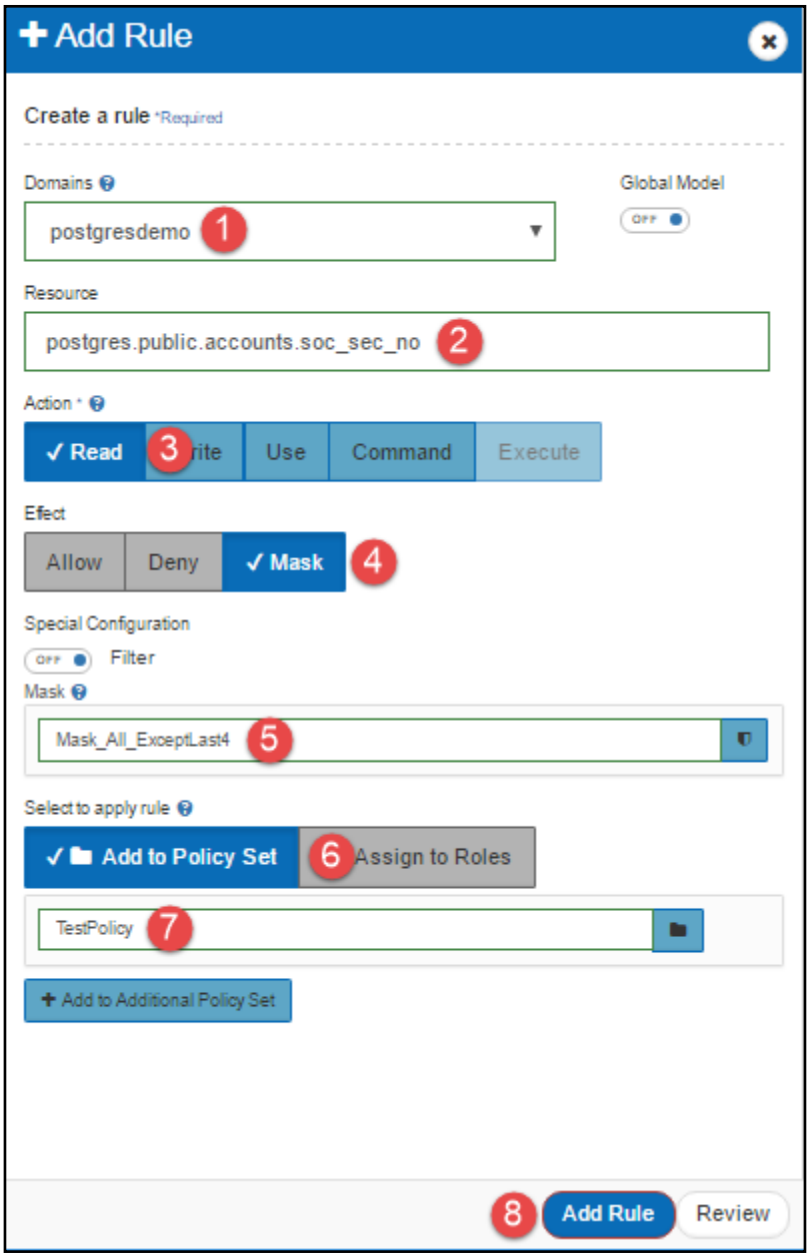
    c. Set the action to Read. See item 3 in the following screenshot.

    d. Set the effect to Mask. See item 4 in the following screenshot.

    e. Set the mask to Mask_All_ExceptLast4. This hides everything except the last four digits. See item 5 in the following screenshot.

    f. Select Add to Policy Set. See item 6 in the following screenshot.

g. Select the policy. Example: TestPolicy. See item 7 in the following screenshot.



5. Click Add Rule. See item 8 in the previous screenshot.
6. Click the Deployment tab and deploy.

Verify Masked Data

Verify the soc_sec_no column is masked by BlueTalon:

1. Connect to the database Enforcement Point using the appropriate database client.

   Example format for PostgreSQL:

   ```
   psql -h <Enforcement Point computer IP address> -p <Enforcement
   Point port> -U alice -d <database name>
   ```

Example for PostgreSQL using the default IP address and port:
```
psql -h 127.0.0.1 -p 1557 -U alice -d postgres
```

The IP address and port are shown in the section Verify Enforcement Point Service is Running on page 70.

If you receive an error stating SSL is off, ensure your database server can receive connections from the Enforcement Point IP address. Example error message for PostgreSQL:
```
psql: FATAL:  no pg_hba.conf entry for host "10.0.1.7", user
"alice", database "postgres", SSL off
```

2. Run the following query and ensure the soc_sec_no column is masked:
```
SELECT * FROM accounts;
```

Example data showing the masked digits of the soc_sec_no column:

```
 id | NAME | phone | birthdate | soc_sec_no | zip | credit_card |
balance
----+------+-------+-----------+------------+-----+-------------+
-------
 0  | 0    | 0     | 0         | XXXXX1527  |   0 |           0 |
0
 0  | 0    | 0     | 0         | XXXXX5483  |   0 |           0 |
0
 0  | 0    | 0     | 0         | XXXXX0766  |   0 |           0 |
0
 0  | 0    | 0     | 0         | XXXXX4141  |   0 |           0 |
0
 0  | 0    | 0     | 0         | XXXXX4670  |   0 |           0 |
0
 0  | 0    | 0     | 0         | XXXXX0690  |   0 |           0 |
0
 0  | 0    | 0     | 0         | XXXXX5814  |   0 |           0 |
0
 0  | 0    | 0     | 0         | XXXXX8723  |   0 |           0 |
0
 0  | 0    | 0     | 0         | XXXXX1100  |   0 |           0 |
0
 0  | 0    | 0     | 0         | XXXXX8766  |   0 |           0 |
0
(10 rows)
```

Columns with a value of zero have no data access rule and the data is blocked. Blocked columns have a value of zero.

Oracle Enforcement Point

This section describes how to set up an Enforcement Point for Oracle.

Collect Information

The following table shows the information you must collect before installing the Enforcement Point.

Item	Description
Database	Oracle service name or SID. Example: ORCL
Host address	Computer hostname or IP address running the database. Example: TestServer.TestCompany.com
Port	Port the database receives JDBC connections. Example: 1521
Schema	Database schema. Only one schema for each connection. Multiple connections allowed.
User ID	User ID for a database account with privileges to obtain metadata.
Password	Password for the database account.

Create Test Data

Create a database user and table. See Create Test Data on page 53. Although the examples are different for Oracle, you can use the previous examples as the basis for your test.

Add Data Domain

Add a data domain. Use Oracle as your data domain type. See Add Data Domain on page 55.

Install Enforcement Point Package

Install the Enforcement Point package:

1. Enter:
   ```
   yum install bluetalon-ep-3.2.4
   ```
2. Follow the prompts.

Configure Enforcement Point Using Setup Script

Configure the Enforcement Point:

1. Run:
   ```
   /opt/bluetalon/3.2.4/pep/scripts/bluetalon-ep-setup
   ```
2. Follow the prompts.

Enforcement Point Files and Service

The following table shows the Enforcement Point files and service.

Item	Detail
Binary files path	/opt/bluetalon/3.2.4/pep/oracle-ep/bin

Configuration file	/etc/bluetalon/pep/oracle-ep/conf/bt-oracle-ep-<data domain name>.conf
Logs	/var/log/bluetalon/pep/oracle-ep/logs/bt-oracle-ep-<data domain name>.log
Service name	bt–oracle-ep-<data domain name>

Test Enforcement Point

See Test Enforcement Point on page 70.

Verify Masked Data

See Verify Masked Data on page 78.

Set Up Enforcement Point for Hadoop Clusters

This section describes how to set up a BlueTalon Enforcement Point for Hadoop clusters.

HDFS Enforcement Point

This section describes how to set up an Enforcement Point for HDFS.

Ordering of Data Domain Set Up

To link other HDFS-related domains (example: Hive) to the primary HDFS domain, you first set up the HDFS Enforcement Point.

Not recommended: If you set up a Hive domain first and then the HDFS domain, then you must remove and reconfigure your HDFS-related domains.

Collect Information

The following table shows the information you must collect before installing the Enforcement Point.

Item	Description
URL	NameNode URL. Example: hdfs://TestServer.TestCompany.com:8020
User ID	User ID for HDFS account with privileges to list directories and files.
Password	Password for user ID.

Create Test Data

In a command line shell, create a test file. Example:

```
vi /tmp/accounts.csv
```

You can use your own file. The example in this section is for illustrative purposes only.

Add example data to the file:

```
147274739-9,Frances Harrison,9-(192)357-
8851,10/27/56,993941527,37726,6393451850134970,52.90

457389751-8,Gregory Patterson,3-(684)454-
2444,11/22/69,233575483,21278,5602245983499780,44.87

360272064-0,Earl James,5-(177)394-
3277,5/23/83,303640766,26595,3534227478434860,3.36

551058833-0,Christine Robertson,5-(437)964-
8463,5/11/88,318794141,50716,5602223026663730,5.22

713553396-8,Daniel Crawford,1-(875)657-
9518,6/1/99,214894670,28693,3571338359613280,22.63

359936857-0,Shirley Holmes,6-(362)871-
3036,4/1/80,964490690,13189,3534219304003200,30.17
```

```
887693755-2,Kelly Adams,9-(887)292-
8810,2/12/88,304475814,84901,5602248578416630,35.09

091619799-9,Bonnie Freeman,2-(163)102-
1214,1/26/71,940518723,73248,3547967170434520,69.74

151872601-1,Joyce Mcdonald,3-(504)572-
0648,9/18/64,919591100,82573,5401856433043540,88.15

956961211-8,Karen Burton,1-(533)256-
2922,3/5/54,479488766,56134,3536534268148670,7.08
```

Save the file and exit vi.

Change permissions on the accounts.csv file:

```
chmod ugo+r /tmp/accounts.csv
```

In the command line shell, perform these commands as the HDFS administration user (typically, the user is named hdfs):

```
useradd alice

passwd alice

hdfs dfs -mkdir /alice

hdfs dfs -put /tmp/accounts.csv /alice

hdfs dfs -chmod ugo+r /alice/accounts.csv

hdfs dfs -ls /alice/accounts.csv

hdfs dfs -cat /alice/accounts.csv
```

Ensure you can see the accounts.csv file data.

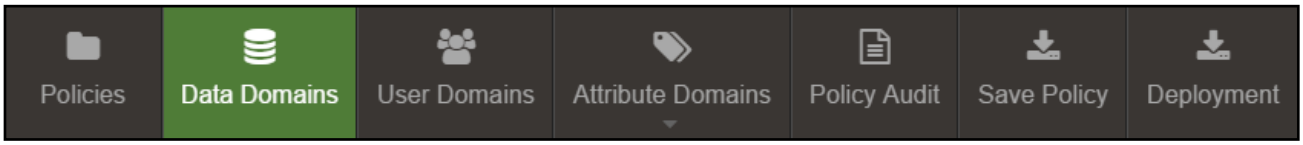
Add Data Domain

A data domain stores information about an external data source.

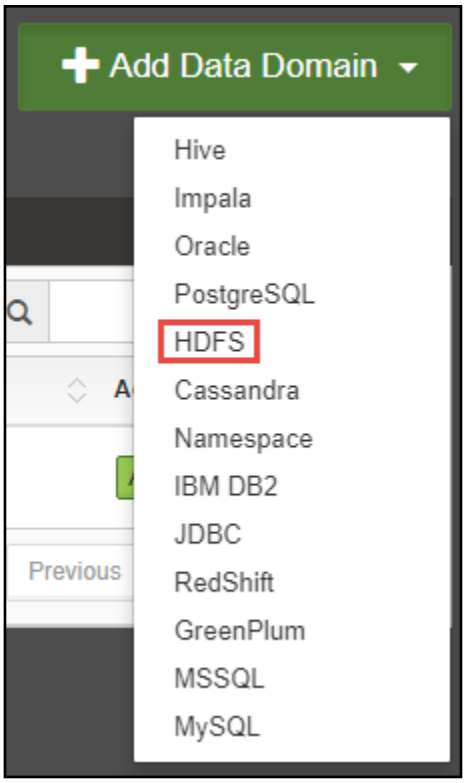
You import the external data source information into a data domain.

Add a data domain using the Policy Console:

1. Open Web browser.
2. Go to the Policy Console.
3. Log in.
4. Click the Data Domains tab.



5. Select Add Data Domains on the right of the screen. Select HDFS.



6. Enter the data domain details. Use the details you obtained earlier in the Collect Information section.

    a. HDFS URL. Example: hdfs://TestServer.TestCompany.com:8020. See item 1 in the following screenshot.

    b. Data domain name. Example: HDFSDataDomain. See item 2 in the screenshot.

    Note the name of your data domain. You will need it later when you configure the Enforcement Point.

    c. Description. See item 3 in the screenshot.

    d. Kerberos. See item 4 in the screenshot.

    If HDFS is secured through Kerberos, then you set the option to on. You must set a Kerberos principal user ID and password. See items 6 and 7 in the screenshot.

    e. Before you create the HDFS data domain for a secure Hadoop cluster, you must ensure the Hadoop XML configuration files are on the computer running the BlueTalon policy server (service is named bt-policy-server).

    The Hadoop XML configuration files are named:
    core-site.xml
    hdfs-site.xml

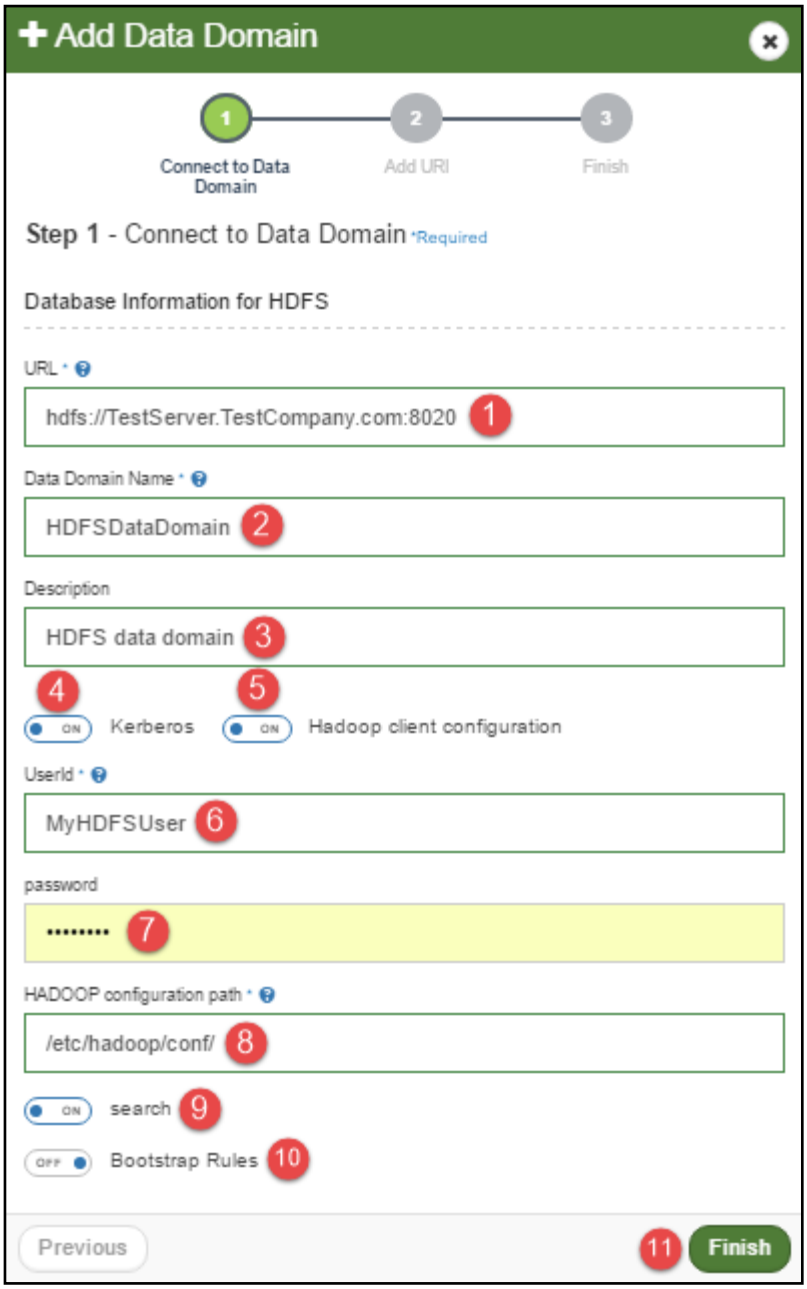
    If the XML files are already on the computer running the BlueTalon policy server, then you can use those files. Examine the directory /etc/hadoop/conf for the XML configuration files. If the XML files are there, you do not copy the files.

If the XML files are not there, then copy the files to the BlueTalon policy server. Copy the files to the directory /etc/bluetalon/policy/pap/<HDFS data domain name>.

f. Hadoop client configuration. See item 5 in the screenshot. When set to on, you set a directory path for the site XML configuration file. You set the path in the HADOOP configuration path parameter. See item 8 in the screenshot. For initial testing, you can leave the option off.

g. Search. Options: On or off. When set to on, data discovery is performed in HDFS. Data discovery means that the files in HDFS are examined. See item 9 in the screenshot. For initial testing, you can leave the option off.

h. Bootstrap Rules. Options: On or off. When set to on, you provide the location of a JSON file with predefined security rules. See item 10 in the screenshot. For initial testing, you can leave the option off.



7. Click Finish. See item 11 in the previous screenshot.
8. Deploy.

You can also use the command line to add a data domain. The following command line example creates an HDFS data domain (non-Kerberos):

```
curl -u btadminuser:P@ssw0rd --header "Content-type: application/json"
--request POST
http://localhost:8111/PolicyManagement/1.0/resource_domains --data '{
  "db_type": "hdfs",
  "url": "hdfs://127.0.0.1:8020",
  "resource_domain_name": "HDFSDataDomain",
  "description": "HDFS data domain",
  "user": "admin",
  "search": true
}'
```

The following command line example creates an HDFS data domain (Kerberos):

```
curl -u btadminuser:P@ssw0rd --header "Content-type: application/json"
--request POST
http://localhost:8111/PolicyManagement/1.0/resource_domains --data '{
  "db_type": "hdfs",
  "url": "hdfs://127.0.0.1:8020",
  "resource_domain_name": "hdfs",
  "description": "ds for hdfs",
  "user": "admin",
  "search": true,
  "securehdfs":true,
  "password":"test"
  "ha_mode":"/etc/hadoop/conf/"
}'
```

Install Enforcement Point Package

Install the Enforcement Point package:

1. Enter:
   `yum install bluetalon-ep-3.2.4`
2. Follow the prompts.

Configure Enforcement Point Using Setup Script

Configure the Enforcement Point:

1. Run:
   `/opt/bluetalon/3.2.4/pep/scripts/bluetalon-ep-setup`
2. Enter the information when you are prompted by the script.

Item	Description
Enforcement Point type	Type of Enforcement Point. Select HDFS.
Kafka Broker hostname	Hostname where the bt-audit-kafka service is running. Example: Kafka-Broker-host.TestServer.TestCompany If you are using an EC2 environment, then you provide the EC2 internal IP address or

	FQDN.
Policy Engine hostname	Computer hostname or IP address on which the BlueTalon Policy Engine is running.
Data domain name	Name of the data domain. Use the value set in the Policy Console for the data domain you created earlier. Example: HDFSDataDomain

3. HDFS permission fallback is enabled by default. Recommendation: Keep the default, which is enabled in the script.
4. Open the following file for editing and read through the file contents:
   `/etc/bluetalon/pep/hdfs-ep/conf/bt-hdfs-site.xml`
5. In the file, set the property bt.hdfs.plugin.union.hdfs.native.perms to true. (This property is not controlled by Ambari.)
6. For an Ambari managed cluster:
   a. In a Web browser, go to the Ambari management console and log in.
   b. Set the following property in the HDFS configuration under custom hdfs-site:
      `dfs.namenode.inode.attributes.provider.class = com.bluetalon.hdfs.core.BlueTalonAccessControlEnforcer`
7. For a non-Ambari managed cluster, open the following file for editing:
   `/etc/hadoop/conf/hdfs-site.xml`

   In the file, set:
   `dfs.namenode.inode.attributes.provider.class = com.bluetalon.hdfs.core.BlueTalonAccessControlEnforcer`
8. Restart the NameNode.
9. Repeat the steps as needed for other NameNodes and secondary NameNodes.

Enforcement Point Files

The following table shows the Enforcement Point files for the HDFS Enforcement Point.

Item	Detail
Binary files path	/opt/bluetalon/3.2.4/pep/hdfs-ep/bin
Configuration files path	/etc/bluetalon/pep/hdfs-ep/conf
Logs	Hadoop NameNode logs
Service name	No service for HDFS EP; it is a plug in for the HDFS NameNode

Verify Data Cannot be Read

Restart the HDFS NameNode. Example:

```
/usr/lib/hadoop/bin/stop-dfs.sh
/usr/lib/hadoop/bin/start-dfs.sh
```

Connect to HDFS secured by the Enforcement Point and attempt to examine the accounts.csv file. Example command:

```
su -l alice -c "hdfs dfs -cat /alice/accounts.csv"
```

Ensure access to the accounts.csv file is denied. Access is denied because BlueTalon prevents data access.

A rule controls data access to a resource and controls the granularity of the access control policy.

To learn how to add HDFS rules to allow access to HDFS data, see the section *HDFS Rules* in the *BlueTalon Security Administration Guide*.

Run the commands in Hadoop Functional Tests on page 162 at the end of this section.

File System Enforcement Point (FSEP)

This section describes how to set up FSEP (File System Enforcement Point).

The following table shows the sections for your FSEP installation type.

Installation Type	Section
FSEP for standard Hadoop HDFS	Configure FSEP for Standard Hadoop HDFS on page 89
FSEP for HDP	Configure FSEP for HDP on page 91
FSEP for CDH	Configure FSEP for CDH on page 126
FSEP for Isilon	Configure FSEP for Isilon on page 137

Hortonworks HDP and Cloudera CDH can be implemented on both Amazon Web Services and Microsoft Azure.

FSEP and Kerberos Support

These software versions are supported by the BlueTalon software for FSEP and Kerberos clusters:

- Ambari 2.2.1.0
- HDP 2.4.3.0
- HDP Stack 2.4.3.0-227
- HDP Utilities 1.1.0.20

Provision Hadoop User

To provision the Hadoop user, perform the following commands as an administrator (typically the "hdfs" user):

```
hdfs dfs -mkdir -p /user/<USER>/.staging
hdfs dfs -chown -R <USER>:<GROUP> /user/<USER>/.staging
```

Examples:

```
hdfs dfs -mkdir -p /user/ambari-qa/.staging
hdfs dfs -chown -R ambari-qa:user /user/ambari-qa/.staging
```

Perform the following commands even if data is not routed through FSEP:

```
hdfs dfs -mkdir /user/<USER>
hdfs dfs -chown <USER>:<GROUP> /user/<USER>
```

Examine Java Home Parameter

Examine file to ensure it contains a JAVA_HOME parameter. Example:

```
grep JAVA_HOME /etc/bluetalon/pep/fs-ep/conf/fsep-env.sh
```

Example JAVA_HOME setting in the file:

```
export JAVA_HOME=/usr/jdk64/jdk1.8.0_60
```

If the file does not contain a JAVA_HOME setting, you must add it to the file.

Data Domain Deployment

You will see how to add a data domain for FSEP shortly. Before you see how to do that, be aware of the following:

- You can create a data domain and deploy it using the Policy Console user interface or the REST API. The instructions you will see later show how to use the Policy Console.
- If you choose to create a data domain using the REST API, ensure you deploy the new data domain.

Example REST API deploy command:

```
curl -u btadminuser:P@ssw0rd  --header "Content-type: application/json"
--request PUT http://localhost:8111/PolicyManagement/1.0/deploy --data
'{"user":"btadminuser","description":"Deployed change"}'
```

Configure FSEP for Standard Hadoop HDFS

To configure FSEP for standard Hadoop HDFS, perform the steps in this section.

Collect Information

Same as for HDFS. See Collect Information on page 82.

Create Test Data

Same as for HDFS. See Create Test Data on page 82.

Add Data Domain

Same as for HDFS. See Add Data Domain on page 83.

Install Enforcement Point Package

Install the Enforcement Point package:

1. Enter:
   ```
   yum install bluetalon-ep-3.2.4
   ```
2. Follow the prompts.

Configure Enforcement Point

Configure the Enforcement Point:

1. Run:
   ```
   /opt/bluetalon/3.2.4/pep/scripts/bluetalon-ep-setup
   ```
2. For the EP set up script, enter the information.

Item	Description
Enforcement Point type	Type of Enforcement Point. Select FSEP.
Kafka Broker hostname	Hostname where the bt-audit-kafka service is running. Example: Kafka-Broker-host.
Policy Engine hostname	Computer hostname or IP address on which the BlueTalon Policy Engine is running.
Data domain name	Name of the data domain. Use the value set in the Policy Console for the data domain you created earlier.

Configure Enforcement Point Using Setup Script in Silent Mode

Ensure you have deployed your data domain before deploying an Enforcement Point.

(Optional) You can also run the script in silent mode, which does not prompt you to enter details.

To configure the Enforcement Point with a preconfigured XML file, run:

```
/opt/bluetalon/3.2.4/pep/scripts/bluetalon-ep-setup -accept -f
/tmp/fsep.xml
```

For the parameter list, see Enforcement Point Silent Mode XML Parameters on page 65.

Example FSEP Silent Mode XML Files

This section shows example FSEP silent mode XML files.

Example FSEP Silent Mode XML File with Simple Authentication

Example FSEP silent mode XML file with simple authentication:

```
<EnforcementPoint xmlns="">
  <db_type>7</db_type>
  <resource_domain_name>HDFSDataDomain</resource_domain_name>
  <audit_ip>127.0.0.1</audit_ip>
  <pdp_hostname>127.0.0.1</pdp_hostname>
  <hadoop_conf_path>/etc/hadoop/conf/</hadoop_conf_path>
  <traffic_divert>yes</traffic_divert>
  <kerberos>no</kerberos>
  <java_home>/usr/lib/jvm/jre-1.6.0-openjdk.x86_64</java_home>
</EnforcementPoint>
```

Example FSEP Silent Mode XML File with Kerberos Authentication

Example FSEP silent mode XML file with Kerberos authentication:

```
<EnforcementPoint xmlns="">
  <db_type>7</db_type>
  <resource_domain_name>HDFSDataDomain</resource_domain_name>
  <audit_ip>127.0.0.1</audit_ip>
  <pdp_hostname>TestServer.TestCompany.com</pdp_hostname>
  <hdfs_principal>hdfs/TestServer.TestCompany.com@EXAMPLE.COM
</hdfs_principal>
  <hdfs_keytab>/home/ec2-user/hdfs.keytab</hdfs_keytab>
  <http_principal>HTTP/TestServer.TestCompany.com@EXAMPLE.COM
</http_principal>
```

```
  <http_keytab>/home/ec2-user/http.keytab</http_keytab>
  <hadoop_conf_path>/etc/hadoop/conf/</hadoop_conf_path>
  <traffic_divert>yes</traffic_divert>
  <kerberos>yes</kerberos>
</EnforcementPoint>
```

Silent mode configuration is complete.

Restart FSEP Service

Restart FSEP service:

```
service bt-fsep restart
```

Verify Data Cannot be Read

Same as for HDFS. See Verify Data Cannot be Read on page 87.

Enforcement Point Files

The following table shows the Enforcement Point files for the File System Enforcement Point.

Item	Detail
Binary files path	/opt/bluetalon/3.2.4/pep/fs-ep/bin
Configuration files path	/etc/bluetalon/pep/fs-ep/conf
Log files path	/var/log/bluetalon/pep/fs-ep/logs
Service name	bt-fsep

Allow Access to HDFS Data

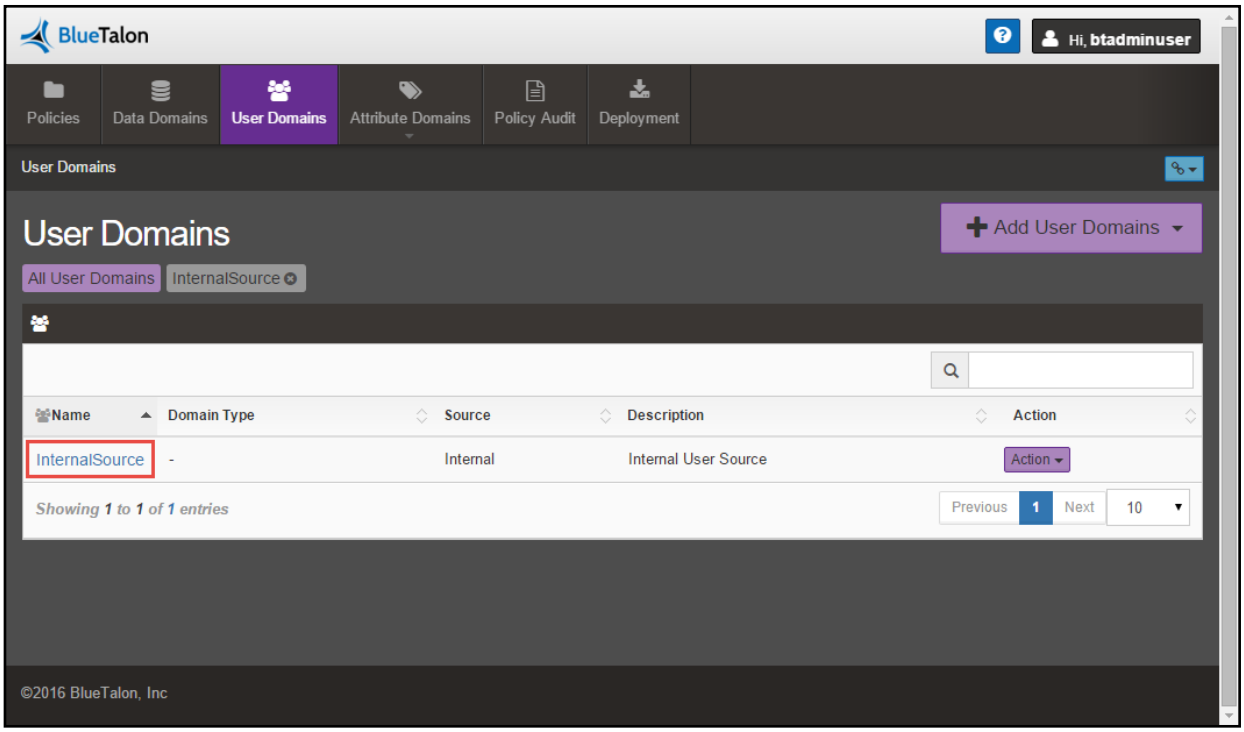
(Optional) By default, HDFS data protected by BlueTalon cannot be accessed. In this section, you will learn how to allow access to HDFS data.

Add User to Internal User Domain

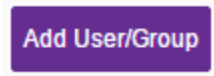
Add a user named alice to the internal user domain:

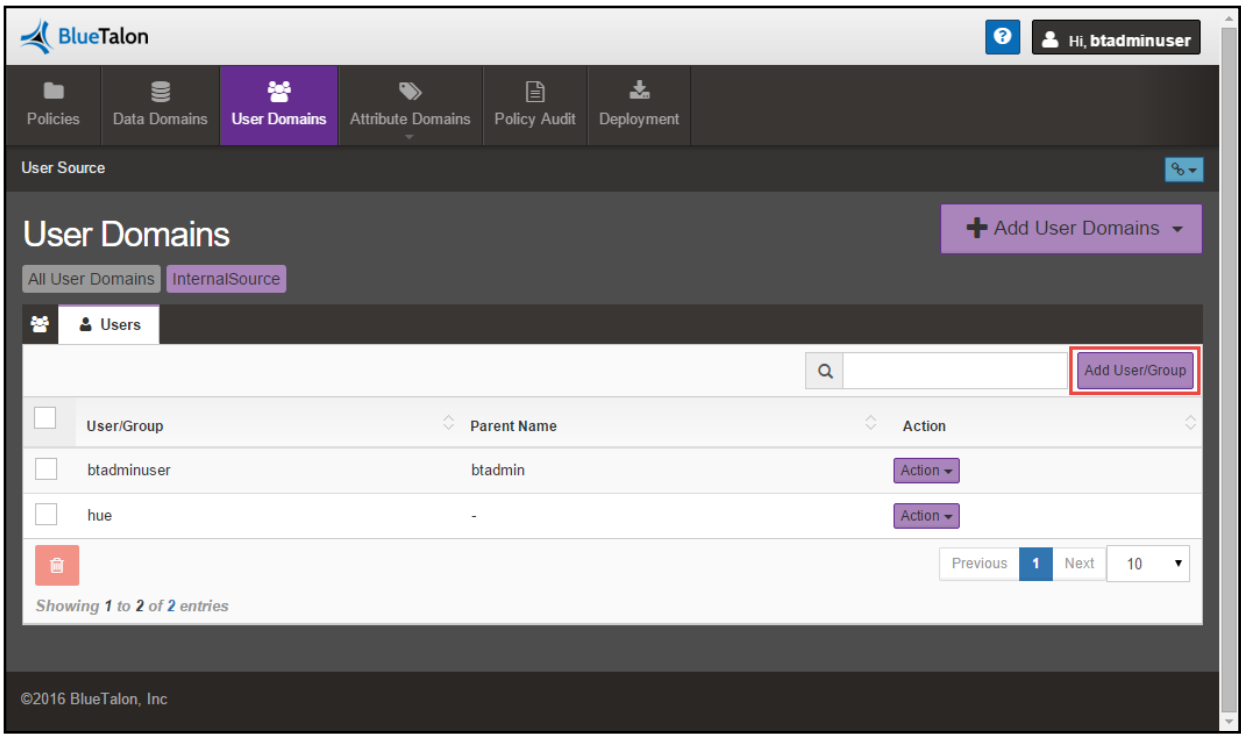
1. Click the User Domains tab.

2. Click the user domain named InternalSource.



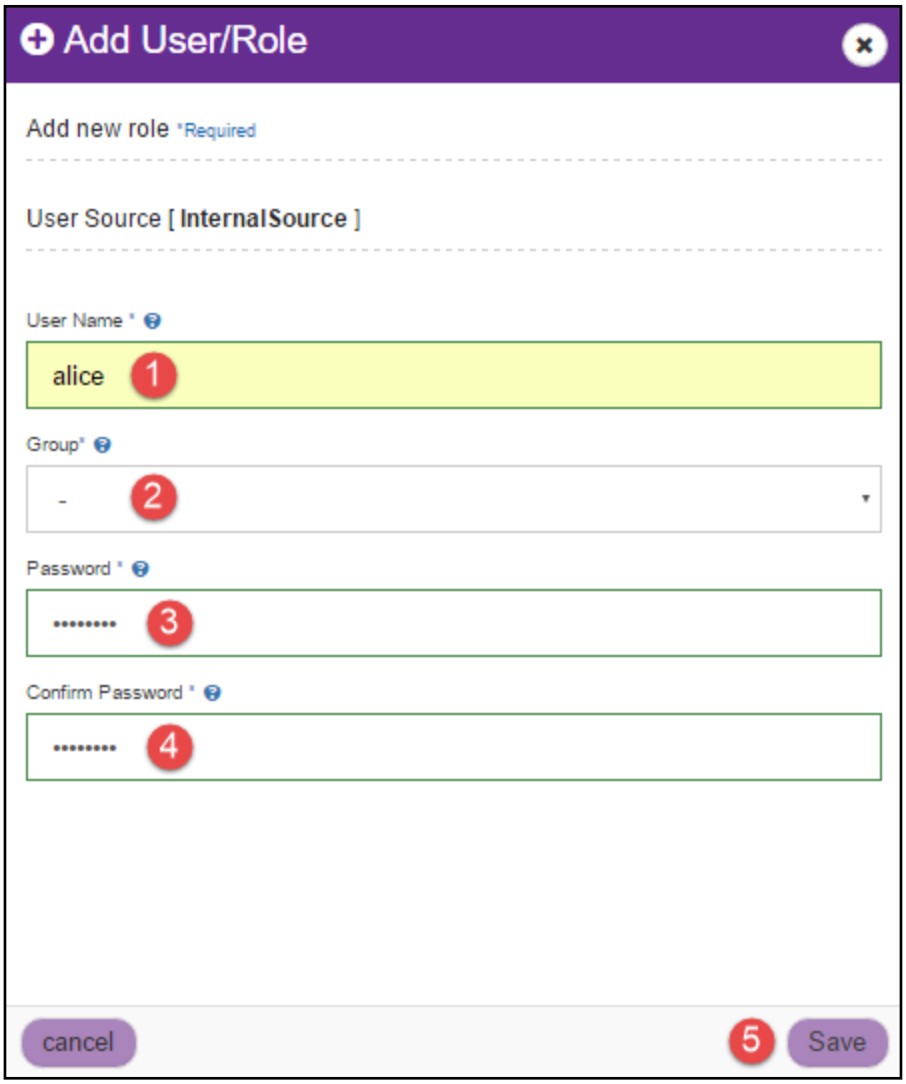
3. Click Add User/Group on the right of the screen.





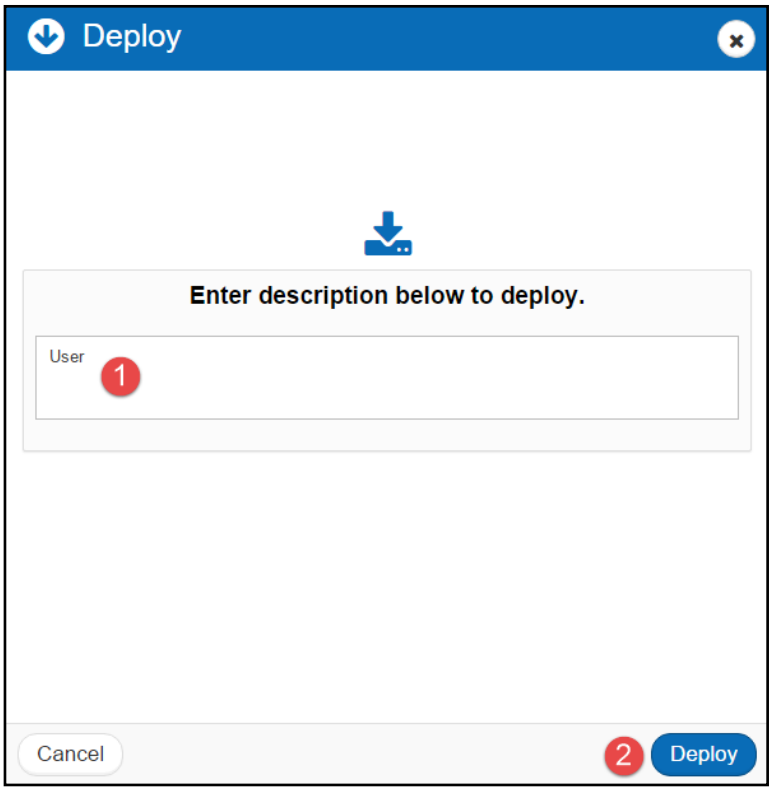
4. Set the user name to alice. See item 1 in the following screenshot.

5. Select "-" from the Group selection. See item 2 in the following screenshot.



6. Set a password. See items 3 and 4 in the previous screenshot.
7. Click Save. See item 5 in the previous screenshot.

8. Click the Deployment tab and deploy.



Add Policy

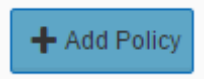
A policy is a set of security enforcement rules that restrict data access.

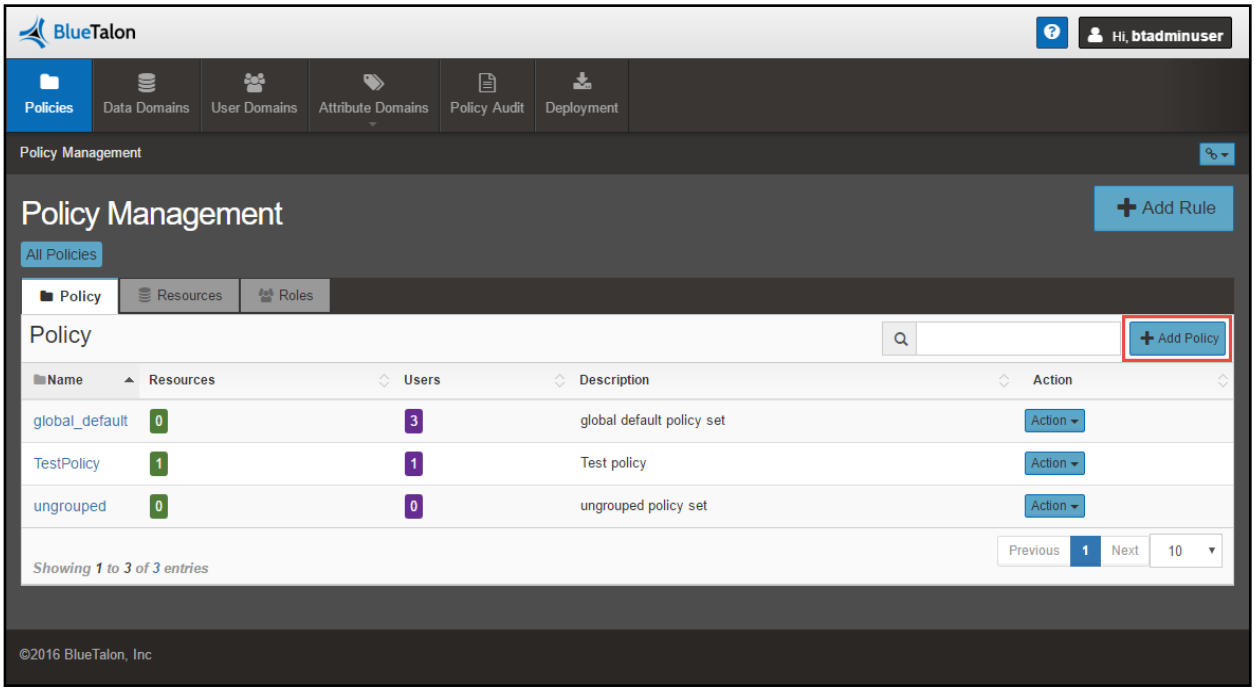
A policy:

- Allows a limited set of users to access the information they need.
- Denies all other access by default.

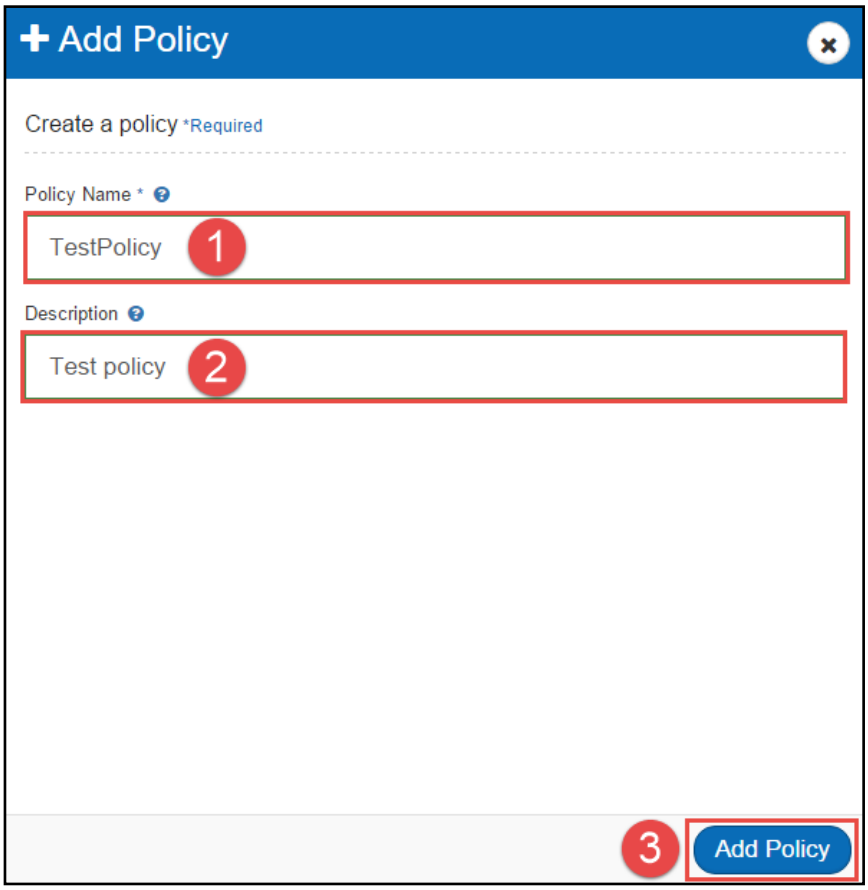
Add a policy:

1. Click the Policies tab.
2. Click Add Policy on the right of the screen.

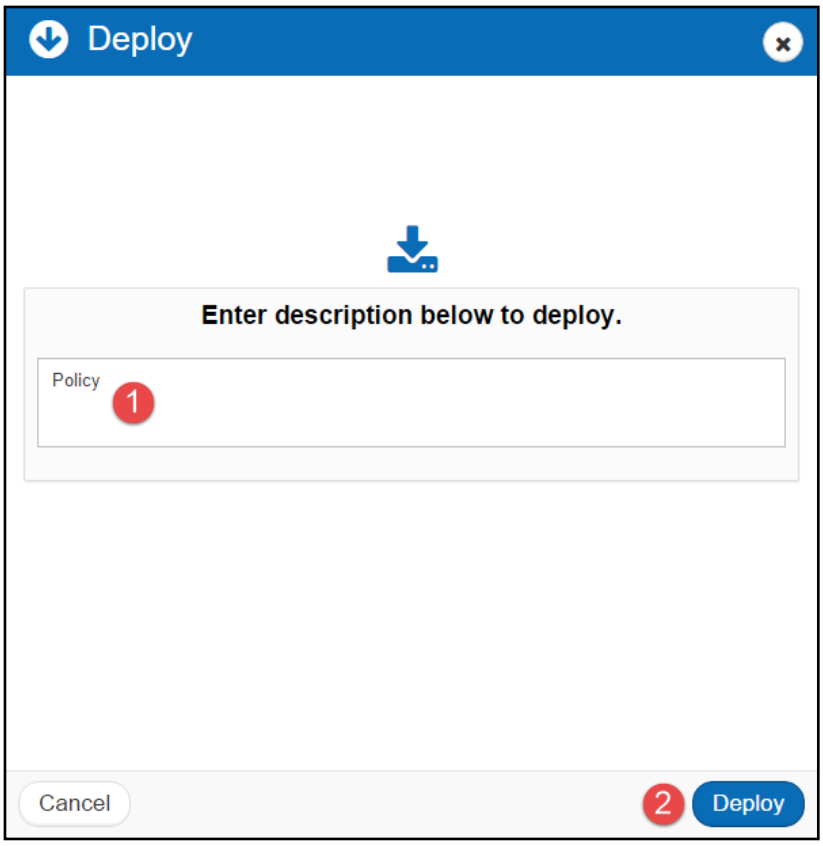




3. Enter a policy name. Example: TestPolicy. See item 1 in the following screenshot.
4. Enter a policy description. Example: Test policy. See item 2 in the following screenshot.



5. Click Add Policy. See item 3 in the previous screenshot.
6. Click the Deployment tab and deploy.

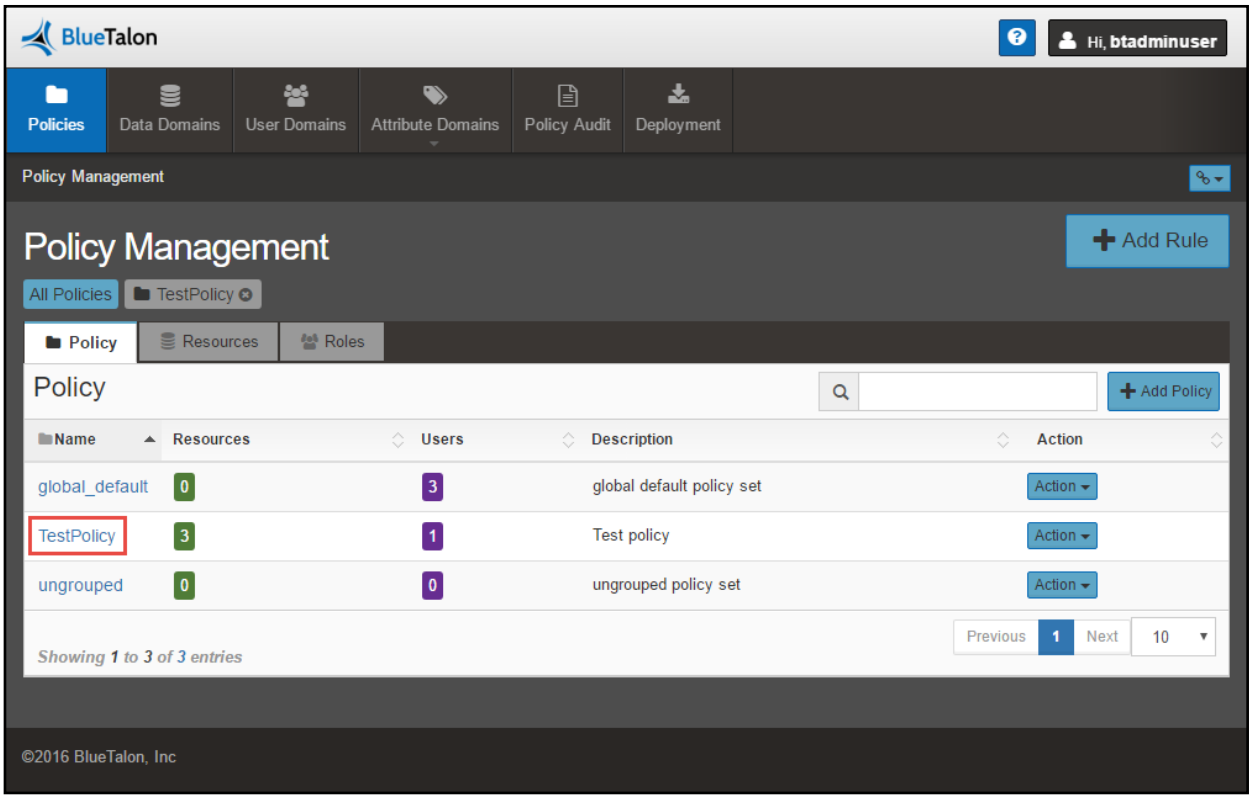


Add User to Policy

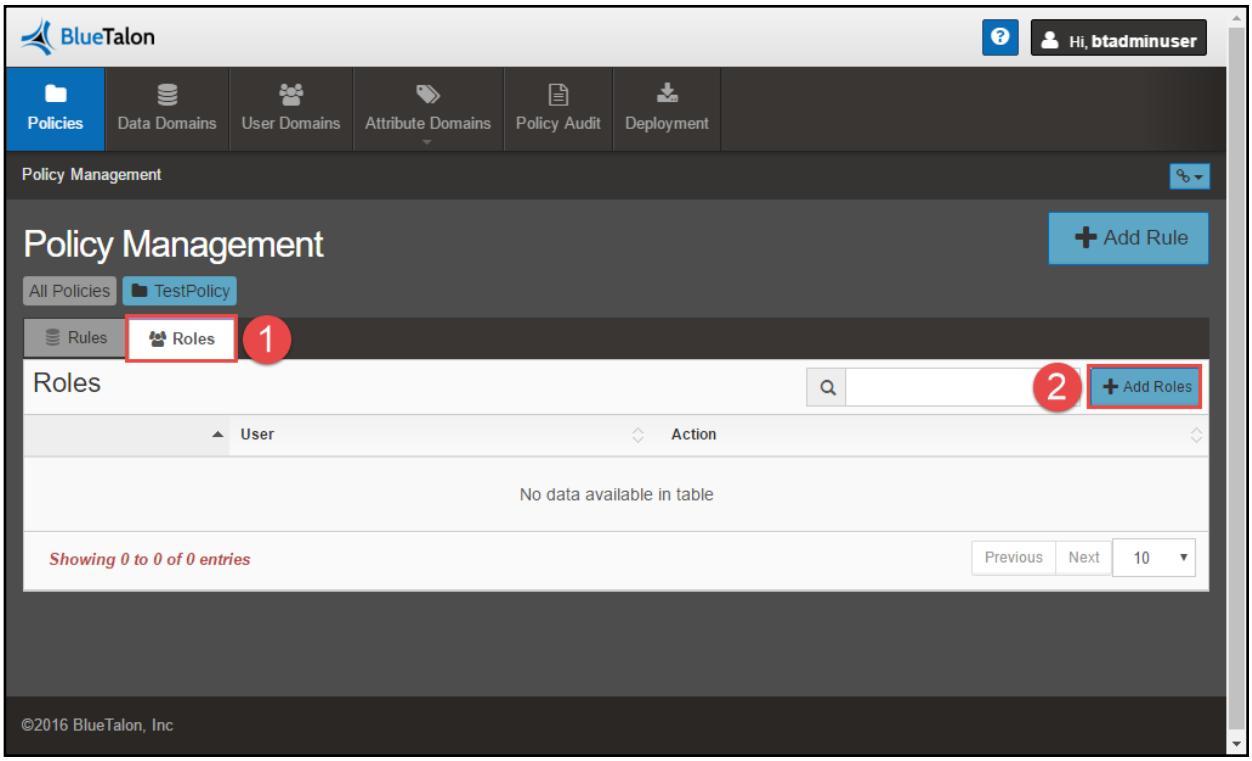
Add the alice user to the policy:

1. Click the Policies tab.

2. Click the policy you created earlier. Example: TestPolicy.

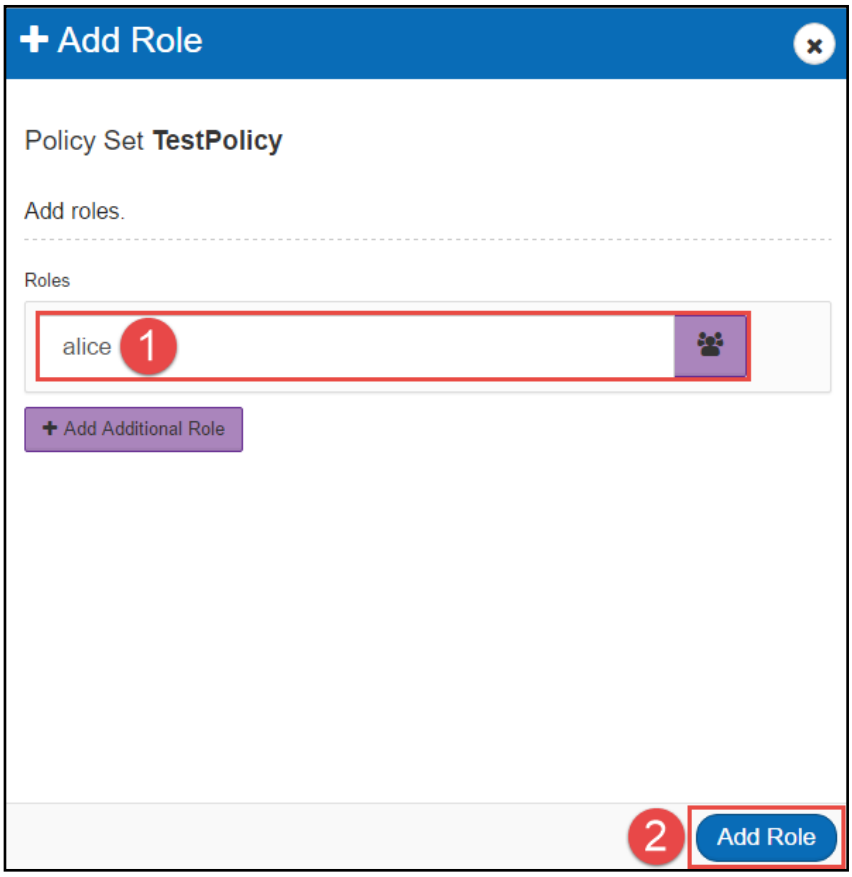


3. Click the Roles subtab. See item 1 in the following screenshot.



4. Click Add Roles. See item 2 in the previous screenshot.

5. Select the alice user. See item 1 in the following screenshot.



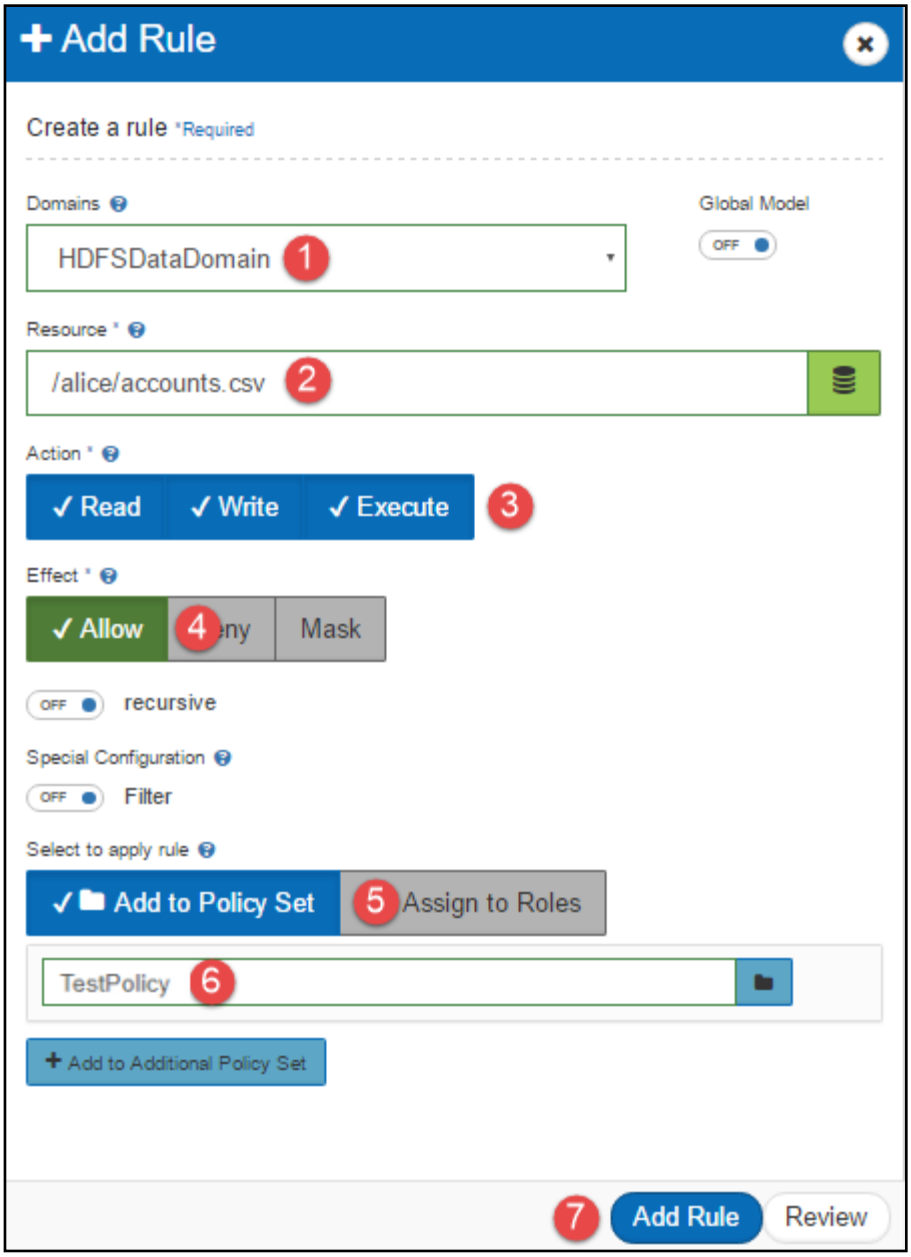
6. Click Add Role. See item 2 in the previous screenshot.
7. Click the Deployment tab and deploy.

Add HDFS Rule

In this section, you will add a rule to allow access to the accounts.csv file.

Add the rule in the Policy Console:

1. Click Policies.
2. Click TestPolicy.
3. Click Add Rule.
4. Set the parameters for the rule. The following screenshot shows example settings.
    a. Data domain name. Example: HDFSDataDomain. See item 1 in the screenshot.
    b. Resource in HDFS. Example: /alice/accounts.csv. See item 2 in the screenshot.
    c. Actions to be performed on the resource. Examples: Read, Write, and Execute. See item 3 in the screenshot. Typically, you should limit the actions to only those required by the users.
    d. Effect for the actions. Example: Allow. See item 4 in the screenshot.
    e. Add to policy set. See item 5 in the screenshot.
    f. Policy to add the rule to. Example: TestPolicy. See item 6 in the screenshot.



5. Click Add Rule. See item 7 in the screenshot.
6. Deploy.

The following screenshot shows the new rule.

Examine Effect of HDFS Rule

Wait two minutes before running the following steps. HDFS rules take a short time to become active.

Connect to HDFS secured by the Enforcement Point and examine the accounts.csv file. Example command:

```
su -l alice -c "hdfs dfs -cat /alice/accounts.csv"
```

Earlier, the command did not succeed because there was no rule to allow access to the HDFS data. Now, the command succeeds because of the new rule.

Only expose HDFS data to users who require access.

Hive

(Optional) You can also set up Hive, as described in this section.

Add Accounts Table

In a command line shell, add an accounts table using the accounts.csv file you created earlier. Use your own URL and port settings in the connect command. Example:

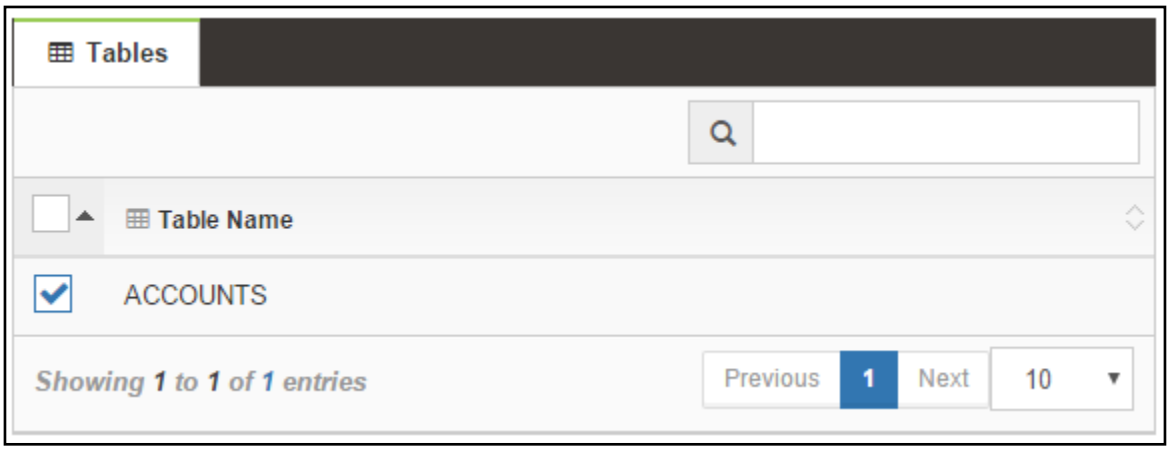
```
hive
!connect jdbc:hive2://localhost:10000/default alice
create table accounts(id string, name string, phone string, birthdate
string, soc_sec_no string, zip bigint, credit_card bigint, balance
decimal(4,2)) row format delimited fields terminated by ',' lines
terminated by '\n' stored as textfile;
load data inpath '/alice/accounts.csv' into table accounts;
select * from accounts;
!close
!quit
```

Install Hive Enforcement Point

You install a Hive Enforcement Point in a similar way to FSEP you installed earlier.

1. In the Policy Console, add a Hive data domain and set the parameters for your Hive database. Example data domain name: HiveDataDomain.

   In step 2 of the add data domain dialog, import the accounts database table you created earlier.



2. Deploy the data domain.

3. In a command line shell, install the Enforcement Point package:
   `yum install bluetalon-ep-3.2.4`
4. Configure the Enforcement Point:
   `/opt/bluetalon/3.2.4/pep/scripts/bluetalon-ep-setup`

   Select Hive as the Enforcement Point type.

   Enter the Hive data domain name you created earlier. Example: HiveDataDomain.
5. In a command line shell, connect to Hive through the Enforcement Point port and perform a query to retrieve the accounts data. Use your own URL and port settings in the connect command. Example:

   ```
   hive
   !connect jdbc:hive2://localhost:10010/default alice
   select * from accounts;
   ```
6. Ensure the query does not return valid data. This is because the BlueTalon software blocks data access when you attempt to retrieve data through the Enforcement Point, and there is no rule added for data access.

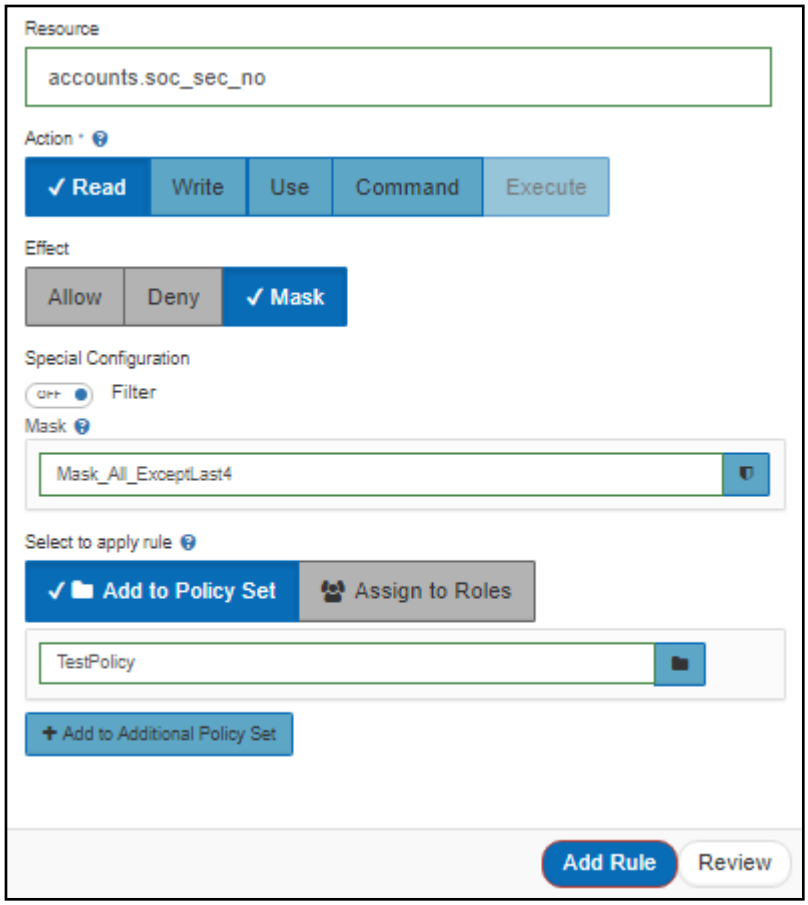
You will see how to add a rule in the next section.

Add Rule for Accounts Table

After you have configured the Hive Enforcement Point, you can add rules using the Policy Console for the accounts database table. You add the rules to the policy you created earlier (example policy is named TestPolicy).

Add a rule:

1. Open Chrome Web browser.
2. Go to the Policy Console.
3. Click Policies.
4. Click TestPolicy.
5. Click Add Rule.
6. Add a rule to HiveDataDomain and TestPolicy. Example rule to only show the last four digits of the social security number in the accounts table:



7. Click Add Rule.
8. Deploy.
9. In a command line shell, connect to Hive through the Enforcement Point port and perform a query to retrieve the accounts data. Use your own URL and port settings to connect to the Enforcement Point. Example:

```
hive

!connect jdbc:hive2://localhost:10010/default alice

select * from accounts;

 id | NAME | phone | birthdate | soc_sec_no | zip | credit_card |
balance

----+------+-------+-----------+------------+-----+-------------+
-------

 0  | 0    | 0     | 0         | XXXXX1527  |   0 |           0 |
0

 0  | 0    | 0     | 0         | XXXXX5483  |   0 |           0 |
0

 0  | 0    | 0     | 0         | XXXXX0766  |   0 |           0 |
0

 0  | 0    | 0     | 0         | XXXXX4141  |   0 |           0 |
0
```

```
 0   | 0     | 0      | 0          | XXXXX4670   |    0 |            0 |
0
 0   | 0     | 0      | 0          | XXXXX0690   |    0 |            0 |
0
 0   | 0     | 0      | 0          | XXXXX5814   |    0 |            0 |
0
 0   | 0     | 0      | 0          | XXXXX8723   |    0 |            0 |
0
 0   | 0     | 0      | 0          | XXXXX1100   |    0 |            0 |
0
 0   | 0     | 0      | 0          | XXXXX8766   |    0 |            0 |
0
```

Columns with a value of zero have no data access rule and the data is blocked. Only the last four digits of the social security number are returned.

You have completed the set up for standard Hadoop.

Configure FSEP for HDP

This section describes how to configure FSEP for HDP.

- BlueTalon provides data-centric security for HDP.
- Apache Ambari makes Hadoop management simpler by provisioning, managing, and monitoring Apache Hadoop clusters.
- BlueTalon software integrates with Ambari to provide security and a seamless experience with HDP.
- BlueTalon enables control over data for each user identity or business role.
- Security is enforced at the file, folder, table, column, row, or cell level.

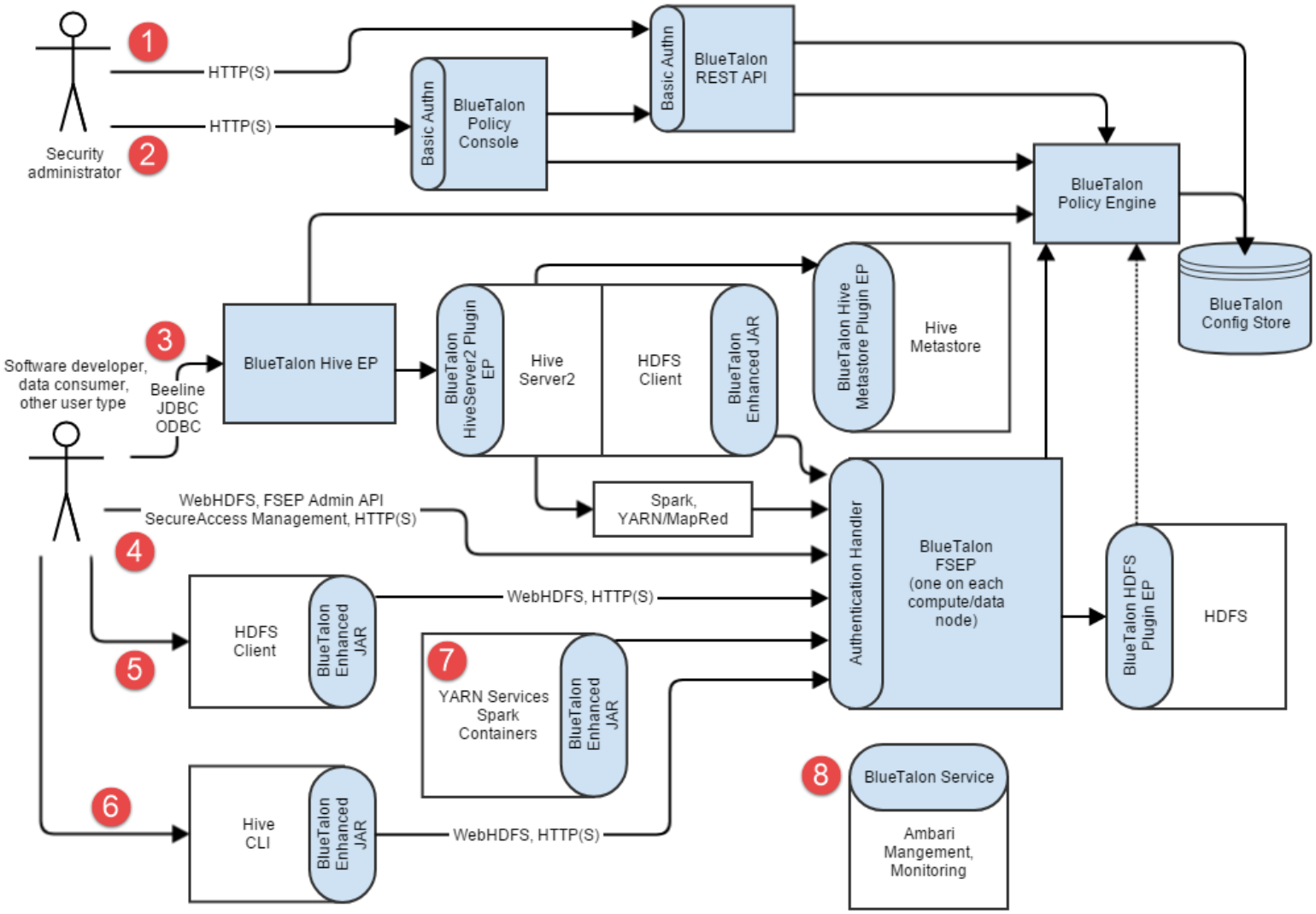
In a typical Hadoop cluster:

- Users specify SQL queries.
- For applications accessing data through Hive, the BlueTalon Hive Enforcement Point transparently proxies HiveServer2 at the network level and provides policy-protected data.
- The BlueTalon Policy Engine performs sophisticated, fine-grained policy decisions based on user and content criteria at run-time by modifying the SQL requests.

Reference Architecture for BlueTalon Software on HDP

The following diagram shows the reference architecture for the BlueTalon software on HDP.

- Blue components are parts of the BlueTalon software that protect the data.
- User performs data requests through the protocols shown in the diagram.
- Data requests are either allowed or denied by the BlueTalon software.
- Table following the diagram describes the data paths, software components, and how BlueTalon enforces security at each point in the paths.



The following table describes the paths identified by the numbers in the diagram. The paths are secured at each point by the BlueTalon software.

Path	Description
1	Security administrator runs the BlueTalon REST API methods to add, edit, delete, or examine security settings. The BlueTalon software enforces security access, and the HTTPS protocol can also be used to further secure the access. The first point in the path is the BlueTalon REST API. BlueTalon REST API The BlueTalon REST API methods are used to add policies, data domains, and so on. The REST API methods can be run from a command line shell or within a program. The path then goes to either the BlueTalon Policy Engine or the BlueTalon Config Store. BlueTalon Policy Engine The BlueTalon Policy Engine is a background service for providing policy decisions used in data access control. The Policy Engine enforces data access rules by parsing all data requests and modifying them based on

	user identity for policy compliance. BlueTalon Config Store The BlueTalon Config Store consists of databases that contain: • Policies added using the REST API or the Policy Console. • Enforceable policies deployed on the Policy Engine.
2	Security administrator logs into the BlueTalon Policy Console. BlueTalon Policy Console The Policy Console is a Web user interface that enables the user to add security policies, data domains, and so on. HTTPS can also be used to secure the access. The path then goes to the BlueTalon REST API, then to the BlueTalon Policy Engine, and then to the BlueTalon Config Store. BlueTalon Policy Engine The BlueTalon Policy Engine is a background service for providing policy decisions used in data access control. The Policy Engine enforces data access rules by parsing all data requests and modifying them based on user identity for policy compliance. BlueTalon Config Store The BlueTalon Config Store consists of databases that contain: • Policies added using the REST API or the Policy Console. • Enforceable policies deployed on the Policy Engine.
3	Software developer, data consumer, or other user type performs queries and other data requests. Security for the data access is enforced at each point in the access path. Beeline is a JDBC client based on the SQLLine command line interface. Beeline can be used to issue queries and other data requests. Java applications can also issue queries and other data requests using JDBC and ODBC. The first point in the path is the BlueTalon Hive Enforcement Point. BlueTalon Hive Enforcement Point The BlueTalon Hive Enforcement Point (EP) is a background service that proxies HDFS. The Hive Enforcement Point: 1. Intercepts queries and data requests. 2. Forwards data requests to the Policy Engine for policy compliance enforcement. 3. Sends the modified policy compliant request received back from the Policy Engine. The next point in the path is the BlueTalon HiveServer2 Plug In EP (Enforcement Point). BlueTalon HiveServer2 Plug In EP The BlueTalon HiveServer2 Plug In EP transparently proxies HiveServer2 at the network level and provides policy-protected data. The next point in the path is HiveServer2. HiveServer2 HiveServer2 is a server interface that enables remote clients to execute queries against Hive and retrieve the results. The next point in the path is the HDFS Client. HDFS Client The HDFS Client is used to access an HDFS cluster. The next point in the path is the BlueTalon Enhanced JAR.

	BlueTalon Enhanced JAR The BlueTalon Enhanced JAR enables the standard HDFS client to use the BlueTalon SecureAccess feature, which uses a shared secret to authenticate users. You will learn about SecureAccess later in this document. See Authenticate End User with SecureAccess on page 216. SecureAccess works with WebHDFS. The next point in the path is the BlueTalon File System Enforcement Point. BlueTalon File System Enforcement Point The BlueTalon File System Enforcement Point (FSEP) protects Hadoop Distributed File System data. FSEP is installed on each compute/data node computer. The next point in the path is HDFS. HDFS The Hadoop Distributed File System (HDFS) is the repository that stores data. The branching paths from HiveServer2 are to the Hive Metastore and to the Spark YARN/MapRed components, which are now described. Hive Metastore The Hive Metastore service stores the metadata for Hive tables and partitions in a relational database. A mapping is performed from the database table name to the HDFS file. Spark YARN/MapRed Containers for Spark YARN/MapRed. YARN has containers that are not directly called. When HDFS data is needed SecureAccess is called. A user submits a job, the framework creates a container, and the user's SecureAccess shared secret is used for the data access that is protected by BlueTalon. The BlueTalon File System Enforcement Point (FSEP) is the next point in the path. BlueTalon File System Enforcement Point The BlueTalon File System Enforcement Point (FSEP) protects Hadoop Distributed File System data. FSEP is installed on each compute/data node computer. The next point in the path is HDFS. HDFS The Hadoop Distributed File System (HDFS) is the repository that stores data.
4	Software developer, data consumer, or other user type performs queries and other data requests. Security for the data access is enforced at each point in the access path. The data access is performed using WebHDFS, the FSEP Admin API, SecureAccess, and/or HTTPS. The next point in the path is the BlueTalon File System Enforcement Point (FSEP), which protects the Hadoop Distributed File System data. BlueTalon File System Enforcement Point The BlueTalon File System Enforcement Point (FSEP) protects Hadoop Distributed File System data. FSEP is installed on each compute/data node computer. The next point in the path is HDFS. HDFS The Hadoop Distributed File System (HDFS) is the repository that stores data.
5	Software developer, data consumer, or other user type performs queries and other data requests. Security for the data access is enforced at each point in the access path. The data access is performed using WebHDFS and HTTP(S). HDFS Client The HDFS Client is used to access an HDFS cluster. The next point in the path is the BlueTalon Enhanced JAR.

	BlueTalon Enhanced JAR The BlueTalon Enhanced JAR enables the standard HDFS client to use the BlueTalon SecureAccess feature, which uses a shared secret to authenticate users. You will learn about SecureAccess later in this document. See Authenticate End User with SecureAccess on page 216. SecureAccess works with WebHDFS. The next point in the path is the BlueTalon File System Enforcement Point. BlueTalon File System Enforcement Point The BlueTalon File System Enforcement Point (FSEP) protects Hadoop Distributed File System data. FSEP is installed on each compute/data node computer. The next point in the path is HDFS. HDFS The Hadoop Distributed File System (HDFS) is the repository that stores data.
6	Software developer, data consumer, or other user type performs queries and other data requests. Security for the data access is enforced at each point in the access path. The data access is performed using WebHDFS and HTTP(S). Hive CLI The Hive Command line interface (CLI) is used to access Hive for performing tasks like queries, creating tables, deleting tables, and other database operations. The next point in the path is the BlueTalon Enhanced JAR. BlueTalon Enhanced JAR The BlueTalon Enhanced JAR enables the standard HDFS client to use the BlueTalon SecureAccess feature, which uses a shared secret to authenticate users. You will learn about SecureAccess later in this document. See Authenticate End User with SecureAccess on page 216. SecureAccess works with WebHDFS. The next point in the path is the BlueTalon File System Enforcement Point. BlueTalon File System Enforcement Point The BlueTalon File System Enforcement Point (FSEP) protects Hadoop Distributed File System data. FSEP is installed on each compute/data node computer. The next point in the path is HDFS. HDFS The Hadoop Distributed File System (HDFS) is the repository that stores data.
7	YARN Services/Spark Containers Containers for Spark YARN. YARN has containers that are not directly called. When HDFS data is needed SecureAccess is called. A user submits a job, the framework creates a container, and the user's SecureAccess shared secret is used for the data access that is protected by BlueTalon. The BlueTalon File System Enforcement Point (FSEP) is the next point in the path. BlueTalon File System Enforcement Point The BlueTalon File System Enforcement Point (FSEP) protects Hadoop Distributed File System data. FSEP is installed on each compute/data node computer. The next point in the path is HDFS. HDFS The Hadoop Distributed File System (HDFS) is the repository that stores data.
8	BlueTalon Service The BlueTalon Service secures access to the Ambari Management and Monitoring Service. Ambari Management and Monitoring The Ambari Management and Monitoring Services are used for provisioning, managing, monitoring, and

	securing Hadoop clusters. BlueTalon and Ambari integration is for FSEP.

Perform the steps in the following subsections to configure FSEP for HDP.

Disable WebHDFS Parameter

Disable the WebHDFS parameter:

1. Edit the HDFS default XML file. Example file:
   `/root/hadoop-2.7.2/share/doc/hadoop/hadoop-project-dist/hadoop-hdfs/hdfs-default.xml`
2. Set the dfs.webhdfs.enabled parameter to false:
```
<property>
  <name>dfs.webhdfs.enabled</name>
  <value>false</value>
  <description>
    Enable WebHDFS (REST API) in Namenodes and Datanodes.
  </description>
</property>
```
3. Save the file.

Disabling the WebHDFS parameter prevents WebHDFS file system commands from being executed, but keeps the NameNode HTTP port open. This ensures the NameNode user interface is accessible.

Hortonworks Sandbox Support

If you are using Hortonworks Sandbox, the following software versions are supported by BlueTalon and must be installed before performing the steps in subsequent sections:

- HDP 2.4
- Ambari 2.2.1.0

Collect Information

Same as for HDFS. See Collect Information on page 82.

Configure FSEP for HDP Using Ambari Plug In Script

To configure FSEP for HDP using the Ambari plug in script, perform the steps in the following section. The script enables you to perform the Ambari configuration using a simplified method.

If you have extensive customization requirements, you might need to use the alternative method that does not use the Ambari plug in script. See Configure FSEP for HDP Without Using Ambari Plug In Script on page 120.

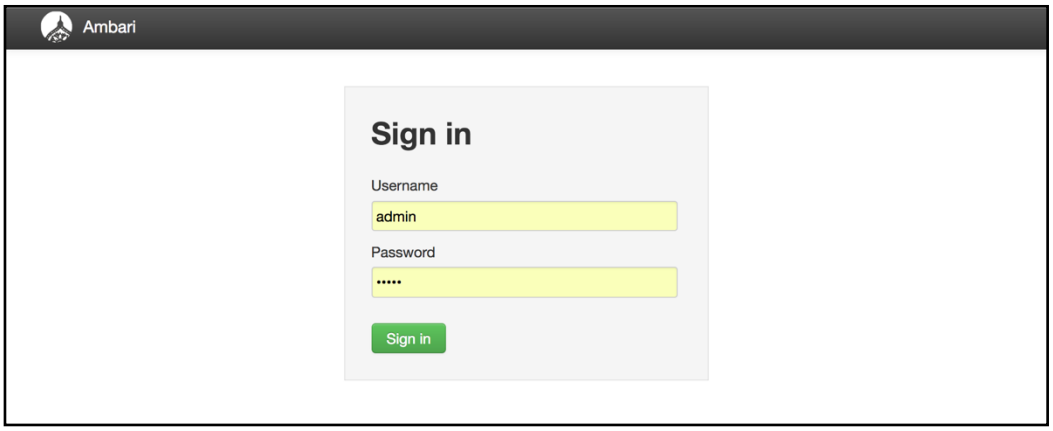
Install Ambari Plug In

Perform the following steps on the Ambari computer node.

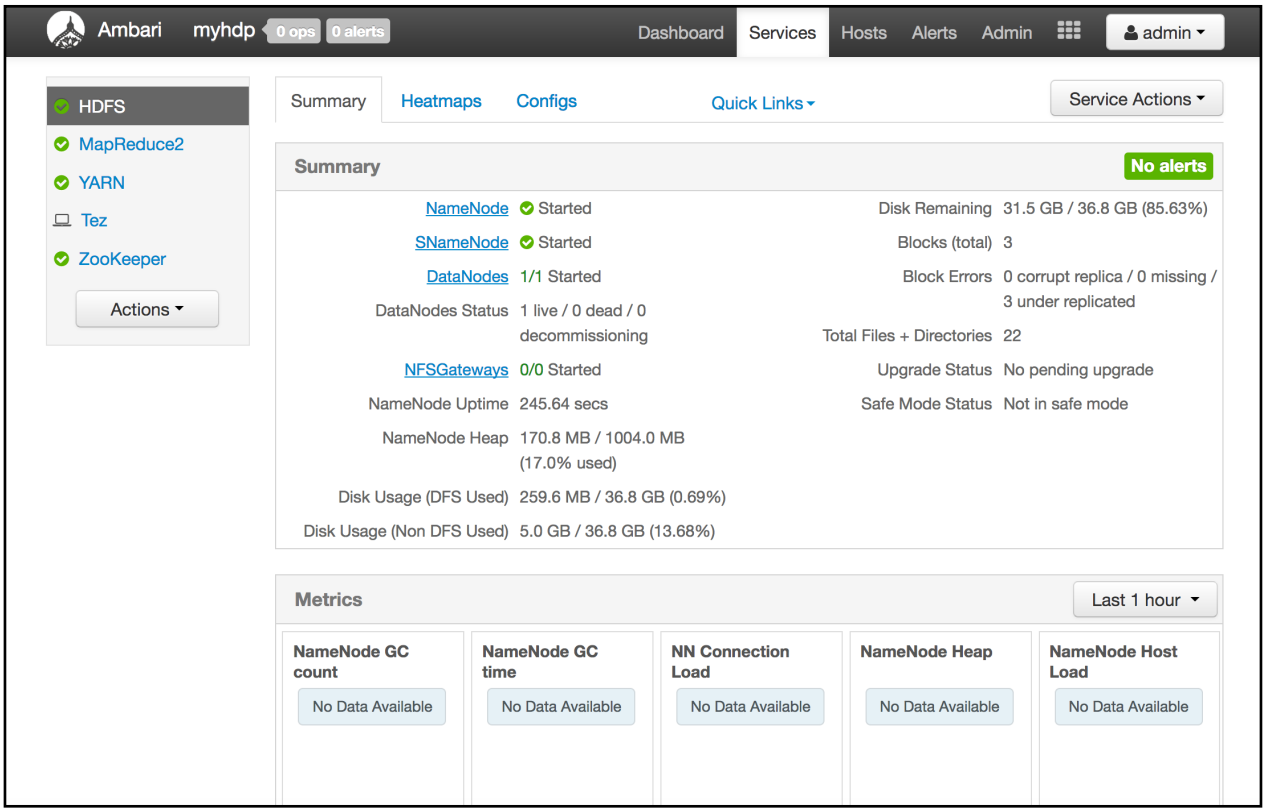
Log in to Ambari Administration Web Console

Log in to the Ambari administration Web console:

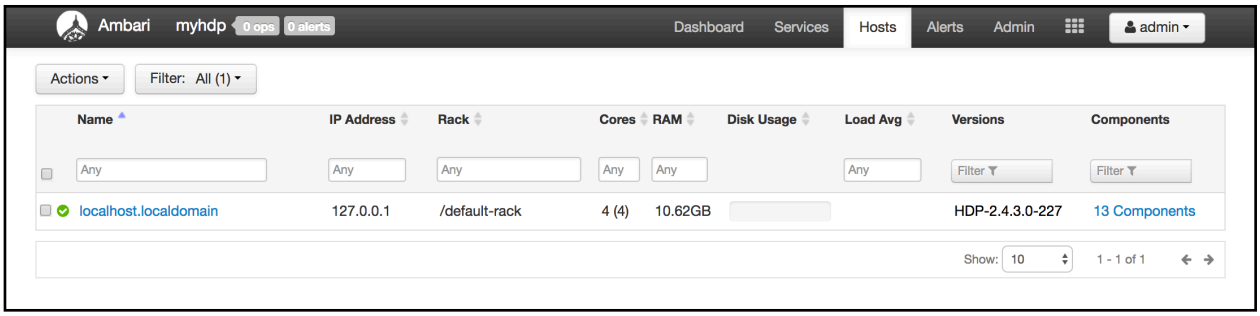
1. Using a Web browser, go to the Ambari administration Web console and log in.



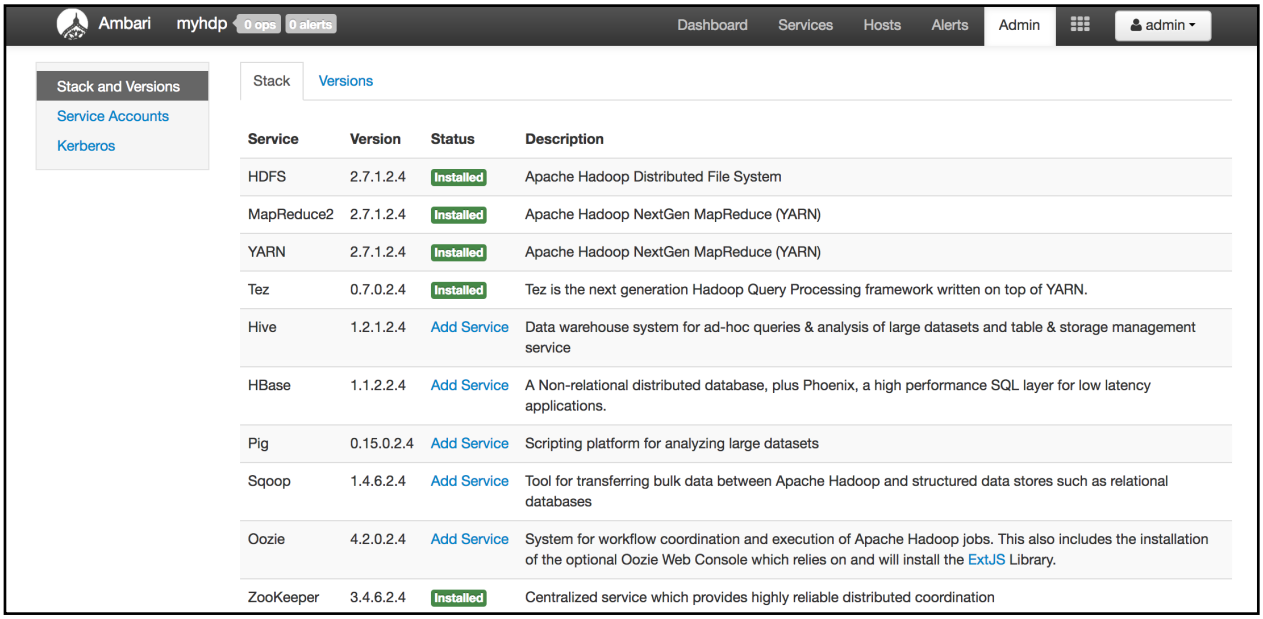
2. Examine the following screen.



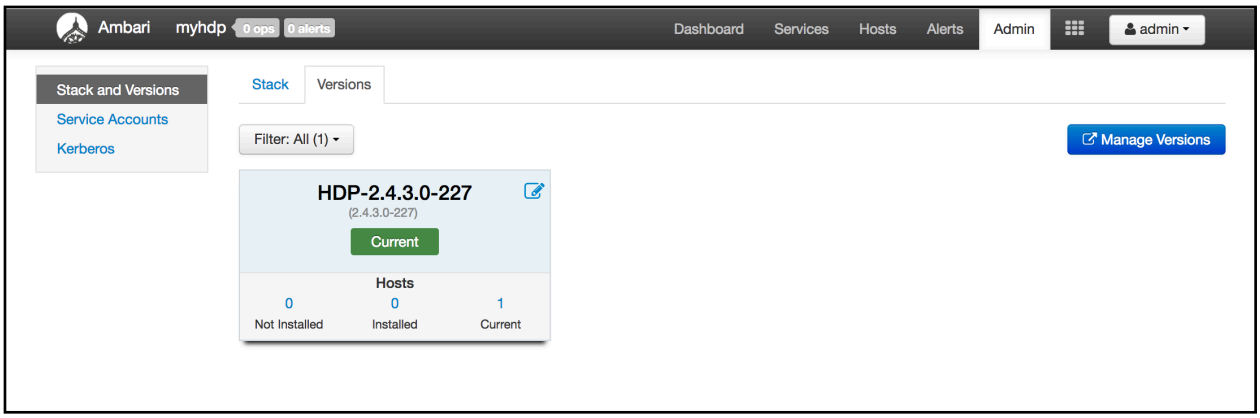
3. Click the Hosts tab.



4. Click the Admin tab. Click Stack and Versions.



5. Click the Versions tab.



Download and Run Ambari Script

Run the following commands to download and run the Ambari script:

```
wget https://repo.cloud.bluetalon.com/Ambari-Plugin/centos6/bt-ambari-plugin-install.sh

chmod +x bt-ambari-plugin-install.sh

./bt-ambari-plugin-install.sh
```

You will be prompted to enter:

- Ambari user name and password.
- BlueTalon software version number. Enter the version number you have purchased. Examples:
    - 3.2.4
    - Latest

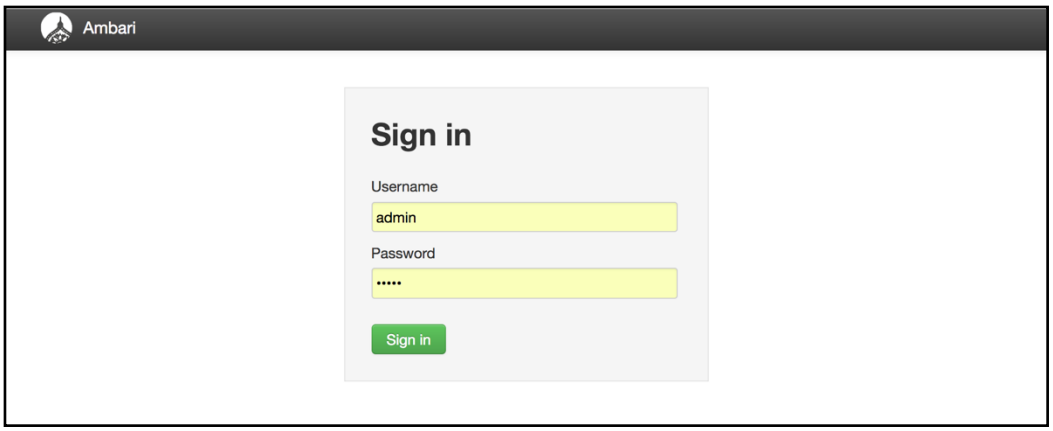
(Optional) You can also run the bt-ambari-plugin-install.sh script in silent non-interactive mode, and you are not prompted to enter the parameters. Example:

```
./bt-ambari-plugin-install.sh -accept -restart_ambari -u <Ambari user name> -p <Ambari password> -v <BlueTalon version>
```

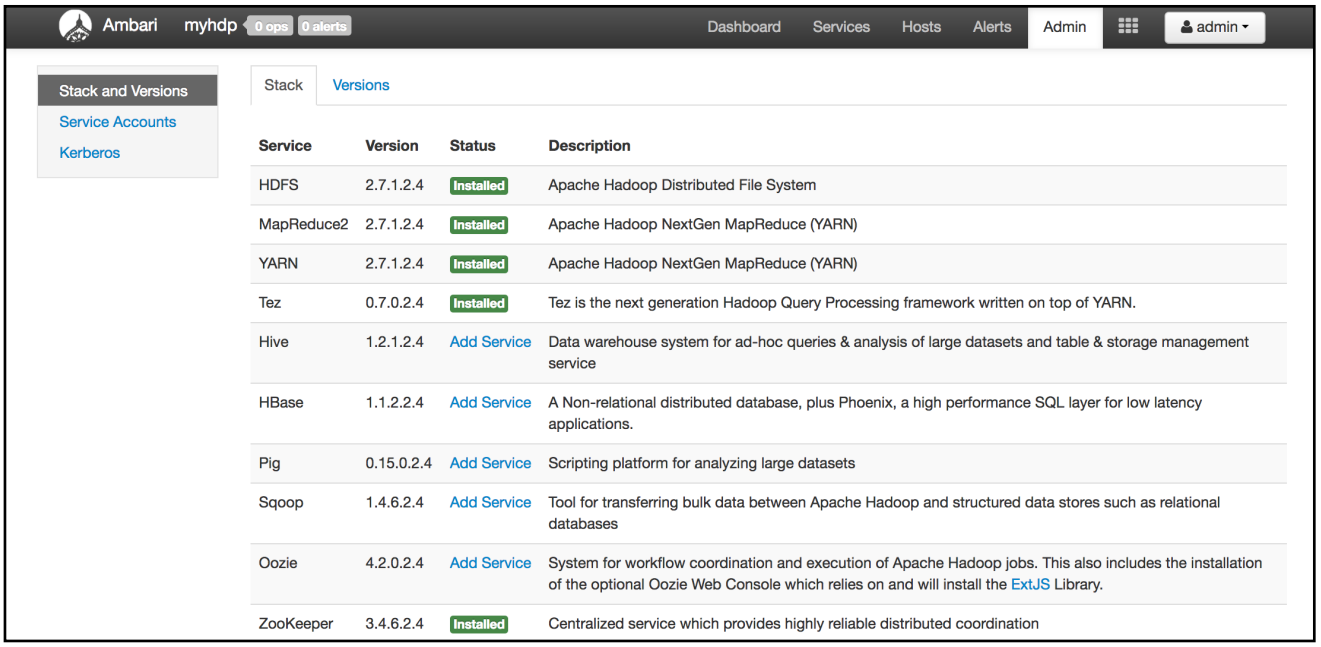
Add BlueTalon Services Using Ambari Administration Web Console

Add BlueTalon services using the Ambari administration Web console:

1. Using a Web browser, go to the Ambari administration Web console and log in.



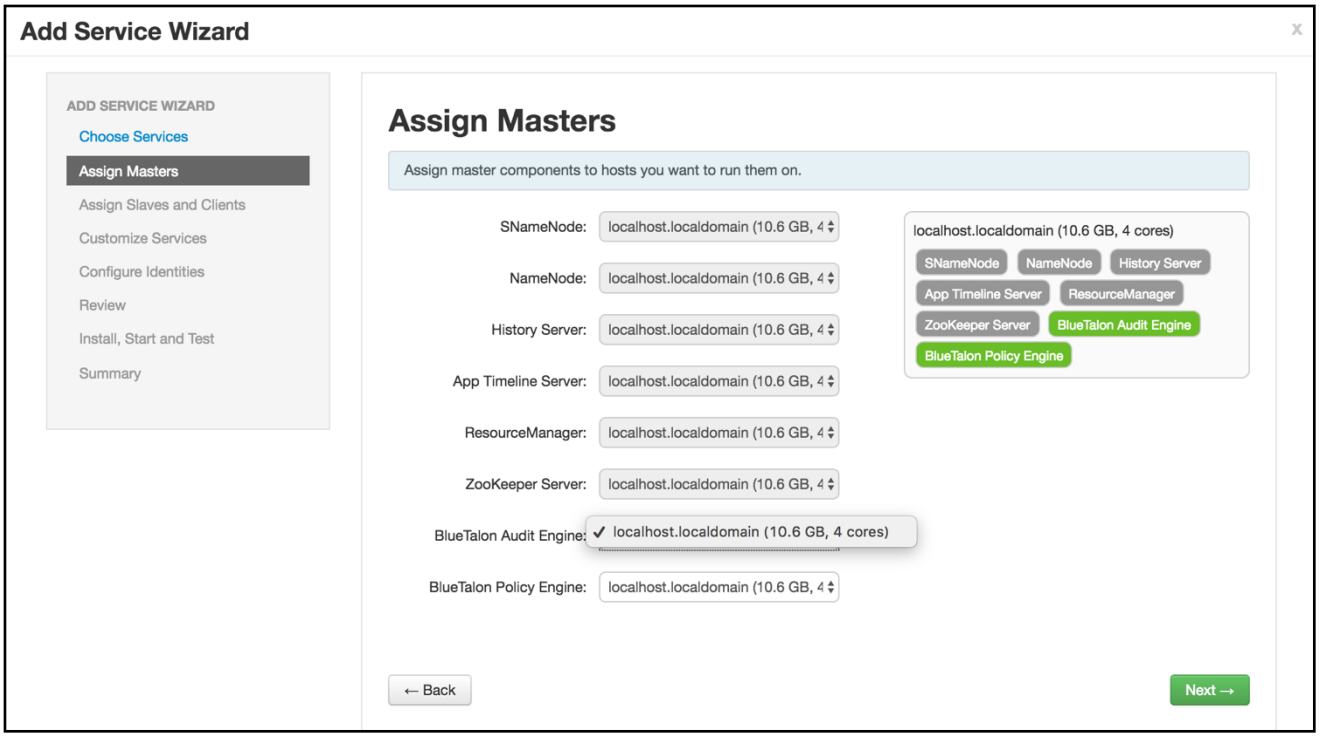
2. Click the Admin tab. Click Stack and Versions.



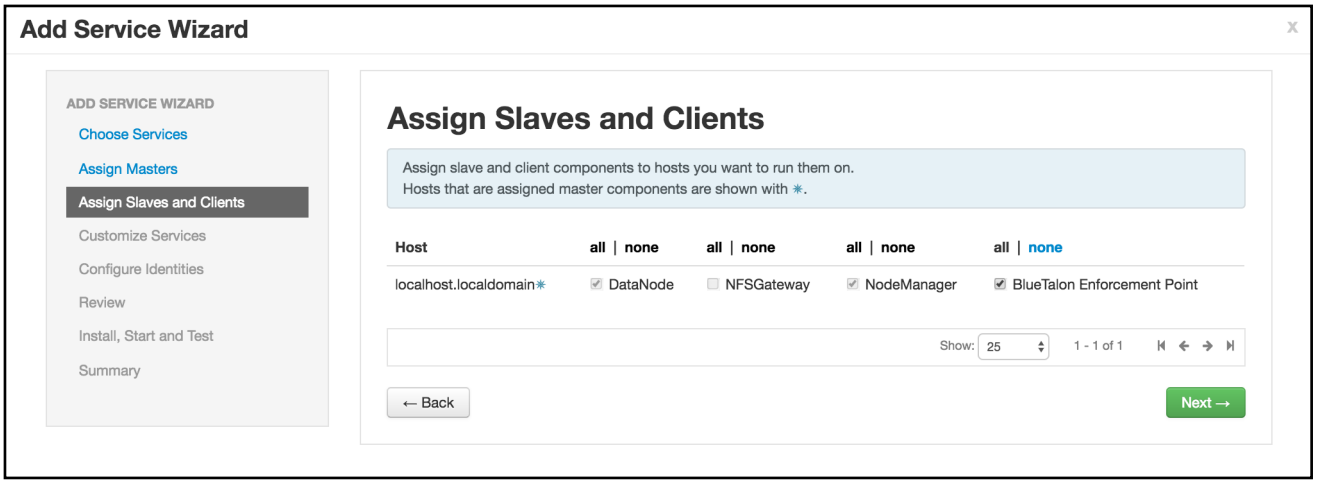
3. Scroll down the page to view the BlueTalon service. (The software version number will vary.)

4. Click Add Service.
5. Select BlueTalon and click Next.

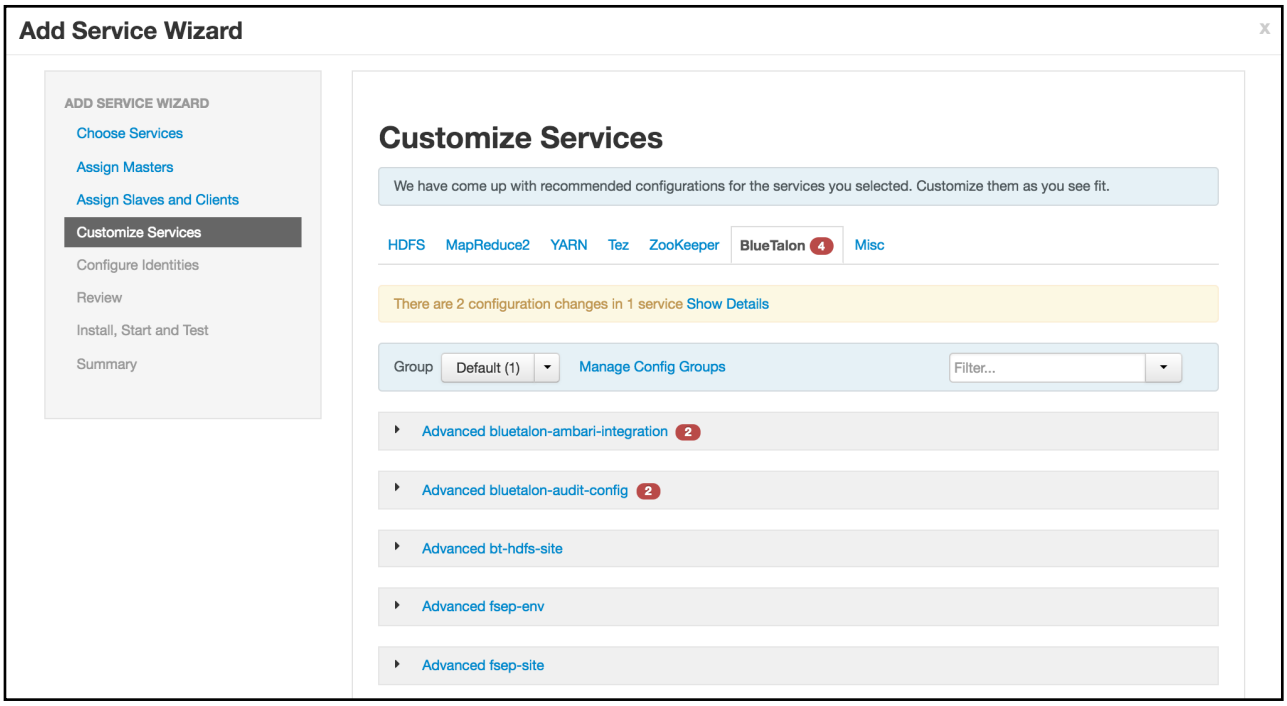
6. Select the computer(s) on which to install the BlueTalon Audit Engine and the Policy Engine. Then, click Next.



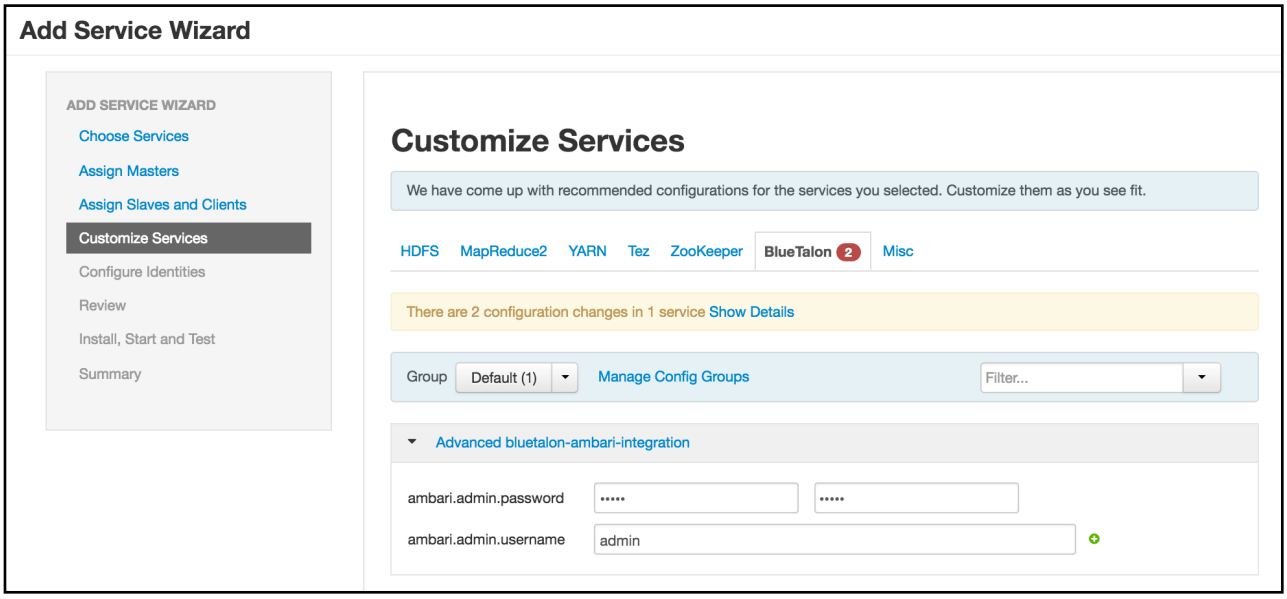
7. Assign the slave and client computers on which to install the BlueTalon Enforcement Point. Ensure you select the BlueTalon Enforcement Point option. Select all the computers if multiple computers are shown. Then, click Next.



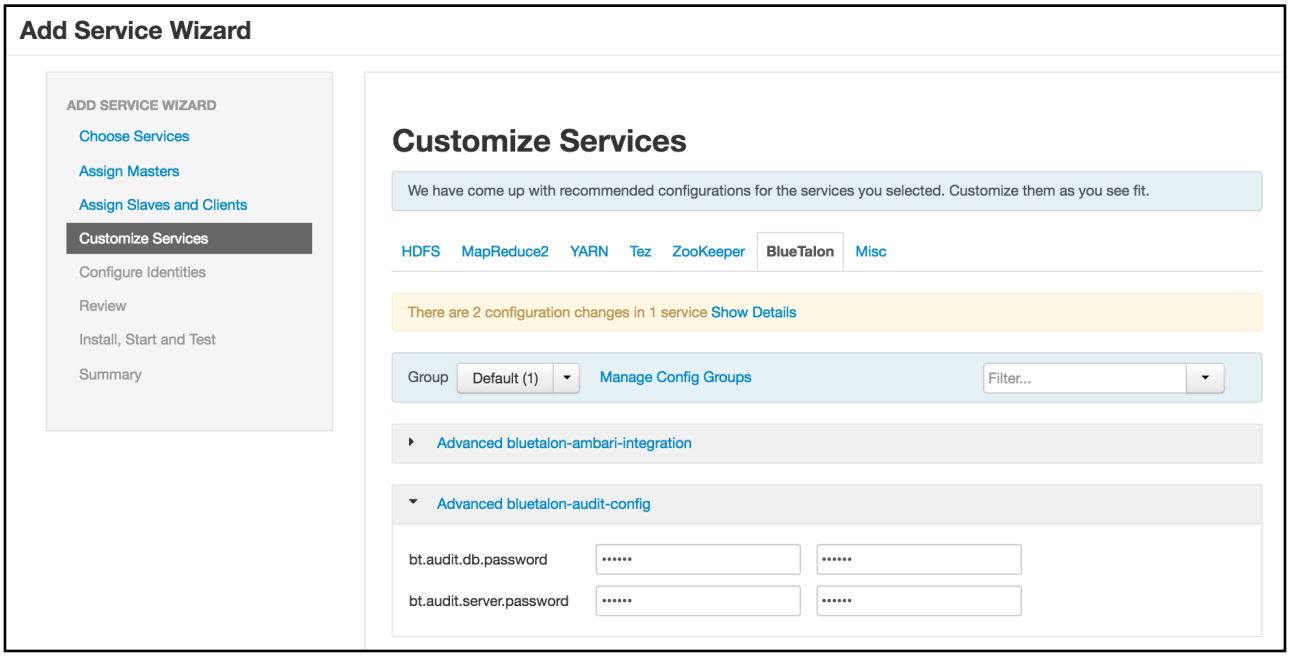
8. Examine the Customize Services screen.



9. Expand the section named Advanced bluetalon-ambari-integration. Enter your Ambari administration user name and password.



10. Expand the section named Advanced bluetalon-audit-config. Set the passwords for your BlueTalon installation. Example: bt#123. Set your own strong passwords.

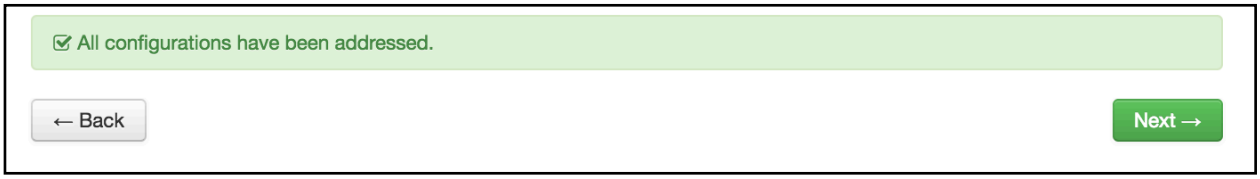


11. (Optional) You can set your own custom JAVA_HOME setting.

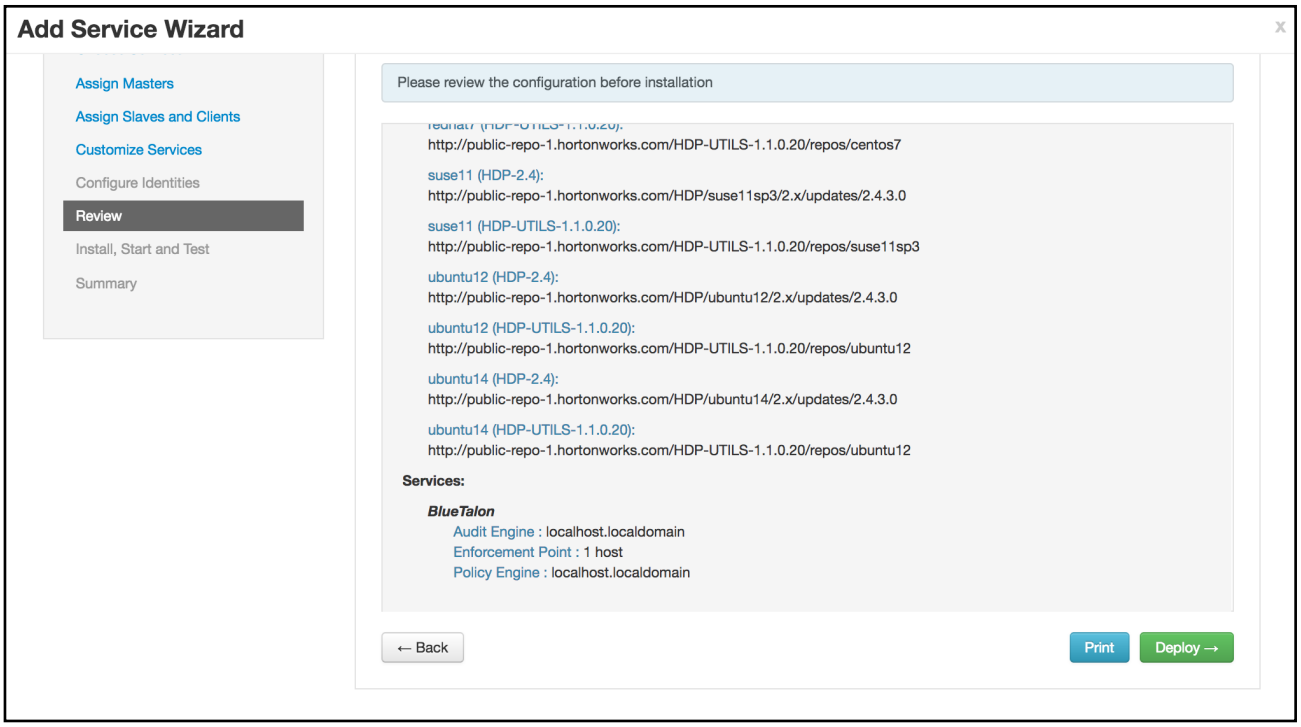
    Example:
    /usr/lib/jvm/jre-1.6.0-openjdk.x86_64

    See Java SE Runtime Environment on page 22 for supported versions.

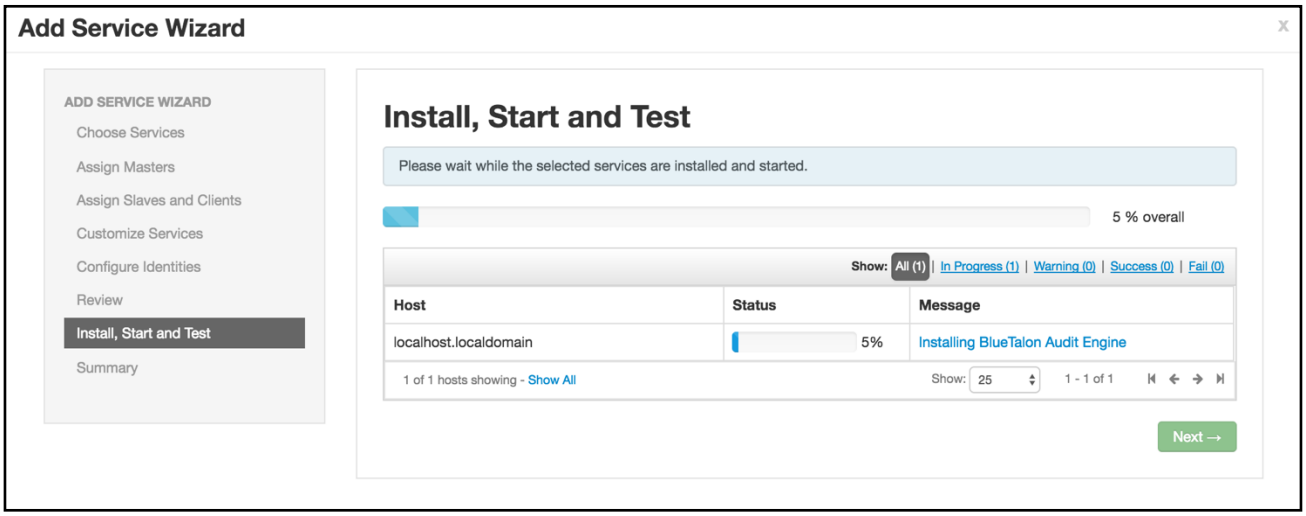
12. Scroll down to the bottom of the screen. Click Next.



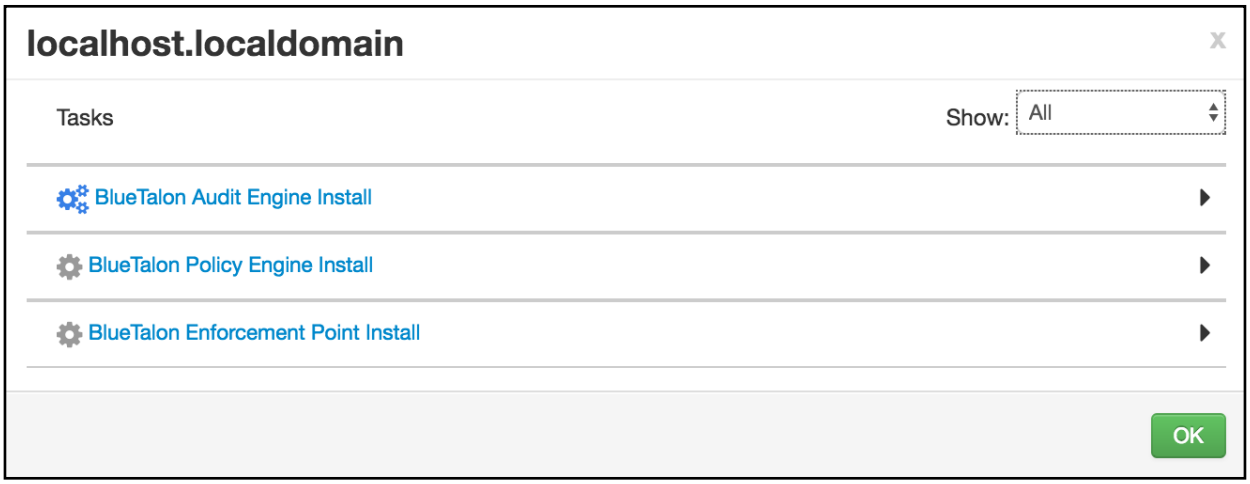
13. Examine the settings. Click Deploy.



14. The deployment begins. After the deployment is complete, click Next.



15. Examine the summary. Click OK. You will add a data domain next.



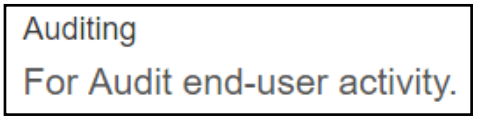
Log in to Audit Console

Log in to the Audit Console:

1. Open a compatible Web browser. See Policy Console and Audit Console Requirements on page 22.
2. Go to the Audit Console. URL format:
   ```
   http://<fully qualified domain name or IP
   address:port>/BlueTalonAudit
   ```

   Examples:
   ```
   http://TestServer.TestCompany.com:8112/BlueTalonAudit
   http://127.0.0.1:8112/BlueTalonAudit
   ```
3. Ensure the Audit Console is accessible. If you cannot access the Audit Console, ensure:
   a. Hostname for the computer is accessible.
   b. Firewall rules are correct.
4. Examine the license agreement.
5. If you accept the license agreement, select Accept.

6. Click Auditing.



7. Log in with the user name and the password.

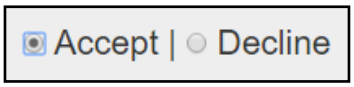
   Defaults:
   btadminuser
   P@ssw0rd

Log in to Policy Console

Log in to the Policy Console:

1. In the Web browser, go to the BlueTalon Policy Console. URL format:
   `http://<fully qualified domain name or IP address:port>/BlueTalonConfig`

   Examples:
   `http://TestServer.TestCompany.com:8111/BlueTalonConfig`
   `http://127.0.0.1:8111/BlueTalonConfig`
2. If you cannot access the Policy Console, ensure:
   a. Hostname for the computer is accessible.
   b. Firewall rules are correct.
3. Examine the license agreement.
4. If you accept the license agreement, select Accept.

5. Click Policies and Auditing.

6. Log in with the user name and the password.

   Defaults:
   btadminuser
   P@ssw0rd

Add Data Domain

A data domain stores information about an external data source.

You import the external data source information into a data domain: For HDFS, you can import folder and file information.

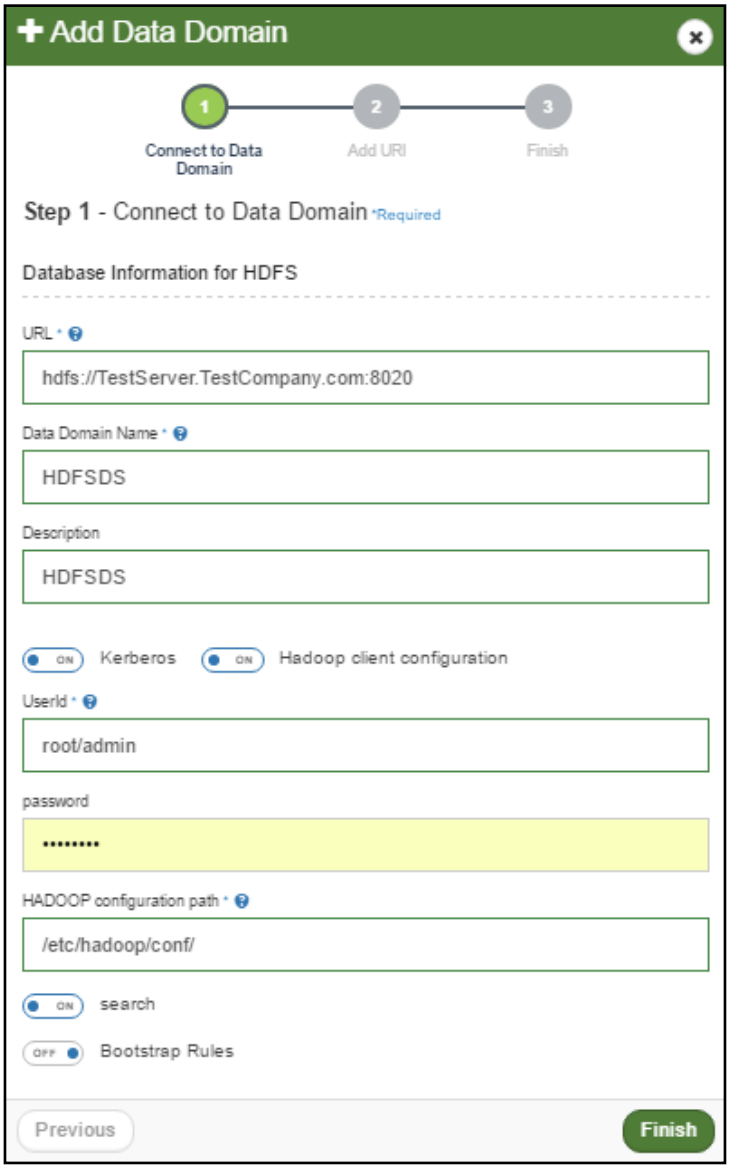
In the BlueTalon Policy Console, add a data domain:

1. Click the Data Domain tab.
2. Add a data domain with the following name:

   HDFSDS

   Use the name HDFSDS unless you explicitly set a different data domain name in the Ambari Web management console in the section bt-hdfs-site. If the data domain names do not match, then the policy and audit functionality will not work.
3. **Always set the Kerberos option on.** Set the option on if you are using a Kerberized or non-Kerberized cluster.

   The following screenshot shows example data domain settings.



4. Click the Deployment tab. Deploy the data domain.

Verify Data Cannot be Read

Same as for HDFS. See Verify Data Cannot be Read on page 87.

Allow Access to HDFS Data

(Optional) You can allow access to HDFS data, see Allow Access to HDFS Data on page 91.

Hive

(Optional) You can set up Hive, see Hive on page 100.

Configuration is complete.

Configure FSEP for HDP Without Using Ambari Plug In Script

(Optional) To configure FSEP for HDP without using the Ambari Plug In script, perform the steps in this section.

Use the method described here if you have extensive customization requirements.

Create Test Data

Same as for HDFS. See Create Test Data on page 82.

Add Data Domain

Same as for HDFS. See Add Data Domain on page 83.

Install Enforcement Point Package

Install the Enforcement Point package:

1. Enter:
   ```
   yum install bluetalon-ep-3.2.4
   ```
2. Follow the prompts.

Configure Enforcement Point

Configure the Enforcement Point:

1. Run:
   ```
   /opt/bluetalon/3.2.4/pep/scripts/bluetalon-ep-setup
   ```
2. Enter the information.

Item	Description
Enforcement Point type	Type of Enforcement Point. Select FSEP.
Kafka Broker hostname	Hostname where the bt-audit-kafka service is running. Example: Kafka-Broker-host.
Policy Engine hostname	Computer hostname or IP address on which the BlueTalon Policy Engine is running.
Data domain name	Name of the data domain. Use the value set in the Policy Console for the data domain you created earlier.

Restart FSEP Service

Restart FSEP service:

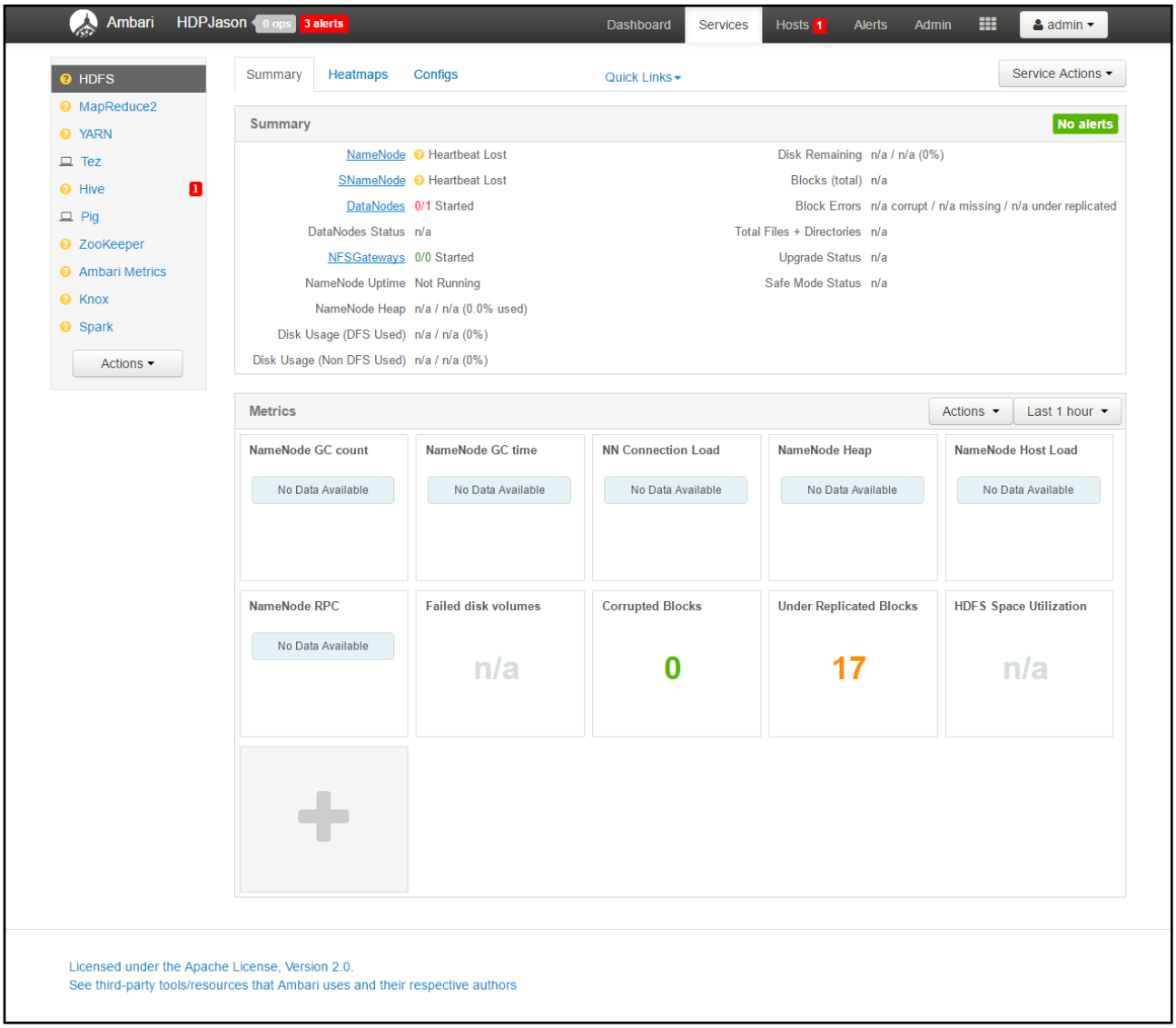
```
service bt-fsep restart
```

Configure Core Sites

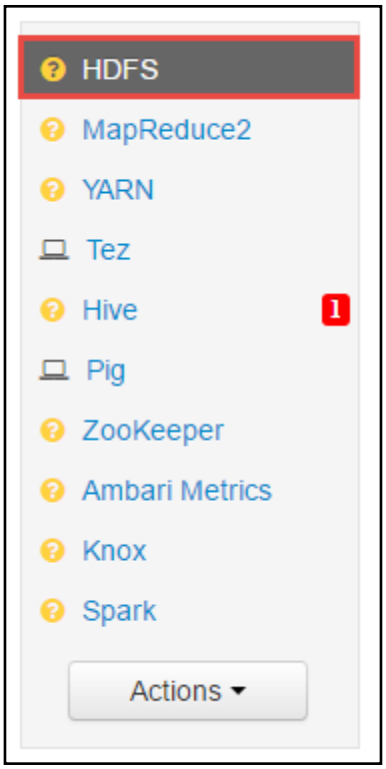
Perform the following steps to ensure your core sites are correctly configured.

1. Open Web browser.

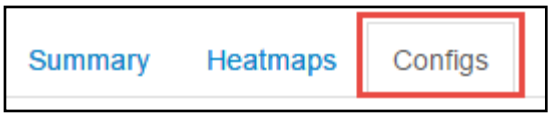
2. Go to the Ambari Web Manager console.



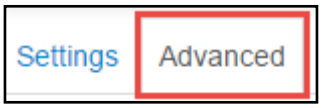
3. Click HDFS on the left of the screen.



4. Click Configs.



5. Click Advanced.



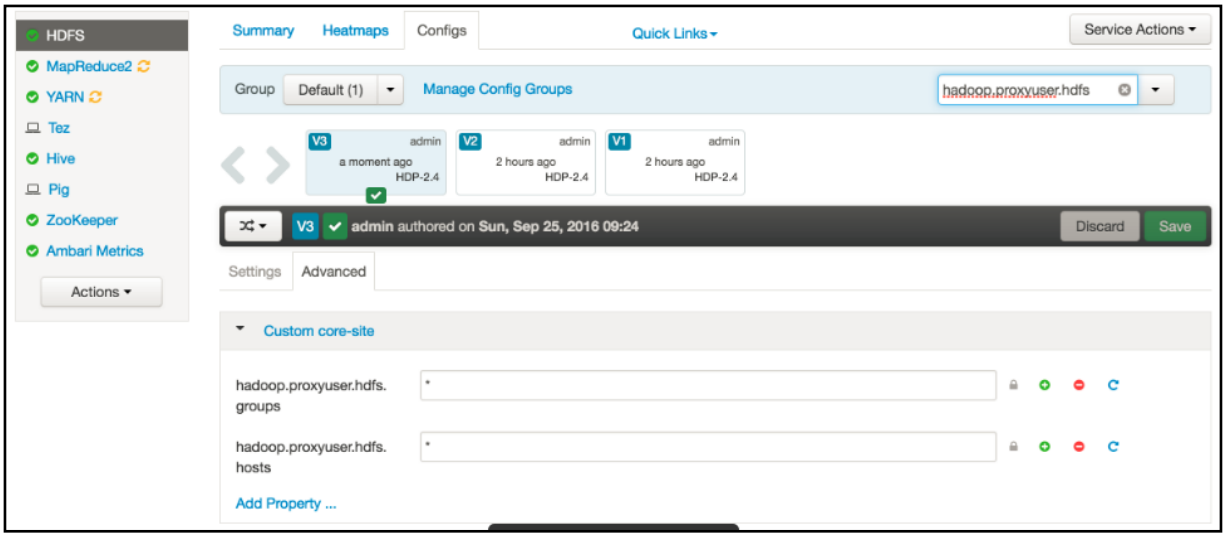
6. Click Advance core-site.

7. Set the property fs.defaultFS to webhdfs://localhost:40070. If the property is not present, add the property.
8. Click HDFS → Configs → Advanced → Custom core-site.

9. Set fs.defaultFS.native to hdfs://<NameNode IP address>:8020. If the property is not present, add the property.

fs.defaultFS.native	hdfs://<NameNode IP address>:8020

10. Set fs.defaultFS.fsep to webhdfs://localhost:40070. Localhost is used for the local FSEP on that node by the Hadoop client. If the property is not present, add the property.
11. If you are using FSEP in Audit mode, set the HDFS groups and HDFS hosts properties:
    a. Set the property hadoop.proxyuser.hdfs.groups to your HDFS groups. Example: Set the property to * for all groups.
    b. Set the property hadoop.proxyuser.hdfs.hosts to your HDFS hosts. Example: Set the property to * for all hosts.



12. **(Optional)** By default, FSEP supports cross cluster traffic. **This is enabled by default, and you should typically leave it enabled**. To disable, use the Ambari Web Manager console:
    a. Add the following core site property and set it to false:
    `btwebhdfs.remote.cluster.support.enabled=false`
    b. Restart HDFS, YARN, and MapRed services, plus any dependent services.

Update BlueTalon FSEP Site Configuration File

Update the BlueTalon FSEP site configuration file:

1. Edit the file:
   `/etc/bluetalon/pep/fs-ep/conf/fsep-site.xml`
2. Add the following lines to the file:

```
<property>
<name>fsep.hadoop.config.dir</name>
<value>/etc/bluetalon/pep/fs-ep/hadoop/conf</value> <!-- changed
from default value of /etc/hadoop/conf-->
</property>
```

Create BlueTalon Core Site Configuration File

Create a BlueTalon FSEP core site configuration file:

1. Create the file /etc/bluetalon/pep/fs-ep/hadoop/conf/core-site.xml.

2. Add the following lines to the file:
```
<configuration>
<property>
<name>fs.defaultFS</name>
<value>hdfs://<Host name or IP address of data storage end point
computer>:8020</value>
</property>
```

Complete FSEP Configuration

To complete the FSEP configuration:

1. Stop all cluster services in Ambari.

   Do not restart the services yet. If you do, Ambari will report that some services are down. Example: NameNode service.

2. Edit the file /etc/bluetalon/pep/fs-ep/conf/fsep-env.sh.
3. Add the following line to the file:
```
export CATALINA_OPTS=" $CATALINA_OPTS -
Dfs.defaultFS.fsep=\$\{fs.defaultFS.native\} "
```
4. Restart the FSEP service:
```
service bt-fsep restart
```
5. Restart all cluster services using Ambari that you stopped in the first step. Restarting the services forces the configuration changes to all of the nodes.

Ensure Services are Running

To ensure the services are running:

1. Open Web browser.
2. Go to the Ambari configuration page.
3. Ensure the services are running. All services should be green. If any service does not start, ensure all BlueTalon policies are correctly defined in the Policy Console.

Verify Data Cannot be Read

Same as for HDFS. See Verify Data Cannot be Read on page 87.

Install HDFS EP after FSEP Install Using Ambari

If you have installed FSEP using Ambari, and also need to install HDFS EP, perform these steps:

1. On the NameNodes and Secondary NameNodes:
```
yum install bluetalon-ep-3.2.4
```
2. Run the EP setup script on the NameNodes and Secondary NameNodes:
```
/opt/bluetalon/current/pep/scripts/bluetalon-ep-setup
```
   a. Follow the script prompts.
   b. Select the HDFS option.
   c. The script automatically enables native HDFS permission fallback.
   Recommendation: Keep the setting.

   (To disable HDFS permission fallback, edit the file `/etc/bluetalon/pep/fs-ep/conf/bt-hdfs-site.xml` and set `bt.hdfs.plugin.union.hdfs.native.perms` to false. The default is true.)

3. For an Ambari managed cluster, set the following property in the HDFS configuration in the "custom hdfs-site" section of the Ambari Web management user interface:
   `dfs.namenode.inode.attributes.provider.class = com.bluetalon.hdfs.core.BlueTalonAccessControlEnforcer`
4. Repeat the previous steps for all NameNodes and Secondary NameNodes.
5. Restart all NameNodes and Secondary NameNodes.

Uninstall HDFS EP

If you need to uninstall the HDFS EP, perform these steps:

1. Run:
   `/opt/bluetalon/current/pep/hdfs-ep/bin/bt-hdfs-undeploy.sh`
2. Restart the NameNode.
3. Repeat the previous steps for all NameNodes and Secondary NameNodes.
4. For an Ambari managed cluster, perform these steps:
   a. Log in to the Ambari server.
   b. Obtain the bluetalon property from the configuration file:
      ```
      /var/lib/ambari-server/resources/scripts/configs.sh -u <Ambari user name> -p <Ambari user password> -port <Ambari port> get <Ambari server IP address or host name> <Ambari cluster name> <configuration file name> | grep bluetalon
      ```
      Example:
      ```
      /var/lib/ambari-server/resources/scripts/configs.sh -u admin -p admin -port 8080 get AmbariServer.TestCompany.internal AMBARI-TESTHDP hdfs-site | grep bluetalon
      ```
      Example output with key name:
      ```
      "dfs.namenode.inode.attributes.provider.class" : "com.bluetalon.hdfs.core.BlueTalonAccessControlEnforcer",
      ```
   c. Delete the bluetalon property using the key name:
      ```
      /var/lib/ambari-server/resources/scripts/configs.sh -u <Ambari user name> -p <Ambari user password> -port <Ambari port> delete <Ambari server IP address or host name> <Ambari cluster name> <configuration file name> <key name>
      ```
      Example:
      ```
      /var/lib/ambari-server/resources/scripts/configs.sh -u admin -p admin -port 8080 delete AmbariServer.TestCompany.internal AMBARI-TESTHDP hdfs-site "dfs.namenode.inode.attributes.provider.class"
      ```
   d. Restart the HDFS service using the Ambari Web management user interface.

Configure Enforcement Point Using Setup Script in Silent Mode

Ensure you have deployed your data domain before deploying an Enforcement Point.

(Optional) You can also configure an Enforcement Point in silent mode, which does not prompt you to enter details.

To configure the Enforcement Point with a preconfigured XML file, run:

```
/opt/bluetalon/3.2.4/pep/scripts/bluetalon-ep-setup -accept -f
/tmp/hdp.xml
```

For the parameter list, see Enforcement Point Silent Mode XML Parameters on page 65.

Example FSEP Silent Mode XML File for HDP Cluster

Example FSEP silent mode XML file for HDP cluster with Ambari post install:

```
<EnforcementPoint xmlns="">
  <db_type>8</db_type>
  <resource_domain_name>HDFSDataDomain</resource_domain_name>
  <audit_ip>127.0.0.1</audit_ip>
  <pdp_hostname>127.0.0.1</pdp_hostname>
  <hadoop_conf_path>/etc/hadoop/conf/</hadoop_conf_path>
  <traffic_divert>yes</traffic_divert>
  <kerberos>no</kerberos>
  <post_install_conf>true</post_install_conf>
  <ambari_ip>127.0.0.1</ambari_ip>
  <ambari_port>8080</ambari_port>
  <ambari_login>admin</ambari_login>
  <ambari_passwd>admin</ambari_passwd>
</EnforcementPoint>
```

HDP configuration is complete.

Configure FSEP for CDH

CDH is Cloudera's open-source platform distribution for Apache Hadoop. BlueTalon is secure certified by Cloudera. To configure FSEP for CDH, perform the steps in this section.

Collect Information

Same as for HDFS. See Collect Information on page 82.

Create Test Data

Same as for HDFS. See Create Test Data on page 82.

Download and Run Cloudera Script

Run the following commands to download and run the Cloudera script:

```
wget https://repo.cloud.bluetalon.com/Cloudera-Manager-
Plugin/centos6/bt-cloudera-manager-install.sh
```

```
chmod +x bt-cloudera-manager-install.sh
```

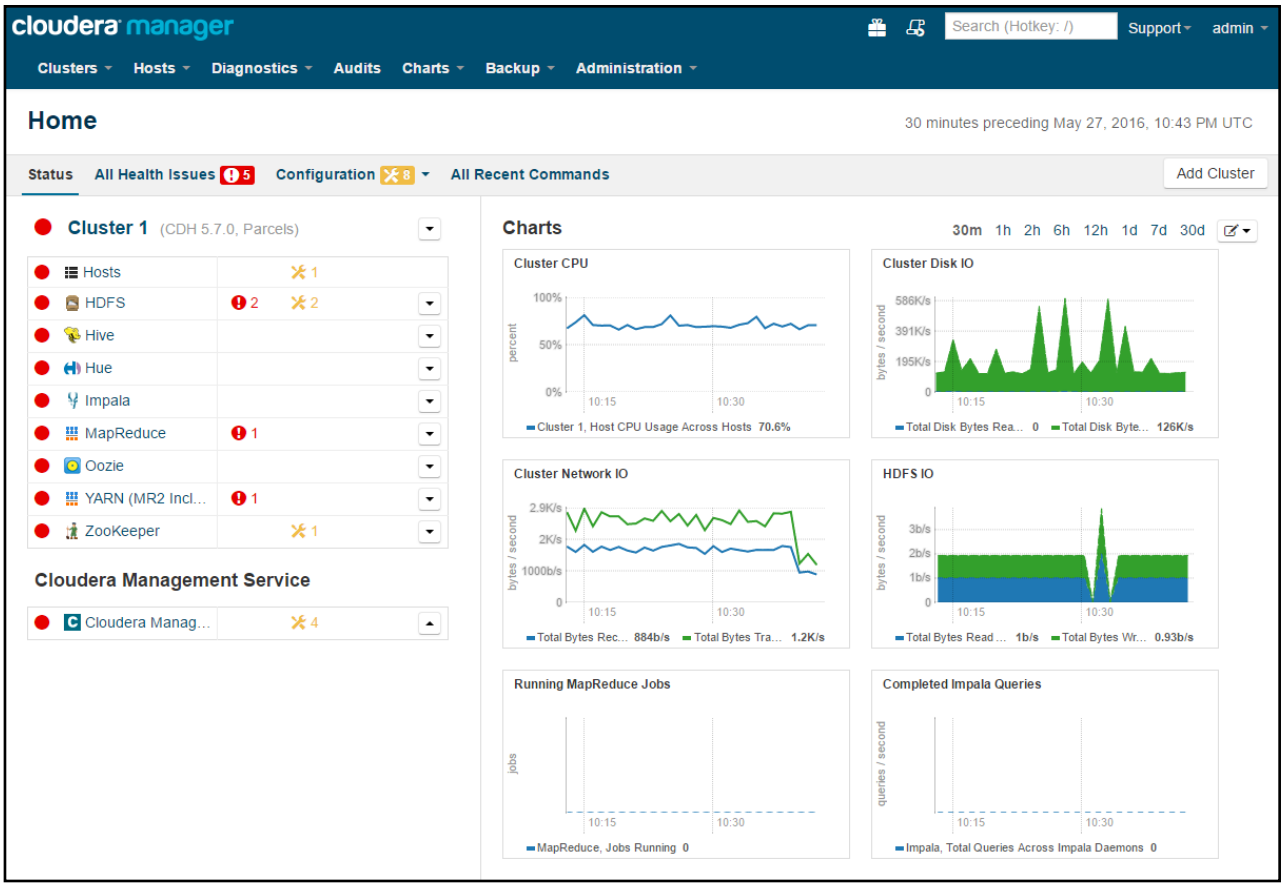
```
./ bt-cloudera-manager-install.sh
```

Follow the prompts.

Install BlueTalon Packages Using Cloudera Manager

To install BlueTalon packages using the Cloudera Manager:

1. Wait a few minutes for the Cloudera Manager Web interface to restart.
2. In a Web browser, log in to Cloudera Manager (default user name: admin, default password: admin). Example:



3. Restart the Cloudera Management Service.
4. Click the Parcels icon at the top of the screen.

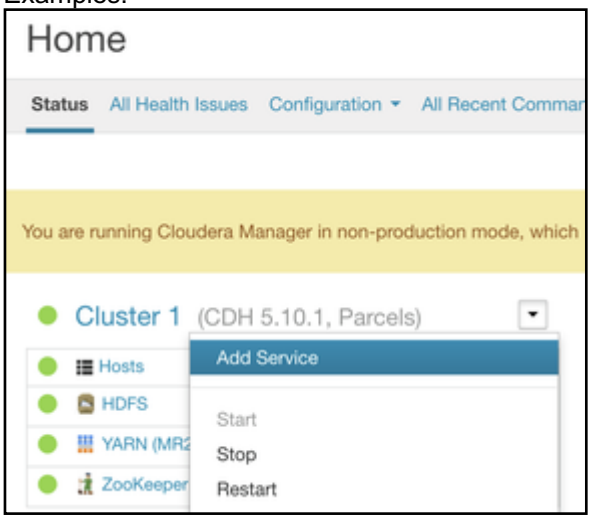
5. Click the Download button for the BlueTalon software.

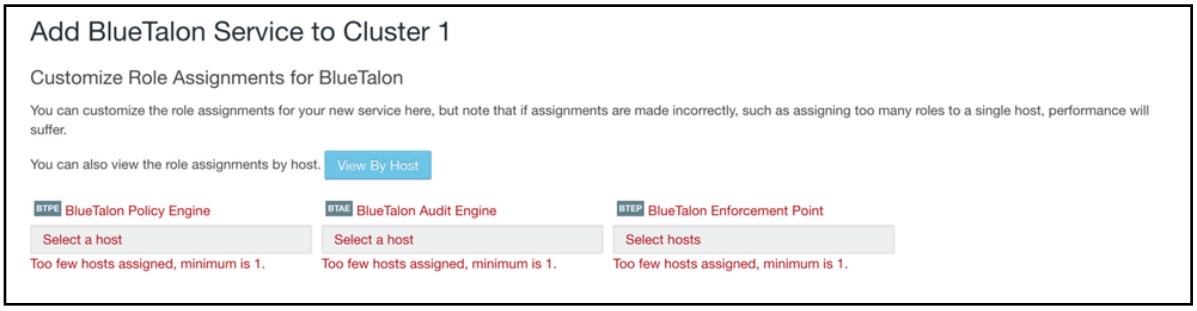
6. After the download is complete, click Distribute.
7. After the distribution is complete, click Activate.
8. Select the computer nodes to install these packages (you must select at least one computer node for each package):
    a. Audit

b. Policy

c. FSEP

Examples:





9. Set the properties:

a. Set the bt.audit.kakfa.host to the computer host name selected for the Audit Engine in the previous step. Example:

b. Set the bt.hdfs.plugin.config.address to the computer host name selected for the Policy Engine in the previous step, plus ":1600" for the port. Example:

c. Set the bt.kafka.metadata.broker.list to the computer host name selected for the Audit Engine in the previous step, plus ":9093" for the port. Example:

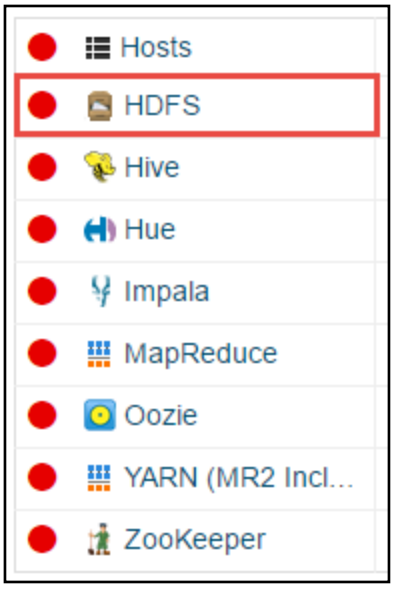
10. Wait for the installation to complete. Example:

After this installation, traffic is not routed through FSEP. To route traffic through FSEP, perform the steps in the following section.

Configure CDH Properties to Route Traffic Through FSEP

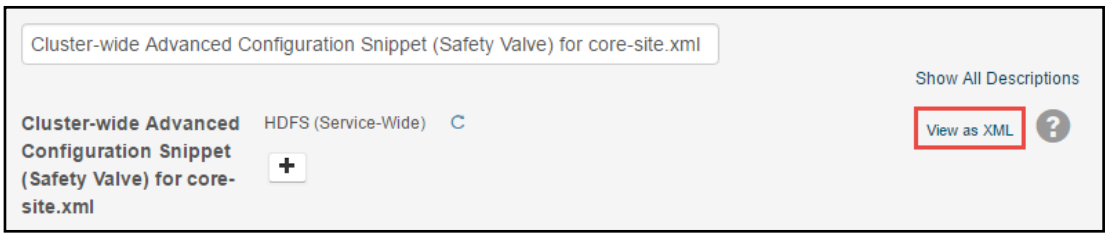
To configure CDH properties to route traffic through FSEP:

1. In a Web browser, log in to Cloudera Manager.
2. Click HDFS on the left of the screen.



3. Click Configuration.

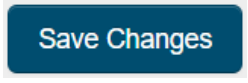
4. Search for the property "Cluster-wide Advanced Configuration Snippet (Safety Valve) for core-site.xml".



5. Click View as XML.
6. Set the property values. In the following example, the computer is localhost. Use your own computer name.

```
<property>
  <name>fs.AbstractFileSystem.webhdfs.impl</name>
  <value>com.bluetalon.btfs.fs.http.client.BtWebHdfs</value>
</property>
<property>
  <name>fs.webhdfs.impl</name>

<value>com.bluetalon.btfs.fs.http.client.BtWebHdfsFileSystem</value>
</property>
<property>
  <name>fs.defaultFS.fsep</name>
  <value>webhdfs://localhost:40070</value>
</property>
<property>
  <name>fs.defaultFS.native</name>
  <value>hdfs://localhost:8020</value>
</property>
<property>
  <name>fs.defaultFS</name>
  <value>${fs.defaultFS.fsep}</value>
</property>
```

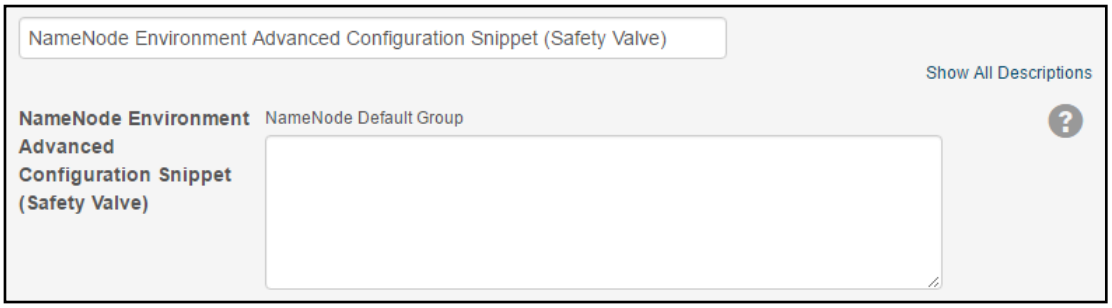
7. Click Save Changes.



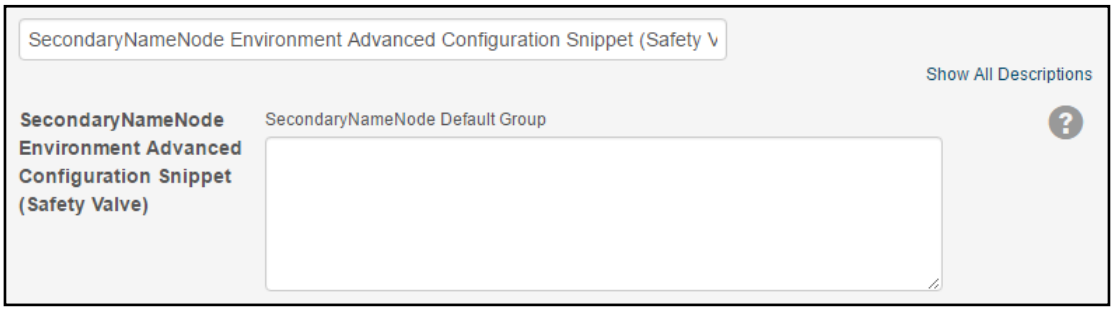
8. In a command line shell, edit the following file:
`/etc/bluetalon/pep/fs-ep/conf/fsep-env.sh`
9. Add the following line to the file and save the file. Localhost is used. Use your own computer name.

```
export CATALINA_OPTS=" -Dfs.defaultFS.fsep=\$\{fs.defaultFS.native\}
${CATALINA_OPTS} "
```

10. In the Web browser, go to Cloudera Manager.
11. Search for the property "NameNode Environment Advanced Configuration Snippet (Safety Valve)".

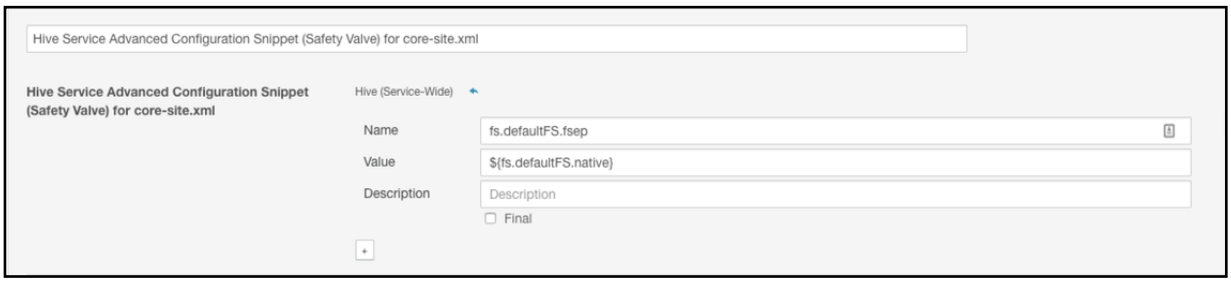


12. Set the property value. In the following example, the computer is localhost. Use your own computer name.
    `HADOOP_NAMENODE_OPTS=-Dfs.defaultFS.fsep=${fs.defaultFS.native}`
13. Click Save Changes.
14. Search for the property "SecondaryNameNode Environment Advanced Configuration Snippet (Safety Valve)".

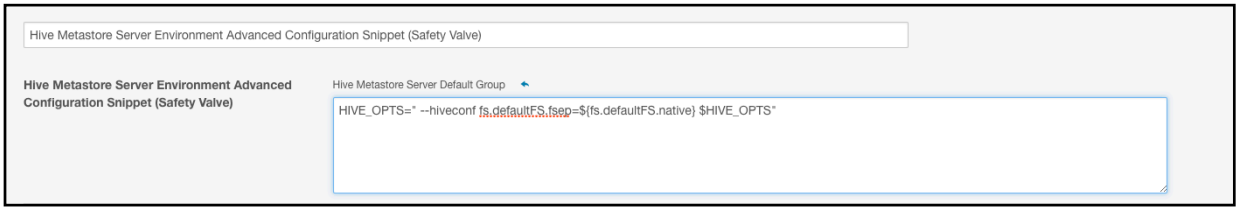


15. Add the property value.
    `HADOOP_SECONDARYNAMENODE_OPTS=-Dfs.defaultFS.fsep=${fs.defaultFS.native}`
16. Click Save Changes.
17. Click Clusters.
18. Click Hive.

19. Search for the property "Hive Service Advanced Configuration Snippet (Safety Valve) for core-site.xml".
20. Add the property value.
    `fs.defaultFS.fsep=${fs.defaultFS.native}`



21. Search for the property "Hive Metastore Server Environment Advanced Configuration Snippet (Safety Valve)".
22. Add the property value.
    `HIVE_OPTS=" --hiveconf fs.defaultFS.fsep=${fs.defaultFS.native} $HIVE_OPTS"`



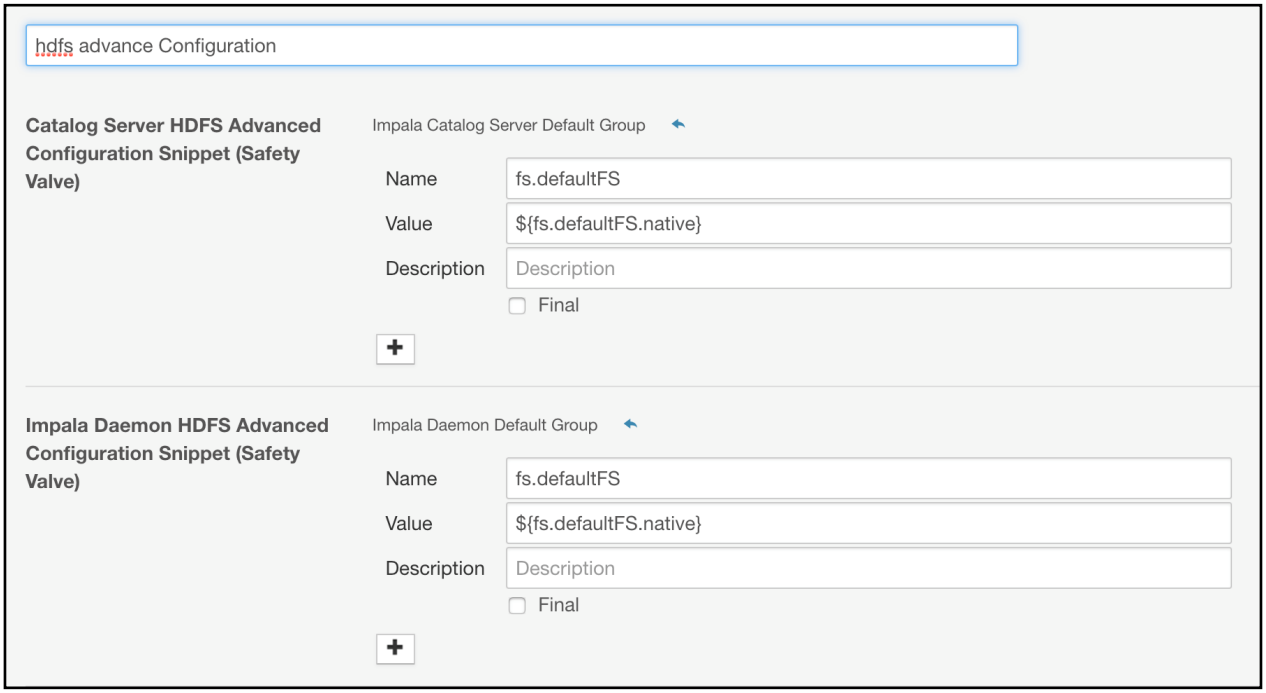
23. Search for the property "HiveServer2 Environment Advanced Configuration Snippet (Safety Valve)".
24. Add the property value.
    `HIVE_OPTS=" --hiveconf fs.defaultFS.fsep=${fs.defaultFS.native} $HIVE_OPTS"`

25. Click Clusters.
26. Click Impala.

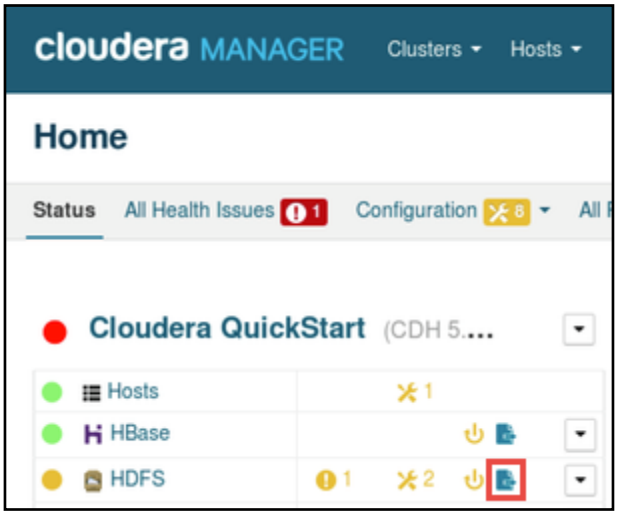
27. Click Configuration.
28. Set these properties:

    Catalog Server HDFS Advanced Configuration Snippet (Safety Valve)
    Impala Daemon HDFS Advanced Configuration Snippet (Safety Valve)

    to this value:
    `fs.defaultFS=${fs.defaultFS.native}`



29. Click Save Changes.
30. To ensure there is no stale configuration, deploy the configuration. Click the blue icon next to HDFS.



31. Click Cloudera Management Service.
32. Click the configuration tab.
33. Search for the property "Java Configuration Options for Service Monitor".
34. Set the property.
`-Dfs.defaultFS.fsep=${fs.defaultFS.native}`

35. Restart the cluster.

Restart FSEP Service

Restart the FSEP service:

1. In Cloudera Manager, click BlueTalon.
2. Click the BlueTalon Enforcement Point.
3. Click the Actions button near the top of the screen, and select the Restart option.

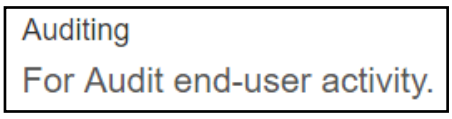
Log in to Audit Console

Log in to the Audit Console:

1. Open a compatible Web browser. See Policy Console and Audit Console Requirements on page 22.
2. Go to the Audit Console. URL format:
   ```
   http://<fully qualified domain name or IP
   address:port>/BlueTalonAudit
   ```

   Examples:
   ```
   http://TestServer.TestCompany.com:8112/BlueTalonAudit
   http://127.0.0.1:8112/BlueTalonAudit
   ```
3. Ensure the Audit Console is accessible. If you cannot access the Audit Console, ensure:
   a. Hostname for the computer is accessible.
   b. Firewall rules are correct.
4. Examine the license agreement.
5. If you accept the license agreement, select Accept.

6. Click Auditing.



7. Log in with the user name and the password.

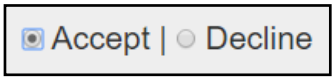
   Defaults:
   btadminuser
   P@ssw0rd

Log in to Policy Console

Log in to the Policy Console:

1. In the Web browser, go to the BlueTalon Policy Console. URL format:
   `http://<fully qualified domain name or IP address:port>/BlueTalonConfig`

   Examples:
   `http://TestServer.TestCompany.com:8111/BlueTalonConfig`
   `http://127.0.0.1:8111/BlueTalonConfig`
2. If you cannot access the Policy Console, ensure:
   a. Hostname for the computer is accessible.
   b. Firewall rules are correct.
3. Examine the license agreement.
4. If you accept the license agreement, select Accept.

5. Click Policies and Auditing.

6. Log in with the user name and the password.

   Defaults:
   btadminuser
   P@ssw0rd

Add Data Domain

A data domain stores information about an external data source.

You import the external data source information into a data domain: For HDFS, you can import folder and file information.

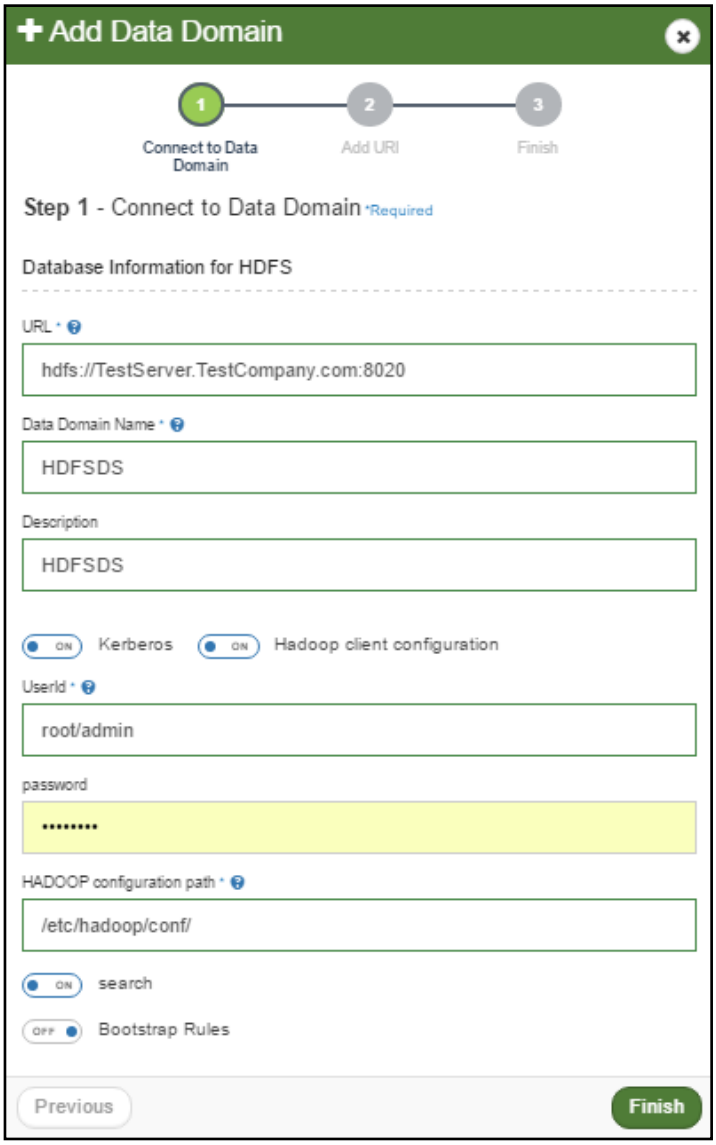
In the BlueTalon Policy Console, add a data domain:

1. Click the Data Domain tab.
2. Add a data domain with the following name:

   HDFSDS

   Use the name HDFSDS unless you explicitly set a different data domain name in Cloudera Manager. If the data domain names do not match, then the policy and audit functionality will not work.
3. **Always set the Kerberos option on.** Set the option on if you are using a Kerberized or non-Kerberized cluster.

   The following screenshot shows example data domain settings.



4. Click the Deployment tab. Deploy the data domain.

Verify Data Cannot be Read

Same as for HDFS. See Verify Data Cannot be Read on page 87.

Manage BlueTalon Services from Cloudera Manager

To manage BlueTalon services from Cloudera Manager:

1. In Cloudera Manager, click BlueTalon.
2. Click a BlueTalon service.
3. Click the Actions button and select an option for the selected service. Examples: Start, stop, restart.

Configure Enforcement Point Using Setup Script in Silent Mode

Ensure you have deployed your data domain before deploying an Enforcement Point.

(Optional) You can also configure an Enforcement Point in silent mode, which does not prompt you to enter details.

To configure the Enforcement Point with a preconfigured XML file, run:

```
/opt/bluetalon/3.2.4/pep/scripts/bluetalon-ep-setup -accept -f
/tmp/cdh.xml
```

For the parameter list, see Enforcement Point Silent Mode XML Parameters on page 65.

Example FSEP Silent Mode XML File for CDH Cluster

Example FSEP silent mode XML file for CDH cluster:

```
<EnforcementPoint xmlns="">
  <db_type>8</db_type>
  <resource_domain_name>HDFSDataDomain</resource_domain_name>
  <audit_ip>127.0.0.1</audit_ip>
  <pdp_hostname>127.0.0.1</pdp_hostname>
  <hadoop_conf_path>/etc/hadoop/conf/</hadoop_conf_path>
  <traffic_divert>yes</traffic_divert>
  <kerberos>no</kerberos>
  <post_install_conf>false</post_install_conf>
</EnforcementPoint>
```

For the parameter list, see Enforcement Point Silent Mode XML Parameters on page 65.

Allow Access to HDFS Data

(Optional) To allow access to HDFS data, see Allow Access to HDFS Data on page 91.

Hive

(Optional) You can set up Hive, see Hive on page 100.

Configuration for CDH is complete.

Configure FSEP for Isilon

Isilon is a network-attached storage platform offered by EMC Corporation for high-volume storage, backup, and archiving of unstructured data. BlueTalon provides security for Isilon. To configure FSEP for Isilon, perform the steps in this section.

Prerequisites

This section contains commands for Isilon.

- Isilon commands are examples and might need modification for your specific implementation.
- Commands should be run by an experienced Isilon administrator.
- Output from the example commands will vary for your Isilon implementation.

- X characters are used to replace IP addresses in the example commands and the output. You will see and use your own IP address.

Ensure Isilon Cluster is Running

```
# isi status -q
Cluster Name: btisilon
Cluster Health:     [  OK ]
Cluster Storage:  HDD                 SSD Storage
Size:             30.4G (54.4G Raw)   0 (0 Raw)
VHS Size:         23.9G
Used:             145.1M (< 1%)       0 (n/a)
Avail:            30.3G (> 99%)       0 (n/a)
                   Health  Throughput (bps)  HDD Storage      SSD
Storage
ID |IP Address     |DASR |  In   Out  Total| Used / Size     |Used /
Size
---+---------------+-----+-----+-----+-----+-----------------+---------
--------
  1|XX.XX.XX.XX    | OK  |65.6k| 715k| 780k|49.9M/18.1G(< 1%)|(No
Storage SSDs)
  2|XX.XX.XX.XX    | OK  |    0| 768k| 768k|49.6M/18.1G(< 1%)|(No
Storage SSDs)
  3|XX.XX.XX.XX    | OK  |    0|    0|    0|45.6M/18.1G(< 1%)|(No
Storage SSDs)
---+---------------+-----+-----+-----+-----+-----------------+---------
--------
Cluster Totals:          |65.6k| 1.5M| 1.5M| 145M/30.4G(< 1%)|(No
Storage SSDs)
     Health Fields: D = Down, A = Attention, S = Smartfailed, R = Read-
Only
```

Ensure HDFS is Licensed on Isilon Cluster

```
# isi license licenses list
Name                  Status     Expiration
-----------------------------------------------------
SmartDedupe           Inactive   -
Swift                 Inactive   -
SmartQuotas           Inactive   -
InsightIQ             Inactive   -
SmartPools            Inactive   -
SmartLock             Inactive   -
Isilon for vCenter    Inactive   -
```

```
CloudPools             Inactive   -
Hardening              Inactive   -
SnapshotIQ             Inactive   -
HDFS                   Evaluation 2016-07-10T20:51:25
SyncIQ                 Inactive   -
SmartConnect Advanced Inactive   -
-------------------------------------------------
Total: 13
```

An evaluation HDFS license is used in the example.

Ensure HDFS is Configured with WebHDFS

```
# isi hdfs setting
              Service: Yes
   Default Block Size: 128M
Default Checksum Type: none
  Authentication Mode: all
       Root Directory: /ifs
      WebHDFS Enabled: Yes
        Ambari Server: -
      Ambari Namenode: -
          Odp Version: -
```

Obtain Access Zone for HDFS Root File System

```
# isi zone zones list
Name   Path
------------
System /ifs
------------
Total: 1
```

The access zone is System.

Obtain Zone ID for Access Zone

```
# isi zone zones view System
                Name: System
                Path: /ifs
            Groupnet: groupnet0
       Map Untrusted: -
      Auth Providers: lsa-local-provider:System, lsa-file-
provider:System
```

```
        NetBIOS Name: -
  User Mapping Rules: -
Home Directory Umask: 0077
  Skeleton Directory: /usr/share/skel
  Cache Entry Expiry: 4H
             Zone ID: 1
```

The zone ID is 1 for the System access zone.

Create HDFS User

```
# isi auth users create --name="hdfs"
# isi auth users list
Name
-------------
Guest
hdfs
root
admin
compadmin
ftp
www
nobody
insightiq
remotesupport
-------------
Total: 11
```

Create Supergroup

```
# isi auth groups create --name="supergroup"
# isi auth groups list
Name
----------------
Administrators
Users
Guests
Backup Operators
Isilon Users
supergroup
wheel
admin
```

```
ftp
guest
ifs
nobody
---------------
Total: 12
```

Create Directories in HDFS Root Directory

```
# isi_run -z 1 mkdir tmp
# isi_run -z 1 chown hdfs:supergroup tmp
# isi_run -z 1 chmod 1777 tmp
# ls -la
total 108
drwxrwxrwx     6 root  wheel        110 Apr 11 21:04 .
drwxrwxrwx     6 root  wheel        110 Apr 11 21:04 ..
drwxrwxr-x    19 root  wheel        649 Apr 11 19:51 .ifsvar
dr-xr-xr-x     2 root  wheel          0 Apr 11 19:30 .snapshot
-rw-r--r--     1 root  wheel       1029 Apr 11 19:30 README.txt
drwxrwxrwx     2 root  wheel          0 Apr 11 19:30 data
drwxrwxr-x     6 root  wheel         94 Apr 11 21:07 home
drwxrwxrwt     2 hdfs  supergroup     0 Apr 11 21:04 tmp

# isi_run -z 1 mkdir user
# isi_run -z 1 chown hdfs:supergroup user
# isi_run -z 1 chmod 755 user
# ls -la /ifs/user
total 6
drwxr-xr-x    2 hdfs  supergroup    0 Apr 11 21:17 .
drwxrwxrwx    7 root  wheel       132 Apr 11 21:17 ..
```

Create Users, Groups, and Directories for YARN or MR2

```
# isi auth users create --name="mapred"
# isi auth groups create --name="mapred"
# isi_run -z 1 mkdir user/history
# isi auth groups create --name="hadoop"
# isi_run -z 1 chown mapred:hadoop user/history
# isi_run -z 1 chmod 777 user/history
# isi_run -z 1 mkdir tmp/logs
# isi_run -z 1 chown mapred:hadoop tmp/logs
```

```
# isi_run -z 1 chmod 775 tmp/logs
```

Create Users, Groups, and Directories for Hive

```
# isi auth users create --name="hive"
# isi auth groups create --name="hive"
# isi_run -z 1 mkdir user/hive
# isi_run -z 1 chown hive:hive user/hive
# isi_run -z 1 chmod 775 user/hive
# isi_run -z 1 mkdir user/hive/warehouse
# isi_run -z 1 chown hive:hive user/hive/warehouse
# isi_run -z 1 chmod 1777 user/hive/warehouse
# isi_run -z 1 mkdir tmp/hive
# isi_run -z 1 chown hive:supergroup tmp/hive
# isi_run -z 1 chmod 777 tmp/hive
# isi auth users create --name="hue"
# isi auth group create --name="hue"
# isi auth users create --name="alice"
# isi auth group create --name="sample"
# isi auth users create --name="sample"
# isi hdfs proxyusers create hdfs
# isi hdfs proxyusers create mapred
# isi hdfs proxyusers create hive
# isi hdfs proxyusers create oozie
# isi hdfs proxyusers create hue
```

Isilon File System API Examples

This section shows some example Isilon file system API calls. The first set of examples show what happens when a user is not authorized to perform the action.

The following example fails because the alice user is not authorized to access the system:

```
$ curl -i -k -u alice
"https://XXX.XXX.XXX.XXX:8080/namespace/ifs/?detail=default"
Enter host password for user 'alice':
HTTP/1.1 401 Authorization Required
Date: Mon, 11 Apr 2016 21:37:49 GMT
Server: Apache/2.2.31 (FreeBSD) mod_ssl/2.2.31 OpenSSL/1.0.1p-fips
mod_fastcgi/2.4.6
WWW-Authenticate: Basic
Last-Modified: Tue, 15 Dec 2015 10:32:14 GMT
ETag: "4e31d-31-526ed4d90e780"
Accept-Ranges: bytes
```

```
Content-Length: 49
Content-Type: application/json
{"errors":[{"message":"authorization required"}]}
```

The following example fails because the alice user is not authorized to access the ifs store:

```
$ curl -i -k -u alice
"https://XXX.XXX.XXX.XXX:8080/namespace/ifs/?detail=default"
Enter host password for user 'alice':
HTTP/1.1 403 Forbidden
Date: Mon, 11 Apr 2016 21:40:44 GMT
Server: Apache/2.2.31 (FreeBSD) mod_ssl/2.2.31 OpenSSL/1.0.1p-fips
mod_fastcgi/2.4.6
Allow: GET, PUT, POST, DELETE, HEAD
x-isi-ifs-spec-version: 1.0
Transfer-Encoding: chunked
Content-Type: application/json
{"errors" : [{"code" : "AEC_FORBIDDEN",
"message" : "Unable to open the store 'ifs' -- permission denied."}]}
```

The following example fails because the hdfs user is not authorized to access the ifs store:

```
$ curl -i -k -u hdfs:hdfs
"https://XXX.XXX.XXX.XXX:8080/namespace/ifs/tmp?detail=default"
HTTP/1.1 403 Forbidden
Date: Mon, 11 Apr 2016 22:25:02 GMT
Server: Apache/2.2.31 (FreeBSD) mod_ssl/2.2.31 OpenSSL/1.0.1p-fips
mod_fastcgi/2.4.6
Allow: GET, PUT, POST, DELETE, HEAD
x-isi-ifs-spec-version: 1.0
Transfer-Encoding: chunked
Content-Type: application/Jon
{"errors" : [{
"code" : "AEC_FORBIDDEN",
"message" : "Unable to open the store 'ifs' -- permission denied."
}]}
```

The following example succeeds because the root user has access to the ifs store:

```
$ curl -i -k -u root:root
"https://XXX.XXX.XXX.XXX:8080/namespace/ifs/tmp?detail=default"
HTTP/1.1 200 Ok
Date: Mon, 11 Apr 2016 22:22:21 GMT
Server: Apache/2.2.31 (FreeBSD) mod_ssl/2.2.31 OpenSSL/1.0.1p-fips
mod_fastcgi/2.4.6
Allow: GET, PUT, POST, DELETE, HEAD
Etag: "4297719830-18446744073709551615-5"
```

```
Last-Modified: Mon, 11 Apr 2016 21:27:08 GMT
x-isi-ifs-access-control: 1777
x-isi-ifs-spec-version: 1.0
x-isi-ifs-target-type: container
Transfer-Encoding: chunked
Content-Type: application/json
{"children":[{
   "group" : "hadoop",
   "last_modified" : "Mon, 11 Apr 2016 21:22:55 GMT",
   "mode" : "0775",
   "name" : "logs",
   "owner" : "mapred",
   "size" : 0,
   "stub" : false,
   "type" : "container"
},{
   "group" : "supergroup",
   "last_modified" : "Mon, 11 Apr 2016 21:27:08 GMT",
   "mode" : "0777",
   "name" : "hive",
   "owner" : "hive",
   "size" : 0,
   "stub" : false,
   "type" : "container"}
```

WebHDFS Examples

The following example retrieves the WebHDFS port:

```
# isi_gconfig -t web-config  webhdfs_port
webhdfs_port (int) = 8082
```

The following example retrieves the file status:

```
$ curl -i -u hdfs:hdfs
"http://XXX.XXX.XXX.XXX:8082/webhdfs/v1/?op=GETFILESTATUS&user.name=hdf
s"
HTTP/1.1 200 OK
Date: Mon, 11 Apr 2016 22:47:24 GMT
Server: Apache/2.2.31 (FreeBSD) mod_ssl/2.2.31 OpenSSL/1.0.1p-fips
mod_fastcgi/2.4.6
Content-Length: 350
Content-Type: application/json; charset=UTF-8
```

```
{
  "FileStatus" : {
      "accessTime" : 1460409425567,
      "blockSize" : 0,
      "childrenNum" : -1,
      "fileId" : 2,
      "group" : "wheel",
      "length" : 0,
      "modificationTime" : 1460409425567,
      "owner" : "root",
      "pathSuffix" : "",
      "permission" : "777",
      "replication" : 0,
      "type" : "DIRECTORY"
   }
}
```


The following example retrieves the file status:

```
$ curl -i
"http://XXX.XXX.XXX.XXX:8082/webhdfs/v1/user/hive?op=GETFILESTATUS&user
.name=hdfs"
```


```
HTTP/1.1 200 OK
Date: Mon, 11 Apr 2016 22:49:15 GMT
Server: Apache/2.2.31 (FreeBSD) mod_ssl/2.2.31 OpenSSL/1.0.1p-fips
mod_fastcgi/2.4.6
Content-Length: 358
Content-Type: application/json; charset=UTF-8
{
   "FileStatus" : {
      "accessTime" : 1460409985801,
      "blockSize" : 0,
      "childrenNum" : -1,
      "fileId" : 4297719842,
      "group" : "hive",
      "length" : 0,
      "modificationTime" : 1460409985801,
      "owner" : "hive",
      "pathSuffix" : "",
      "permission" : "775",
```

```
      "replication" : 0,
      "type" : "DIRECTORY"
   }
}
```

The following example retrieves the file status:

```
$ curl -i
"http://XXX.XXX.XXX.XXX:8082/webhdfs/v1/user/hive?op=GETFILESTATUS&user
.name=alice"
HTTP/1.1 200 OK
Date: Mon, 11 Apr 2016 22:49:22 GMT
Server: Apache/2.2.31 (FreeBSD) mod_ssl/2.2.31 OpenSSL/1.0.1p-fips
mod_fastcgi/2.4.6
Content-Length: 358
Content-Type: application/json; charset=UTF-8
{
   "FileStatus" : {
      "accessTime" : 1460409985801,
      "blockSize" : 0,
      "childrenNum" : -1,
      "fileId" : 4297719842,
      "group" : "hive",
      "length" : 0,
      "modificationTime" : 1460409985801,
      "owner" : "hive",
      "pathSuffix" : "",
      "permission" : "775",
      "replication" : 0,
      "type" : "DIRECTORY"
   }
}
```

HDFS Examples

The following example uses the alice user:

```
# su -l alice -c 'hdfs dfs -ls hdfs://XXX.XXX.XXX.XXX:8020/'
Warning: fs.defaultFs is not set when running "ls" command.
Found 6 items
drwxrwxr-x   - root wheel               0 2016-04-11 19:51
hdfs://XXX.XXX.XXX.XXX:8020/.ifsvar
-rw-r--r--   3 root wheel            1029 2016-04-11 19:30
hdfs://XXX.XXX.XXX.XXX:8020/README.txt
```

```
drwxrwxrwx   - root wheel               0 2016-04-11 19:30
hdfs://XXX.XXX.XXX.XXX:8020/data

drwxrwxr-x   - root wheel               0 2016-04-11 21:30
hdfs://XXX.XXX.XXX.XXX:8020/home

drwxrwxrwt   - hdfs supergroup          0 2016-04-11 21:27
hdfs://XXX.XXX.XXX.XXX:8020/tmp

drwxr-xr-x   - hdfs supergroup          0 2016-04-11 21:25
hdfs://XXX.XXX.XXX.XXX:8020/user
```

The following example performs an ls command using the hdfs user:

```
# su -l hdfs  -c 'hdfs dfs -ls hdfs://XXX.XXX.XXX.XXX:8020/'

Warning: fs.defaultFs is not set when running "ls" command.

Found 6 items

drwxrwxr-x   - root wheel               0 2016-04-11 19:51
hdfs://XXX.XXX.XXX.XXX:8020/.ifsvar

-rw-r--r--   3 root wheel            1029 2016-04-11 19:30
hdfs://XXX.XXX.XXX.XXX:8020/README.txt

drwxrwxrwx   - root wheel               0 2016-04-11 19:30
hdfs://XXX.XXX.XXX.XXX:8020/data

drwxrwxr-x   - root wheel               0 2016-04-11 21:30
hdfs://XXX.XXX.XXX.XXX:8020/home

drwxrwxrwt   - hdfs supergroup          0 2016-04-11 21:27
hdfs://XXX.XXX.XXX.XXX:8020/tmp

drwxr-xr-x   - hdfs supergroup          0 2016-04-11 21:25
hdfs://XXX.XXX.XXX.XXX:8020/user
```

The Isilon commands are complete and you can begin the FSEP steps in the following sections.

Collect Information

Same as for HDFS. See Collect Information on page 82.

Create Test Data

Same as for HDFS. See Create Test Data on page 82.

Add Data Domain

Same as for HDFS. See Add Data Domain on page 83.

Install Enforcement Point Package

Install the Enforcement Point package:

```
yum install bluetalon-ep-3.2.4
```

Configure Enforcement Point

Configure the Enforcement Point:

1. Run:
   `/opt/bluetalon/3.2.4/pep/scripts/bluetalon-ep-setup`

2. Enter the information.

Item	Description
Enforcement Point type	Type of Enforcement Point. Select FSEP.
Kafka Broker hostname	Hostname where the bt-audit-kafka service is running. Example: Kafka-Broker-host.
Policy Engine hostname	Computer hostname or IP address on which the BlueTalon Policy Engine is running.
Data domain name	Name of the data domain. Use the value set in the Policy Console for the data domain you created earlier.

Restart FSEP Service

Restart FSEP service:

```
service bt-fsep restart
```

Verify Data Cannot be Read

Same as for HDFS. See Verify Data Cannot be Read on page 87.

Specify Policy Engine Tag with FSEP

(Optional) When using FSEP, you can specify the Policy Engine to use in the file bt-hdfs-site.xml using a tag.

The Enforcement Point uses the tag when retrieving the Policy Engine list. Only the Policy Engine that matches the tag is returned and used by the Enforcement Point.

You set the tag in the following file:

```
/etc/bluetalon/pep/hdfs-ep/conf/bt-hdfs-site.xml
```

Example file extract showing the tag:

```
<property>
  <name>bt.hdfs.plugin.pdp.tag</name>
  <value>TAG-NAME</value>
  <description>BlueTalon PDP HA tag</description>
  <type>string</type>
</property>
```

If you change the tag, save the file and then restart the Enforcement Point service:

```
service bt-fsep restart
```

FSEP configuration is complete.

Hive Enforcement Point

Apache Hive is a data warehouse for Hadoop. Hive provides data summarization, and a SQL style interface to query data stored in databases and file systems that integrate with Hadoop. BlueTalon provides security for Hive.

This section describes how to set up an Enforcement Point for Hive.

Collect Information

The following table shows the information you must collect before installing the Enforcement Point.

Item	Description
Database	Database name. Example: TestDatabase
Host address	Computer hostname or IP address running the database. Example: TestServer.TestCompany.com
Port	Port the database receives JDBC connections. Example: 10000 for binary mode and 10001 for http mode.
Link to HDFS data domain	Whether to link to the HDFS data domain. Options: • On • Off When set to on, the SELECT or INSERT rules set for the tables in the data domain are automatically applied as READ or WRITE rules on the HDFS data directories that store the Hive data.
Database credentials	Authentication type. Options: • Login • No login • Kerberos
User ID	User ID for a database account with privileges to obtain metadata. Only required if the credentials option is set to Login or Kerberos. For Kerberos, the user ID must be a user principal for a Kerberos realm that can access HiveServer2.
Password	Password for the database account.
Kerberos service principle	Kerberos service principle. Only required if the Kerberos database credential option is selected. Example: hive/TestServer.ec2.internal@EXAMPLE.COM
HTTP mode	Whether to use HTTP mode. Options: • On • Off
Connection parameter	Parameter for HTTP mode connection. Example: hive.server2.transport.mode=http;hive.server2.thrift.http.path=bluetalon

Create Test Data

Create a CSV file named accounts.csv in the /tmp directory with the following content:

```
147274739-9,Frances Harrison,9-(192)357-8851,10/27/56,993-94-
1527,37726,6393451850134970,52.90

457389751-8,Gregory Patterson,3-(684)454-2444,11/22/69,233-57-
5483,21278,5602245983499780,44.87

360272064-0,Earl James,5-(177)394-3277,5/23/83,303-64-
0766,26595,3534227478434860,3.36

551058833-0,Christine Robertson,5-(437)964-8463,5/11/88,318-79-
4141,50716,5602223026663730,5.22

713553396-8,Daniel Crawford,1-(875)657-9518,6/1/99,214-89-
4670,28693,3571338359613280,22.63

359936857-0,Shirley Holmes,6-(362)871-3036,4/1/80,964-49-
0690,13189,3534219304003200,30.17

887693755-2,Kelly Adams,9-(887)292-8810,2/12/88,304-47-
5814,84901,5602248578416630,35.09

091619799-9,Bonnie Freeman,2-(163)102-1214,1/26/71,940-51-
8723,73248,3547967170434520,69.74

151872601-1,Joyce Mcdonald,3-(504)572-0648,9/18/64,919-59-
1100,82573,5401856433043540,88.15

956961211-8,Karen Burton,1-(533)256-2922,3/5/54,479-48-
8766,56134,3536534268148670,7.08
```

Create a database table named accounts:

```
hive

CREATE TABLE accounts (id STRING, name STRING, phone STRING, birthdate
STRING, soc_sec_no STRING, zip BIGINT, credit_card BIGINT, balance
DECIMAL(4,2)) ROW FORMAT DELIMITED FIELDS TERMINATED BY ',' LINES
TERMINATED BY '\n' STORED AS TEXTFILE;
```

Load the accounts.csv file data into the accounts table. Example:

```
LOAD DATA INPATH '/tmp/accounts.csv' INTO TABLE accounts;
```

Examine the rows in the accounts table:

```
SELECT * FROM accounts;
```

Add Data Domain

A data domain stores information about an external data source.

You import the external data source information into a data domain. For relational database sources, you can import table and column information.

Add a data domain. Use Hive as your data domain type. See Add Data Domain on page 55.

Install Enforcement Point Package

Install the Enforcement Point package:

1. Run:
   ```
   yum install bluetalon-ep-3.2.4
   ```

2. Follow the prompts.

Configure Enforcement Point

Configure the Enforcement Point:

1. Run:
   `/opt/bluetalon/3.2.4/pep/scripts/bluetalon-ep-setup`
2. Enter the information.

Item	Description
Enforcement Point type	Type of Enforcement Point. Select Hive.
Enforcement Point port	JDBC end point port that client applications use to obtain secure data. Typically use 10010.
Hostname	Database computer hostname or IP address. Use the value set in the Policy Console for the data domain you created earlier.
Port	Database port. Use the value set in the Policy Console for the data domain you created earlier.
Enforcement Point mode	Options: • Forward only • Audit only • Enforce only • Audit and Enforce
Security DB TCP hostname	Computer hostname or IP address on which the BlueTalon Policy Engine is running.
Security DB TCP port	Port for the BlueTalon Policy Engine. Typically use the default 1600.
Data domain name	Name of the data domain. Use the value set in the Policy Console for the data domain you created earlier.

Enforcement Point Files and Service

The following table shows the Enforcement Point files and service.

Item	Detail
Binary files path	/opt/bluetalon/3.2.4/pep/hive-ep/bin
Configuration file	/etc/bluetalon/pep/hive-ep/conf/bt-hive-ep-<data domain name>.conf
Logs	/var/log/bluetalon/pep/hive-ep/logs/bt-hive-ep-<data domain name>.log
Service name	bt–hive-ep-<data domain name>

Test Enforcement Point

See Test Enforcement Point on page 70. Perform similar steps for Hive (the Hive Enforcement Point service name is shown in the previous table).

Verify Masked Data

See Verify Masked Data on page 78. Perform similar steps for the Hive database accounts table.

Run the commands in Hadoop Functional Tests on page 162 at the end of this section.

Provision Users for Running Hive Command Line Jobs

On a newly provisioned cluster with FSEP in the path of all traffic, and only for HDFS, YARN, and Hive services, you must perform the following steps to enable a user to run Hive command line interface jobs:

1. Add the user to BlueTalon in the InternalSource user domain or discover the user from the OpenLDAP or Windows AD user domain using the Policy Console or the REST API. If you do not do this, the user is denied access to all HDFS data.
2. Add the user to the policy set for Hive. If you do not do this, the user will be able to run map reduce job and show tables, but will get the error messages shown in the following sections.

In the following sections, assume example HDFS data is to be loaded at /apps/hive/warehouse/accounts by the root user.

Load Data

If a user attempts to perform an HDFS mkdir command without the previous steps having already been performed, the user receives an error. Example:

```
FAILED: Execution Error, return code 1 from
org.apache.hadoop.hive.ql.exec.DDLTask. MetaException(message:Got
exception: org.apache.hadoop.security.AccessControlException Permission
denied: user=root, access=MKDIRS, inode=/apps/hive/warehouse/accounts)
```

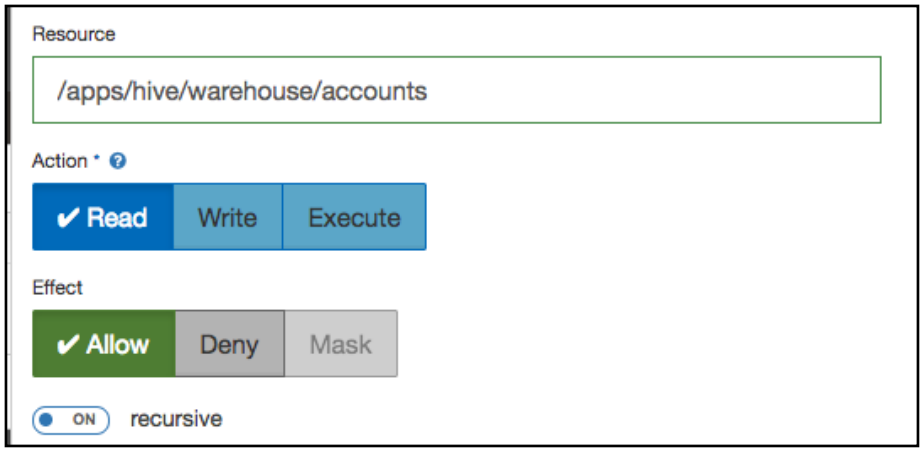
When the user has the correct permissions, new accounts data can be stored in HDFS.

Read Data

If a user attempts to read data from the example HDFS accounts location without read permission, the user receives an error. Example:

```
Failed with exception
java.io.IOException:org.apache.hadoop.security.AccessControlException:
Permission denied: user=root, access=LISTSTATUS,
inode=/apps/hive/warehouse/accounts
```

Example read permission:



When the user has the read permission, they can read the accounts data from HDFS. Example:

```
accounts.id accounts.name     accounts.phone    accounts.birthdate
      accounts.soc_sec_no     accounts.zip      accounts.credit_card
      accounts.balance

147274739-9 Frances Harrison  9-(192)357-8851   10/27/56
      993-941527              37726             6393451850134970
      52.9

457389751-8 Gregory Patterson 3-(684)454-2444   11/22/69
      233-575483              21278             5602245983499780
      44.87

360272064-0 Earl James        5-(177)394-3277   5/23/83
      303-640766              26595             3534227478434860
      3.36

      … other data omitted for brevity …
```

Configure Enforcement Point Using Setup Script in Silent Mode

Ensure you have deployed your data domain before deploying an Enforcement Point.

(Optional) You can also configure an Enforcement Point in silent mode, which does not prompt you to enter details.

To configure the Enforcement Point with a preconfigured XML file, run:

```
/opt/bluetalon/3.2.4/pep/scripts/bluetalon-ep-setup -accept -f
/tmp/hive.xml
```

For the parameter list, see Enforcement Point Silent Mode XML Parameters on page 65.

Example Hive Silent Mode XML Files

This section shows example Hive silent mode XML files.

Example Hive Silent Mode XML File with Simple Authentication

Example Hive silent mode XML file with simple authentication:

```
<EnforcementPoint xmlns="">
  <hostname>ec2-184-73-22-224.compute-1.amazonaws.com</hostname>
  <port>10000</port>
```

```
  <db_type>2</db_type>
  <mode>4</mode>
  <resource_domain_name>hivetest</resource_domain_name>
  <custom_name>hivetest1</custom_name>
  <audit_ip>sandbox.bluetalon.com</audit_ip>
  <ep_port>2004</ep_port>
  <dbtcp_ip>sandbox.bluetalon.com</dbtcp_ip>
  <kerberos>no</kerberos>
</EnforcementPoint>
```

Example Hive Silent Mode XML File with Kerberos Authentication

Example Hive silent mode XML file with Kerberos authentication:

```
<EnforcementPoint xmlns="">
  <hostname>sandbox.bluetalon.com</hostname>
  <port>10000</port>
  <db_type>2</db_type>
  <resource_domain_name>hivekrb</resource_domain_name>
  <custom_name>hivekrb</custom_name>
  <audit_ip>172.30.0.229</audit_ip>
  <ep_port>2011</ep_port>
  <dbtcp_ip>sandbox.bluetalon.com</dbtcp_ip>
  <pwd_auth>no</pwd_auth>
  <proxy_auth>no</proxy_auth>
  <endpoint_tag>DEFAULT</endpoint_tag>
  <kerberos>yes</kerberos>
  <bt_keytab_path>/home/ec2-user/bluetalon.keytab</bt_keytab_path>
</EnforcementPoint>
```

Hive configuration is complete.

Impala Enforcement Point

Impala is an open-source parallel processing query engine for clustered systems like Apache Hadoop. Impala provides a SQL style query engine for HDFS. BlueTalon provides an Enforcement Point for Impala.

This section describes how to set up an Enforcement Point for Impala.

Download and Install Impala JDBC Driver

The following subsections describe how to download and install the Impala JDBC driver.

Files

You can use either of these Impala JDBC driver JAR files:

- ImpalaJDBC3.jar
- ImpalaJDBC4.jar

Download Impala JDBC Driver

For Impala version 2.2.0 on CDH 5.4.9, download ImpalaJDBC3.jar for Cloudera Enterprise from http://www.cloudera.com/downloads/connectors/impala/jdbc/2-5-31.html

For others, download the JAR file from http://www.simba.com/connectors/apache-impala-driver

Install Impala JDBC Driver

To install the Impala JDBC driver:

1. Copy the JAR file to these locations (overwrite the existing JAR file if necessary):
   - /opt/bluetalon/3.2.4/policy/pap/webapps/PolicyManagement/WEB-INF/lib
   - /opt/bluetalon/3.2.4/policy/pap/webapps/BlueTalonConfig/WEB-INF/lib
2. In a command line shell, run these commands as a privilieged user:
   ```
   service bt-policy-server restart
   service bt-webserver restart
   ```

Installation of the JAR file is complete.

Collect Information

The following table shows the information you must collect before installing the Enforcement Point.

Item	Description
Database	Database name. Examples: • TestDatabase • default
Host address	Computer hostname or IP address running the database. Examples: • TestServer.TestCompany.com • 127.0.0.1
Port	Port the database receives JDBC connections. Example: 21050.
Link to HDFS data domain	Whether to link to the HDFS data domain. Options: • On • Off When set to on, the SELECT or INSERT rules set for the tables in the data domain are automatically applied as READ or WRITE rules on the HDFS data directories that store the data.

Database credentials	Authentication type. Options: • Login • No login • Kerberos
User ID	User ID for a database account with privileges to obtain metadata. Only required if the credentials option is set to Login or Kerberos. For Kerberos, the user ID must be a user principal for a Kerberos realm.
Password	Password for the database account.

Create Test Data

On the computer running Impala, connect to the database:

```
impala-shell -i localhost --quiet
```

Create a database table named accounts and add example data:

```
CREATE TABLE accounts (id STRING, name STRING, phone STRING, birthdate
STRING, soc_sec_no STRING, zip INT, credit_card BIGINT, balance
DECIMAL(4,2));

INSERT INTO accounts (id, name, phone, birthdate, soc_sec_no, zip,
credit_card, balance) VALUES ('147274739-9', 'Frances Harrison', '9-
(192)357-8851', '10/27/1956', '993941527', 37726, 6393451850134970,
52.90);

INSERT INTO accounts (id, name, phone, birthdate, soc_sec_no,
zip,credit_card, balance) VALUES ('457389751-8', 'Gregory Patterson',
'3-(684)454-2444', '11/22/1969', '233575483', 21278, 5602245983499780,
44.87);

INSERT INTO accounts (id, name, phone, birthdate, soc_sec_no, zip,
credit_card, balance) VALUES ('360272064-0', 'Earl James', '5-(177)394-
3277', '05/23/1983', '303640766', 26595, 3534227478434860, 3.36);

INSERT INTO accounts (id, name, phone, birthdate, soc_sec_no, zip,
credit_card, balance) VALUES ('551058833-0', 'Christine Robertson', '5-
(437)964-8463', '05/11/1988', '318794141', 50716, 5602223026663730,
5.22);

INSERT INTO accounts (id, name, phone, birthdate, soc_sec_no, zip,
credit_card, balance) VALUES ('713553396-8', 'Daniel Crawford', '1-
(875)657-9518', '06/01/1999', '214894670', 28693, 3571338359613280,
22.63);

INSERT INTO accounts (id, name, phone, birthdate, soc_sec_no, zip,
credit_card, balance) VALUES ('359936857-0', 'Shirley Holmes', '6-
(362)871-3036', '04/01/1980', '964490690', 13189, 3534219304003200,
30.17);

INSERT INTO accounts (id, name, phone, birthdate, soc_sec_no, zip,
credit_card, balance) VALUES ('887693755-2', 'Kelly Adams', '9-
(887)292-8810', '02/12/1988', '304475814', 84901, 5602248578416630,
35.09);
```

```
INSERT INTO accounts (id, name, phone, birthdate, soc_sec_no, zip,
credit_card, balance) VALUES ('091619799-9', 'Bonnie Freeman', '2-
(163)102-1214', '01/26/1971', '940518723', 73248, 3547967170434520,
69.74);

INSERT INTO accounts (id, name, phone, birthdate, soc_sec_no, zip,
credit_card, balance) VALUES ('151872601-1', 'Joyce Mcdonald', '3-
(504)572-0648', '09/18/1964', '919591100', 82573, 5401856433043540,
88.15);

INSERT INTO accounts (id, name, phone, birthdate, soc_sec_no, zip,
credit_card, balance) VALUES ('956961211-8', 'Karen Burton', '1-
(533)256-2922', '03/05/1954', '479488766', 56134, 3536534268148670,
7.08);
```

Examine the rows in the accounts table:

```
SELECT * FROM accounts;
```

You will create a rule to mask the social security number stored in the soc_sec_no column later.

Add Data Domain

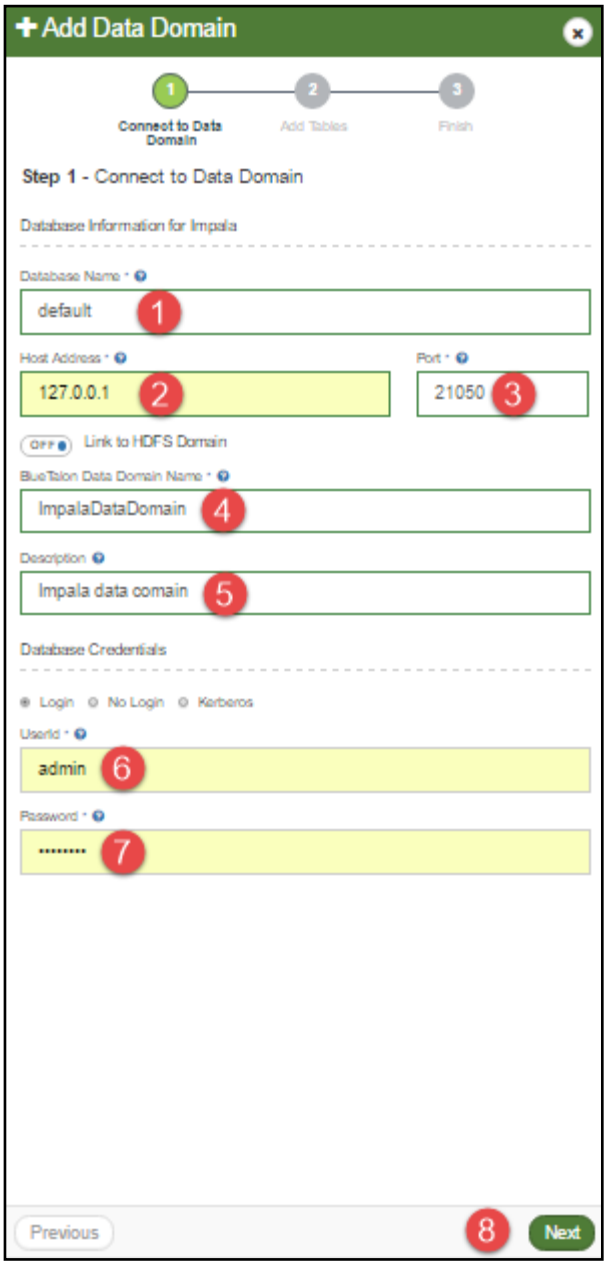
You import the external data source information into a data domain: For relational database sources, you can import table and column information.

Add a data domain:

1. See Add Data Domain on page 55. That section contains similar steps that you will use here.
2. Use Impala as your data domain type.
3. Enter the data domain details. Use the details you obtained earlier in the Collect Information section. See the following screenshot for an example.

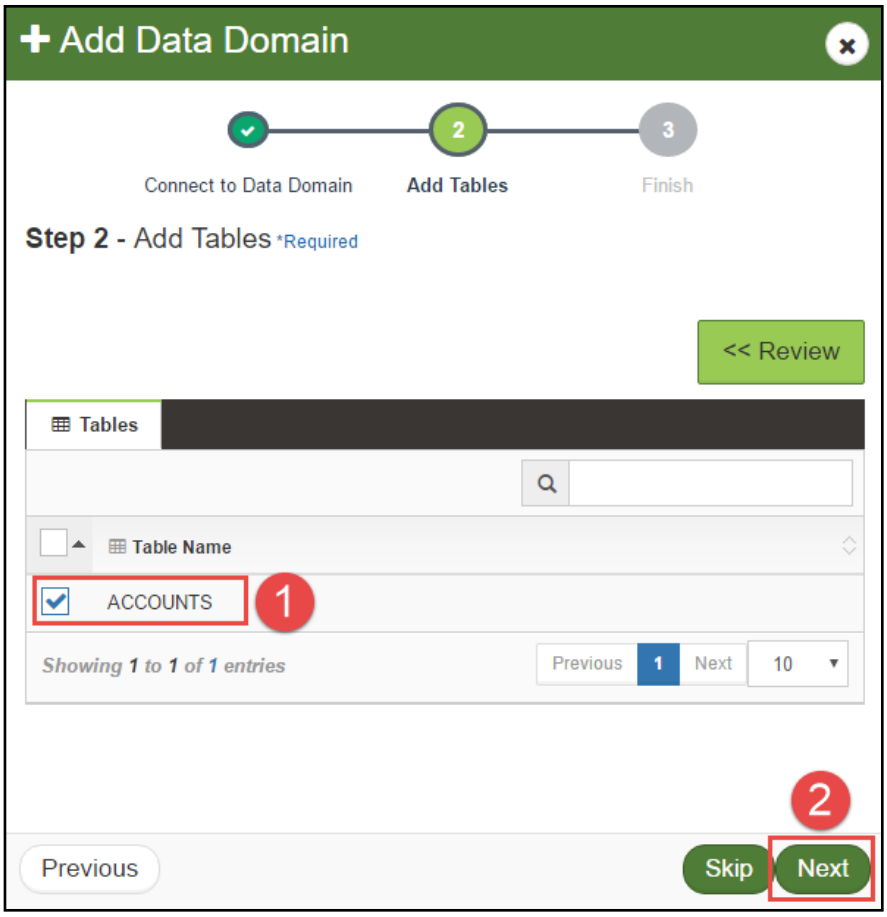
   Example data domain name: ImpalaDataDomain.

   Note the name of your data domain. You will need it later when you configure the Enforcement Point.


Add Data Domain
1
Connect to Data Domain
2
Add Tables
3
Finish
Step 1 - Connect to Data Domain
Database Information for Impala
Database Name *
default
1
Host Address *
127.0.0.1
2
Port *
21050
3
OFF
Link to HDFS Domain
BlueTalon Data Domain Name *
ImpalaDataDomain
4
Description
Impala data comain
5
Database Credentials
Login
No Login
Kerberos
Userid *
admin
6
Password *
.....
7
Previous
8
Next

4. Click Next.

5. Select the accounts table you created earlier. See item 1 in the following screenshot.



6. Click Next. See item 2 in the previous screenshot.
7. Deploy.

Install Enforcement Point Package

Install the Enforcement Point package:

1. Enter:
   `yum install bluetalon-ep-3.2.4`
2. Follow the prompts.

Configure Enforcement Point

Configure the Enforcement Point:

1. Run:
   `/opt/bluetalon/3.2.4/pep/scripts/bluetalon-ep-setup`
2. Accept the license agreement.
3. Enter the information.

Item	Description
Enforcement Point type	Type of Enforcement Point.

	Select Impala.
Enforcement Point port	JDBC end point port that client applications use to obtain secure data. Example: 1557
Enforcement Point port for Impala shell	Enforcement Point port for Impala shell. Example: 1558
Hostname	Database computer hostname or IP address. Use the value set in the Policy Console for the data domain you created earlier.
Port	Hostname port. Example: 21050
Impala Port	Impala port. Example: 21000
Enforcement Point mode	Options: • Forward only • Audit only • Enforce only • Audit and Enforce
BlueTalon Policy package hostname	Computer hostname or IP address on which the BlueTalon Policy Engine is running. Example: 10.0.0.70
bt-policy-configuration service port	Port for the BlueTalon Policy Engine. Example: 1600
Data domain name	Name of the data domain. Use the value set in the Policy Console for the data domain you created earlier. Example: ImpalaDataDomain
Kafka Broker hostname	Hostname where the bt-audit-kafka service is running. Example: Kafka-Broker-host.TestServer.TestCompany

Enforcement Point Files and Service

The following table shows the Enforcement Point files and service.

Item	Detail
Binary files path	/opt/bluetalon/3.2.4/pep/hive-ep/bin
Configuration file	/etc/bluetalon/pep/hive-ep/conf/bt-impala-ep-<data domain name>.conf
Logs	/var/log/bluetalon/pep/hive-ep/logs/bt-impala-ep-<data domain name>.log
Service name	bt–impala-ep-<data domain name>

Test Enforcement Point

Initially:

1. Add a user named alice. See Add User to Internal User Domain on page 48.
2. Add a policy named TestPolicy. See Add Policy on page 70.
3. Add the alice user to TestPolicy. See Add User to Policy on page 73.

Add Rule to Policy

A rule controls data access to a resource and controls the granularity of the access control policy. For a relational database, a resource can be a table or a column.

Add a rule to the test policy:

1. See Add Rule to Policy on page 75.
2. Set the data domain to ImpalaDataDomain.
3. Set the resource to the soc_sec_no column of the accounts table.

   The resource has the format Database.Schema.Table.Column.

   Example:
   default.default.accounts.soc_sec_no

   The database name and schema are assumed to be "default" and you use "default" twice in the resource field. If your database name or schema are different, you use your own settings.
4. Set the action to Read.
5. Set the effect to Mask.
6. Set the mask to Mask_All_ExceptLast4 to hide everything except the last four digits of the social security number.
7. Select Add to Policy Set.
8. Add the rule to TestPolicy.
9. Click Add Rule.
10. Deploy.

Verify Masked Data

Verify masked data:

1. Start Beeline:
   ```
   beeline
   ```
2. Connect to the Enforcement Point computer and port.
   Example format:
   ```
   !connect jdbc:hive2://<Enforcement Point computer IP
   address>:<Enforcement Point port>/default;auth=noSasl
   ```

   Example with IP address and port:
   ```
   !connect jdbc:hive2://10.0.0.70:1557/default;auth=noSasl
   ```
3. Enter the user name and password. Example: User alice with password mypassword.

   If you are using LDAP or Kerberos with Impala, then an authenticated user name and password must be provided.

4. Run the following query and ensure the soc_sec_no column is masked:
   ```
   SELECT soc_sec_no FROM accounts;
   ```

Run the commands in Hadoop Functional Tests on page 162.

Configure Enforcement Point Using Setup Script in Silent Mode

Ensure you have deployed your data domain before deploying an Enforcement Point.

(Optional) You can also configure an Enforcement Point in silent mode, which does not prompt you to enter details.

To configure the Enforcement Point with a preconfigured XML file, run:

```
/opt/bluetalon/3.2.4/pep/scripts/bluetalon-ep-setup -accept -f
/tmp/impala.xml
```

For the parameter list, see Enforcement Point Silent Mode XML Parameters on page 65.

Example Impala Silent Mode XML Files

This section shows example Impala silent mode XML files.

Example Impala Silent Mode XML File with Simple Authentication

Example Impala silent mode XML file with simple authentication:

```
<EnforcementPoint xmlns="">
  <hostname>172.30.1.6</hostname>
  <port>21050</port>
  <impala_shell_ep_port>2005</impala_shell_ep_port>
  <db_type>3</db_type>
  <resource_domain_name>impa</resource_domain_name>
  <custom_name>impas</custom_name>
  <audit_ip>172.30.0.229</audit_ip>
  <ep_port>2004</ep_port>
  <dbtcp_ip>172.30.0.228</dbtcp_ip>
  <pwd_auth>yes</pwd_auth>
  <proxy_auth>yes</proxy_auth>
  <endpoint_tag>DEFAULT</endpoint_tag>
</EnforcementPoint>
```

Example Impala Silent Mode XML File with Kerberos Authentication

Example Impala silent mode XML file with Kerberos authentication:

```
<EnforcementPoint xmlns="">
  <hostname>sandbox.bluetalon.com</hostname>
  <port>21050</port>
  <impala_shell_ep_port>2013</impala_shell_ep_port>
  <db_type>3</db_type>
```

```
  <resource_domain_name>impakrb</resource_domain_name>
  <custom_name>impakrb</custom_name>
  <audit_ip>172.30.0.229</audit_ip>
  <ep_port>2012</ep_port>
  <dbtcp_ip>sandbox.bluetalon.com</dbtcp_ip>
  <pwd_auth>no</pwd_auth>
  <proxy_auth>no</proxy_auth>
  <endpoint_tag>DEFAULT</endpoint_tag>
  <kerberos>yes</kerberos>
  <bt_keytab_path>/home/ec2-user/bluetalon.keytab</bt_keytab_path>
</EnforcementPoint>
```

Silent mode configuration is complete.

Configure Impala for Isilon

This section describes how to configure Impala for Isilon.

- Impala must be configured to use HDFS, not FSEP.
- Hive must also use HDFS because Hive and Impala share the same schema definition in the Hive metastore.
- The schema definition includes the absolute data path.
- HDFS in AIM is backed by Isilon and only the service users are allowed to connect directly to the Isilon file system.
- The end user running the Hive CLI cannot run Hive queries.
- To resolve, the FSEP client driver is enhanced to route the request to FSEP for HDFS URIs.
- Configuration is required to load the FSEP driver for HDFS URIs.
- The Hive metastore path must include the full HDFS URI to ensure the table or database created through the Hive CLI uses HDFS as the storage location, and not WebHDFS.

Perform the configuration described in the following sections:

- Use the Cloudera Manager Web user interface.
- View the properties as XML and paste the XML shown in the following sections.

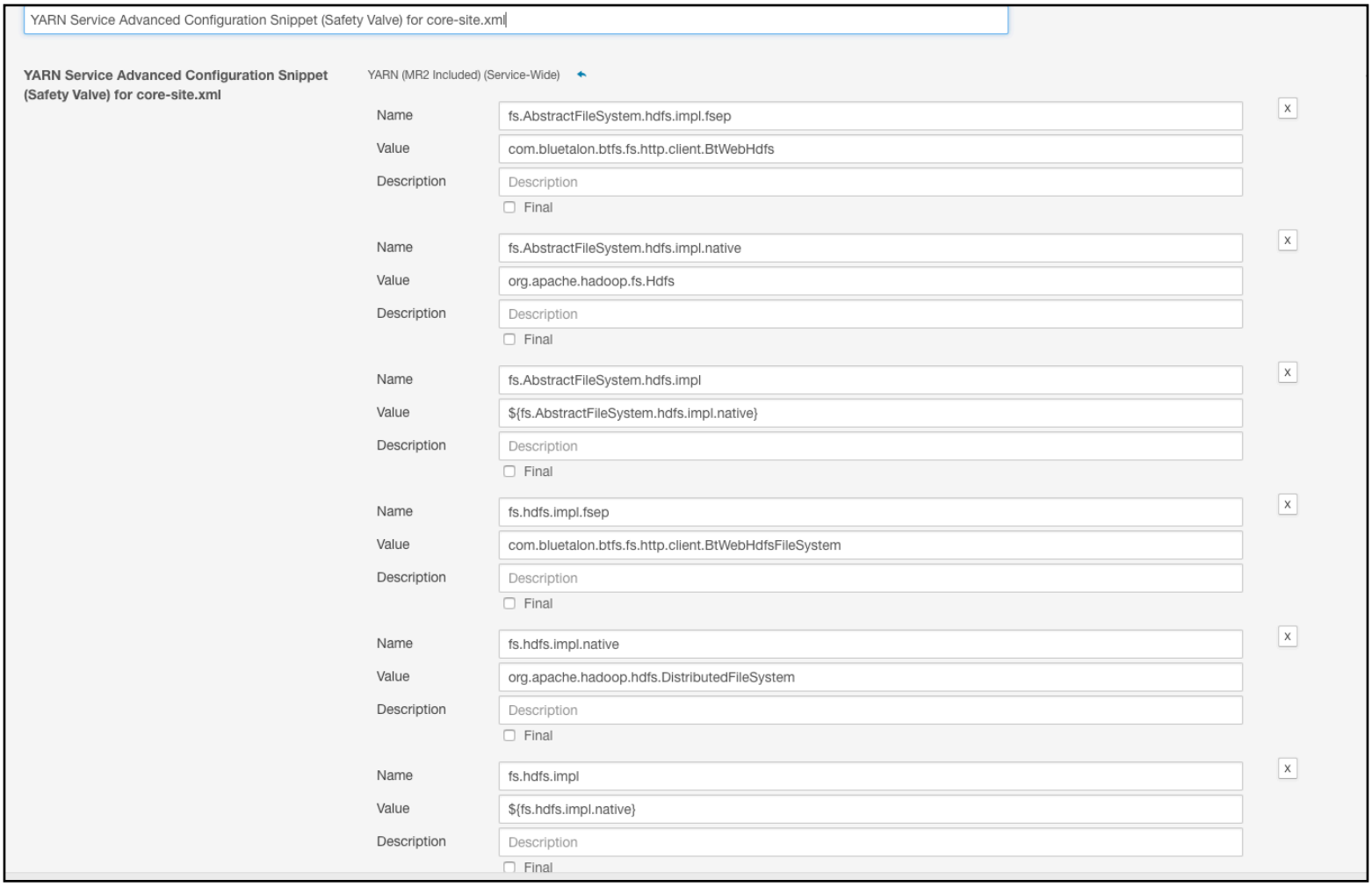
Configure YARN

To configure YARN, add following properties in "YARN Service Advanced Configuration Snippet (Safety Valve) for core-site.xml":

```
<property>
  <name>fs.AbstractFileSystem.hdfs.impl.fsep</name>
  <value>com.bluetalon.btfs.fs.http.client.BtWebHdfs</value>
</property>
<property>
  <name>fs.AbstractFileSystem.hdfs.impl.native</name>
```

```
  <value>org.apache.hadoop.fs.Hdfs</value>
</property>
<property>
  <name>fs.AbstractFileSystem.hdfs.impl</name>
  <value>${fs.AbstractFileSystem.hdfs.impl.native}</value>
</property>
<property>
  <name>fs.hdfs.impl.fsep</name>
  <value>com.bluetalon.btfs.fs.http.client.BtWebHdfsFileSystem</value>
</property>
<property>
  <name>fs.hdfs.impl.native</name>
  <value>org.apache.hadoop.hdfs.DistributedFileSystem</value>
</property>
<property>
  <name>fs.hdfs.impl</name>
  <value>${fs.hdfs.impl.native}</value>
</property>
```

The following screenshot shows the settings.

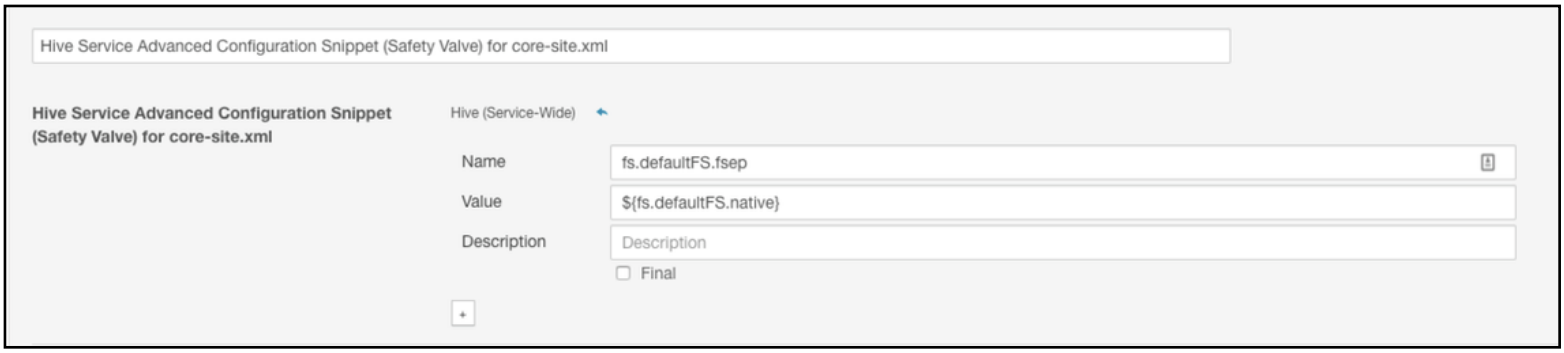


Configure Hive

To configure Hive, add following property in "Hive Service Advanced Configuration Snippet (Safety Valve) for core-site.xml":

```
<property>
  <name>fs.defaultFS.fsep</name>
  <value>${fs.defaultFS.native}</value>
</property>
```

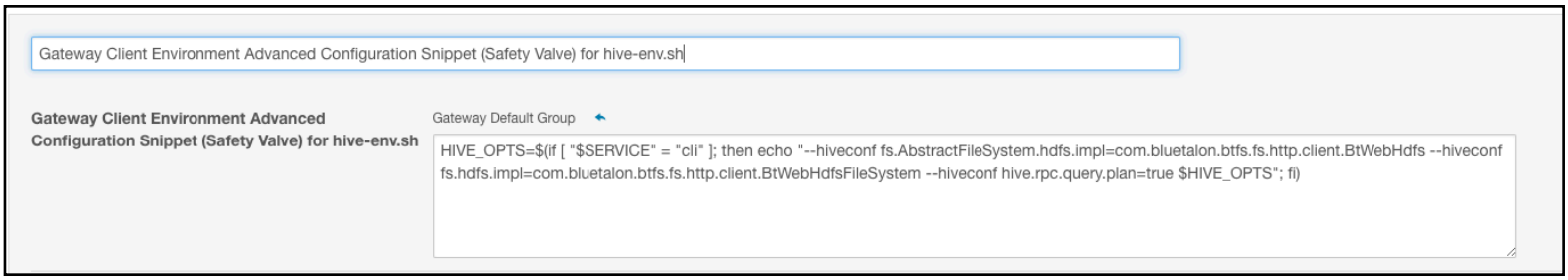
The following screenshot shows the settings.



Set "Gateway Client Environment Advanced Configuration Snippet (Safety Valve) for hive-env.sh" to:

```
HIVE_OPTS=$(if [ "$SERVICE" = "cli" ]; then echo "--hiveconf
fs.AbstractFileSystem.hdfs.impl=com.bluetalon.btfs.fs.http.client.BtWeb
Hdfs --hiveconf
fs.hdfs.impl=com.bluetalon.btfs.fs.http.client.BtWebHdfsFileSystem --
hiveconf hive.rpc.query.plan=true $HIVE_OPTS"; fi)
```

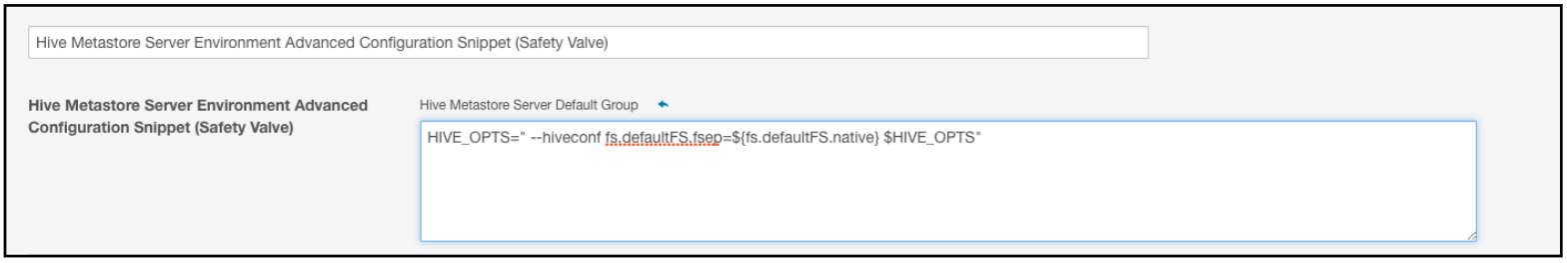
The following screenshot shows the settings.



Set "Hive Metastore Server Environment Advanced Configuration Snippet (Safety Valve)" to:

```
HIVE_OPTS=" --hiveconf fs.defaultFS.fsep=${fs.defaultFS.native}
$HIVE_OPTS"
```

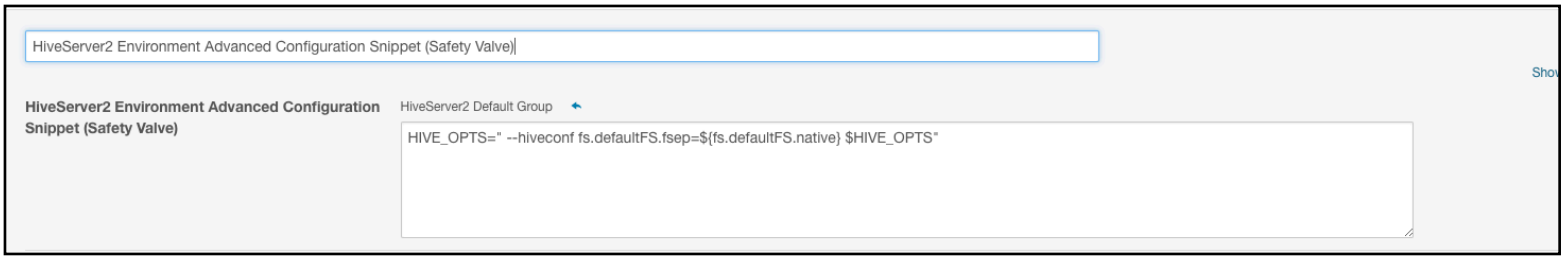
The following screenshot shows the settings.



Set "HiveServer2 Environment Advanced Configuration Snippet (Safety Valve)" to:

```
HIVE_OPTS=" --hiveconf fs.defaultFS.fsep=${fs.defaultFS.native}
$HIVE_OPTS"
```

The following screenshot shows the settings.



Set "Hive Client Advanced Configuration Snippet (Safety Valve) for hive-site.xml" to:

```
<property>
  <name>hive.metastore.warehouse.dir</name>
  <value><HDFS-URI>/user/hive/warehouse</value>
</property>
```

The following screenshot shows the settings.

Configuration for Impala on Isilon is complete.

Hadoop Functional Tests

Perform the tests in this section to ensure your Hadoop installation is correctly implemented. Only perform the steps for the components you have installed.

Create Hadoop User

Create a Hadoop user named hduser1.

HDFS Test

Perform the test commands in the following subsections.

Allow Access Scenario for HDFS

Ensure data access is allowed for the following scenario.

Connect to the database directly. Do not access the database through the BlueTalon Enforcement Point.

```
[root@mycluster1-master-0 ~]# sudo -u hdfs hdfs dfs -ls /
Found 5 items
```

```
-rw-r--r--   1 root  hadoop              0 2014-08-05 05:59
/THIS_IS_ISILON
drwxr-xr-x   - hbase hbase             148 2014-08-05 06:06 /hbase
drwxrwxr-x   - solr  solr                0 2014-08-05 06:07 /solr
drwxrwxrwt   - hdfs  supergroup        107 2014-08-05 06:07 /tmp
drwxr-xr-x   - hdfs  supergroup        184 2014-08-05 06:07 /user
[root@mycluster1-master-0 ~]# sudo -u hdfs hdfs dfs -put -f /etc/hosts
/tmp
[root@mycluster1-master-0 ~]# sudo -u hdfs hdfs dfs -cat /tmp/hosts
127.0.0.1 localhost
[root@mycluster1-master-0 ~]# sudo -u hdfs hdfs dfs -rm -skipTrash
/tmp/hosts
[root@mycluster1-master-0 ~]# su - hduser1
[hduser1@mycluster1-master-0 ~]$ hdfs dfs -ls /
Found 5 items
-rw-r--r--   1 root  hadoop              0 2014-08-05 05:59
/THIS_IS_ISILON
drwxr-xr-x   - hbase hbase             148 2014-08-05 06:28 /hbase
drwxrwxr-x   - solr  solr                0 2014-08-05 06:07 /solr
drwxrwxrwt   - hdfs  supergroup        107 2014-08-05 06:07 /tmp
drwxr-xr-x   - hdfs  supergroup        209 2014-08-05 06:39 /user
```

Deny Access Scenario for HDFS

Perform the following steps and ensure access is denied by BlueTalon:

1. Add a deny rule using the BlueTalon Policy Console to prevent access to the HDFS root folder ("/") for all users.
2. Connect to HDFS through the BlueTalon Enforcement Point.
3. Ensure the following command fails with an access denied error:
   [root@mycluster1-master-0 ~]# sudo -u hdfs hdfs dfs -ls /
4. Ensure the following command fails with an access denied error:
   [root@mycluster1-master-0 ~]# sudo -u hdfs hdfs dfs -cat /tmp/hosts

YARN / MapReduce Test

Perform the test commands in the following subsections.

Allow Access Scenario for YARN / MapReduce

Ensure data access is allowed for the following scenario.

Connect to the database directly. Do not access the database through the Enforcement Point.

```
[hduser1@mycluster1-master-0 ~]$ hadoop jar \
/usr/lib/hadoop-mapreduce/hadoop-mapreduce-examples.jar \
pi 10 1000
```

```
...
Estimated value of Pi is 3.14000000000000000000
[hduser1@mycluster1-master-0 ~]$ hadoop fs -mkdir in
```

Add a file to the "in" directory. Example:

```
[hduser1@mycluster1-master-0 ~]$ hadoop fs -put -f /etc/hosts in
[hduser1@mycluster1-master-0 ~]$ hadoop fs -ls in
...
[hduser1@mycluster1-master-0 ~]$ hadoop fs -rm -r out
[hduser1@mycluster1-master-0 ~]$ hadoop jar \
/usr/lib/hadoop-mapreduce/hadoop-mapreduce-examples.jar \
wordcount in out
...
[hduser1@mycluster1-master-0 ~]$ hadoop fs -ls out
Found 4 items
-rw-r--r--   1 hduser1 hduser1          0 2014-08-05 06:44 out/_SUCCESS
-rw-r--r--   1 hduser1 hduser1         24 2014-08-05 06:44 out/part-r-
00000
-rw-r--r--   1 hduser1 hduser1          0 2014-08-05 06:44 out/part-r-
00001
-rw-r--r--   1 hduser1 hduser1          0 2014-08-05 06:44 out/part-r-
00002
[hduser1@mycluster1-master-0 ~]$ hadoop fs -cat out/part*
localhost     1
127.0.0.1     1
```

Browse to the YARN Resource Manager Web console. Example:

```
http://mycluster1-master-0.lab.example.com:8088/
```

Browse to the MapReduce History Server Web console. Example:

```
http://mycluster1-master-0.lab.example.com:19888/
```

Ensure you can view the logs for the commands.

Deny Access Scenario for YARN / MapReduce

Perform the following steps and ensure access is denied by BlueTalon:

1. Add a rule for the Alice user in the BlueTalon Policy Console: Allow read, write, and execute recursive permissions for the HDFS root directory ("/").
2. Import a user named Bob (or use any other existing user in your BlueTalon system). Click the User Domains tab in the Policy Console to see the users. The default access for Bob should be deny access to the HDFS root directory (recursive).
3. Deploy the changes.
4. Create a text file named space.txt and add a line of text to the file.

5. Connect to the Enforcement Point and perform the following commands (as the Alice user, these commands should succeed because you added a rule to allow access):
```
su -l alice -c 'hdfs dfs -put space.txt /user/alice'
su -l alice -c 'yarn jar hadoop-mapreduce-examples.jar wordcount
/user/alice/space.txt /user/alice/space'
su -l alice -c 'hdfs dfs -cat /user/alice/space/part-r-00000'
su -l alice -c 'hdfs dfs -rm -f -skipTrash /user/alice/space/*'
su -l alice -c 'hdfs dfs -rmdir /user/alice/space/'
su -l alice -c 'hdfs dfs -rm -skipTrash /user/alice/space.txt'
```
6. Connect to the Enforcement Point and repeat the previous commands as the Bob user (or any other user you have). The commands should fail with an access denied error because, by default, BlueTalon prevents access when there is no access rule for a user.

Hive Test

Perform the test commands in the following subsections.

Allow Access Scenario for Hive

Ensure data access is allowed for the following scenario.

Connect to the database directly. Do not access the database through the Enforcement Point.

```
[hduser1@mycluster1-master-0 ~]$ hadoop fs -mkdir -p sample_data/tab1
[hduser1@mycluster1-master-0 ~]$ cat - > tab1.csv
1,true,123.123,2012-10-24 08:55:00
2,false,1243.5,2012-10-25 13:40:00
3,false,24453.325,2008-08-22 09:33:21.123
4,false,243423.325,2007-05-12 22:32:21.33454
5,true,243.325,1953-04-22 09:11:33
Type <Control+D>.
[hduser1@mycluster1-master-0 ~]$ hadoop fs -put -f tab1.csv
sample_data/tab1
[hduser1@mycluster1-master-0 ~]$ hive
hive>
DROP TABLE IF EXISTS tab1;
CREATE EXTERNAL TABLE tab1
(
   id INT,
   col_1 BOOLEAN,
   col_2 DOUBLE,
   col_3 TIMESTAMP
)
ROW FORMAT DELIMITED FIELDS TERMINATED BY ','
LOCATION '/user/hduser1/sample_data/tab1';
```

```
DROP TABLE IF EXISTS tab2;

CREATE TABLE tab2
(
   id INT,
   col_1 BOOLEAN,
   col_2 DOUBLE,
   month INT,
   day INT
)
ROW FORMAT DELIMITED FIELDS TERMINATED BY ',';

INSERT OVERWRITE TABLE tab2
SELECT id, col_1, col_2, MONTH(col_3), DAYOFMONTH(col_3)
FROM tab1 WHERE YEAR(col_3) = 2012;
...
OK
Time taken: 28.256 seconds

hive> show tables;
OK
tab1
tab2
Time taken: 0.889 seconds, Fetched: 2 row(s)

hive> select * from tab1;
OK
1      true   123.123       2012-10-24 08:55:00
2      false  1243.5        2012-10-25 13:40:00
3      false  24453.325     2008-08-22 09:33:21.123
4      false  243423.325    2007-05-12 22:32:21.33454
5      true   243.325       1953-04-22 09:11:33
Time taken: 1.083 seconds, Fetched: 5 row(s)

hive> select * from tab2;
OK
1      true   123.123       10     24
```

```
2       false   1243.5          10      25
Time taken: 0.094 seconds, Fetched: 2 row(s)

hive> select * from tab1 where id=1;
OK
1       true    123.123         2012-10-24 08:55:00
Time taken: 15.083 seconds, Fetched: 1 row(s)

hive> select * from tab2 where id=1;
OK
1       true    123.123         10      24
Time taken: 13.094 seconds, Fetched: 1 row(s)

hive> exit;
```

Deny Access Scenario for Hive

Perform the steps in the section Hive Enforcement Point on page 149 and ensure access to the soc_sec_no column of the test Hive accounts table is masked by BlueTalon for the Alice user.

Pig Test

Ensure data access is allowed for the following scenario.

Connect to the database directly. Do not access the database through the Enforcement Point.

```
[hduser1@mycluster1-master-0 ~]$ pig
grunt> a = load 'in';
grunt> dump a;
...
Success!
...
grunt> quit;
```

HBase Test

Ensure data access is allowed for the following scenario.

Connect to the database directly. Do not access the database through the Enforcement Point.

```
[hduser1@mycluster1-master-0 ~]$ hbase shell
hbase(main):001:0> create 'test', 'cf'
0 row(s) in 3.3680 seconds
=> Hbase::Table - test
hbase(main):002:0> list 'test'
TABLE
```

```
test
1 row(s) in 0.0210 seconds
=> ["test"]
hbase(main):003:0> put 'test', 'row1', 'cf:a', 'value1'
0 row(s) in 0.1320 seconds
hbase(main):004:0> put 'test', 'row2', 'cf:b', 'value2'
0 row(s) in 0.0120 seconds
hbase(main):005:0> scan 'test'
ROW                          COLUMN+CELL
 row1
column=cf:a,timestamp=1407542488028,value=value1
 row2
column=cf:b,timestamp=1407542499562,value=value2
2 row(s) in 0.0510 seconds
hbase(main):006:0> get 'test', 'row1'
COLUMN                       CELL
 cf:a                         timestamp=1407542488028,value=value1
1 row(s) in 0.0240 seconds
hbase(main):007:0> quit
```

Ambari Service Test

Ambari has functional tests for each component. The tests are executed automatically when you install your cluster with Ambari.

To execute the tests after installation:

1. Log in to the Ambari Web management console.
2. Select the service to test.
3. Click the Service Actions button.
4. Select Run Service Check.

Hue with Hive and Impala Enforcement Points

This section describes how to configure Hue with Hive and Impala Enforcement Points.

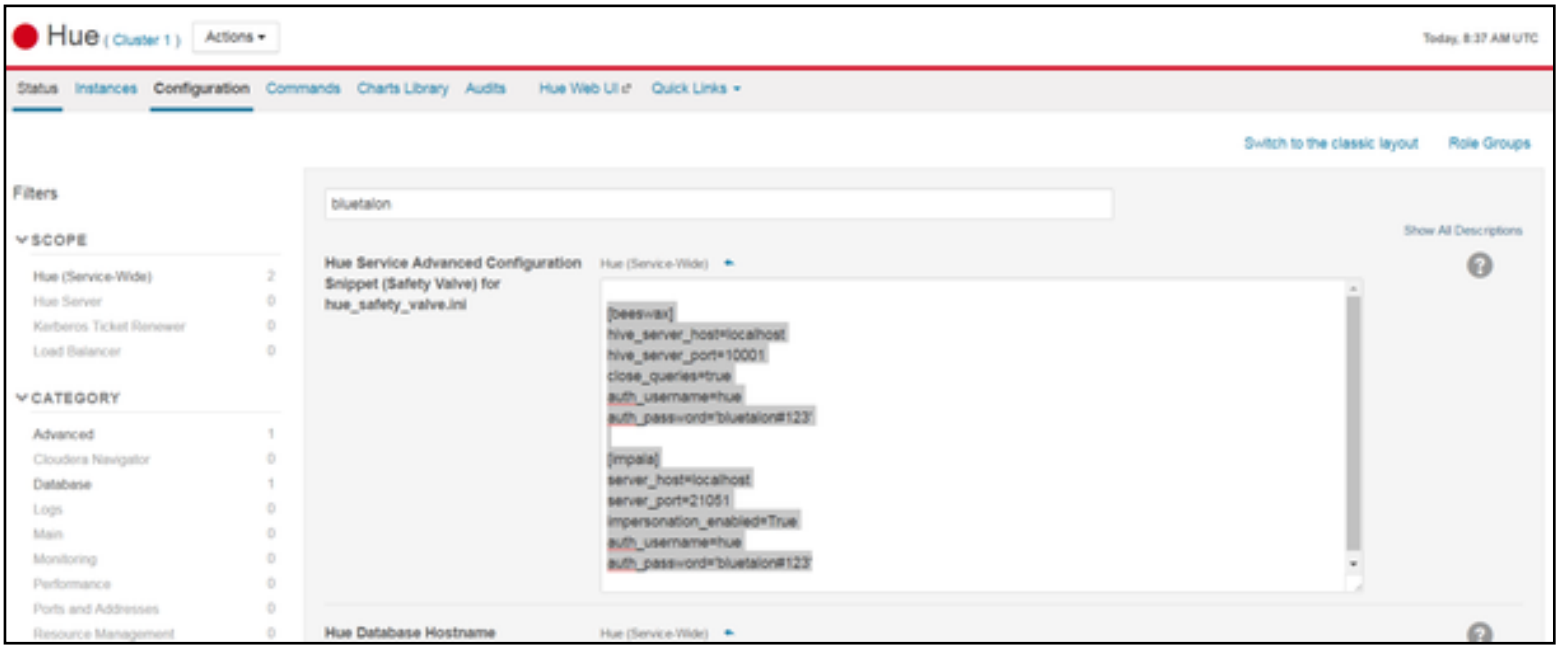
- BlueTalon released a patch for software version 3.2.2.
- The example in this section is applied to BlueTalon software version 3.2.0.
- To obtain the patch, contact your BlueTalon sales person.
- Ask your sales person if the patch is required for your installed version of the BlueTalon software.

Configure Hue

In the Cloudera Manager Web console, set the section "Hue Service Advanced Configuration Snippet (Safety Valve) for hue_safety_valve.ini" to:

```
[beeswax]
hive_server_host=localhost
hive_server_port=10001
close_queries=true
auth_username=hue
auth_password='bluetalon#123'

[impala]
server_host=localhost
server_port=21051
impersonation_enabled=True
auth_username=hue
auth_password='bluetalon#123'
```

The following screenshot shows the configuration.

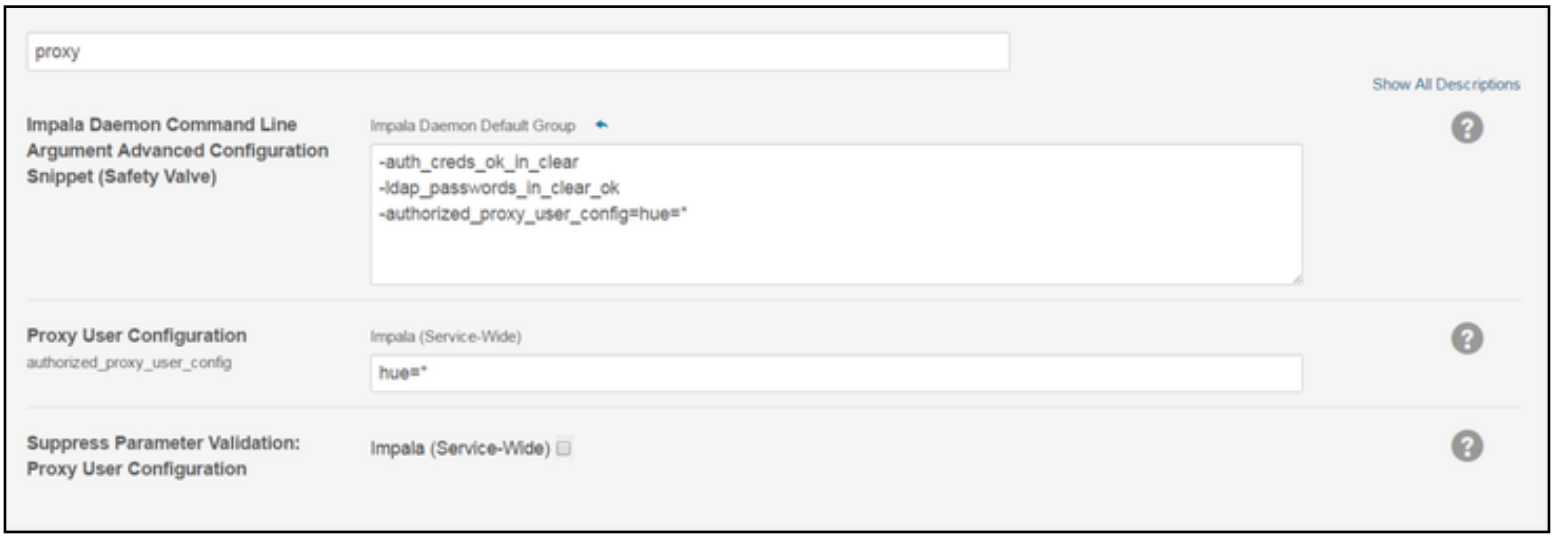


Configure Impala

Set the configuration for "Impala Daemon Command Line Argument Advanced Configuration Snippet (Safety Valve)" to:

```
-auth_creds_ok_in_clear
-ldap_passwords_in_clear_ok
-authorized_proxy_user_config=hue=*
```

The following screenshot shows the configuration.

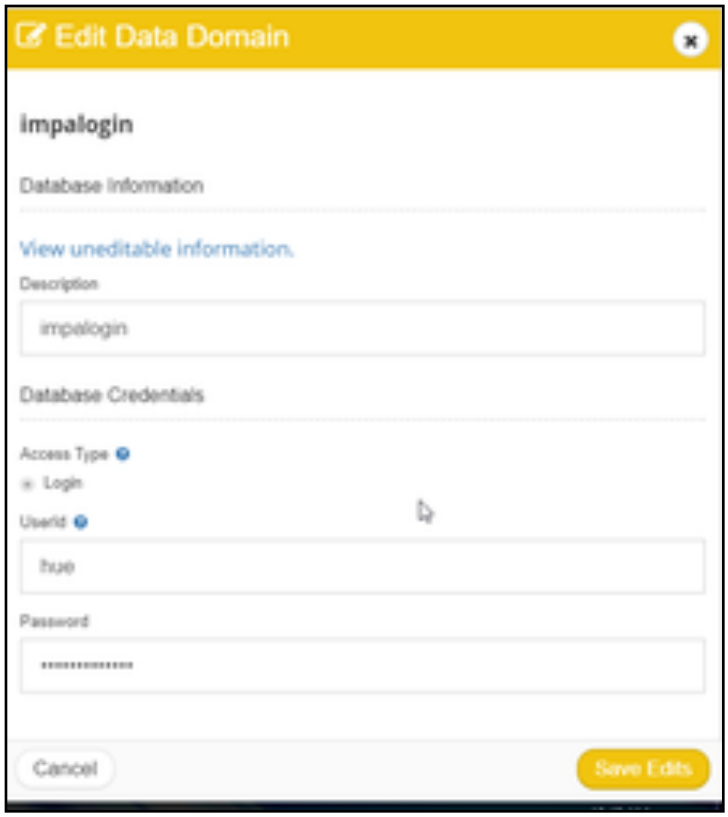


Configure Hive

Set the configuration for Hive as shown in the following screenshot.

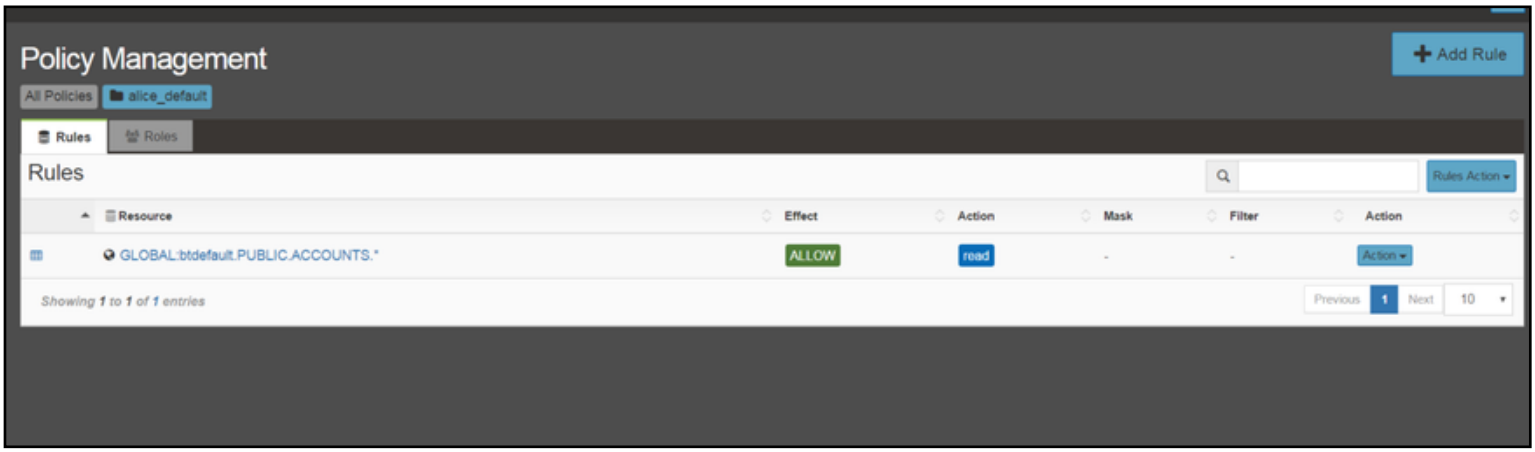
Impala Data Domain

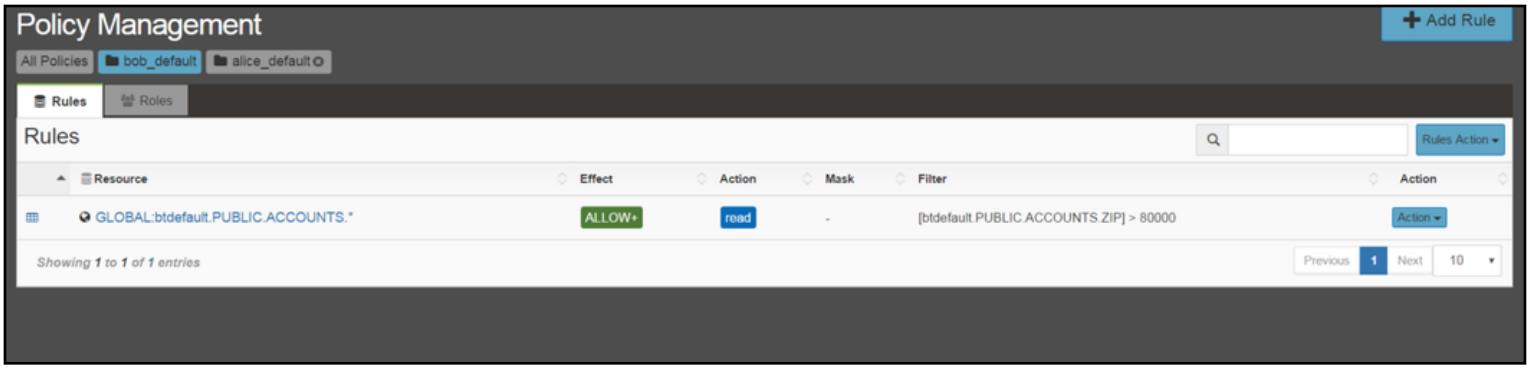
In the BlueTalon Policy Console, create a data domain for Impala. The following screenshot shows an example Impala data domain.



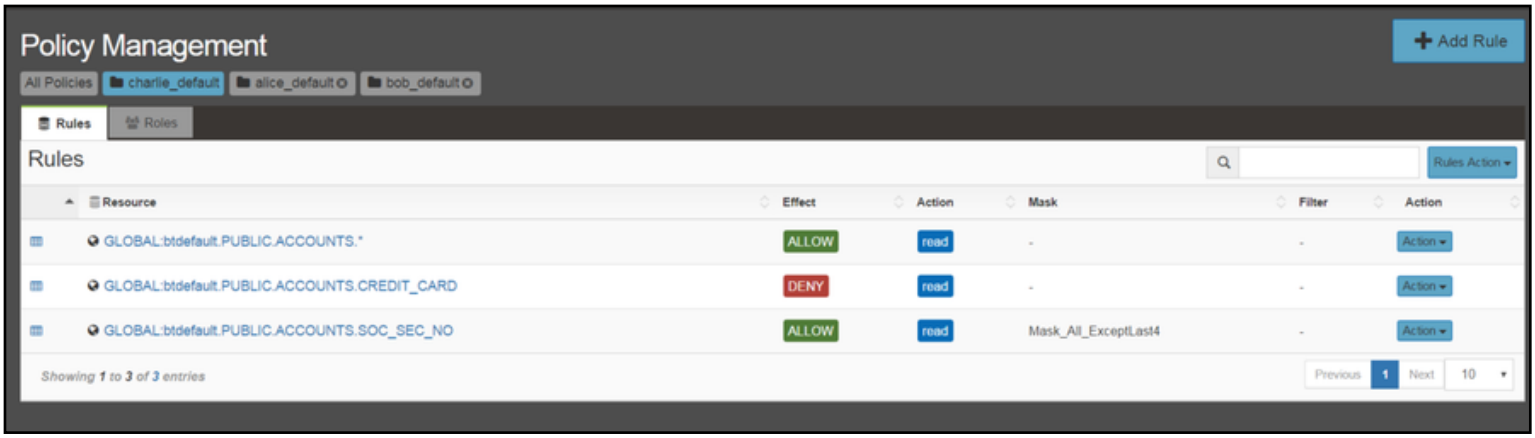
Rules

In the Policy Console, create rules for testing. The following screenshots show example rules for the accounts table.



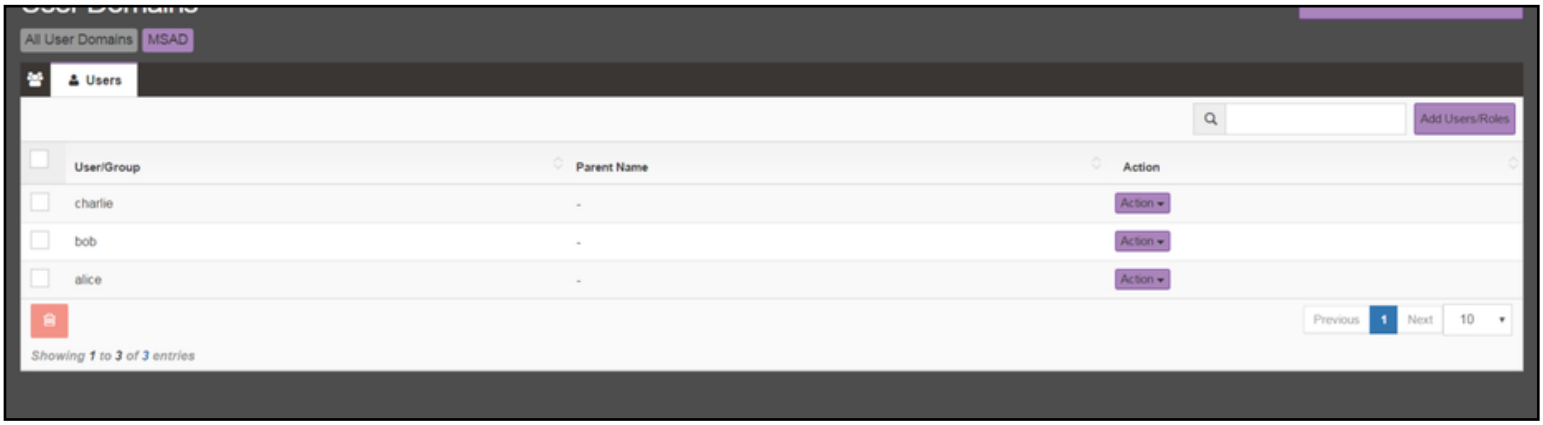


Notice in the following screenshot that read access to the credit_card column is denied.



End Users

In the Policy Console, configure end users for testing. The following screenshots show example users imported from Active Directory.



The following screenshot shows the Hue user.

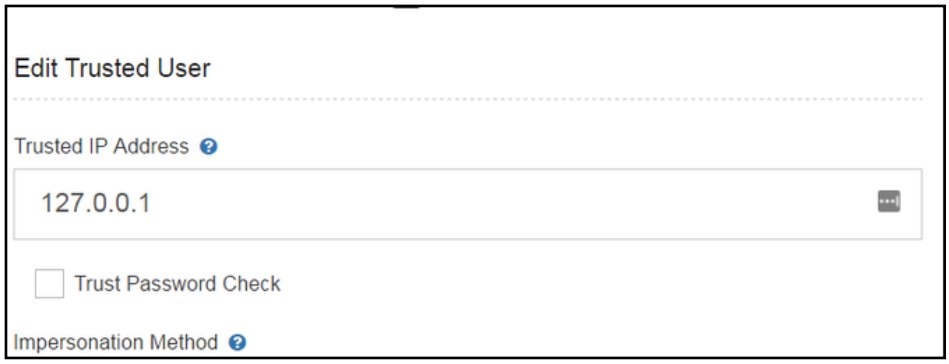
You must configure the Hue user in the Policy Console as a trusted user and set the IP address of the Hue server instance (you can use a 0.0.0.0/32 address style if you do not know the IP address of the Hue server).

Hue can be a user in the InternalSource user domain or a user in an AD/OpenLDAP user domain.

A password must be set for the Hue user if you are using the Hue user stored in the InternalSource user domain. The default password for the InternalSource Hue user is "hue".

The tested version is 5.9 for Cloudera CDH, and is configured with Hive and Impala using the "no login" method. You can also configure Hive and Impala for "login mode" with AD/OpenLDAP.

The following screenshot shows the configuration for the Hue user set as a trusted user on the local host computer (127.0.0.1).



To obtain the IP address to set for the Trusted IP Address:

1. Configure only the host_address and port in the hue.ini file and hue settings. File is: /etc/hue/conf/hue.ini
2. Restart Hue.
3. Run a Hive query as any user and attempt to retrieve the credit_card column from the accounts table (connect to Hive through the Enforcement Point). The query will fail because there is no allow rule in BlueTalon to permit data access for that column.
4. Log in to the Audit Console and examine the IP address for the failed query in the logs.

Configure Enforcement Point

Configure the BlueTalon Enforcement Point as follows:

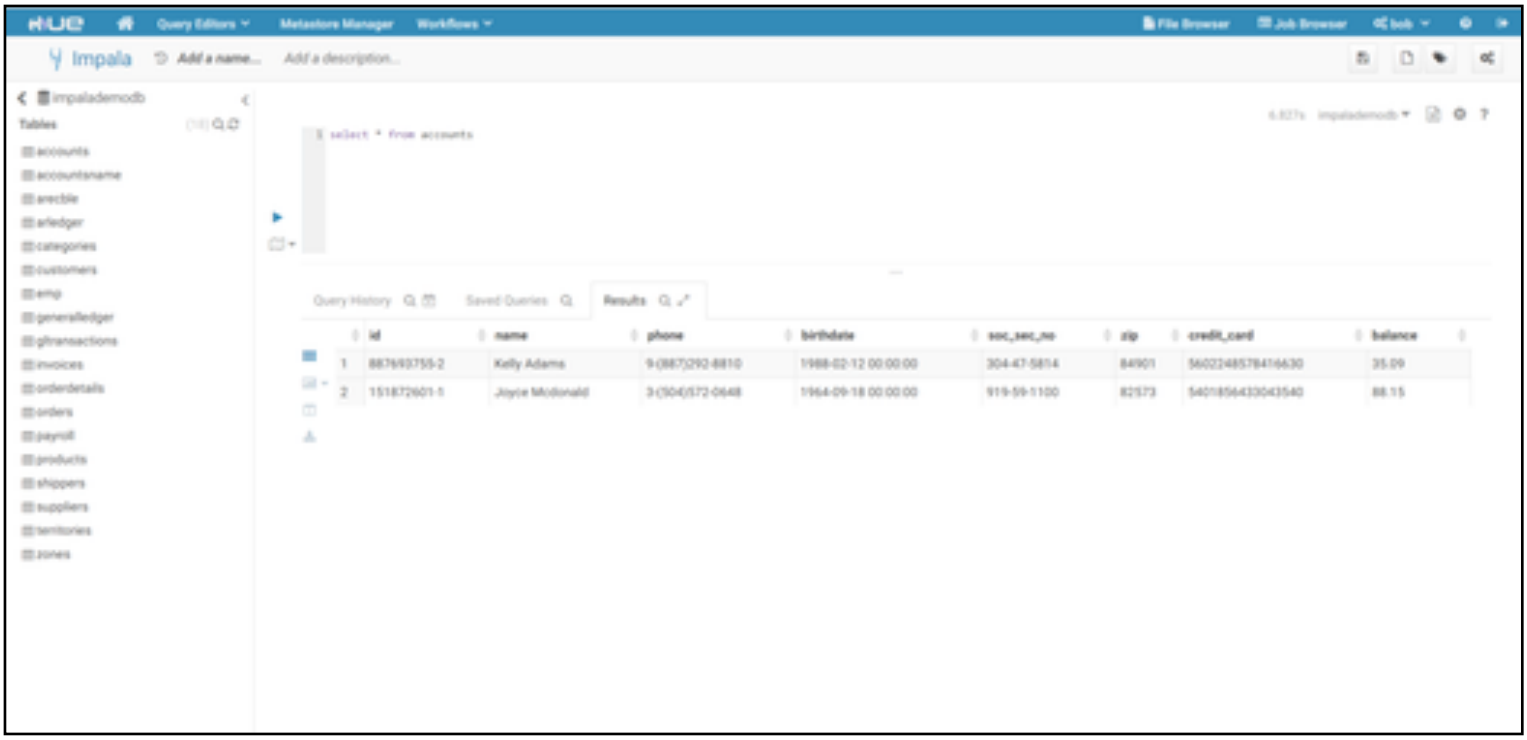
- Disable delegation for "no login" scenario (or enable delegation for "login" scenario).
- Enable password authentication.
- Enable token authorization.

Example:

```
[BTACCOUNTDELEGATION]
Value=FALSE;
[PASSWORDAUTH]
Value=TRUE;
[TOKENBASEDAUTHORIZATION]
Value=True;
```

Example Run with Hue

The following screenshot shows an example run with Hue. The data is retrieved using the Bob user.



This is the end of the section.

Set Up Enforcement Point for Cloud Relational Databases

BlueTalon Enforcement Points intercept queries from applications and sends them to the BlueTalon Policy Engine. The Policy Engine parses the queries according to defined policies and rules. A policy-compliant query is returned to the Enforcement Points and a policy-compliant dataset result is returned to the user.

This section describes how to set up a BlueTalon Enforcement Point for cloud relational databases.

Redshift Enforcement Point

Redshift is a petabyte-scale data warehouse for analyzing data using existing business intelligence tools. BlueTalon provides an Enforcement Point for Redshift.

This section describes how to set up an Enforcement Point for Redshift.

Create Test Data

Create a database user and table. See Create Test Data on page 53.

Add Data Domain

A data domain stores information about an external data source.

You import the external data source information into a data domain. For relational database sources, you can import table and column information.

Add a data domain. Use RedShift as your data domain type. See Add Data Domain on page 55.

Install Enforcement Point Package

Install the Enforcement Point package:

1. Enter:
   ```
   yum install bluetalon-ep-3.2.4
   ```
2. Follow the prompts.

Configure Enforcement Point

Configure the Enforcement Point:

1. Run:
   ```
   /opt/bluetalon/3.2.4/pep/scripts/bluetalon-ep-setup
   ```
2. Follow the prompts. You select the PostgreSQL Enforcement Point type for RedShift.

Test Enforcement Point

See Test Enforcement Point on page 70.

Verify Masked Data

See Verify Masked Data on page 78.

Cassandra Enforcement Point

This section describes how to set up an Enforcement Point for Cassandra.

Collect Information

The following table shows the information you must collect before installing the Enforcement Point.

Item	Description
Database name	Database name. Examples: • TestDatabase • CassandraDatabase
Host address	Computer hostname or IP address running the database. Examples: • TestServer.TestCompany.com • 127.0.0.1
Database port	Port the database receives JDBC connections. Example: 5432
User ID	User ID for a database account with privileges to obtain metadata.
Password	Password for the database account.

Create Test Data

This section contains example commands to create test data. You can use your own data source.

Log in as a database administration user:

```
cqlsh -u cassandra -p cassandra
```

Create a user named alice (use your own strong password):

```
CREATE USER alice WITH PASSWORD 'bt#123';
```

Create a keyspace named sales and use it:

```
CREATE KEYSPACE sales WITH REPLICATION = { 'class' : 'SimpleStrategy',
'replication_factor' : 1 };

USE sales;
```

Create a table named accounts:

```
CREATE TABLE accounts (id TEXT PRIMARY KEY, name TEXT, phone TEXT,
birthdate VARCHAR, soc_sec_no VARCHAR, zip INT, credit_card TEXT,
balance FLOAT);
```

Create an index named accounts_zip:

```
CREATE INDEX accounts_zip ON accounts(zip);
```

Insert rows into the accounts table:

```
INSERT INTO accounts (id, name, phone, birthdate, soc_sec_no, zip,
credit_card, balance) VALUES ('147274739-9', 'Frances Harrison', '9-
(192)357-8851', '10/27/1956', '993941527', 37726, '6393451850134970',
52.90);

INSERT INTO accounts (id, name, phone, birthdate, soc_sec_no, zip,
credit_card, balance) VALUES ('457389751-8', 'Gregory Patterson', '3-
(684)454-2444', '11/22/1969', '233575483', 21278, '5602245983499780',
44.87);

INSERT INTO accounts (id, name, phone, birthdate, soc_sec_no, zip,
credit_card, balance) VALUES ('360272064-0', 'Earl James', '5-(177)394-
3277', '05/23/1983', '303640766', 26595, '3534227478434860', 3.36);

INSERT INTO accounts (id, name, phone, birthdate, soc_sec_no, zip,
credit_card, balance) VALUES ('551058833-0', 'Christine Robertson', '5-
(437)964-8463', '05/11/1988', '318794141', 50716, '5602223026663730',
5.22);

INSERT INTO accounts (id, name, phone, birthdate, soc_sec_no, zip,
credit_card, balance) VALUES ('713553396-8', 'Daniel Crawford', '1-
(875)657-9518', '06/01/1999', '214894670', 28693, '3571338359613280',
22.63);

INSERT INTO accounts (id, name, phone, birthdate, soc_sec_no, zip,
credit_card, balance) VALUES ('359936857-0', 'Shirley Holmes', '6-
(362)871-3036', '04/01/1980', '964490690', 13189, '3534219304003200',
30.17);

INSERT INTO accounts (id, name, phone, birthdate, soc_sec_no, zip,
credit_card, balance) VALUES ('887693755-2', 'Kelly Adams', '9-
(887)292-8810', '02/12/1988', '304475814', 84901, '5602248578416630',
35.09);

INSERT INTO accounts (id, name, phone, birthdate, soc_sec_no, zip,
credit_card, balance) VALUES ('091619799-9', 'Bonnie Freeman', '2-
(163)102-1214', '01/26/1971', '940518723', 73248, '3547967170434520',
69.74);

INSERT INTO accounts (id, name, phone, birthdate, soc_sec_no, zip,
credit_card, balance) VALUES ('151872601-1', 'Joyce Mcdonald', '3-
(504)572-0648', '09/18/1964', '919591100', 82573, '5401856433043540',
88.15);

INSERT INTO accounts (id, name, phone, birthdate, soc_sec_no, zip,
credit_card, balance) VALUES ('956961211-8', 'Karen Burton', '1-
(533)256-2922', '03/05/1954', '479488766', 56134, '3536534268148670',
7.08);
```

Examine the rows in the accounts table:

```
SELECT * FROM accounts;
```

You will create a rule to mask the social security number stored in the soc_sec_no column later.

Add Data Domain

A data domain stores information about an external data source.

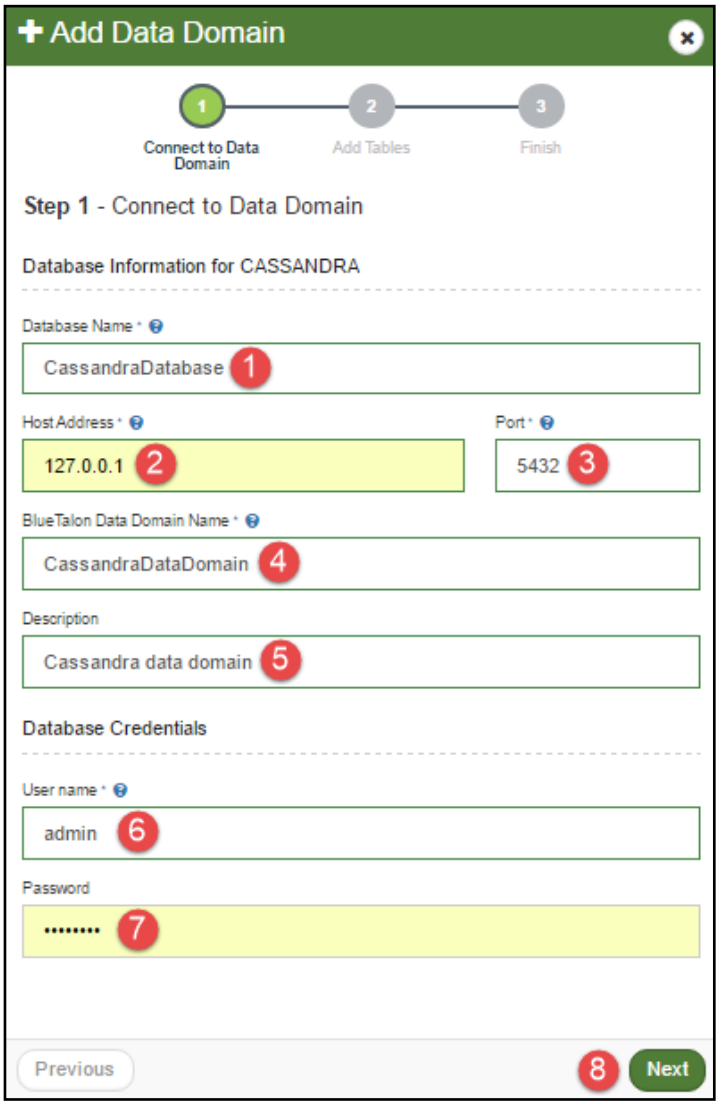
You import the external data source information into a data domain. For relational database sources, you can import table and column information.

Add a data domain:

1. See Add Data Domain on page 55. That section contains similar steps that you will use here.
2. Use Cassandra as your data domain type.
3. Enter the data domain details. Use the details you obtained earlier in the Collect Information section. See the following screenshot for an example.

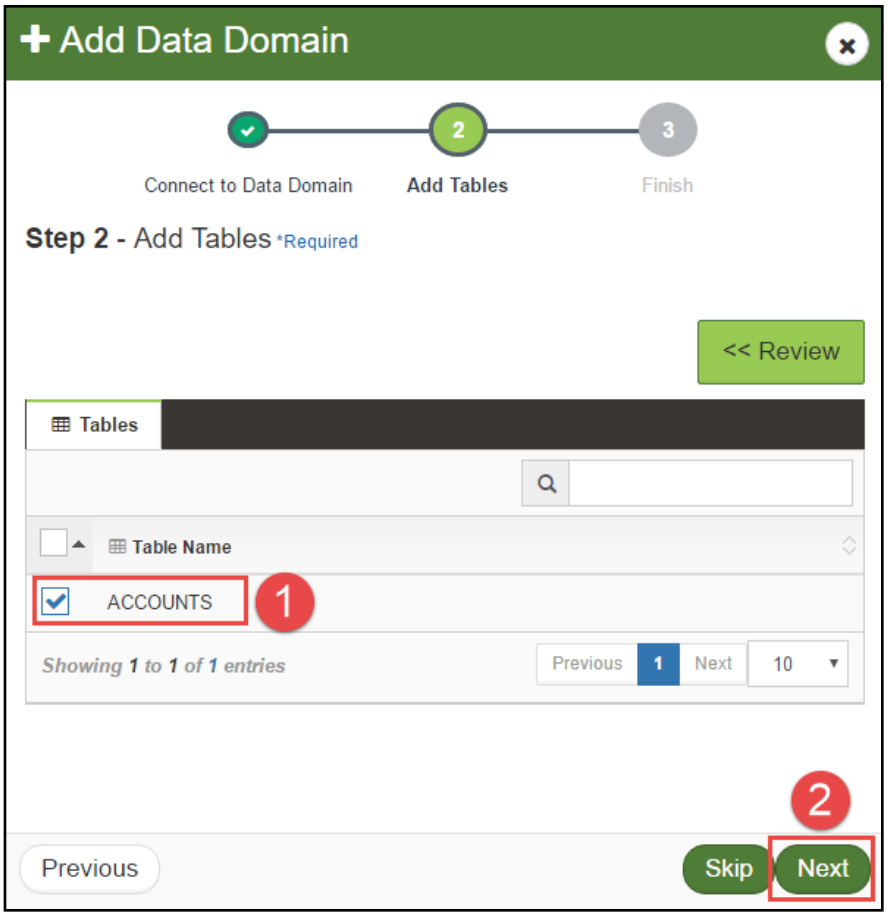
   Example data domain name:
   CassandraDataDomain

   Note the name of your data domain. You will need it later when you configure the Enforcement Point.



4. Click Next.

5. Select the accounts table you created earlier. See item 1 in the following screenshot.



6. Click Next. See item 2 in the previous screenshot.
7. Deploy.

Install Enforcement Point Package

Install the Enforcement Point package:

1. Enter:
   `yum install bluetalon-ep-3.2.4`
2. Follow the prompts.

Configure Enforcement Point

Configure the Enforcement Point:

1. Run:
   `/opt/bluetalon/3.2.4/pep/scripts/bluetalon-ep-setup`
2. Accept the license agreement.
3. Enter the information.

Item	Description
Enforcement Point type	Type of Enforcement Point.

	Select Cassandra.
Policy Engine hostname	Hostname or IP address for the computer running the Policy Engine. Example: 10.0.0.70
Policy Engine port	Policy Engine port. Example: 1555
Data domain name	Name of the data domain. Use the value set in the Policy Console for the data domain you created earlier. Example: CassandraDataDomain
Cassandra hostname	Hostname or IP address for the computer running Cassandra. Example: 10.0.0.68
Cassandra Port	Cassandra port. Example: 5432
Enforcement Point mode	Options: • Forward only • Audit only • Enforce only • Audit and Enforce

Enforcement Point Files

The following table shows the Enforcement Point files.

Item	Detail
Binary files path	/opt/bluetalon/3.2.4/pep/cs-ep/bin
Configuration files path	/etc/bluetalon/pep/cs-ep/conf
Log files path	/var/log/cassandra

Test Enforcement Point

Initially:

1. Add a user named alice. See Add User to Internal User Domain on page 48.
2. Add a policy named TestPolicy. See Add Policy on page 70.
3. Add the alice user to TestPolicy. See Add User to Policy on page 73.

Add Rule to Policy

You add a rule to the policy to allow access to a resource. For a relational database source, a resource is a table or column.

Add a rule to the test policy:

1. See Add Rule to Policy on page 75.
2. Set the data domain to CassandraDataDomain.

3. Set the resource to the soc_sec_no column of the accounts table.

   The resource has the format:
   Database.Schema.Table.Column

   Example:
   CassandraDatabase.sales.accounts.soc_sec_no
4. Set the action to Read.
5. Set the effect to Mask.
6. Set the mask to Mask_All_ExceptLast4 to hide everything except the last four digits of the social security number.
7. Select Add to Policy Set.
8. Add the rule to TestPolicy.
9. Click Add Rule.
10. Deploy.

Verify Masked Data

Verify masked data:

1. Connect to the database as the alice user. Example:
   ```
   cqlsh -u alice -p bt#123
   ```
2. Run the following commands and ensure the soc_sec_no column is masked:
   ```
   USE sales;
   SELECT soc_sec_no FROM accounts;
   ```

Spark Enforcement Point

This section describes how to set up an Enforcement Point for Spark.

Collect Information

The following table shows the information you must collect before installing the Enforcement Point.

Item	Description
Database name	Database name. Examples: • TestDatabase • default
Host address	Computer hostname or IP address running the database. Examples: • TestServer.TestCompany.com • 127.0.0.1
Database port	Port the database receives JDBC connections. Example: 10002

User ID	User ID for a database account with privileges to obtain metadata.
Password	Password for the database account.

Create Test Data

This section contains example commands to create test data. You can use your own data source.

Create a CSV file named accounts.csv in the /tmp directory with the following content:

```
147274739-9,Frances Harrison,9-(192)357-
8851,10/27/1956,993941527,37726,6393451850134970,52.90

457389751-8,Gregory Patterson,3-(684)454-
2444,11/22/1969,233575483,21278,5602245983499780,44.87

360272064-0,Earl James,5-(177)394-
3277,05/23/1983,303640766,26595,3534227478434860,3.36

551058833-0,Christine Robertson,5-(437)964-
8463,05/11/1988,318794141,50716,5602223026663730,5.22

713553396-8,Daniel Crawford,1-(875)657-
9518,06/01/1999,214894670,28693,3571338359613280,22.63

359936857-0,Shirley Holmes,6-(362)871-
3036,04/01/1980,964490690,13189,3534219304003200,30.17

887693755-2,Kelly Adams,9-(887)292-
8810,02/12/1988,304475814,84901,5602248578416630,35.09

091619799-9,Bonnie Freeman,2-(163)102-
1214,01/26/1971,940518723,73248,3547967170434520,69.74

151872601-1,Joyce Mcdonald,3-(504)572-
0648,09/18/1964,919591100,82573,5401856433043540,88.15

956961211-8,Karen Burton,1-(533)256-
2922,03/05/1954,479488766,56134,3536534268148670,7.08
```

Create a database table named accounts:

```
beeline

!connect jdbc:hive2://<Hostname or IP address of
computer>:<Port>/default

CREATE TABLE accounts (id STRING, name STRING, phone STRING, birthdate
STRING, soc_sec_no STRING, zip BIGINT, credit_card BIGINT, balance
DECIMAL(4,2)) ROW FORMAT DELIMITED FIELDS TERMINATED BY ',' LINES
TERMINATED BY '\n' STORED AS TEXTFILE;

!q
```

The final command in the previous list exits from beeline.

Create a directory named tmp in HDFS and load the accounts.csv file data into the HDFS tmp directory:

```
hdfs dfs -mkdir /tmp

hdfs dfs -put /tmp/accounts.csv /tmp
```

Load the data from HDFS into the accounts database table:

```
beeline
```

```
!connect jdbc:hive2://<Hostname or IP address of
computer>:<Port>/default
```

```
LOAD DATA INPATH '/tmp/accounts.csv' into table accounts;
```

Examine the rows in the accounts table:

```
SELECT * FROM accounts;
```

You will create a rule to mask the social security number stored in the soc_sec_no column later.

Exit beeline:

```
!q
```

Add Data Domain

A data domain stores information about an external data source.

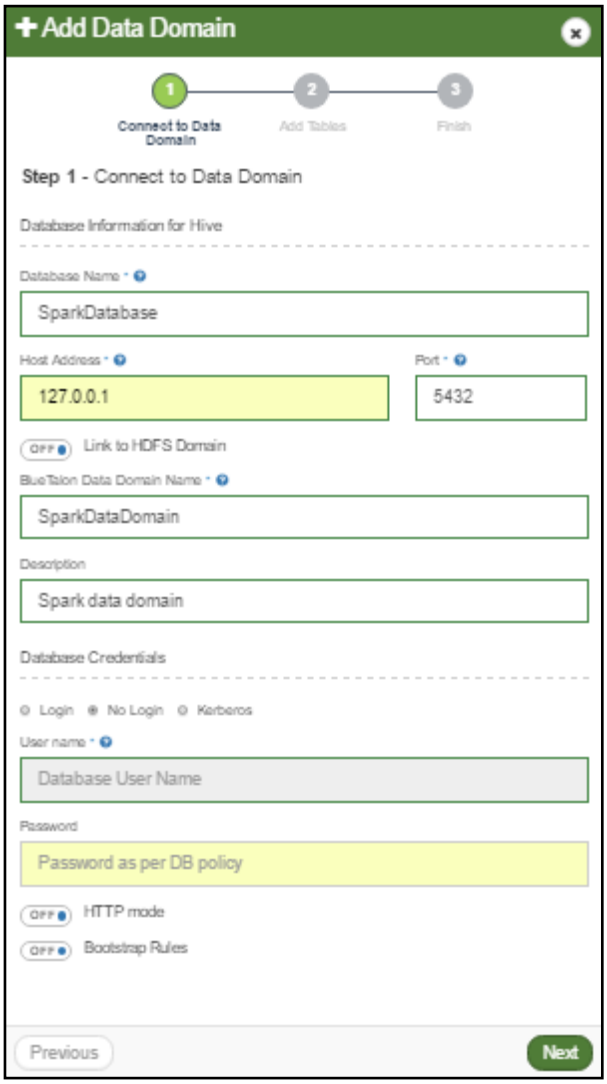
You import the external data source information into a data domain: For relational database sources, you can import table and column information.

Add a data domain:

1. See Add Data Domain on page 55. That section contains similar steps that you will use here.
2. Use Hive as your data domain type. You select Hive even though the database is Spark.
3. Enter the data domain details. Use the details you obtained earlier in the Collect Information section. See the following screenshot for an example.

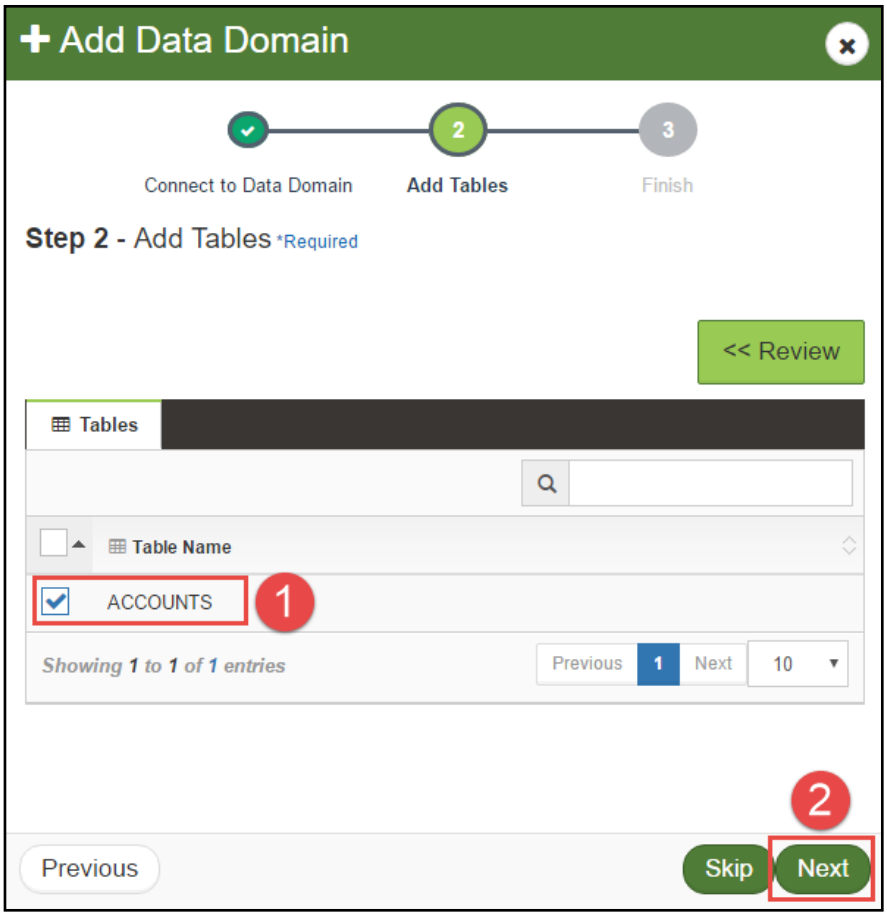
   Example data domain name:
   SparkDataDomain

   Note the name of your data domain. You will need it later when you configure the Enforcement Point.


Add Data Domain
1
2
3
Connect to Data Domain
Add Tables
Finish
Step 1 - Connect to Data Domain
Database Information for Hive
Database Name *
SparkDatabase
Host Address *
127.0.0.1
Port *
5432
OFF
Link to HDFS Domain
BlueTalon Data Domain Name *
SparkDataDomain
Description
Spark data domain
Database Credentials
Login
No Login
Kerberos
User name *
Database User Name
Password
Password as per DB policy
OFF
HTTP mode
OFF
Bootstrap Rules
Previous
Next

4. Click Next.

5. Select the accounts table you created earlier. See item 1 in the following screenshot.



6. Click Next. See item 2 in the previous screenshot.
7. Deploy.

Install Enforcement Point Package

Install the Enforcement Point package:

1. Enter:
   `yum install bluetalon-ep-3.2.4`
2. Follow the prompts.

Configure Enforcement Point

Configure the Enforcement Point:

1. Run:
   `/opt/bluetalon/3.2.4/pep/scripts/bluetalon-ep-setup`
2. Accept the license agreement.
3. Enter the information.

Item	Description
Enforcement Point type	Type of Enforcement Point.

	Select Spark SQL.
Enforcement Point port	JDBC end point port that client applications use to obtain secure data. Example: 1557
Spark hostname	Hostname or IP address for the computer running Spark. Example: 10.0.0.45
Spark port	Spark port. Example: 5432
BlueTalon Policy package hostname	Computer hostname or IP address on which the BlueTalon Policy Engine is running. Example: 10.0.0.70
bt-policy-configuration service port	Port for the BlueTalon Policy Engine. Example: 1600
Data domain name	Name of the data domain. Use the value set in the Policy Console for the data domain you created earlier. Example: SparkDataDomain

Enforcement Point Files and Service

The following table shows the Enforcement Point files and service.

Item	Detail
Binary files path	/opt/bluetalon/3.2.4/pep/sparksql-ep/bin
Configuration files path	/etc/bluetalon/pep/sparksql-ep/conf
Log files path	/var/log/bluetalon/pep/sparksql-ep/logs
Service name	bt–sparksql-ep-<data domain name>

Test Enforcement Point

Initially:

1. Add a user named alice. See Add User to Internal User Domain on page 48.
2. Add a policy named TestPolicy. See Add Policy on page 70.
3. Add the alice user to TestPolicy. See Add User to Policy on page 73.

Add Rule to Policy

You add a rule to the policy to allow access to a resource. For a relational database source, a resource is a table or column.

Add a rule to the test policy:

1. See Add Rule to Policy on page 75.
2. Set the data domain to SparkDataDomain.

3. Set the resource to the soc_sec_no column of the accounts table.

   The resource has the format:
   Database.Schema.Table.Column

   Examples:
   default.default.accounts.soc_sec_no
   SparkDatabase.SparkDatabase.accounts.soc_sec_no
4. Set the action to Read.
5. Set the effect to Mask.
6. Set the mask to Mask_All_ExceptLast4 to hide everything except the last four digits of the social security number.
7. Select Add to Policy Set.
8. Add the rule to TestPolicy.
9. Click Add Rule.
10. Deploy.

Verify Masked Data

Verify masked data:

1. Connect to the database through the port for the Enforcement Point. Example:
   ```
   beeline
   !connect jdbc:hive2://<Hostname or IP address of
   computer>:<BlueTalon Enforcement Point port>/default
   ```
2. Log in as the alice user.
3. Run the following query and ensure the soc_sec_no column is masked:
   ```
   SELECT soc_sec_no FROM accounts;
   ```

Configure Enforcement Point Using Setup Script in Silent Mode

Ensure you have deployed your data domain before deploying an Enforcement Point.

(Optional) You can also configure an Enforcement Point in silent mode, which does not prompt you to enter details.

To configure the Enforcement Point with a preconfigured XML file, run:

```
/opt/bluetalon/3.2.4/pep/scripts/bluetalon-ep-setup -accept -f
/tmp/spark.xml
```

For the parameter list, see Enforcement Point Silent Mode XML Parameters on page 65.

Example Spark Silent Mode XML File

Example Spark silent mode XML file:

```
<EnforcementPoint xmlns="">
  <hostname>172.30.1.142</hostname>
  <port>10015</port>
  <db_type>14</db_type>
  <mode>3</mode>
```

```
  <resource_domain_name>sparktest</resource_domain_name>
  <custom_name>sparktest</custom_name>
  <ep_port>2003</ep_port>
  <dbtcp_ip>172.30.0.228</dbtcp_ip>
  <pwd_auth>yes</pwd_auth>
  <proxy_auth>yes</proxy_auth>
  <endpoint_tag>DEFAULT</endpoint_tag>
</EnforcementPoint>
```

Spark configuration is complete.

Configure Installation

This section describes how to configure your BlueTalon installation.

Set Strong Passwords

Always change the default passwords and set strong passwords for your installation.

The password modification scripts are described in the following table.

Script	Description
/opt/bluetalon/current/policy/pap/scripts/reset-pap-db-password	Change the Policy Database password. If you change the Policy Database password, you must then change the Policy Engine password using the script on the following row.
/opt/bluetalon/current/policy/pdp/scripts/update-pdp-password	Change the Policy Engine password.
/opt/bluetalon/current/policy/pap/scripts/reset-pap-ui-password	Change the Policy Console password.
/opt/bluetalon/current/audit/bluetalon-audit-visualize-basic/scripts/reset-audit-ui-password	Change the Audit Console password.

You can run the scripts in interactive mode or silent mode.

Interactive Mode Password Change

In interactive mode, you enter the passwords when the script prompts you:

1. Run the script. Example:
   `/opt/bluetalon/current/audit/bluetalon-audit-visualize-basic/scripts/reset-audit-ui-password`
2. The script confirms that you want to reset the password.
3. Enter a new password.
4. Prompts you to confirm.
5. Password is changed.

Silent Mode Password Change

In slient mode, you enter the passwords on the command line when you run the script:

1. Run the script and specify the new password. Example:
   `/opt/bluetalon/current/audit/bluetalon-audit-visualize-basic/scripts/reset-audit-ui-password -confirm -password <new password>`
2. Password is changed.

User Names

The following table shows the default access details for your BlueTalon Policy Console and Audit Console.

Application	URL	User Name	Default

			Password
Policy Console	<fully qualified domain name or IP address:port>/BlueTalonConfig Examples: • http://TestServer.TestCompany.com:8111/BlueTalonConfig • 127.0.0.1:8111/BlueTalonConfig	btadminuser	P@ssw0rd
Audit Console	<fully qualified domain name or IP address:port>/BlueTalonAudit Examples: • http://TestServer.TestCompany.com:8112/BlueTalonAudit • 127.0.0.1:8112/BlueTalonAudit	btadminuser	P@ssw0rd

Repositories

The BlueTalon Policy and Audit data are stored in PostgreSQL repositories. The contents of the repositories are for BlueTalon use only and should not be modified.

The following table shows the repositories that store the BlueTalon data along with default passwords.

Repository	Description	Database	User Name	Default Password	Example JDBC Connection String
Policy	Compiled policy data	btrepo	btuser	bt#123	jdbc:postgresql://127.0.0.1:5433/btrepo
Audit	Audit data	btauditrepo	btaudituser	bt#123	jdbc:postgresql://127.0.0.1:5433/btauditrepo
Design	Policy design data	btuirepo	btuiuser	bt#123	jdbc:postgresql://127.0.0.1:5433/btuirepo

Connect to Policy Repository

Use SSH to connect to the computer that has the Policy package installed.

Connect to the policy repository:

```
psql -h 127.0.0.1 -p 5433 -U btuser -d btrepo
password : bt#123
```

Connect to Audit Repository

Use SSH to connect to the computer that has the Policy package installed.

Connect to the audit repository:

```
psql -h 127.0.0.1 -p 5434 -U btaudituser -d btauditrepo
password : bt#123
```

Connect to Design Repository

Use SSH to connect to the computer that has the Policy package installed.

Connect to the design repository:

```
psql -h 127.0.0.1 -p 5433 -U btuiuser -d btuirepo
password : bt#123
```

Access Policy Console and Audit Console

You use the Policy Console to manage:

- Policies
- Data sources
- Rules
- Users
- Security attributes

You use the Audit Console to:

- Examine audit events.
- Run audit reports.

Access the BlueTalon Policy Console and Audit Console:

1. Open Web browser.
2. Go to the Policy Console:
   ```
   http://<fully qualified domain name or IP
   address:port>/BlueTalonConfig
   ```

   The default port is 8111.
3. The following table shows the default user name and password.

User Name	Default Password
btadminuser	P@ssw0rd

4. Go to the Audit Console:
   ```
   http://<fully qualified domain name or IP
   address:port>/BlueTalonAudit
   ```

   The default port is 8112.
   The Audit Console has the same default user name and password shown in the previous table.

Commands to Manage Services

This section describes the commands to manage the BlueTalon services.

Examine Service Status

Examine the BlueTalon service status:

1. Run the service command and search for BlueTalon services:
   ```
   service --status-all | grep bt
   ```
2. View the status of the services. Each service has a name in the form:
   ```
   bt-<service name>
   ```

Manage Services

The following table shows a subset of the BlueTalon services and management commands.

Service	Command
Policy Engine	service bt-policy-engine start | stop | status | restart
Audit Monitor	service bt-audit-monitor star t |stop | status | restart
Policy Server	service bt-policy-server start | stop | status | restart
PostgreSQL Enforcement Point	service bt-postgresql-ep-<data domain name> start | stop | status | restart

Use Windows Active Directory Users to Access BlueTalon Consoles

This section describes how to configure Windows Active Directory (AD) with SSL (LDAPS) and BlueTalon. After you have performed the steps in this section, you will be able to use a Windows AD user to log in to the BlueTalon Policy Console and Audit Console.

- If you have not exported a certificate, then see Export Certificate on page 197.
- If you have exported a certificate, then see Import Certificate on page 198.

Export Certificate

This section describes how to export the root certificate.

Prerequisite steps:

1. If you do not have the Certificate Server Role in Windows AD on the Domain Controller computer, then install the Certificate Server Role.
2. If you installed the Certificate Server Role, reboot the computer. The reboot is mandatory and causes the server certificate to be requested, generated, and installed.

You use the Windows user interface or the command line to perform the export.

To export the certificate using the Windows user interface:

1. On the Domain Controller computer, click Start.
2. Select Administrative Tools.
3. Click Certification Authority.
4. Right-click on your Certificate Authority.
5. Select Properties.
6. In the Properties window, click View Certificate.
7. In the Certificate window, click the Details tab.
8. Click Copy to File.
9. In the Certificate Export Wizard window, click Next.
10. Select Base-64 encoded X.509 (.CER).
11. Click Next.
12. Enter the path and file name. Example: C:\BTCorpRootCa.cer.
13. Click Next.
14. Click Finish.

To export the certificate using the command line:

1. Run the command:
   `certutil -ca.cert DomainController.cer`
2. Copy the certificate to the Linux computer. Example certificate file name:
   `/home/ec2-user/BTCorpRootCa.cer`
3. Edit the LDAP configuration file on the Linux computer to ensure the ldapsearch utility can operate:
   `vi /etc/openldap/ldap.conf`

Example ldap.conf file (set your own host IP address and port):

```
#
# LDAP Defaults
#


# See ldap.conf(5) for details
# This file should be world readable but not world writable.


#BASE   dc=example,dc=com
#URI    ldap://ldap.example.com ldap://ldap-master.example.com:666
#SIZELIMIT      12
#TIMELIMIT      15
#DEREF          never
HOST XX.XX.XXX.XXX
PORT XXX
TLS_CACERT  /home/ec2-user/BTCorpRootCa.cer
TLS_REQCERT  allow
TLS_CACERTDIR  /etc/openldap/certs
```

Import Certificate

To import the certificate:

1. Run the command:
   `keytool -import -alias DomainController -file /home/ec2-user/DomainController.cer -keystore /opt/bluetalon/current/shared/jre/lib/security/cacerts`
2. Enter the keystore password. The default password is changeit.

Configure BlueTalon Policy Service

To configure the BlueTalon Policy service:

1. Add the realm in the file /opt/bluetalon/current/policy/pap/conf/server.xml. The realm in the file has the format:
   `<Realm className="org.apache.catalina.realm.JNDIRealm" connectionURL="ldaps://<AD_HOST>:<AD_PORT>" debug="10" connectionName="<AD_USER@FQDN>"`

```
connectionPassword="<PWD_OF_AD_USER>" referrals="follow"
userBase="<BASE_DN>" userSearch="(sAMAccountName={0})"
userSubtree="true" roleBase="<BASE_DN>" roleName="name"
roleSubtree="true" roleSearch="(member={0})"
allRolesMode="authOnly"/>
```

Example setting (set your own connection parameters):

```
<Realm className="org.apache.catalina.realm.JNDIRealm"
connectionURL="ldaps://XX.XX.XXX.XXX:XX" debug="10"
connectionName="Administrator@example.com" connectionPassword="XXXX"
referrals="follow" userBase="dc=corp,dc=example,dc=com"
userSearch="(sAMAccountName={0})" userSubtree="true"
roleBase="dc=corp,dc=example,dc=com" roleName="name"
roleSubtree="true" roleSearch="(member={0})"
allRolesMode="authOnly"/>
```

2. Restart the service:
   ```
   service bt-policy-server restart
   ```
3. You can use a Windows AD user with SSL to log in to the BlueTalon Policy Console. Example user ID: Administrator.

Configure BlueTalon Audit Service

To configure the BlueTalon Audit service:

1. Edit the file /opt/bluetalon/3.2.4/audit/bluetalon-audit-visualize-basic/conf/server.xml.
2. Add the realm to the file. The realm has the format:
   ```
   <Realm className="org.apache.catalina.realm.JNDIRealm"
   connectionURL="ldaps://<AD_HOST>:<AD_PORT>" debug="10"
   connectionName="<AD_USER@FQDN>"
   connectionPassword="<PWD_OF_AD_USER>" referrals="follow"
   userBase="<BASE_DN>" userSearch="(sAMAccountName={0})"
   userSubtree="true" roleBase="<BASE_DN>" roleName="name"
   roleSubtree="true" roleSearch="(member={0})"
   allRolesMode="authOnly"/>
   ```

   Example setting (remember to set your IP address and password):

   ```
   <Realm className="org.apache.catalina.realm.JNDIRealm"
   connectionURL="ldaps://XX.XX.XXX.XXX:XXX" debug="10"
   connectionName="Administrator@corp.bluetalon.com"
   connectionPassword="XXXX" referrals="follow"
   userBase="dc=corp,dc=bluetalon,dc=com"
   userSearch="(sAMAccountName={0})" userSubtree="true"
   roleBase="dc=corp,dc=bluetalon,dc=com" roleName="name"
   roleSubtree="true" roleSearch="(member={0})"
   allRolesMode="authOnly"/>
   ```
3. Restart the service:
   ```
   service bt-audit-server restart
   ```
4. You can use a Windows AD user with SSL to log in to the BlueTalon Audit Console. Example user ID: Administrator.

Use BlueTalon Consoles with Self-Signed SSL Certificates

This section describes how to use the BlueTalon Policy Console and Audit Console with self-signed SSL certificates.

Generate Certificate Request and Import Certificate

In a production environment, generate a certificate request and import the certificate:

1. Generate Certificate Signing Request (CSR).
2. Import certificate into your key store.

Example request for Cloudera:

```
/opt/bluetalon/current/shared/jre/bin/keytool -genkey -keyalg RSA -
alias btpolicyselfsigned -keystore /home/cloudera/btui.jks -validity
360 -keysize 2048
```

Use your own system-specific implementation to generate a request and import the certificate into your key store.

Edit Policy Server Configuration XML File

Edit the Policy Server configuration XML file:

1. Edit file:
   ```
   vi /opt/bluetalon/current/policy/pap/conf/server.xml
   ```
2. Add a comment around the 8111 connector to disable the connector:
   ```
   <!--
   <Connector port="8111" protocol="HTTP/1.1"
     connectionTimeout="20000"
     redirectPort="8443" />
   -->
   ```
3. Remove the comment around the 8443 connector to enable the connector:
   ```
   <Connector port="8443"
   protocol="org.apache.coyote.http11.Http11NioProtocol"
     maxThreads="150" SSLEnabled="true" scheme="https" secure="true"
     clientAuth="false" sslProtocol="TLS" />
   ```
4. Add your keystoreFile and keystorePass details to the 8443 connector. Example for Cloudera:
   ```
   <Connector port="8443"
   protocol="org.apache.coyote.http11.Http11NioProtocol"
     maxThreads="150" SSLEnabled="true" scheme="https" secure="true"
     clientAuth="false" sslProtocol="TLS"
     keystoreFile="/home/cloudera/btui.jks" keystorePass="bt#123"/>
   ```
5. Ensure port 8443 is open and accessible. If you must use different connector port, you can change the connector port in the file.
6. Save file.

Edit Policy Server and Policy Management Web XML Files

Edit files:

1. Edit file:
   ```
   vi /opt/bluetalon/current/policy/pap/webapps/BlueTalonConfig/WEB-INF/web.xml
   ```
2. Add the following user-data-constraint element shown in bold to the file:
   ```
   ...
   <web-resource-collection>
     <web-resource-name>Public Area</web-resource-name>
       <url-pattern>/resource/images/*</url-pattern>
         <url-pattern>/resource/js/*</url-pattern>
         <url-pattern>/resource/css/*</url-pattern>
     </web-resource-collection>
     <user-data-constraint>
       <transport-guarantee>CONFIDENTIAL</transport-guarantee>
     </user-data-constraint>
   </security-constraint>
   <security-constraint>
     <web-resource-collection>
       <web-resource-name>Public Area</web-resource-name>
   ...
   ```
3. Save file.
4. Edit file:
   ```
   vi /opt/bluetalon/current/policy/pap/webapps/PolicyManagement/WEB-INF/web.xml
   ```
5. Add the following user-data-constraint element shown in bold to the file:
   ```
   ...
     <auth-constraint>
       <role-name>*</role-name>
     </auth-constraint>
     <user-data-constraint>
       <transport-guarantee>CONFIDENTIAL</transport-guarantee>
     </user-data-constraint>
   </security-constraint>
   ...
   ```
6. Save file.
7. Restart service:
   ```
   service bt-policy-server restart
   ```

Open Policy Management Console

Open the Policy Management console:

1. Open Web browser.
2. Go to Policy Management console:
   ```
   https://<computer hostname or IP address>:8443
   ```

   Example:
   ```
   https://127.0.0.1:8443
   ```
3. Add a security exception for the URL. A self-signed OpenSSL certificate is used.
4. Click Advanced link.
5. Click Proceed as unsafe.
6. Examine the license agreement.

7. If you accept the license agreement, select Accept.

8. Click Policies and Auditing.

9. Log in with the user name and the password.

   Defaults:
   btadminuser
   P@ssw0rd

   See the following screenshot.

Edit Policy Server JavaScript Files

After you log in, you might see some issues with the Policy Console. This is because the application is calling the REST API methods using the HTTP protocol.

To resolve the issues, edit the Policy Server JavaScript files:

1. Change directories:
   ```
   cd
   /opt/bluetalon/current/policy/pap/webapps/BlueTalonConfig/resource/j
   s
   ```
2. Find files to edit:
   ```
   grep http:// *.js
   ```
3. For each file:
   a. Edit file.
   b. Search for http.

c. Change http to https. Example for constant.js file:
```
var HTTP = 'https://'
```

d. Save file.

4. Restart service:
```
service bt-policy-server restart
```

Add Test Data Domain

To ensure the Policy Console is operating correctly, add a new data domain:

1. In a Web browser, go to the Policy Console.
2. Click the Data Domains tab.
3. Add a test data domain.
4. Deploy the test data domain.

You can also access the REST API methods using HTTPS.

Edit Audit Server Configuration XML File

Edit the Audit Server configuration XML file:

1. Edit file:
```
vi /opt/bluetalon/current/audit/bluetalon-audit-visualize-
basic/conf/server.xml
```

2. Add a comment around the 8112 connector to disable the connector:
```
<!--
<Connector port="8112" protocol="HTTP/1.1"
  connectionTimeout="20000"
  redirectPort="8443" />
-->
```

3. Remove the comment around the 8443 connector to enable the connector:
```
<Connector port="8443"
protocol="org.apache.coyote.http11.Http11NioProtocol"
  maxThreads="150" SSLEnabled="true" scheme="https" secure="true"
  clientAuth="false" sslProtocol="TLS" />
```

4. Add your keystoreFile and keystorePass details to the 8443 connector. Example for Cloudera:
```
<Connector port="8443"
protocol="org.apache.coyote.http11.Http11NioProtocol"
  maxThreads="150" SSLEnabled="true" scheme="https" secure="true"
  clientAuth="false" sslProtocol="TLS"
  keystoreFile="/home/cloudera/btui.jks" keystorePass="bt#123"/>
```

5. Change the port from 8443 to 8444 (typically, you run the Policy Console on port 8443). If you did not install the Policy package and Audit package on the same computer, then you can use port 8443.
6. Save file.

Edit Audit Server Web XML File

Edit the Audit Server Web XML file:

1. Edit file:
   ```
   vi /opt/bluetalon/current/audit/bluetalon-audit-visualize-
   basic/webapps/BlueTalonAudit/WEB-INF/web.xml
   ```
2. Add the following user-data-constraint element shown in bold to the file:
   ```
   ...
   <web-resource-collection>
     <web-resource-name>Public Area</web-resource-name>
       <url-pattern>/resource/images/*</url-pattern>
         <url-pattern>/resource/js/*</url-pattern>
         <url-pattern>/resource/css/*</url-pattern>
     </web-resource-collection>
     <user-data-constraint>
       <transport-guarantee>CONFIDENTIAL</transport-guarantee>
     </user-data-constraint>
   </security-constraint>
   <security-constraint>
     <web-resource-collection>
       <web-resource-name>Public Area</web-resource-name>
   ...
   ```
3. Save file.
4. Restart service:
   ```
   service bt-audit-server restart
   ```

Edit Audit Server JavaScript Files

Edit the Audit Server JavaScript files:

1. Change directories:
   ```
   cd /opt/bluetalon/current/audit/bluetalon-audit-visualize-
   basic/webapps/BlueTalonAudit/resource/js
   ```
2. Find the files to edit:
   ```
   grep http:// *.js
   ```

   Do not change the select2.min.js file.
3. For each file:
   a. Edit file.
   b. Search for http.
   c. Change http to https. Example for constant.js file:
      ```
      var HTTP = 'https://'
      ```
   d. Save file.
4. Change directories:
   ```
   cd /opt/bluetalon/current/audit/bluetalon-audit-visualize-
   basic/webapps/BlueTalonAudit/jsp
   ```
5. Edit file:
   ```
   vi home.jsp
   ```
6. If necessary, add the following script elements shown in bold before the </head> element in home.jsp:
   ```
   ...
   <script src="resource/js/select2.min.js"></script>
   <script type="text/javascript" src="resource/js/jquery/jquery-
   2.1.0.min.js"></script>
   ```

```
</head>
...
```

7. Save file.
8. Change directories:
   ```
   cd /opt/bluetalon/current/audit/bluetalon-audit-visualize-basic/webapps/ROOT
   ```
9. Run:
   ```
   grep "licenseaccept" index.jsp
   ```

   Example return result showing where you must change http to https:
   ```
   var URL =
   'http://'+window.location.host+'/BlueTalonAudit/licenseaccept'
   var vData = {"action":"licenseaccept","data":value}
   ```
10. Edit file:
    ```
    vi index.jsp
    ```
11. Change http to https.
12. Save file.
13. Restart service:
    ```
    service bt-audit-server restart
    ```

Configuration is complete.

Configure Proxy Authentication

For PostgreSQL and Cassandra Enforcement Points, you can use proxy authentication to simplify user management. Proxy authentication can be enabled or disabled when you initially set up the Enforcement Point, or have already set up the Enforcement Point.

Proxy Authentication Enabled

If proxy authentication is enabled:

1. Database client application sends the user name and password for the database account to the Enforcement Point. The account must exist in the database.
2. Enforcement Point receives the user name and password and forwards them to the database.
3. Database examines the user name and password as it would if the Enforcement Point was not present. If the user name and password are valid, the connection is permitted. Otherwise, an authentication error is returned.

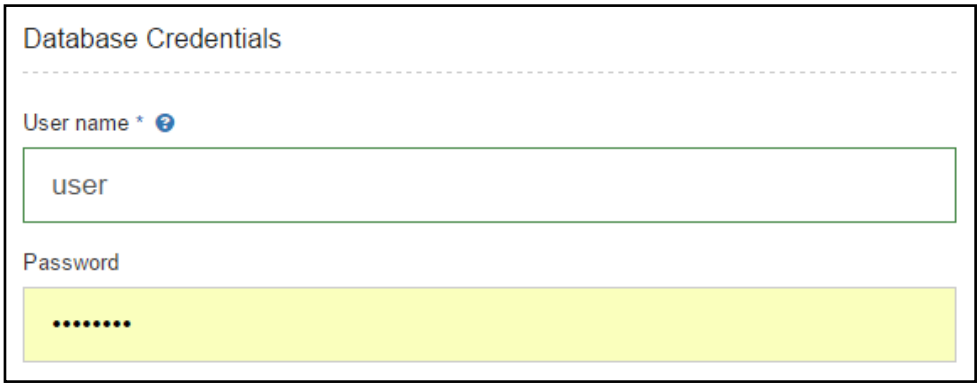
Proxy Authentication Disabled

If proxy authentication is disabled:

1. Database client application sends the user name and password for the end user account to the Enforcement Point. The account does not need to exist in the database.
2. Enforcement Point receives the user name to verify. Before the client application connects to the database, the Enforcement Point sends the user name and password to the user domain system (examples, Windows AD, LDAP) for verification.
3. If Windows AD/LDAP verifies that the user name and password are valid, then the Enforcement Point permits the connection to database to be performed with a preconfigured

account (known as the service ID account) that exists in the database. If Windows AD/LDAP returns an authentication error, then the Enforcement Point prevents the connection to the database.

You configure the service ID account that is sent to the database in the BlueTalon Policy Console when you set the user name and password for the data domain. Example:



Proxy authentication benefits:

- Database only requires the service ID account, and does not have to store end users from different applications.
- Enforcement Point integrates with Windows AD/LDAP for password management.

Configure Enforcement Point for Proxy Authentication

This section describes the different methods you can use to configure an Enforcement Point for proxy authentication. You can choose the configuration method based on your requirements.

Use REST API to Configure Proxy Authentication

To perform the configuration using the REST API, perform the following method call:

```
http://<host name or IP address of computer running Policy
Console>:8111/PolicyManagement/1.0/resource_domains/:domain/enforcement
_points
```

To turn **on** proxy authentication, set pwd_auth and proxy_auth to **yes** in the body of the API method call.

To turn **off** proxy authentication, set pwd_auth and proxy_auth to **no** in the body of the API method call.

See https://policy.readme.io/docs/enforcement_point for details on setting the body of the API call.

Use Script in Interactive Mode to Configure Proxy Authentication

To perform the configuration using the Enforcement Point script in interactive mode, run:

```
/opt/bluetalon/3.2.4/pep/scripts/bluetalon-ep-setup
```

The script prompts you to answer questions, including these:

- Enforcement point can validate a password against a User Domain or delegate authentication to the database server.
  Do you want the enforcement point to also validate the password (yes/no)? [no]

Note: This question asks if you want the Enforcement Point to take the password provided to the database client application and check the password with the user domain.

- For validation of password by the enforcement point, the user ID and password can be the values entered by the end user or can be provided at the time of defining the Data Domain as a proxy user. This user account must be present on the database also.
  Do you want to configure enforcement point to use the proxy account (yes/no)? [no]

  Note: This question asks if you want the Enforcement Point to take the user ID and password set in the data domain. A matching user account must be created in the database.

To turn **on** proxy authentication, answer **yes** to both questions.

To turn **off** proxy authentication, answer **no** to both questions. No is the default.

Use Script with XML File to Configure Proxy Authentication

To perform the configuration using a file with the Enforcement Point script:

1. Create a file named ep_conf.xml.
2. To turn on proxy authentication, set the following parameters in the file:
   ```
   <pwd_auth>yes</pwd_auth>
   <proxy_auth>yes</proxy_auth>
   ```
3. To turn off proxy authentication, set the following parameters in the file:
   ```
   <pwd_auth>no</pwd_auth>
   <proxy_auth>no</proxy_auth>
   ```
4. Save the file.
5. Run:
   ```
   /opt/bluetalon/3.2.4/pep/scripts/bluetalon-ep-setup -f ep_conf.xml
   ```

Change Proxy Authentication After Enforcement Point is Set Up

To change proxy authentication after the Enforcement Point is set up:

1. On the computer where the Enforcement Point is installed, log in using ssh with sudo privileges.
2. Edit the data domain configuration file. Example file name format for a PostgreSQL data domain:
   ```
   /opt/bluetalon/current/pep/psql-ep/conf/bt-postgresql-ep-<data
   domain name>.conf
   ```

   Example file name where the data domain is named postgresdemo:
   ```
   /opt/bluetalon/current/pep/psql-ep/conf/bt-postgresql-ep-
   postgresdemo.conf
   ```
3. To turn on proxy authentication, set the following parameters in the file:
   ```
   [BTACCOUNTDELEGATION] Value=TRUE;
   [PASSWORDAUTH] Value=TRUE;
   ```
4. To turn off proxy authentication, set the following parameters in the file:
   ```
   [BTACCOUNTDELEGATION] Value=FALSE;
   [PASSWORDAUTH] Value=FALSE;
   ```
5. Save the file.
6. Restart the Enforcement Point service. Command format for a PostgreSQL Enforcement Point service:
   ```
   service bt-postgresql-ep-<data domain name> restart
   ```

Example command:
```
service bt-postgresql-ep-postgresdemo restart
```

Configure Policy Enforcement

Each database has different settings for the Policy Engine. The Policy Engine enforces data access rules by parsing all data requests and modifying them based on user identity for policy compliance.

Example: Configure HDFS for Policy Enforcement

Configure HDFS for policy enforcement:

1. Open Web browser.
2. Go to the Policy Console.
3. Click the Data Domains tab.
4. Click Add Data Domain. Select HDFS.
5. Enter the details. To enable a search on the policy rules you add, set the search selection to ON.
6. To add a policy, select the Policy tab and click Add Policy.
7. Select the URI. Set the permissions and attach the permissions to the user. The URI has the format:
   ```
   <data domain:URI>
   ```

   Example:
   ```
   hdfsds:/user
   ```
8. Click the Deployment tab and deploy the changes.

Offline Configuration for Second Enforcement Point

You can use an offline configuration to run a second Enforcement Point for the same database.

1. For PostgreSQL, edit the file:
   ```
   /etc/bluetalon/pep/psql-ep/conf/bt-postgres-ep-<data domain name>.conf
   ```

   Example:
   ```
   /etc/bluetalon/pep/psql-ep/conf/bt-postgres-ep-postgresdemo.conf
   ```
2. Edit the parameter values. The following table shows the settings for PostgreSQL.

Parameter Name	Default Value	Modified Value
ARCDBSIGN	AUTODBSIGN_101	Database signature of the data source. Obtain the signature setting from the repository.
ONLINECONFIG	TRUE	FALSE
ARCGSIP	127.0.0.1	IP address of computer running the Policy Engine.
ARCDSIP	127.0.0.1	IP address of current system.
ARCDSPORT	1234	Port the Enforcement Point uses.

PSQLSRVIP	127.0.0.1	Target database computer IP address.
ARCDEFDB	demodb	Target database name.
PSQLSRVPORT	5432	Target database port number.
ARCDEFSCH	demodb	Target schema name.

Example: Configure Hive for Policy Enforcement

This section describes how to configure Hive to work with BlueTalon policy enforcement.

Add Data Domain, User, and Rule

Add a data domain, user, and rule:

1. Open Web browser.
2. Go to the BlueTalon Policy Console.
3. Click the Data Domains tab.
4. Click Add Data Domain and select HIVEDB.
5. Complete the details.
6. Click the User Domains tab.
7. Add a database user to the internal source or import a user from an LDAP/Kerberos source.
8. Add a rule to the accounts table.
9. Mask the soc_sec_no column.
10. Click Deployment and deploy.

Test Hive Enforcement Point (Non-Kerberos)

Test the Hive Enforcement Point (non-Kerberos):

1. Start Beeline.
2. Connect to the Hive Enforcement Point:
   ```
   !connect jdbc:hive2://<Hive Enforcement Point IP
   address>:<port>/<database name> <user name>
   ```

   Example:
   ```
   !connect jdbc:hive2://127.0.0.1:2005/default alice
   ```
3. Execute the query and ensure the soc_sec_no column is masked:
   ```
   SELECT * FROM accounts;
   ```
4. Exit:
   ```
   !quit
   ```

Test Connection to HiveServer2

The Policy Engine connects to HiveServer2 to obtain the policy.

Test the connection using Beeline on the computer where the Policy Console is installed:

1. Start Beeline.
2. Test the connection. Use the !connect command with the appropriate connection string, user name, and password.

3. Close the session with !quit.

Test Hive Binary Mode without Authentication

Test Hive running in binary mode without authentication:

```
!connect jdbc:hive2://localhost:10000/default
```

Leave the user name and password blank. Entries are ignored by HiveServer2.

Test Hive HTTP Mode without Authentication

Test Hive running in HTTP mode without authentication:

```
!connect
jdbc:hive2://localhost:10001/default;?hive.server2.transport.mode=http;
hive.server2.thrift.http.path=/cliservice
```

Leave the user name and password blank. Entries are ignored by HiveServer2.

Test LDAP Binary Mode with Authentication

Test LDAP authentication running in binary mode:

```
!connect jdbc:hive2://localhost:10000/default username password
```

Test LDAP HTTP Mode with Authentication

Test LDAP authentication running in HTTP mode:

```
!connect
jdbc:hive2://localhost:10001/default;?hive.server2.transport.mode=http;
hive.server2.thrift.http.path=/cliservice username password
```

Test Kerberos Binary Mode with Authentication

Test Kerberos authentication running in binary mode:

```
!connect jdbc:hive2://localhost:10000/default username password
```

Test Kerberos HTTP Mode with Authentication

Test Kerberos authentication running in HTTP mode:

```
!connect
jdbc:hive2://localhost:10001/default;?hive.server2.transport.mode=http;
hive.server2.thrift.http.path=/cliservice username password
```

Install FsShell Enforcement Point

Install FsShell Enforcement Point:

1. Run:
   ```
   yum install bluetalon-ep-3.2.4
   ```
2. Enter the hostname of computer where you installed the BlueTalon Audit package.
3. Select the FsShell option.
4. Follow the prompts to complete the installation.
5. Run:
   ```
   hdfs dfs -ls
   ```

You can see the log of HDFS commands in the Audit Console.

Configure Enforcement Point for Audit Only

If you configure the Enforcement Point to perform audits only using Kafka or the file system, then log in attempts and queries are not sent to the Policy Engine.

Before performing the steps in this section, you must have installed and started the Enforcement Point.

Use Hive for Audit

The Policy Engine supports implicit HDFS permissions for Hive tables.

After the BlueTalon rules are applied, and if you have already configured HDFS on your tables, then BlueTalon enforces the existing allow and deny permissions.

If there are no permissions for BlueTalon to enforce, then enforcement is performed through the existing Access Control List.

Verify Kafka Service

The Kafka service runs brokers to push audit logs from the Policy Engine or Enforcement Points.

Verify the Kafka service is running:

```
service bt-audit-kafka status
```

Verify ZooKeeper Service

ZooKeeper provides the consumer end point from which the audit activity Java daemon consumes the audit logs and sends the information to the PostgreSQL database instance.

Verify the ZooKeeper service is running:

```
service bt-audit-zookeeper status
```

Example Firewall Rule

If your organization does not allow incoming ports except 80, you can use the following example firewall rule.

Firewall Rule	Description
iptables -t nat -A PREROUTING -p tcp --dport 80 -j REDIRECT --to-port 8111	Redirect the Policy Console URL running on port 8111.

Run MapReduce Jobs with BlueTalon

Without BlueTalon, an administrator must create the HDFS directory for a user and set the ownership permissions before the user can run a MapReduce job.

Example commands that create a directory and set the ownership:

```
su -l hdfs -c 'hdfs dfs -mkdir -p /user/root'
su -l hdfs -c 'hdfs dfs -chown root /user/root'
```

With BlueTalon, an end user can create their own directory if they are a part of a user domain connected to BlueTalon because of the HDFS rules created during the BlueTalon software installation process.

Example MapReduce Jobs

Example HDP job:

```
yarn jar /usr/hdp/current/hadoop-mapreduce-client/hadoop-mapreduce-
examples.jar pi 3 4
```

Example CDH job:

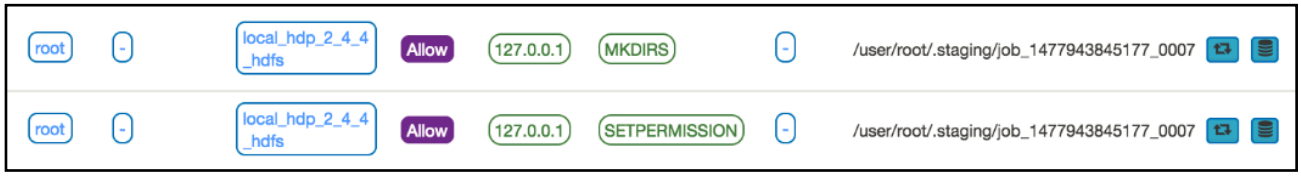
```
yarn jar /opt/cloudera/parcels/CDH/lib/hadoop-mapreduce/hadoop-
mapreduce-examples.jar pi 3 4
```

Example MapReduce Job Output

Example HDP job with output (some output omitted for brevity):

```
yarn jar /usr/hdp/current/hadoop-mapreduce-client/hadoop-mapreduce-
examples.jar pi 3 4
Number of Maps  = 3
Samples per Map = 4
Wrote input for Map #0
Wrote input for Map #1
Wrote input for Map #2
Starting Job
...
Job Finished in 47.265 seconds
Estimated value of Pi is 3.66666666666666666667
```

The following screenshot shows the job details in the Audit Console.



Example MapReduce Error Scenarios

This section describes example error scenarios when running MapReduce jobs.

Permission Denied Error

Example permission denied error:

```
org.apache.hadoop.ipc.RemoteException(org.apache.hadoop.security.Access
ControlException): Permission denied: user=root, access=GETFILESTATUS,
inode=/tmp/hive
```

If you see the previous error in the log:

1. Log in to the BlueTalon Policy Console.
2. Add the user to a user domain.
3. Add a policy.
4. Add an access rule for the HDFS file to the policy.
5. Add the user to the policy.

You can add HDFS file access rules to the HDFS global_default data domain, or create your own HDFS data domain and add a file access rule.

Java Heap Space Error

If you see the following Java heap space error in the log, then increase the memory available on the cluster:

```
INFO mapreduce.Job: Task Id : attempt_1477943845177_0007_m_000002_0,
Status : FAILED
Error: Java heap space
Container killed by the ApplicationMaster.
Container killed on request. Exit code is 143
Container exited with a non-zero exit code 143
```

Configure Hue for HDFS Enforcement Point

To configure Hue for the HDFS Enforcement Point:

1. Add the HDFS Data Domain using the BlueTalon Policy Console.
2. In a command line shell, run the following script and select FSEP:
   `/opt/bluetalon/current/pep/scripts/bluetalon-ep-setup`
3. Edit the file /etc/hue/conf/hue.ini:
   a. In the beeswax section, set the FSEP host computer name and port:
      `fs_defaultfs=webhdfs://<FSEP computer name or IP address>:<FSEP port>`
   b. Uncomment webhdfs_url and set the FSEP host computer name and port:
      `webhdfs_url=http://<FSEP computer name or IP address>:<FSEP port>/webhdfs/v1`
4. Restart the Hue service using a command line shell (do not use the Cloudera manager to start Hue because the settings will not be applied).
5. Edit the file /etc/bluetalon/pep/fs-ep/conf/fsep-site.xml. Add the following text to the configuration section:

```
<!-- Hue HttpFS proxy user setting -->
  <property>
    <name>fsep.proxyuser.hue.hosts</name>
    <value>*</value>
  </property>
  <property>
    <name>fsep.proxyuser.hue.groups</name>
    <value>*</value>
  </property>
  <property>
    <name>fsep.proxyuser.root.hosts</name>
    <value>*</value>
  </property>
  <property>
    <name>fsep.proxyuser.root.groups</name>
    <value>*</value>
  </property>
```

6. Restart the FSEP service using a command line shell:
   `service bt-fsep restart`
7. Add the following rules using the BlueTalon Policy Console:
   a. Global Default:
      i. Allow read and execute on the root folder "/".

ii. Allow read and execute on the folder "/user".

iii. Allow read and execute on the folder "/user/".

b. For each individual user:

i. Allow read, write, and execute recursive on the folder "/user/<user>".

ii. Allow read, write, and execute recursive on the folder "/user/<user>/".

Configure BlueTalon to Protect Hive Tables in Amazon EMR Cluster

This section describes how to configure BlueTalon to protect Hive tables in an Amazon Elastic MapReduce (EMR) cluster.

For full details about BlueTalon and Amazon EMR, see https://aws.amazon.com/blogs/big-data/using-bluetalon-with-amazon-emr.

Install Hive Enforcement Point

To install the Hive Enforcement Point, choose one of these methods:

- Install the Hive Enforcement Point within the Amazon EMR cluster that points to the Docker PE and AE.
- Create a container for the Hive Enforcement Point that accesses the Hive installation in the Amazon EMR cluster.

Regardless of which method you use, run these commands:

1. Create test data. See Create Test Data on page 150.
2. Add a data domain. Use Hive as your data domain type. See Add Data Domain on page 55.
3. In a command line shell, run the installation script as a privileged user:
   ```
   yum install bluetalon-ep-3.2.4
   ```
4. Run the EP set up script with an XML configuration file.

   Example command with XML file named hiveep.xml:
   ```
   /opt/bluetalon/3.2.4/pep/scripts/bluetalon-ep-setup -accept -f
   /tmp/hiveep.xml
   ```

   Example hiveep.xml file (replace items in bold with your own settings):
   ```
   <EnforcementPoint xmlns="">
          <hostname>hostname</hostname>
          <port>10000</port>
          <db_type>2</db_type>
          <mode>4</mode>
          <resource_domain_name>hivedatadomain</resource_domain_name>
          <custom_name>hivedatadomain</custom_name>
          <audit_ip>sandbox.bluetalon.com</audit_ip>
          <ep_port>2004</ep_port>
          <dbtcp_ip>sandbox.bluetalon.com</dbtcp_ip>
          <kerberos>no</kerberos>
   </EnforcementPoint>
   ```
5. Add a policy. See Add Policy on page 70.
6. Add a user to the policy. See Add User to Policy on page 73.

Test Hive Enforcement Point

To test the Hive Enforcement Point:

1. In a command line shell, run Beeline as a privileged user.
2. In Beeline, connect to the database through the Enforcement Point as the Alice user.
   Example:
   ```
   !connect jdbc:hive2://hostname:2004/hivedemodb alice mypassword
   ```
3. Run the following query:
   ```
   select * from accounts;
   ```
4. Ensure the soc_sec_no column of the accounts table is masked.

Preroute Traffic from Outside Management Node

To preroute traffic coming from outside the management node with HiveServer2 and BlueTalon, perform the steps in the following subsections.

Ensure Iptables Service is Installed

To ensure the iptables service is installed and running with default permissions, run the following command in a shell as a privileged user:

```
service iptables status
```

The following example output shows the service is installed and running. Your output will differ. If you see an error message, install and start the iptables service.

```
Table: filter
Chain INPUT (policy ACCEPT)
num  target     prot opt source               destination
1    ACCEPT     all  --  0.0.0.0/0            0.0.0.0/0           state
RELATED,ESTABLISHED
2    ACCEPT     icmp --  0.0.0.0/0            0.0.0.0/0
3    ACCEPT     all  --  0.0.0.0/0            0.0.0.0/0
4    ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0           state
NEW tcp dpt:22
5    REJECT     all  --  0.0.0.0/0            0.0.0.0/0
reject-with icmp-host-prohibited

Chain FORWARD (policy ACCEPT)
num  target     prot opt source               destination
1    REJECT     all  --  0.0.0.0/0            0.0.0.0/0
reject-with icmp-host-prohibited

Chain OUTPUT (policy ACCEPT)
num  target     prot opt source               destination

Table: nat

Chain PREROUTING (policy ACCEPT)
num  target     prot opt source               destination
1    REDIRECT   tcp  --  10.0.2.15            0.0.0.0/0           tcp
dpt:10000 redir ports 2017

Chain POSTROUTING (policy ACCEPT)
num  target     prot opt source               destination
```

```
Chain OUTPUT (policy ACCEPT)

num  target     prot opt source               destination
1    REDIRECT   tcp  --  10.0.2.15            0.0.0.0/0           tcp
dpt:10000 redir ports 2017
```

Next, add rules to route traffic.

Add Rules to Route Traffic

In this section, you add iptable rules to route HiveServer2 port traffic to the BlueTalon Hive Enforcement Point port.

If a user attempts to connect to HiveServer2 from the same computer as HiveServer2 and the Hive Enforcement Point, then they are unable to connect. The user must cancel the session and connect from another computer node.

To add rules:

1. Run the following command in a shell as a privileged user to identify the ethernet interfaces on the computer:
   ```
   ip link show
   ```
2. Set the iptable rules for each ethernet interface:
   ```
   iptables -t nat -I PREROUTING -p tcp -i eth0 --dport <port of
   HiveServer2>  -j  REDIRECT --to-ports <port of BlueTalon Hive
   Enforcement Point>
   ```

   Example:
   ```
   iptables -t nat -I PREROUTING -p tcp -i eth0 --dport 10000  -j
   REDIRECT --to-ports 10002
   ```
3. Save the rules:
   ```
   service iptables save
   ```

Traffic routing is complete.

Authenticate End User with SecureAccess

SecureAccess uses a shared secret to authenticate users wth Hadoop WebHDFS.

Notes:

- FSEP supports SecureAccess authentication to protect access to storage and compute jobs in WebHDFS.
- YARN authenticates a user with simple authentication:
  - A standard Java job client (JobClient) for YARN relies on the local shell user as the authenticated entity. You can use PAM authentication. Users cannot SU or change ENV variables.
  - A user can write a custom JobClient driver and fake an authenticated identity that YARN trusts. Also applies to Kerberos.
- All HDFS clients require a JAR file. Therefore, all HDFS clients that are not installed by a controller require a manual JAR file. This ensures untrusted HDFS clients cannot interact with the software.

Example BlueTalon SecureAccess scenario:

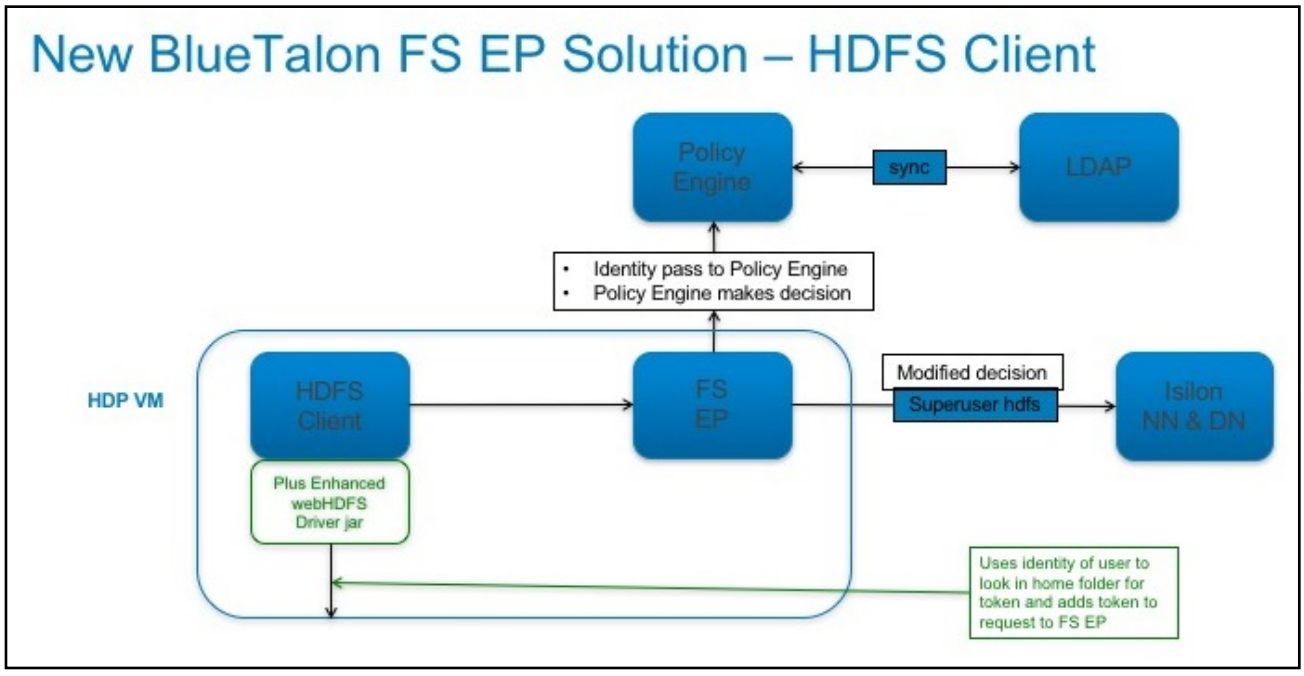
- /home/bob/.fsepUserSecret is provisioned by an administrator on the nodes.

- /home/bob/.fsepUserSecret can only be written to by the root user.
- /home/bob/.fsepUserSecret can only be read by the user (bob) and FSEP (root).
- User can log in to any computer using strong PAM authentication only.

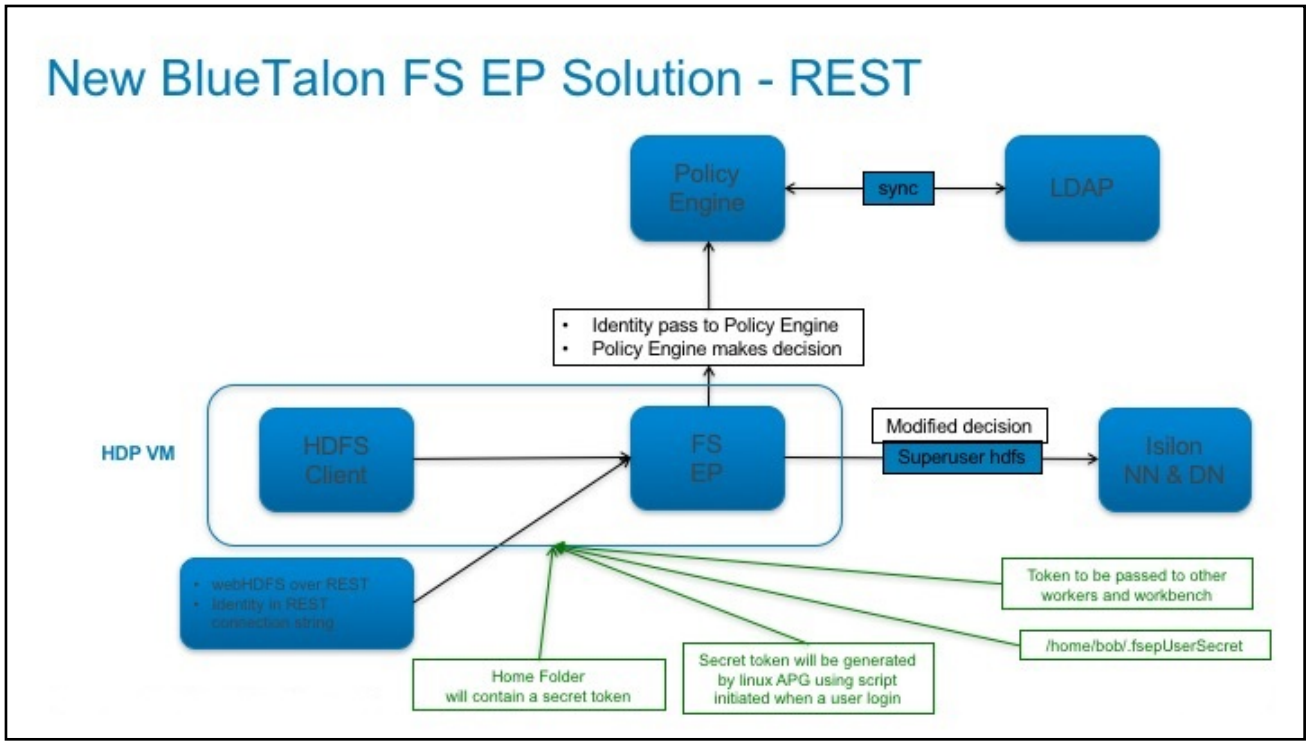
LDAP integration for management servers:

- Uses Cloudera manager LDAP integration.
- Uses Ambari LDAP integration.
- Only workspace owners can log in to the management servers.
- Only need to authorize users to start and stop services.
- No configuration changes can be performed. Therefore, a user cannot perform changes to circumvent the security controls.

The following diagram shows the BlueTalon FSEP solution with the HDFS client.



The following diagram shows the BlueTalon FSEP solution with the REST interface.



The following subsections describe how to configure and test SecureAccess.

Configure SecureAccess

To configure SecureAccess:

1. Open a command line shell and log in as a privileged user.
2. Change directories:
   `cd /opt/bluetalon/current/pep/scripts`
3. Create an XML file with the appropriate settings for installing a File System Enforcement Point (FSEP).

   Example file named fsep-silent-install-input.xml:

```
<EnforcementPoint xmlns="">
  <db_type>8</db_type>
  <resource_domain_name>hdfsds</resource_domain_name>
  <audit_ip>sandbox.hortonworks.com</audit_ip>
  <pdp_hostname>sandbox.hortonworks.com</pdp_hostname>
  <policy_conf_list>sandbox.hortonworks.com:1600</policy_conf_list>
  <hadoop_conf_path>/etc/hadoop/conf/</hadoop_conf_path>
  <traffic_divert>yes</traffic_divert>
  <kerberos>no</kerberos>
  <java_home>/usr/lib/jvm/jre-1.6.0-openjdk.x86_64</java_home>
<fsep_authentication_sharedsecret_file>/home/${USER}/.fsepSharedSecr
et</fsep_authentication_sharedsecret_file>
<fsep_authentication_sharedsecret_enabled>true</fsep_authentication_
sharedsecret_enabled>
<provision_shared_secrets>bluetalon:bluetalon123,hdfs:hdfs123,yarn:y
arn123,mapred:mapred123,hive:hive123,ambari-qa:ambari-
qa123</provision_shared_secrets>
```

```
  <post_install_conf>true</post_install_conf>
  <ambari_ip>127.0.0.1</ambari_ip>
  <ambari_port>8080</ambari_port>
  <ambari_login>admin</ambari_login>
  <ambari_passwd>admin</ambari_passwd>
</EnforcementPoint>
```

You need to set your own parameters in the file. The parameters you need to change are shown in bold in the previous example.

The policy_conf_list section enables FSEP to operate with multiple Policy Engines by obtaining the list of active Policy Engines from the Policy Configuration service. Values are a comma separated list of **host:port** settings.

The provision_shared_secrets section shows example user names and passwords for the shared secret (example user name bluetalon with a password of bluetalon123). Set your own strong passwords.

4. Run the Enforcement Point set up script in silent mode using your XML file as input.
   Example:
   ```
   ./bluetalon-ep-setup -accept -f fsep-silent-install-input.xml
   ```
5. Run the Ambari configuration script (default user name is admin with a password of admin).
   Example:
   ```
   ./bluetalon-post-install-conf-ambari admin admin --no-restart
   ```
6. Using a Web browser:
   a. Log in to the Ambari Web management console.
   b. Restart the HDFS service.
7. Using a command line shell, restart the FSEP service:
   ```
   service bt-fsep restart
   ```
8. Using the Web browser showing the Ambari Web management console, restart these services:
   a. YARN
   b. MapReduce2

You can now test SecureAccess.

Test SecureAccess

To test SecureAccess:

1. In a command line shell, switch to an account you use for loading content into HDFS.
   Example:
   ```
   su ambari-qa
   ```
2. List the contents of the user directory:
   ```
   hdfs dfs -ls /user/ambari-qa
   ```
3. Run:
   ```
   cd
   pwd
   ```

   Example output:
   ```
   /home/ambari-qa
   ```

4. Add content to a test file and add the file to HDFS. Example:
   ```
   echo hello > hello.txt
   hdfs dfs -put hello.txt /user/ambari-qa
   ```
5. Examine the contents of the test file:
   ```
   hdfs dfs -cat /user/ambari-qa/hello.txt
   ```

   Example output:
   ```
   hello
   ```
6. Examine the shared secret file:
   ```
   cat .fsepSharedSecret
   ```

   Example output:
   ```
   ambari-qa123
   ```

If the previous commands succeed, then SecureAccess is operating correctly.

Examine Shared Secret Using REST API

Example REST API command that returns the shared secret for the ambari-qa user:

```
curl -i -k -X GET -u "bluetalon:bluetalon123"
"http://sandbox.hortonworks.com:40070/webhdfs/fsepadmin/1.0/sharedsecre
t/ambari-qa"
```

Note:

- In the example, the user name is bluetalon with a password of bluetalon123.
- Set your own parameters for the computer host name, port, user name, and password.

Example output showing the shared secret:

```
ambari-qa123
```

SecureAccess configuration and testing is complete.

Monitor System

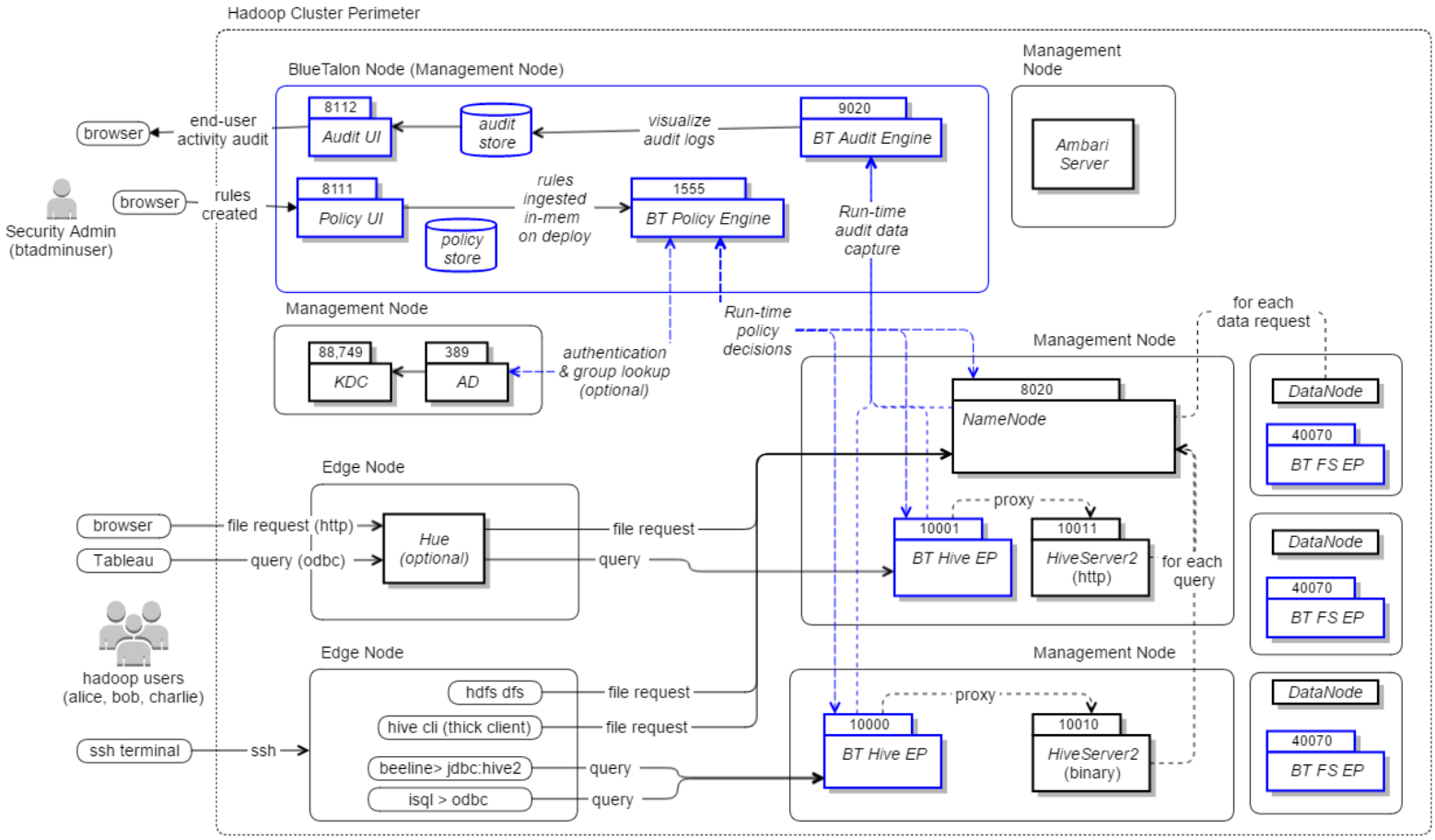
This section describes how to monitor your BlueTalon system.

Example BlueTalon System Implementation

The following table shows the components for an example BlueTalon system implementation.

BlueTalon Component	Purpose	Computer and Port
Policy Engine	Provides policy decisions to Enforcement Points at run-time.	<BlueTalon computer>:1555 For C++ process
Audit Engine	Collects audit data from Enforcement Points at run-time.	<BlueTalon computer>:9093 For Kafka <BlueTalon computer>:2182 For ZooKeeper
Hive Enforcement Point	Intercepts requests to enforce policy decisions and capture audit information for Hive queries.	<Hive Server 2 Node>:10000 For JDBC end point
HDFS Enforcement Point	Intercepts requests to enforce policy decisions and capture audit information for file requests. Runs on each worker node and reads data from local HDFS or Isilon implementation.	localhost:40070 For HDFS/WebHDFS end point
Policy Console	Web application for security administrators to create rules in BlueTalon.	<BlueTalon computer>:8111 For Tomcat Web application
Audit Console	Web application for security administrators to visualize audit data in BlueTalon.	<BlueTalon computer>:8112 For Tomcat Web application
Policy Store database	PostgreSQL database for storing rules.	<BlueTalon computer>:5433 For PostgreSQL database
Audit Store database	PostgreSQL database for storing audit details.	<BlueTalon computer>:5434 For PostgreSQL database

The following diagram shows an example system implementation.



The following sections describe how to monitor the system components.

Ensure Policy Console is Running

To ensure the Policy Console is running:

1. Open the Policy Console URL in a Web browser:
   `http://<fully qualified domain name or IP address of computer running Policy Console:port>/BlueTalonConfig`

   Examples:
   `http://TestServer.TestCompany.com:8111/BlueTalonConfig`
   `http://127.0.0.1:8111/BlueTalonConfig`
2. Log in as btadminuser.

You will see the Policy Console.

Ensure Policy Service is Running

To ensure the policy service is running, you run a command from a:

- Local computer on which the policy service was installed.
- Remote computer with HTTP access to the Policy Console.

From a local computer:

1. Run the following command:
   ```
   service bt-policy-server status
   ```
2. The process ID is returned. Example:
   ```
   bt-policy-server (pid   618 ) is running
   ```

From a remote computer:

1. Run the following command:
   ```
   curl -u <user name>:<password> --request GET  http: //<hostname or
   IP address of computer running Policy
   Console>:8111/PolicyManagement/1.0/version
   ```
2. The software version is returned. Example:
   ```
   [
        {
            "api_version" :  "1.0" ,
            "build_no" :  "32" ,
            "build_version" :  "3.2.4" ,
            "company_name" :  "BlueTalon, Inc" ,
            "company_website" :  "www.bluetalon.com" ,
            "revision_no" :  "2650" ,
            "schema_revision" :  "50" ,
            "update_date" :  "02/11/17"
        }
   ]
   ```

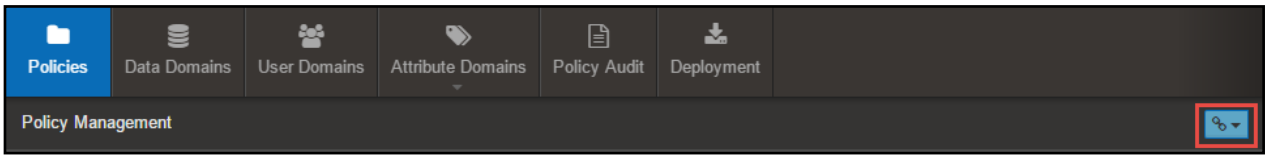
Ensure Policy Engine is Running

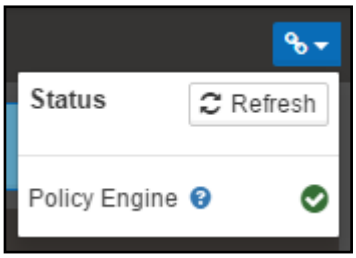
To ensure the Policy Engine is running, you:

- Use the Policy Console.
- Run a command from a local computer on which the Policy Engine service was installed.

Use the Policy Console:

1. Click the Policy Engine Status button on the right of the screen.





2. Click Refresh. If the Policy Engine is running, you will see a green icon (see bottom right in previous screenshot for example icon).

From a local computer:

1. Run the following command:
   ```
   service bt-policy-engine status
   ```
2. The process ID is returned. Example:
   ```
   bt-policy-engine (pid   614 ) is running
   ```

Ensure Audit Console is Running

To ensure the Audit Console is running:

1. Open the Audit Console URL in a Web browser:
   ```
   http://<fully qualified domain name or IP address running Audit
   Console:port>/BlueTalonAudit
   ```

   Examples:
   ```
   http://TestServer.TestCompany.com:8112/BlueTalonAudit
   http://127.0.0.1:8112/BlueTalonAudit
   ```
2. Log in as btadminuser.

You will see the Audit Console.

Ensure Audit Service is Running

To ensure the audit service is running:

1. Run the following command from the computer on which the service was installed:
   ```
   service bt-audit-server status
   ```
2. The process ID is returned. Example:
   ```
   bt-audit-server (pid   467 ) is running
   ```

Ensure Policy Store Database Service is Running

To ensure the policy store database service is running:

1. Run the following command from the computer on which the service was installed:
   ```
   service bt-policy-database status
   ```
2. The process ID is returned. Example:
   ```
   bt-policy-database (pid   431 ) is running
   ```

Ensure Audit Store Database Service is Running

To ensure the audit store database service is running:

1. Run the following command from the computer on which the service was installed:
   ```
   service bt-audit-database status
   ```
2. The process ID is returned. Example:
   ```
   bt-audit-database (pid   439 ) is running
   ```

Ensure Hive Enforcement Point Service is Running

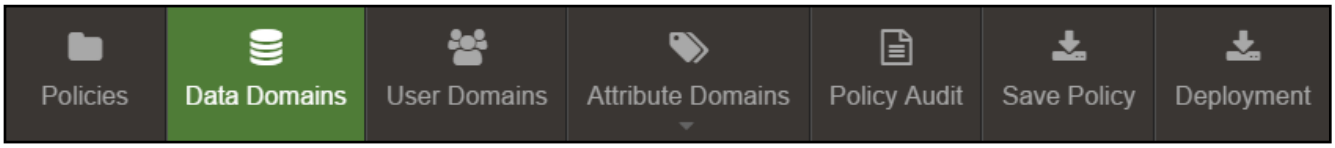
To ensure the Hive Enforcement Point service is running:

- Use the Policy Console to obtain the Hive data domain name.

- Run a command from a local computer on which the Hive Enforcement Point service was installed to examine the service status.

Obtain the Hive data domain name:

1. In the Policy Console, click the Data Domains tab.



2. Examine the data domains and obtain the name for your Hive data domain. Example: HIVEDS.

To ensure the Hive Enforcement Point service is running:

1. Run the following command from the computer on which the service was installed:
   `service bt-hive-ep-HIVEDS status`
   The service name includes the Hive data domain name: bt-hive-ep-<data domain name>.
2. The process ID is returned. Example:
   `bt-hive-ep-HIVEDS (pid  523 ) is running`

Ensure HDFS Enforcement Point Service is Running

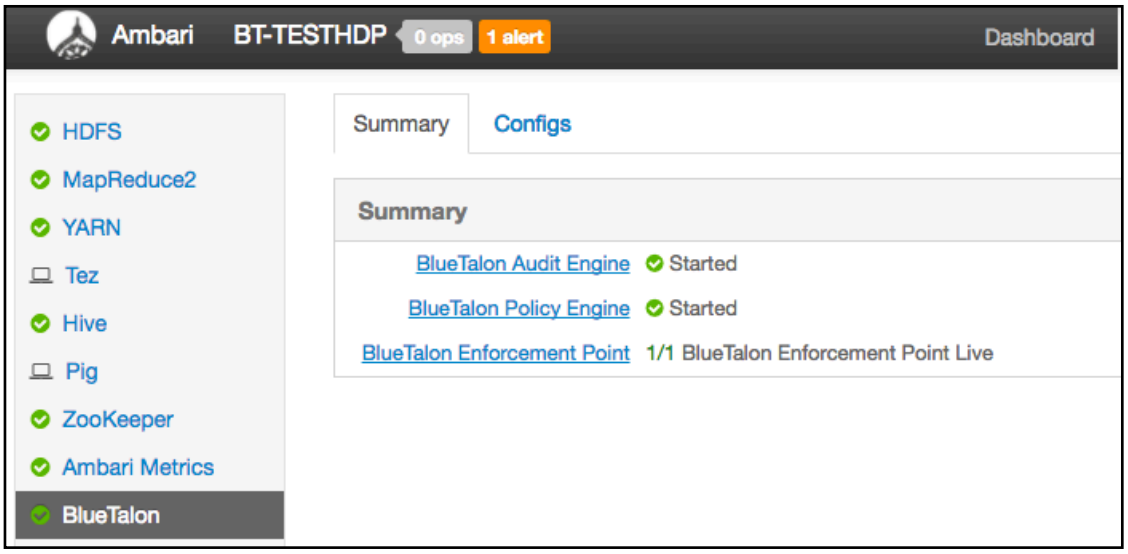
To ensure the HDFS Enforcement Point service is running:

- Use the Ambari Web management console to ensure the HDFS Enforcement Point and other BlueTalon services are running.
- Run a command from a local computer on which the HDFS Enforcement Point service was installed to examine the service status.

Use the Ambari Web management console:

1. Open a Web browser and go to the Ambari Web management console.

2. Click BlueTalon.



3. Ensure the BlueTalon Enforcement Point and other BlueTalon services are running.

To check the FSEP service status from a local computer:

1. Run the following command:
   `service bt-fsep status`
2. The process ID is returned. Example:
   `bt-fsep (pid  597 ) is running`

Ensure Enforcement Point Service is Running

To ensure Enforcement Point service is running:

1. Open a command line shell and log in as a privileged user.
2. Run:
   `service --status-all | grep bt`
3. Examine the status for your Enforcement Point. The service name format for a PostgreSQL Enforcement Point is:
   `bt-postgresql-ep-<data domain name>`
4. Example service status result for a data domain named postgresdemo:
   `bt-postgresql-ep-postgresdemo (pid 3960) is running...`
5. Note the process ID shown as "pid" in the previous step. Example pid: 3960.
6. If the service is not running, run:
   `service bt-postgresql-ep-<data domain name> start`
7. Examine the network statistics and search for the process ID:
   `netstat -plnt | grep 3960`
8. View the IP address and port in the results from the previous step. Example:
   `tcp 0 0 10.0.1.7:1557 0.0.0.0:* LISTEN 3960/arcdspsql`
9. Note the IP address and port in the previous step. Examples:
   IP address: `10.0.1.7`
   Port: `1557`

Change Policy Engine Log Level

The Policy Engine log file is:

```
/var/log/bluetalon/policy/pdp/logs/bt-policy-engine.log
```

The Policy Engine configuration file is:

```
/opt/bluetalon/current/policy/pdp/conf/bt-policy-engine.conf
```

To change Policy Engine log level:

1. Edit the Policy Engine configuration file.
2. Change the ARCGSMODE parameter. The following table shows the parameter values you can use. ERRORLOG generates the least amount of information. DEBUG generates the most.

Value	Description
ERRORLOG	Error messages only.
NORMAL	For typical daily use. The default.
DEBUG	Maximum information.

3. Restart the bt-policy-engine service:
   ```
   service bt-policy-engine restart
   ```

Troubleshoot System

This section describes how to troubleshoot your BlueTalon system.

Enable Policy Engine and Enforcement Point Debug Logs

This section describes how to enable debug logs for the Policy Engine and Enforcement Points.

Policy Engine Debug Logs

To enable debug logs for the Policy Engine:

1. Edit file:
   `/etc/bluetalon/policy/pdp/conf/log4j.xml`
2. Change the value of the priority parameter as decribed in the following table.

Parameter Example	Description	Value	Use	Log File Path
<priority value="all" />	Policy Engine log level.	error	Error messages (default value set at installation).	/var/log/bluetalon/policy/pdp/logs/bt-policy-engine.log Log file contains time-stamp based messages that can be used to calculate the time differences between the steps.
		info	Normal use.	
		all	Maximum information. **Set this value when debugging.**	

3. Edit file:
   `/etc/bluetalon/policy/pdp/conf/bt-policy-engine.conf`
4. Change the value of the ARCDEBUGLOG parameter as decribed in the following table.

Parameter Example	Description	Value	Use	Log File Path
[ARCDEBUGLOG] Value=YES;	Enables logging. Logs are used for debugging: • Policy issues. • Syntax error issues generated by the Policy Engine.	Value=NO;	Normal use (default value set at installation).	Logs files are stored in the the path specified by the following parameter: [DEBUGPATH] Default value: /opt/bluetalon/current/policy/pdp/logs/arcquerylog A separate log file is created for each SQL database.
		Value=YES;	Enable log. **Set this value when debugging.**	

5. Restart the bt-policy-engine service:
   `service bt-policy-engine restart`

Enforcement Point Debug Logs

To enable debug logs for an Enforcement Point:

1. Edit appropriate Enforcement Point file (use your Enforcement Point directory, type, and data domain):
   `/etc/bluetalon/pep/<EP directory>/conf/bt-<EP type>-ep-<data domain name>-log4j.xml`
2. Change the value of the priority parameter as described in the following table.

Parameter Example	Description	Value	Use	Log File Path
<priority value="all" />	Policy Engine log level.	error	Error messages (default value set at installation).	/var/log/bluetalon/pep/<EP directory>/conf/bt-<EP type>-ep-<data domain name>.log Log file contains time-stamped messages that can be used to calculate the time differences between the steps.
		info	Normal use.	
		all	Maximum information. **Set this value when debugging.**	

3. Edit appropriate Enforcement Point service file (use your Enforcement Point directory, type, and data domain):
   `/etc/bluetalon/pep/<EP directory>/conf/bt-<EP type>-ep-<data domain name>.conf`
4. Change the value of the PACKETLOG parameter as described in the following table.

Parameter Example	Description	Value	Use	Log File Path
[PACKETLOG] Value=TRUE;	Controls logging of packets sent between the client and the database.	Value=FALSE;	Normal use (default value set at installation).	Log file location: /var/log/bluetalon/pep/<EP directory>/logs/ Each client database session creates a separate log file. Log file contains time-stamp based messages that can be used to calculate the time differences between the steps. For PostgreSQL EP, you must specify the PACKETLOGPATH for generating packet logs. Set the parameter to the log file path. Example: [PACKETLOGPATH] Value=/var/log/bluetalon/pep/<EP directory>/logs/
		Value=TRUE;	Enable packet log. **Set this value when debugging.**	

5. Restart the appropriate Enforcement Point service (use your Enforcement Point type and data domain):
   `service bt-<EP type>-ep-<data domain name> restart`

Concurrency Issue

If multiple users attempt to deploy a configuration state or access an Enforcement Point at the same time, you might see the following error message in the Policy Engine log file:

```
[ERROR] [ARCQuery] [WriteActivityLog] [0] Failed to write activity log
on Kafka Cluster
```

To resolve this issue:

1. Increase ulimit (maximum can be 100000). Example:
   ```
   ulimit -n 64536
   ```
2. Reuse TCP sockets. Example:
   ```
   echo "1" >/proc/sys/net/ipv4/tcp_tw_reuse
   echo "1" >/proc/sys/net/ipv4/tcp_tw_recycle
   ```

Snapshot Deployment Log File Extract

The following listing shows a log file extract for a snapshot deployment. The elapsed time for the snapshot deployment is shown.

```
|  2017-07-21 06:41:56.562-+0000 | INFO |
BTPolicySDK.com.mapper.publish.PublishObject |
doPublishOnServerRepository() |  DEPLOYMENT | Publish on server
repository starts for . Deployment : 1 |

|  2017-07-21 06:41:56.570-+0000 | DEBUG |
BTPolicySDK.com.mapper.publish.PublishObject |
doPublishOnServerRepository() |  DEPLOYMENT | Connected to server
repository. Elapsed time [7] in millisecond(s). Deployment : 1 |

|  2017-07-21 06:41:56.571-+0000 | DEBUG |
BTPolicySDK.com.mapper.publish.PublishObject |
doPublishOnServerRepository() |  DEPLOYMENT | Exporting the data from
server repository to file
[/opt/bluetalon/current/policy/pap///logs/backup/autoexport/BSR_snapsho
tNo121.varc]. Deployment : 1 |
```

Manually Set IP Addresses

Occasionally, the Policy Engine does not automatically detect computer IP addresses. If the IP addresses are not automatically detected, then you perform the steps in this section.

Set IP Address for Policy Engine Service

Set the IP address for the Policy Engine service:

1. Delete the database entry. Example:
   ```
   psql -h 127.0.0.1 -p 5433 -U btuser -d btrepo
   password : bt#123
   DELETE FROM arcservers WHERE servertype='ARCMUX';
   ```
2. Edit the file:
   ```
   cd /etc/bluetalon/policy/pdp/conf
   vi bt-policy-engine.conf
   ```
3. Change the following parameter from Auto to the computer IP address:
   ```
   [ARCGSIP]
   Value=<IP address>;
   ```
4. Restart the Policy Engine service:
   ```
   service bt-policy-engine restart
   ```

Change Port for PostgreSQL Enforcement Point Service

To change the port for the PostgreSQL Enforcement Point service:

1. Switch user:
   ```
   su bluetalon
   ```
2. Run:
   ```
   source /etc/profile
   ```
3. Change directories:
   ```
   cd /etc/bluetalon/pep/psql-ep/conf
   ```
4. The configuration is stored in a file with a name in the format:
   ```
   bt-postgresql-ep-<data domain name>.conf
   ```

   Edit the file. Example:
   ```
   vi bt-postgresql-ep-postgresdemo.conf
   ```
5. Change the following parameter to the same port as the PostgreSQL data domain. Example:
   ```
   [ARCDSPORT]
   Value=65535;
   ```
6. The service name has the format:
   ```
   bt-postgresql-ep-<data domain name>
   ```

   Restart the service. Example:
   ```
   service bt-postgresql-ep-postgresdemo restart
   ```

Set IP Address for Other Enforcement Points

If you created other data domains using the Policy Console, then you set the IP addresses for those Enforcement Point computers. Example data domains shown in this section: Hive, Impala, and Oracle.

Set IP Address for Hive

Set the IP address for Hive:

1. Edit the file:
   ```
   cd /etc/bluetalon/pep/hive-ep/conf
   vi bt-hive-ep-<data domain name>.conf
   ```
2. Change the following parameter from Auto to the computer IP address:
   ```
   [SECDBTCPIP]
   Value=<IP address>;
   ```
3. Restart the service:
   ```
   service bt-hive-ep-<data domain name> restart
   ```

Set IP Address for Impala

Set the IP address for Impala:

1. Edit the file:
   ```
   cd /etc/bluetalon/pep/hive-ep/conf
   vi bt-impala-ep-<data domain name>.conf
   ```

   Note: Impala file is stored in the Hive directory.
2. Change the following parameter from Auto to the computer IP address:
   ```
   [SECDBTCPIP]
   Value=<IP address>;
   ```
3. Restart the service:
   ```
   service bt-impala-ep-<data domain name> restart
   ```

Set IP Address for Oracle

Set the IP address for Oracle:

1. Edit the file:
   ```
   cd /etc/bluetalon/pep/oracle-ep/conf
   vi bt-oracle-ep-<data domain name>.conf
   ```
2. Change the following parameter from Auto to the computer IP address:
   ```
   [SECDBTCPIP]
   Value=<IP address>;
   ```
3. Restart the service:
   ```
   service bt-oracle-ep-<data domain name> restart
   ```

Data Domain Template Files

When you create a new data domain in the Policy Console, you also create an associated configuration file (.conf extension) from the template file (.template extension). The template file has an Auto parameter to assign IP addresses.

If you change Auto to an IP address in the template file, then the associated configuration file also has that IP address.

Set IP Address in Hadoop Template

Set the IP address in the Hadoop template:

1. Edit the file:
   ```
   cd /opt/bluetalon/3.2.4/pep/hive-ep/conf.template
   vi bt-hadoop-ep.conf.template
   ```
2. Change the parameter:
   ```
   [SECDBTCPIP]
   Value=<IP address>;
   ```

Set IP Address in Oracle Template

Set the IP address in the Oracle template:

1. Edit the file:
   ```
   cd /opt/bluetalon/3.2.4/pep/oracle-ep/conf.template
   vi bt-oracle-ep.conf.template
   ```
2. Change the parameter:
   ```
   [SECDBTCPIP]
   Value=<IP address>;
   ```

Configure Stack Size

If any of the following services terminate with an error then a stack core dump is generated:

- Policy Engine service.
- Enforcement Point services. Examples: Hive, Spark, Impala, PostgreSQL.

You should run the following command to enable core dumps of any size:

```
ulimit -c unlimited
```

The core dumps are used to examine the failure reason.

Services Installed by Audit and Policy Packages

The following subsections describe the services installed by the Audit and Policy packages.

Services Installed by Audit Package

The following table describes the services installed by the Audit package.

Service	Description
bt-audit-database	PostgreSQL database service that stores the audit logs.
bt-audit-server	Audit user interface service that runs on Tomcat.
bt-audit-kafka	Kafka broker service that receives audit entries from the Enforcement Points.
bt-audit-zookeeper	ZooKeeper service that locates the Kafka broker service.
bt-audit-activity-monitor	ETL service that reads data from Kafka, transforms, and then loads the data into the database for the user interface.

Services Installed by Policy Package

The following table describes the services installed by the Policy package.

Service	Description
bt-policy-database	PostgreSQL database service that stores the user interface database, along with the Policy Engine database that contains the configuration and rules.
bt-policy-server	Policy user interface service that runs on Tomcat that is connected to policy-database.
bt-policy-configuration	C++ binary service that reads from the policy database and enables the Policy Engine to allow the Enforcement Points self-configure. The protocol is BlueTalon specific.
bt-policy-engine	C++ binary service that reads from the policy database and responds to decision requests from the Enforcement Points.

Audit Services

This section describes troubleshooting for the Audit services.

Audit Database Service

Service name:

bt-audit-database

Configuration file location:

/opt/bluetalon/3.2.4/audit/bluetalon-audit-store-psql/data/postgresql.conf

Log file location:

/opt/bluetalon/3.2.4/audit/logs

If the service is not started:

1. Examine the latest log file.

2. Decrease the shared buffer size. Edit the shared_buffers parameter in the postgresql.conf file.

Audit Monitor Service

Service name:

bt-audit-monitor

Configuration file location:

/opt/bluetalon/3.2.4/audit/conf/bluetalon-audit.xml

Log file:

/opt/bluetalon/3.2.4/audit/logs/bt-audit-database.log

If the service is not started:

1. Examine the log file.
2. Examine the status of the following services:
    a. bt-audit-database
    b. bt-audit-kafka
    c. bt-audit-zookeeper

Audit Kafka Service

Service name:

bt-audit-kafka

Configuration file:

/opt/bluetalon/3.2.4/audit/bluetalon-audit-aggregate/config/server.properties

Log files:

/opt/bluetalon/3.2.4/audit/bluetalon-audit-aggregate/logs/kafkaServer.log

/opt/bluetalon/3.2.4/audit/bluetalon-audit-aggregate/logs/kafka-request.log

/opt/bluetalon/3.2.4/audit/bluetalon-audit-aggregate/logs/server.log

If the service is not started:

1. Examine the log file.
2. Examine the status of the service.
3. Open two command line shells.
4. In shell 1, write to Kafka:
   ```
   /opt/bluetalon/3.2.4/audit/bluetalon-audit-aggregate/bin/kafka-console-producer.sh --broker-list localhost:9093 --topic ActivityMonitor
   ```
5. In shell 2, read from Kafka:
   ```
   /opt/bluetalon/3.2.4/audit/bluetalon-audit-aggregate/bin/kafka-console-consumer.sh --zookeeper localhost:2182 --topic ActivityMonitor --from-beginning
   ```
6. In shell 1, enter the following line (update the timestamp to the current date and time on your computer):
   ```
   {"LoggedUser":"Test", "GroupName":"Test", "OrignalQuery":"Test",
   ```

```
"FinalQuery":"Test", "Timestamp":"2016-01-20|14:15:45.000",
"DataBase":"-", "Schema":"-", "Client":"-", "PolicySet":"Test",
"SessionID":"-", "UniqueID":"-", "Action":"Test",
"ClientIp":"1.1.1.1", "AuditParams":"-", "Effect":"Authorized",
"ColumnList":"-" }
```

7. In the Web browser, go to the Audit Console. View the timestamp entry in the User Audit tab.

Audit ZooKeeper Service

Service name:

bt-audit-zookeeper

Configuration file:

/opt/bluetalon/3.2.4/audit/bluetalon-audit-aggregate/config/zookeeper.properties

Log file:

/opt/bluetalon/3.2.4/audit/bluetalon-audit-aggregate/logs/zookeeper.out

If the service is not started, examine the log file.

Audit Server

Service name:

bt-audit-server

Configuration file:

/opt/bluetalon/3.2.4/audit/bluetalon-audit-visualize-basic/conf/BTAuditConnInfo.xml

Log file location:

/opt/bluetalon/3.2.4/audit/bluetalon-audit-visualize-basic/logs

If the service is not started:

1. Examine the log file.
2. Examine the available memory.
3. Examine the status of the bt-audit-database service.

Audit Records Not Shown in Audit Console

If audit records are not shown in the Audit Console:

1. Examine the bt-audit-monitor service status:
   ```
   service bt-audit-monitor status
   ```
2. Restart the bt-audit-monitor service if it is not running:
   ```
   service bt-audit-monitor restart
   ```
3. Restart other services:
   ```
   service bt-audit-server restart
   service bt-audit-kafka restart
   service bt-audit-zookeeper restart
   service bt-audit-database restart
   ```
4. Perform some SQL queries that exercise BlueTalon security rules. The queries should generate audit records.

5. Ensure the following directory contains recent audit log files:
   `cd /opt/bluetalon/3.2.4/audit/blueatlon-audit-aggregate/logs`
6. Examine the Audit Console to ensure audit records are shown.

Policy Services

This section describes troubleshooting for the Policy services.

PostgreSQL Service

Service name:

bt-postgresql-ep-<data domain name>

Configuration file location:

/opt/bluetalon/3.2.4/pep/psql-ep/conf/bt-postgresql-ep-<data domain name>.conf

Log file location:

/var/log/bluetalon/pep/psql-ep/logs

If service is not started:

1. Examine the latest log file.
2. Decrease the shared buffer size. Edit the shared_buffers parameter in the bt-postgresql-ep-<data domain name>.conf file.

Policy Engine Service

Service name:

bt-policy-engine

Configuration file:

/etc/bluetalon/policy/pdp/conf/bt-policy-engine.conf

Log file:

/var/log/bluetalon/policy/pdp/logs/bt-policy-engine.log

If the service is not started:

1. Examine the log file.
2. Examine the status of the bt-postgresql-ep-<data domain name> service.

Policy Server Service

Service name:

bt-policy-server

Configuration file:

/opt/bluetalon/3.2.4/policy/pap/webapps/BlueTalonConfig/config/BTConnectionList.xml

Log file:

/opt/bluetalon/3.2.4/policy/pap/logs/bt-policy-server.log

If the service is not started:

1. Examine the log file.

2. Examine the available memory.
3. Examine the status of the bt-postgresql-ep-<data domain name> service.

Service Configuration Files

The following table shows the service configuration files and their locations.

Service	Configuration File
Kafka	/opt/bluetalon/3.2.4/audit/bluetalon-audit-aggregate/config/server.properties
ZooKeeper	/opt/bluetalon/3.2.4/audit/bluetalon-audit-aggregate/config/zookeeper.properties
Spark EP	/etc/bluetalon/pep/sparksql-ep/conf/bt-sparksql-ep-<data domain name>.conf
Hive EP	/etc/bluetalon/pep/hive-ep/conf/bt-hive-ep-<data domain name>.conf
FSEP	/etc/bluetalon/pep/fs-ep/conf/fsep-log4j.properties
Policy Engine	/etc/bluetalon/policy/pdp/conf/bt-policy-engine.conf
Policy Config	/etc/bluetalon/policy/pap/conf/bt-policy-configuration.conf

Enforcement Point Services

Configuration file:

/etc/bluetalon/pep/<EP Package Name>/conf/bt-<EP name>-ep-<data domain name>.conf

Example for PostgreSQL Enforcement Point:

/etc/bluetalon/pep/psql-ep/conf/bt-postgresql-ep-postgresdemo.conf

Log file:

/var/log/bluetalon/pep/<EP Package Name>/logs/bt-<EP name>-ep-<data domain name>.log

Example for PostgreSQL Enforcement Point:

/var/log/bluetalon/pep/psql-ep/logs/bt-postgresql-ep-postgresdemo.log

<Data Domain Name> is the name of data domain set in the Policy Console.

The following table shows the details for the other Enforcement Point variables.

Database Type	EP Package Name	EP Name
PostgreSQL	psql-ep	postgresql
Hive	hive-ep	hive
Impala	impala-ep	impala
Oracle	oracle-ep	oracle
DB2	db2-ep	db2
Fsshell	fsshell	fsshell

HDFS	hdfs-ep	hdfs
FSEP	fs-ep	fsep
Redshift	redshift-ep	redshift
Greenplum	greenplum-ep	greenplum
MySQL	mysql-ep	mysql
MSSQL	mssql-ep	mssql
Cassandra	cs-ep	cs
Spark	sparksql-ep	sparksql
Custom	custom-ep	custom

Start and Stop All BlueTalon Services

To start all BlueTalon services, run the following command as a privileged user:

```
/opt/bluetalon/current/shared/scripts/bt-services -p -all -start
```

To stop all BlueTalon services, run the following command as a privileged user:

```
/opt/bluetalon/current/shared/scripts/bt-services -p -all -stop
```

PostgreSQL Shared Memory

If the PostgreSQL pgstartup.log file has these messages:

```
FATAL: could not create shared memory segment: Invalid argument
DETAIL: Failed system call was shmget(key=5433001, size=74342400,
03600).
```

The error indicates that the PostgreSQL request for a shared memory segment exceeded the operating system kernel SHMMAX parameter.

To resolve the error, you can perform one or both of the following tasks:

- Reduce the PostgreSQL shared memory request size. In the example messages, the request size is 74342400 bytes.
- Reconfigure the operating system kernel with a larger SHMMAX value.

Reduce PostgreSQL Shared Memory Request Size

The PostgreSQL configuration is stored in the following files:

- /opt/bluetalon/3.2.4/policy/pap/dep/pgsql/data/postgresql.conf
- /opt/bluetalon/data/audit/data/postgresql.conf
- /opt/bluetalon/data/pap/data/postgresql.conf
- /var/lib/pgsql/data/postgresql.conf

To reduce the PostgreSQL shared memory request size:

1. Reduce the PostgreSQL shared memory usage. Example: Reduce the shared_buffers or max_connections parameters. Change these settings in the postgresql.conf files:
    a. Change shared_buffers from 100MB to 25MB.
    b. Change max_connections from 250 to 50.
2. Restart the postgresql service.

If the shared memory request size is already small, it is possible that the request is less than the kernel SHMMIN parameter:

1. Raise the request size or lower the SHMMIN parameter.
2. Restart the system.

Examine the PostgreSQL documentation for information about the shared memory configuration.

Reconfigure Kernel with Larger SHMMAX Value

To reconfigure the kernel with a larger SHMMAX value:

1. Examine the current kernel SHMMAX value:
```
cat /proc/sys/kernel/shmmax
33554432
```
2. Set SHMMAX to a larger value.
3. Restart the system.

FSEP and HDFS Commands Issue

This issue affects BlueTalon version 2.8 and below. It does not affect version 2.9.1 and higher.

The core-site.xml file:

- Contains the configuration settings for Hadoop Core.
- Sets where the NameNode runs in the cluster.

If the core-site.xml file is changed to use the FSEP URL of webhdfs:40070 (a self-reference), then the FSEP service will start but HDFS commands will halt and never complete.

To resolve the issue:

1. Edit the core-site.xml file. Set the parameter Fs.DefaultFS to the correct NameNode value.
2. Restart the FSEP service:
```
service bt-fsep restart
```
3. After the service has restarted, edit the core-site.xml file. Set the parameter Fs.DefaultFS to the localhost value.

FSEP and HDFS Polling Interval

The polling interval is the time period that FSEP examines the Policy Engine for policy changes made after the last polling interval.

The polling interval property is stored in the following file:

`/etc/bluetalon/pep/fs-ep/conf/bt-hdfs-site.xml`

You can edit the file and change the polling interval value. Example:

```
<property>
  <name>bt.hdfs.plugin.policy.redeploy.polling.interval.secs</name>
  <value>300</value>
  <description>HDFS plug in polling interval to detect whether policies
have been redeployed, default is 300 seconds</description>
</property>
```

If a policy change is detected:

- Locally cached policies are removed.
- Policies are fetched again based on resource access.

Guidelines for setting the polling interval value:

- In a production environment:
    - You set the polling interval based on how frequently policies are changed and the acceptable delay between policy deployment and the policy being effective at the Enforcement Point.
    - Setting a small polling interval causes policies to become effective sooner, but polling is performed more frequently and reduces performance.
- In a test environment:
    - You set the polling interval to a small value.
    - Example: Ten seconds. This means you wait ten seconds to test policy changes.

Do not set the polling interval to less than five seconds. Polling reduces performance and can cause FSEP to become unresponsive.

Restart FSEP Service Issue

If you need to restart the bt-fsep service multiple times, add sleep statements. Example:

```
service bt-fsep restart; sleep 20; service bt-fsep stop; sleep 20;
service bt-fsep restart
```

You must add sleep statements to allow Tomcat to start and accept connections.

If you do not add sleep statements, you will see an error. Example:

```
service bt-fsep restart; service bt-fsep stop; service bt-fsep restart
Stopping bt-fsep service: [ OK ]
Starting bt-fsep service: [ OK ]
Stopping bt-fsep service: Dec 21, 2016 9:28:03 PM
org.apache.catalina.startup.Catalina stopServer
SEVERE: Could not contact localhost:40071. Tomcat may not be running.
Dec 21, 2016 9:28:03 PM org.apache.catalina.startup.Catalina stopServer
SEVERE: Catalina.stop:
java.net.ConnectException: Connection refused
```

MapReduce Connection Issue

MapReduce is a programming model for processing and generating large data sets with a parallel distributed algorithm on a cluster.

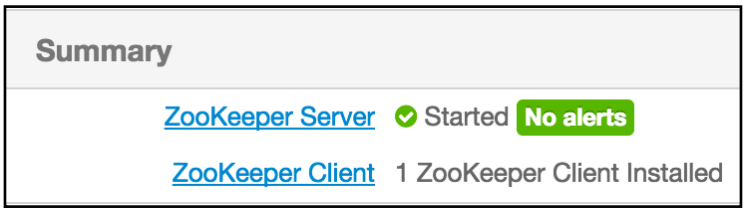
When running a MapReduce job, if you see an error "ipc.Client: Retrying connect to server" with an end point at port 8050, then restart the ZooKeeper service.

Example error scenario:

```
yarn jar /usr/hdp/current/hadoop-mapreduce-client/hadoop-mapreduce-
examples.jar pi 3 4

Number of Maps  = 3

Samples per Map = 4

Wrote input for Map #0

Wrote input for Map #1

Wrote input for Map #2

Starting Job

INFO impl.TimelineClientImpl: Timeline service address:
http://localhost.localdomain:8188/ws/v1/timeline/

INFO client.RMProxy: Connecting to ResourceManager at
localhost.localdomain/127.0.0.1:8050

INFO ipc.Client: Retrying connect to server:
localhost.localdomain/127.0.0.1:8050. Already tried 0 time(s); retry
policy is RetryUpToMaximumCountWithFixedSleep(maxRetries=50,
sleepTime=1000 MILLISECONDS)
```

To resolve the ZooKeeper issue:

1. Log in to the Ambari Web management console.
2. Restart ZooKeeper.
3. Ensure ZooKeeper is running.



Ensure File System Enforcement Point Sends Messages to Audit Engine

The BlueTalon File System Enforcement Point (FSEP) service runs on each Hadoop cluster node. The service is named bt-fsep.

On the computer where FSEP is installed, edit the file fsep-log4j.properties and set the DEBUG property. The file is:

```
/etc/bluetalon/pep/fs-ep/conf/fsep-log4j.properties
```

Set the DEBUG property in the file:

```
log4j.logger.kafka.producer=DEBUG, fsep
```

Run an HDFS command that will exercise FSEP. Example:

```
hdfs dfs -ls webhdfs://localhost:40070/

ls: Permission denied: user=root, access=GETFILESTATUS, inode=/
```

Examine the FSEP log file for errors. Example:

```
grep ERROR /var/log/bluetalon/pep/fs-ep/logs/*
```

```
/var/log/bluetalon/pep/fs-ep/logs/fsep.log:2016-09-08 17:26:11,227
ERROR DefaultEventHandler [][root:] GETFILESTATUS Failed to send
requests for topics ActivityMonitor with correlation ids in [0,18]

/var/log/bluetalon/pep/fs-ep/logs/fsep.log:2016-09-08 17:26:11,228
ERROR ProducerSendThread [][root:] GETFILESTATUS Error in handling
batch of 1 events
```

If an error message is shown, then ensure the audit services and the FSEP service are running:

```
# service --status-all | grep bt-audit
bt-audit-database (pid  982) is running...
bt-audit-etl (pid  1773) is running...
bt-audit-kafka (pid  1654) is running...
bt-audit-server (pid  1279) is running...
bt-audit-zookeeper (pid  1439) is running...

# service --status-all | grep bt-fsep
bt-fsep (pid  1007) is running...
```

Restart any services that are stopped.

Protocol Error with OpenLDAP and Policy Engine

If you encounter a protocol error with OpenLDAP and the Policy Engine, then enable bind_v2 in OpenLDAP.

Examine the ldif file. Example:

```
cat /home/allowv2.ldif
dn: cn=config
add: olcAllows
olcAllows: bind_v2
```

Run the following command:

```
ldapmodify -xWD cn=config -f /home/allowv2.ldif
```

You can also add an entry for olcAllows:bind_v2 to the end of the ldif file.

To enable LDAPv2 client connections:

```
allow bind_v2
```

Restart OpenLDAP:

```
/etc/init.d/slapd restart
```

Oozie Server Error

The following subsections show how to resolve an Oozie server error.

Error Condition

Example error file:

```
stderr: /var/lib/ambari-agent/data/errors-1107.txt
```

If you see the following error, perform the steps in the following subsections:

```
Traceback (most recent call last):
  File "/var/lib/ambari-agent/cache/common-
services/OOZIE/4.0.0.2.0/package/scripts/service_check.py", line 138,
in <module>
    OozieServiceCheck().execute()
  File "/usr/lib/python2.6/site-
packages/resource_management/libraries/script/script.py", line 219, in
execute
    method(env)
  File "/var/lib/ambari-agent/cache/common-
services/OOZIE/4.0.0.2.0/package/scripts/service_check.py", line 61, in
service_check
...
resource_management.core.exceptions.Fail: Execution of
'/var/lib/ambari-agent/tmp/oozieSmoke2.sh redhat
/usr/hdp/current/oozie-client /usr/hdp/current/oozie-client/conf
/usr/hdp/current/oozie-client/bin http://ip-172-31-33-252.us-west-
2.compute.internal:11000/oozie /usr/hdp/current/oozie-client/doc
/usr/hdp/current/hadoop-client/conf /usr/hdp/current/hadoop-client/bin
ambari-qa False' returned 1. source /usr/hdp/current/oozie-
client/conf/oozie-env.sh ; /usr/hdp/current/oozie-client/bin/oozie -
Doozie.auth.token.cache=false job -oozie http://ip-172-31-33-252.us-
west-2.compute.internal:11000/oozie -config /usr/hdp/current/oozie-
client/doc/examples/apps/map-reduce/job.properties -run
Error: E0507 : E0507: Could not access to
[${fs.defaultFS.override}/user/${user.name}/${examplesRoot}/apps/map-
reduce/workflow.xml], User: oozie is not allowed to impersonate ambari-
qa
Invalid sub-command: Missing argument for option: info
```

Prerequisites

Prerequisites:

- FSEP installed on Oozie client computer.
- All traffic routed through FSEP.

Update Ambari Script

Update the Ambari script on the Ambari node computer:

1. Edit file:
   ```
   vi /var/lib/ambari-server/resources/common-
   services/OOZIE/4.0.0.2.0/package/scripts/params_linux.py
   ```
2. Search for this text line:
   ```
   fs_root = config['configurations']['core-site']['fs.defaultFS']
   ```
3. Add this text immediately after the previous text line:
   ```
   # block added to resolve Oozie issue
   if fs_root == '${fs.defaultFS.fsep}':
     fs_root = config['configurations']['core-
   site']['fs.defaultFS.fsep']
   elif fs_root == '${fs.defaultFS.native}':
   ```

```
    fs_root = config['configurations']['core-
site']['fs.defaultFS.native']
```

Restart Ambari Server

Restart Ambari server:

```
ambari-server restart
```

Example output:

```
[root@sandbox scripts]# ambari-server restart
Using python  /usr/bin/python2
Restarting ambari-server
Using python  /usr/bin/python2
Stopping ambari-server
Ambari Server stopped
Using python  /usr/bin/python2
Starting ambari-server
Ambari Server running with administrator privileges.
Organizing resource files at /var/lib/ambari-server/resources...
Server PID at: /var/run/ambari-server/ambari-server.pid
Server out at: /var/log/ambari-server/ambari-server.out
Server log at: /var/log/ambari-server/ambari-server.log
Waiting for server start.....................
Ambari Server 'start' completed successfully.
```

Error is resolved.

ORC Table Error

The following subsections show how to resolve an error when creating BlueTalon rules for an ORC table.

Error Condition

Example error condition in the Policy Server log:

```
0 | FATAL | BTPolicySDK.com.bluetalon.ds.GenericSqlDs |
getTableColumnsFromShellStatement() | DATADOMAIN | Failed to executing
describe patient statement. cause: java.lang.AssertionError: assertion
failed
```

You might also see the following error when performing a DESCRIBE command on an ORC table:

```
Exception Caught : java.lang.AssertionError: assertion failed
|java.sql.SQLException: java.lang.AssertionError: assertion failed
```

Example ORC Table

Example ORC table:

```
create table Physician(DocID INT,DocName STRING,Specialist
STRING,DocSSN INT) clustered by (DocID) into 4 buckets row format
delimited fields terminated by ',' stored as ORC
TBLPROPERTIES('transactional'='true');
```

Set Ambari ORC Parameter to False

To resolve the error, set the Ambari ORC parameter to false:

1. In a Web browser, log in to the Ambari Web management console.
2. Go to the section Custom-Spark-Hive-Site-Override.
3. Set the parameter spark.sql.hive.convertMetastoreOrc to false.

Error is resolved.

Administer System

This section is an overview of BlueTalon system administration. For more information, see the *BlueTalon Security Administration Guide*.

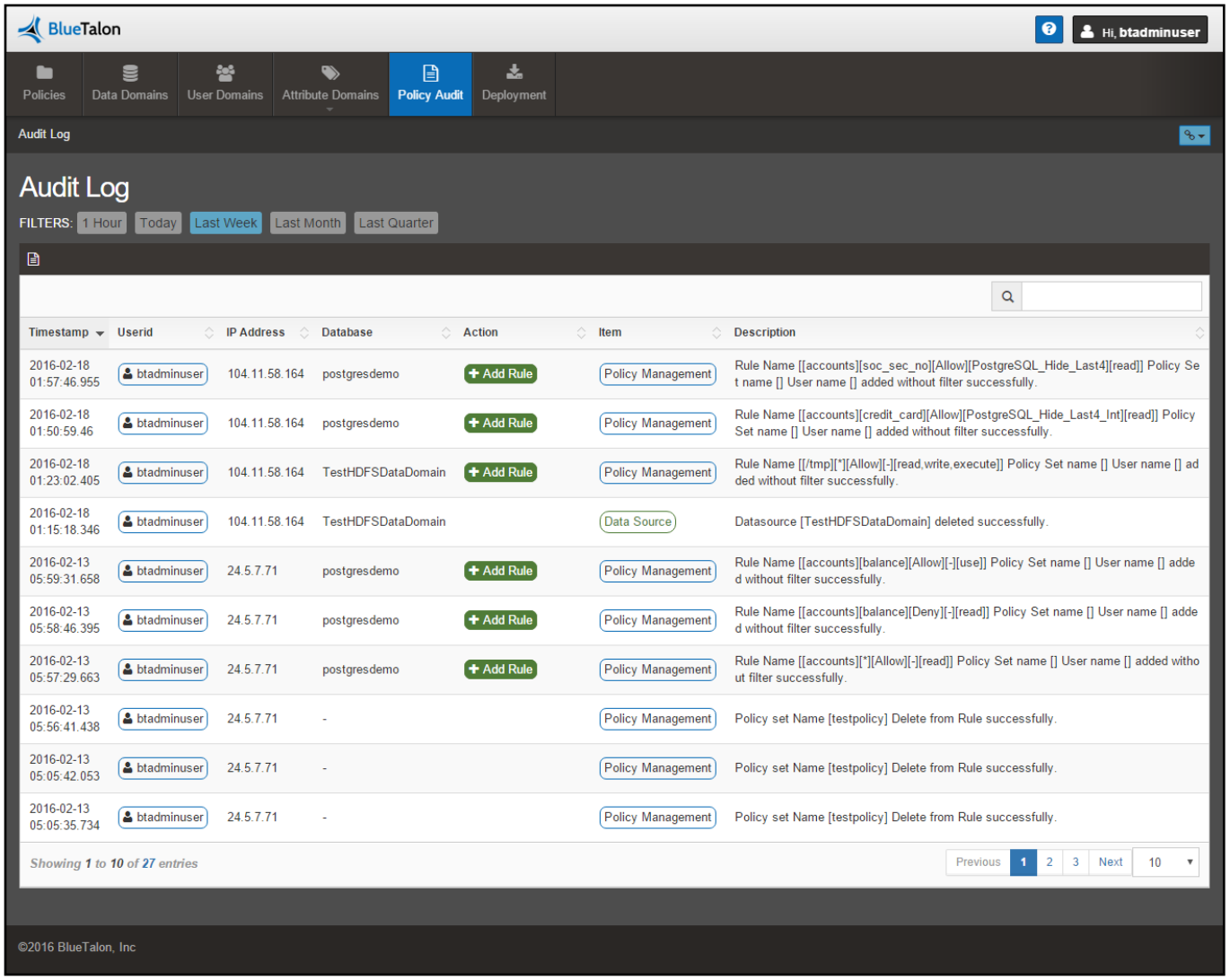
Examine Policy Audit Logs

You can use the Policy Console to view the policy audit logs. You can view the changes that administrators have made to policies, tables, rules, filters, permissions, and other security settings.

Examine the policy audit logs:

1. Open Web browser.
2. Go to the Policy Console.
3. Click the Policy Audit tab.
4. Choose a time span.
5. Sort the columns and search for an event.

The following screenshot shows example audit events. A timespan of last week is selected.



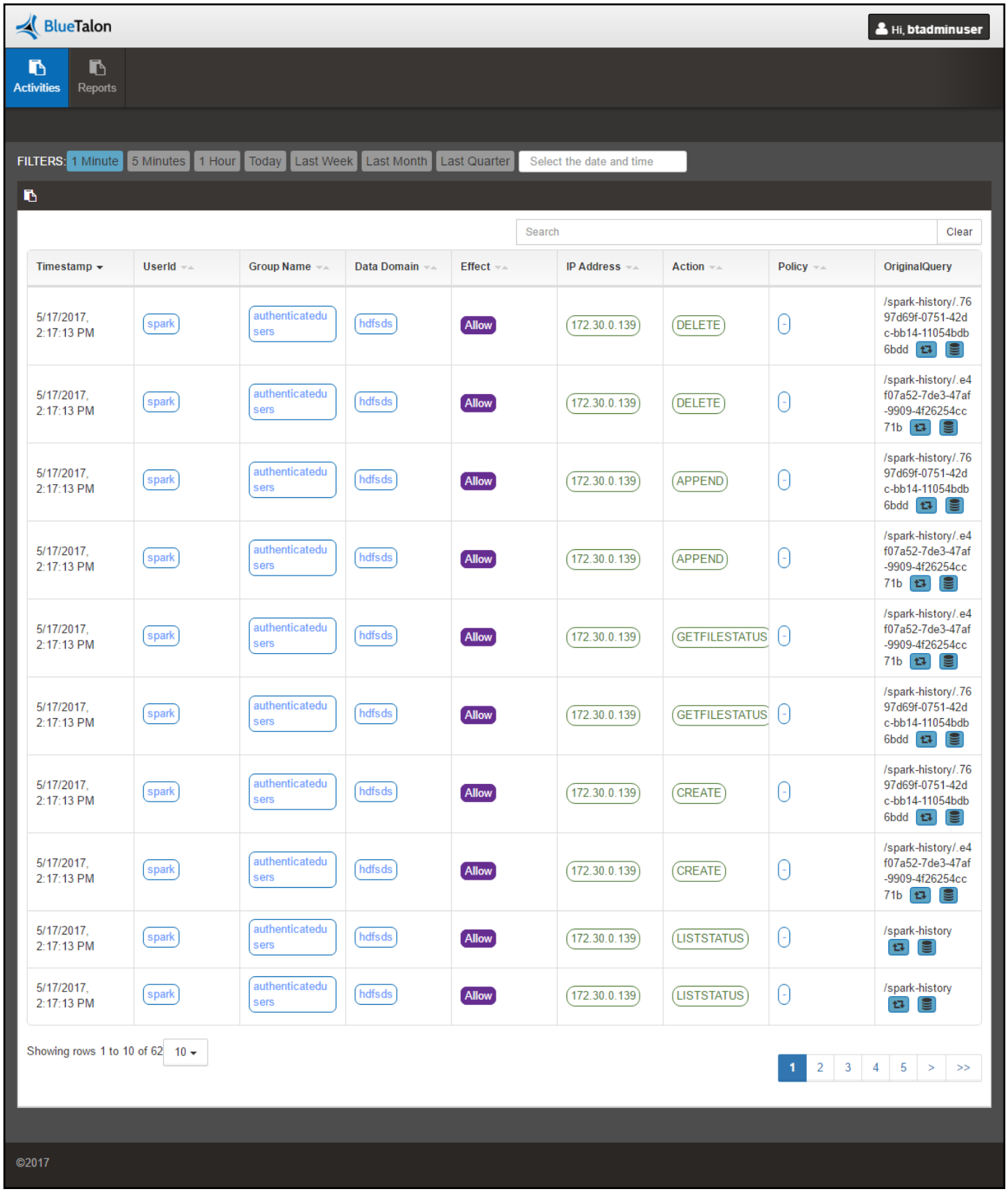
Examine User Audit Log

You use the Audit Console to view the user audit log. You can see user data requests and the original queries submitted by users. You can examine user audit logs to monitor unusual user activity and potential permission issues.

Examine the user audit log:

1. Open Web browser.
2. Go to the Audit Console.
3. Click the Activities tab.
4. (Optional) Choose a time span.
5. (Optional) Sort the columns and search for an event.

The following screenshot shows an example log.


BlueTalon
Hi, btadminuser
Activities
Reports
FILTERS: 1 Minute 5 Minutes 1 Hour Today Last Week Last Month Last Quarter Select the date and time
Search
Clear
Timestamp
UserId
Group Name
Data Domain
Effect
IP Address
Action
Policy
OriginalQuery
5/17/2017, 2:17:13 PM
spark
authenticatedusers
hdfsds
Allow
172.30.0.139
DELETE
/spark-history/.7697d69f-0751-42dc-bb14-11054bdb6bdd
/spark-history/.e4f07a52-7de3-47af-9909-4f26254cc71b
APPEND
GETFILESTATUS
CREATE
LISTSTATUS
/spark-history
Showing rows 1 to 10 of 62
10
1 2 3 4 5 > >>
©2017

Index

D

E

F

H

I

J

K

M

N

O

P

R

S

T

U

V