Protegrity Log Forwarding Guide 9.1.0.5

Created on: Nov 19, 2024

Notice

Copyright

Table of Contents

Chapter 1

Introduction to This Guide

This document provides information about the new logging architecture and contains information about forwarding logs.

> **Note:** The *Protegrity Log Management Guide* is renamed to the *Protegrity Log Forwarding Guide*.

1.1 Sections contained in this Guide

The guide is broadly divided into the following sections

- Section *Introduction to This Guide* defines the purpose and scope for this guide. In addition, it explains how information is organized in this guide.
- Section *Introduction to Logging* describes an overview and the architecture of the new logging ecosystem. It also describes the various components used for forwarding and storing logs.
- Section *Initializing the td-agent* provides information for setting up the *td-agent* service.
- Section *Verifying your Installation* lists the procedure for viewing the files, folders, and settings to ensure that the logging system is set up as required.
- Section *Managing td-agent and Log Forwarder Configurations* lists the steps for updating the configuration files for the *td-agent* and the *Log Forwarder*.
- Section *Sending Logs to an External Security Information and Event Management (SIEM)* lists the steps for collecting and sending logs to an external SIEM.
- Section *Troubleshooting* lists the solutions or workarounds to resolve problems while working with the logging system.

1.2 Accessing the Protegrity documentation suite

This section describes the methods to access the *Protegrity Documentation Suite* using the *My.Protegrity* portal.

Chapter 2

Introduction to Logging

Logs store information about your system or events that take place on a system. These entries are time stamped to track when an activity occurred. In addition, logs might also store additional information for tracking, monitoring, and solving system issues.

This section covers the importance of logging. It also covers the overview of logging in Protegrity products.

2.1 Logging Architecture

Logging follows a fixed routine. The system generates logs, which are collected and then forwarded to the Audit Store. The Audit Store holds the logs and these log records are used in various areas, such as, Forensics, alerts, reports, dashboards, and so on. This section explains the logging architecture.

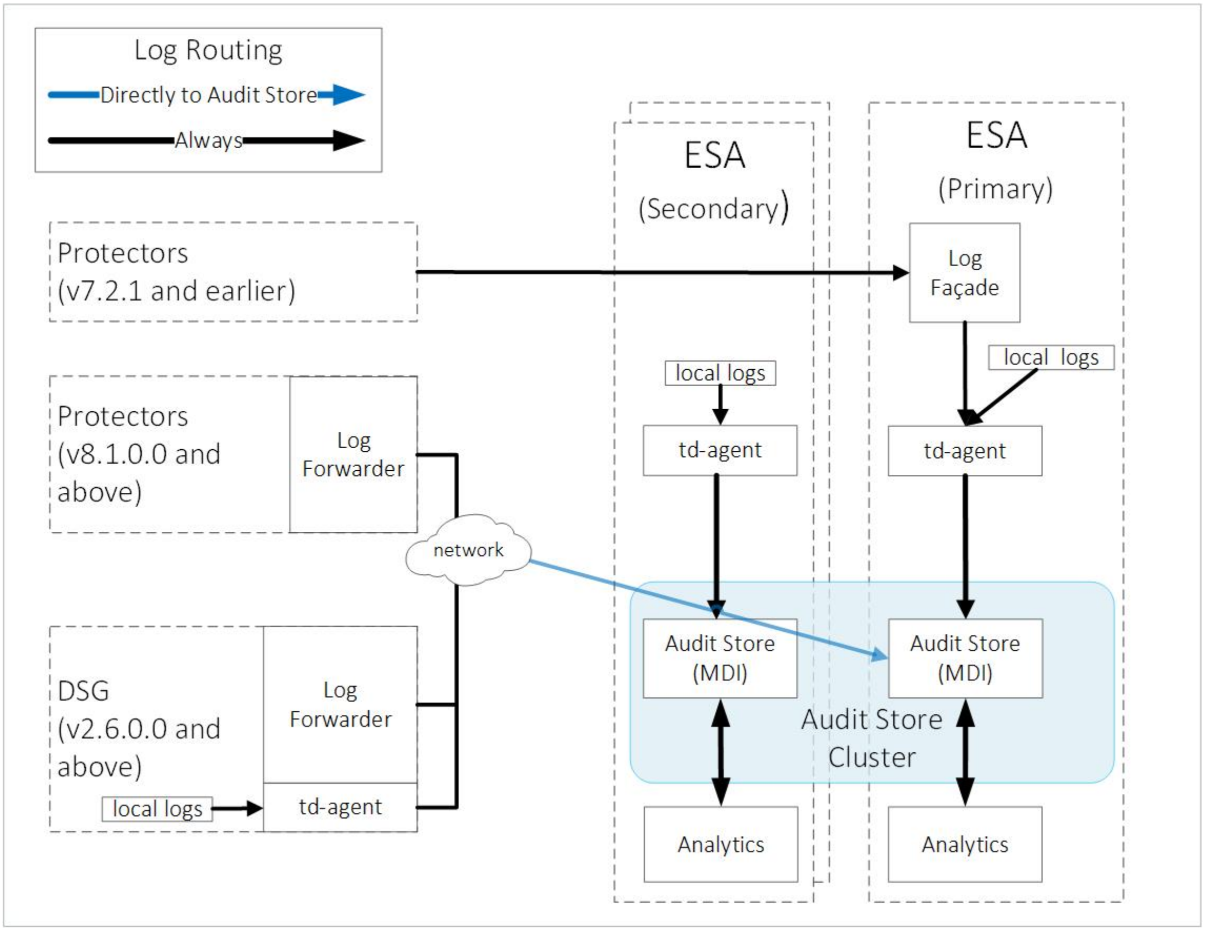


Figure 2-1: Architecture: Overview

- **ESA:**

  The ESA has the *td-agent* service installed for receiving and sending logs to the Audit Store. Additionally, the ESA has Protegrity Analytics and the Audit Store installed on it. The ESA receives the logs and stores it in the Audit Store. From this store, the logs are analyzed by Analytics and used in various areas, such as, Forensics, Alerts, Reports, Dashboards, and so on. Alerts are displayed for events as per the policies configured. Additionally, logs are collected from the log files generated by the P11 Gateway, Hub Controller, and Membersource services and sent to the Audit Store. By default, all Audit Store nodes have all node roles, that is, Master-eligible, Data, and Ingest. The Audit Store node on the primary ESA can be set to have the Master-eligible node role only, which allows it to function as the Master node that manages the Audit Store cluster, when it gets elected as a master.

  For more information about roles, refer to the section *Working with Roles* in the *Audit Store Guide 9.1.0.5*.

  > **Note:** A minimum of 3 ESAs are required for creating a dependable Audit Store cluster to protect it from system crashes. The architecture diagram shows 3 ESAs. Your architecture might also contain PSUs v9.1.0.0. If you are using the PSUs, then ensure that you keep them updated with all the available patches and fixes.

- **Protectors:**

  For v8.1.0.0 and later protectors, the new logging system is configured to send logs to the Audit Store on the ESA using the Log Forwarder.

  For Protectors, the Log Forwarder component is configured in the *pepserver.cfg* file.

For more information about the Log Forwarder-related logging configuration for protectors, refer to the section *Appendix: PEP Server Configuration File* in the *Protegrity Installation Guide 9.1.0.5*.

For v8.0.0.0 Protectors and earlier, the new logging system is integrated in the Log Facade on the ESA. Using the Log Facade, the logs are received and processed by the *td-agent* service on the ESA 9.1.0.5. Processing involves checking the integrity of the logs and adding audit-related information. The logs are then stored in the Audit Store.

- **DSG 3.1.0.1:**

  The DSG 3.1.0.1 has the *td-agent* service installed. The *td-agent* forwards the appliance logs to the Audit Store on the ESA. The *Log Forwarder* service forwards the data security operations-related logs, namely protect, unprotect, and reprotect, and the PEP server logs to the Audit Store on the ESA.

  For more information about configuring *td-agent* to forward logs, refer to the section *Forwarding System Logs to the Audit Store*.

  For more information about configuring *Log Forwarder* to forward logs, refer to the section *Appendix A: PEP Server Configuration File* in the *Protegrity Installation Guide 9.1.0.5*.

2.2 Components of the Logging Architecture

Protegrity software generates comprehensive logs. The logging infrastructure generates a huge number of log entries that take time to read. The enhanced logging architecture consolidates logs in the Audit Store and provides tools to make it easier to view and analyze log data. This section explains the logging components.

When you perform some operation using Protegrity software, or interact with Protegrity software directly, or indirectly using a different software or interface, a log entry is generated. This entry is stored in an Audit Store with other similar entries. A log entry contains valuable information about the interaction between the Protegrity software and a user or other systems.

A log entry might contain the following information:

- Date and time of the operation
- User who initiated the operation
- Operation that was initiated
- Systems involved in the operation
- Files that were modified or accessed

As the transactions build up, the quantum of logs generated also increases. Every day a lot of business transactions, inter-process activities, interactivity-based activities, system level activities, and other transactions take place resulting in a huge number of log entries. All these logs take up a lot of time and space to store and process. To save time, space, and to increase efficiency, Protegrity processes the log entries and saves them to the Audit Store. This processing ensures that the validity and quality of logs are maintained.

The solution for collecting and forwarding the logs to the Audit Store composes the logging architecture. The various components are installed on an appliance or an ESA. Based on your setup and configuration, the logging architecture installer adapts and uses specific ports.

For more information about the required ports, refer to the section *Open Listening Ports* in the *Protegrity Appliances Overview Guide 9.1.0.5*.

A brief overview of the components is provided in the following figure.

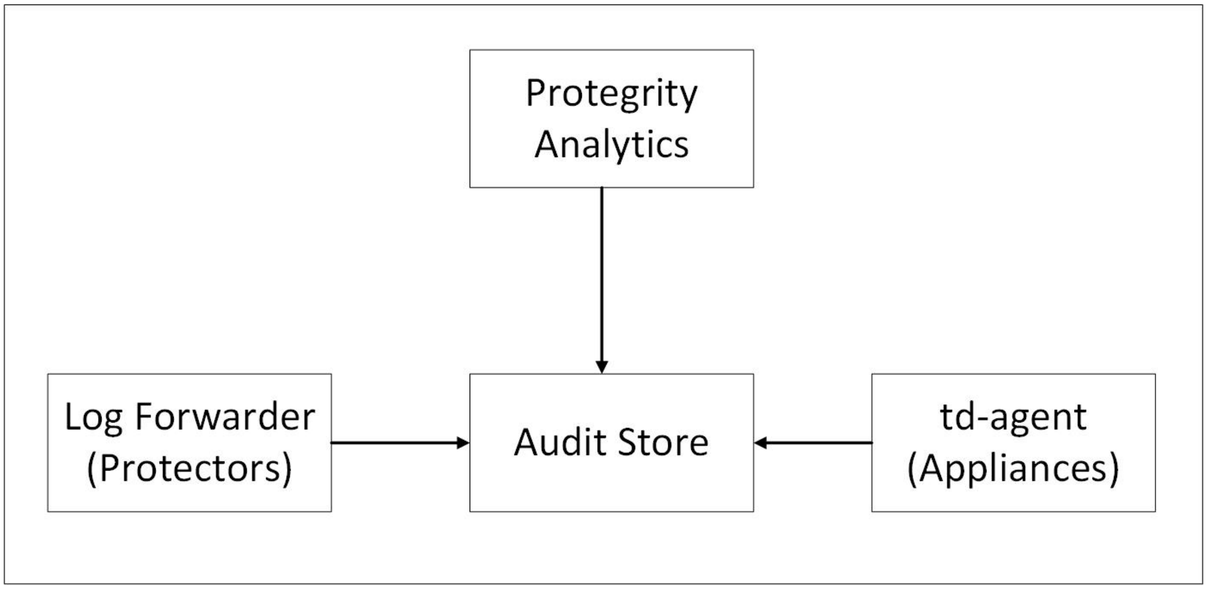


Figure 2-2: Components

The logging architecture contains the following components:

- **Protegrity Analytics:**

  Protegrity Analytics is a set of tools that help you analyse and study the log data in the Audit Store. Protegrity Analytics processes the log data and displays them using reports, graphs, and charts. This information makes it easy to view trends and anomalies in the log transactions. You can create and use JSON and SQL-like queries to retrieve and view the log data from the Audit Store.

  For more information about Protegrity Analytics, refer to the *Protegrity Analytics Guide 9.1.0.5*.

- **Audit Store:**

  The Audit Store is a central repository for storing all the logs. You can also create an Audit Store cluster by grouping multiple Audit Store nodes together. The Audit Store has the capability to query and retrieve large sets of data quickly. This provides real-time access to log data and reports to help you take quick decisions.

  For more information about Audit Store, refer to the *Audit Store Guide 9.1.0.5*.

- **td-agent:**

  The *td-agent* contains the internal logging configurations. It collects logs from Appliances, such as, the ESA or the DSG, and forwards them to the Audit Store.

  For more information about *td-agent*, refer to the *Protegrity Log Forwarding Guide 9.1.0.5*.

- **Log Forwarder:**

  The Log Forwarder contains the internal logging configurations. It collects logs from the protectors and forwards them to the Audit Store. In the case of pre-8.0.0.0 protectors, the logs are forwarded by the PEP server to the Log Facade, on the ESA, where they are processed. The Log Facade adds the required information to upgrade the log quality for Forensics and sends the logs to the Audit Store. The additional information added to the logs helps maintain the integrity of the data. It makes log tracing easy so that data that is tampered, deleted, or added can be easily identified.

  For more information about the Log Forwarder, refer to the *Protegrity Log Forwarding Guide 9.1.0.5*.

2.2.1 Understanding Protegrity Analytics

The logs stored in the Audit Store hold valuable data. This information is very useful if you use it effectively. To view the information in an effective way, Protegrity Analytics provides you with tools to view and analyze the data in the Audit Store.

Protegrity Analytics is a component that you need to configure when you set up the ESA. After it is installed, the tools provided by Protegrity Analytics are accessible using the ESA.

For more information about the tools in Protegrity Analytics, refer to the *Protegrity Analytics Guide 9.1.0.5*.

2.2.2 Understanding the Audit Store

The Audit Store is the center of the logging ecosystem. The main task of the Audit Store is to process all the logs that are received by the system, store them, and provide the information when log-related data is requested. It is very versatile and processes data fast.

The Audit Store is a component that is installed on the ESA during the installation of v9.1.0.5. The Audit Store is scalable, hence, additional nodes can be added to the Audit Store cluster.

For more information about the Audit Store, refer to the *Audit Store Guide 9.1.0.5*.

2.2.3 Understanding the td-agent

The *td-agent* forms an integral part of logging architecture. It is responsible for sending logs from the Appliance to an Audit Store. It is the *td-agent* service that is configured to send and receive logs. The service is installed by default when you install or upgrade components, such as, the ESA, to v9.1.0.5 components. It is also available on the DSG.

Based on your installation, the following configurations are performed for the *td-agent*:

- Audit Store on the local system: In this case, the *td-agent* is configured to collect the logs and send it to the Audit Store on the local system.
- Audit Store on a remote system: In this case, the Audit Store service is not installed locally, such as DSG 3.1.0.1, but it is installed on the ESA. The *td-agent* is configured to forward logs securely to the Audit Store in the ESA.

  For more information about forwarding logs, refer to the section *Forwarding System Logs to the Audit Store*.

2.2.4 Understanding the Log Forwarder

The Log Forwarder is responsible for forwarding data security operation logs to the Audit Store in the ESA. In cases when the Audit Store is unreachable, the Log Forwarder handles the logs.

For Linux-based protectors, such as the Oracle Database Protector for Linux, if the connection to the ESA is lost, then the Log Forwarder starts collecting the logs in the memory cache. If the Audit Store is still unreachable after the cache is full, then the Log Forwarder continues collecting the logs and stores in the disk. When the connection to the Audit Store is restored, the logs in the cache are forwarded to the Audit Store.

> **Important:** The default memory cache for collecting logs is 256 MB.
>
> For information about updating the cache limit, refer to the section *Configuring the Disk Space on the Log Forwarder.*

The following table provides information about how the Log Forwarder handles logs in different situations.

Table 2-1: Log Forwarder Behavior

If..	then the Log Forwarder...
Connection to Audit Store is lost	Starts collecting logs in the in-memory cache based on the cache limit defined.
Connection to Audit Store is lost and the cache is full	In case of Linux-based protectors, the Log Forwarder continues to collect the logs and stores in disk. If the disk space is full, then all the cache files will be emptied and the Log Forwarder will continue to run. For Windows-based protectors, the Log Forwarder starts throwing away the logs. **Important:** If the filesystem for Linux protectors is not EXT4 or XFS, then the logs will not be saved to the disk after the cache is full.
Connection to Audit Store is restored	Forwards logs to the Audit Store

2.3 Understanding the Log Aggregation

This section describes the architecture, the configurations, and the workflow of the Log Aggregation.

The following diagram describes the architecture and the workflow of the Log aggregation.

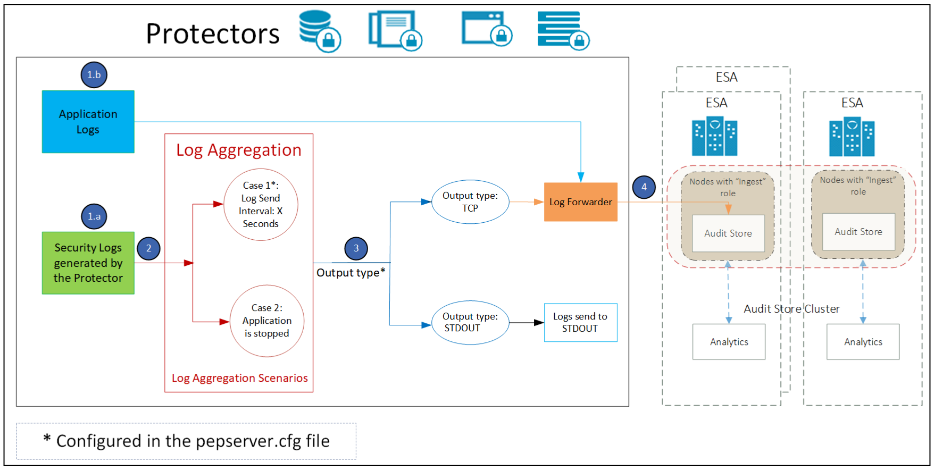


Figure 2-3: Architecture: Overview

1. a. The security logs generated by the protectors are aggregated in the protectors.
   b. The application logs are not aggregated and they are sent directly to the Log Forwarder.
2. The security logs are aggregated based on one of the following cases that happens first:
   - Time interval set using the *logsendinterval* setting in the *pepserver.cfg* file to send the security logs has been reached.

     For more information about configuring the *logsendinterval* setting, refer to the section *Appendix A: PEP Server Configuration File* in the *Protegrity Installation Guide 9.1.0.5*

   - Application is stopped.
3. The aggregated logs can be forwarded to the Log Forwarder, sent to the standard output screen, or stored in a file based on the *output* type setting in the *pepserver.cfg* file.
   - If the *output* type is set to *tcp* (default output type), then the security logs are forwarded to the Log Forwarder.
   - If the *output* type is set to *stdout*, then the security logs are sent to the standard output screen.

Caution:

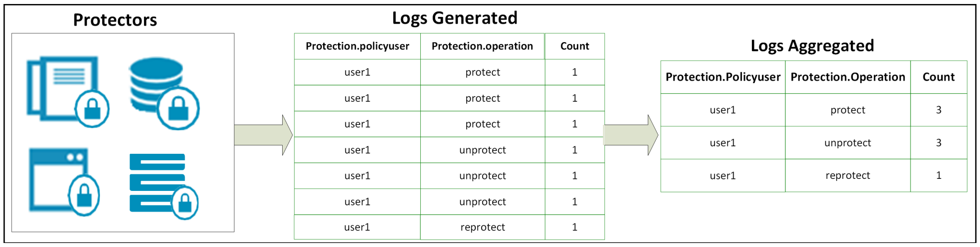
> Do not use the *output=stdout* setting in a production environment. Use this setting for debugging only.
>
> If the *output=stdout* setting is configured, then the aggregated logs are not sent to audit store.

For more information about configuring the *output* type setting, refer to the section *Appendix A: PEP Server Configuration File* in the *Protegrity Installation Guide 9.1.0.5*

4. The aggregated security logs from the Log Forwarder are forwarded to the Audit Store.

   For more information about the Audit Store, refer to the section *Understanding the Audit Store.*

The following diagram illustrates how similar logs are aggregated.



Protection.policyuser	Protection.operation	Count
user1	protect	1
user1	protect	1
user1	protect	1
user1	unprotect	1
user1	unprotect	1
user1	unprotect	1
user1	reprotect	1

Protection.Policyuser	Protection.Operation	Count
user1	protect	3
user1	unprotect	3
user1	reprotect	1

Figure 2-4: Pictorial Representation

The similar security logs are aggregated after the log send interval or when an application is stopped.

For example: If you are performing 30 similar protect operations simultaneously, then a single log will be generated with a count of 30.

Chapter 3

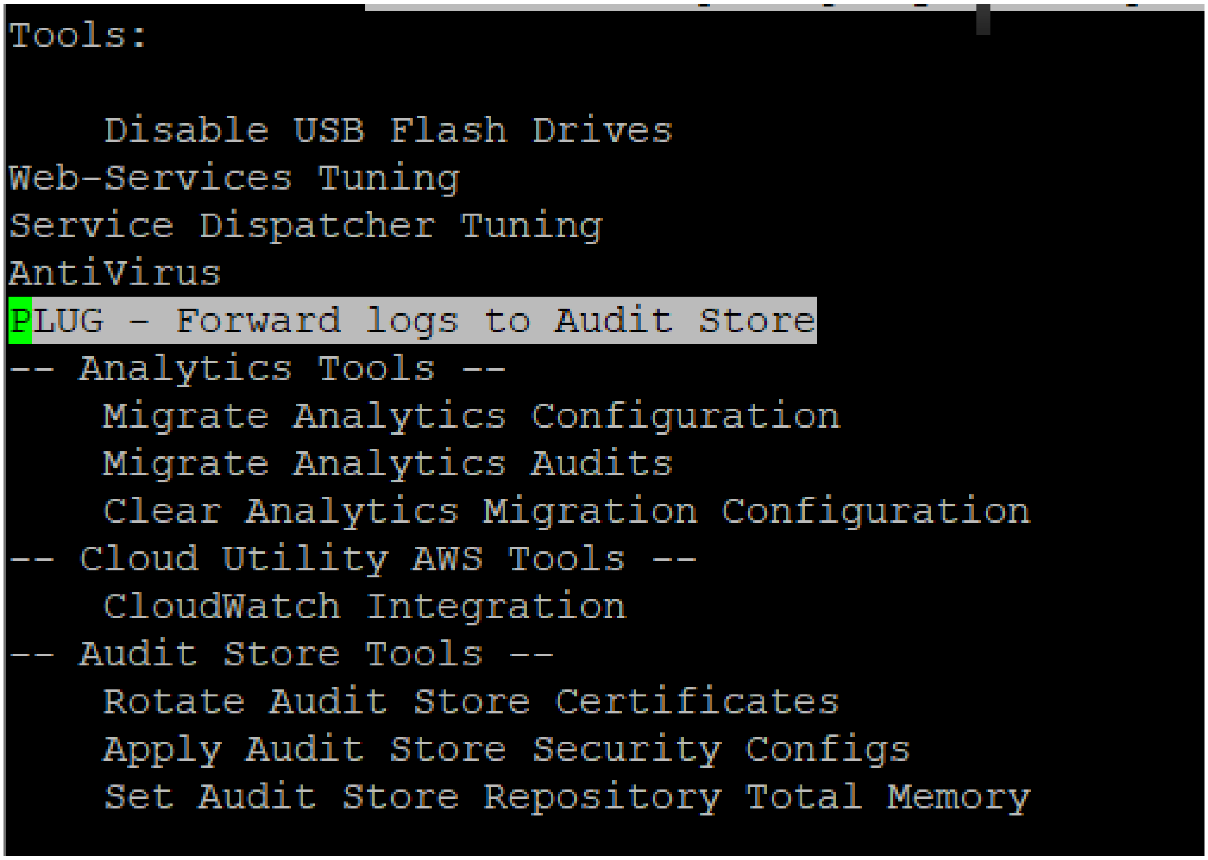
Managing td-agent and Log Forwarder Configurations

This section covers the configurations for forwarding logs. It specifies the options available for working with the *td-agent* and the *Log Forwarder*.

3.1 Forwarding System Logs to the Audit Store

When the logging components are configured on the ESA or the appliance, system logs are sent to the Audit Store. Complete the following steps to send the system logs to the Audit Store.

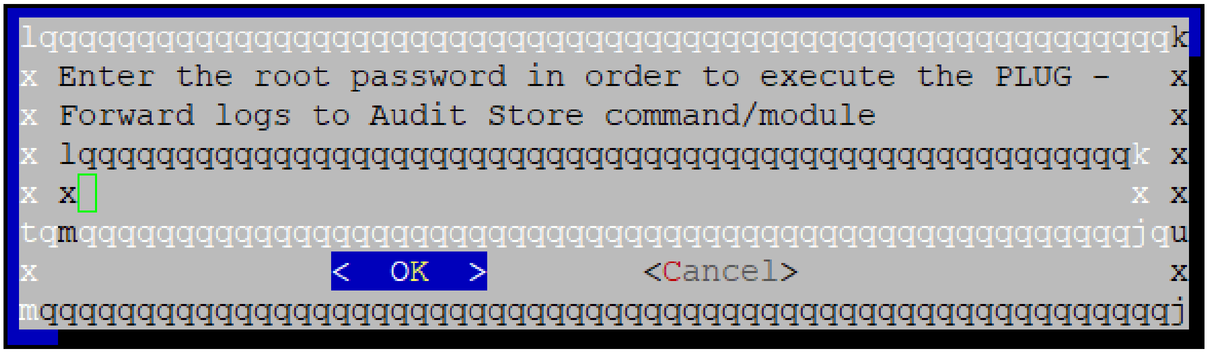
1. Login to the CLI Manager on the ESA or the Appliance.
2. Navigate to **Tools** > **PLUG - Forward logs to Audit Store**.



```
Tools:

    Disable USB Flash Drives
Web-Services Tuning
Service Dispatcher Tuning
AntiVirus
PLUG - Forward logs to Audit Store
-- Analytics Tools --
    Migrate Analytics Configuration
    Migrate Analytics Audits
    Clear Analytics Migration Configuration
-- Cloud Utility AWS Tools --
    CloudWatch Integration
-- Audit Store Tools --
    Rotate Audit Store Certificates
    Apply Audit Store Security Configs
    Set Audit Store Repository Total Memory
```

Figure 3-1: Forwarding Logs

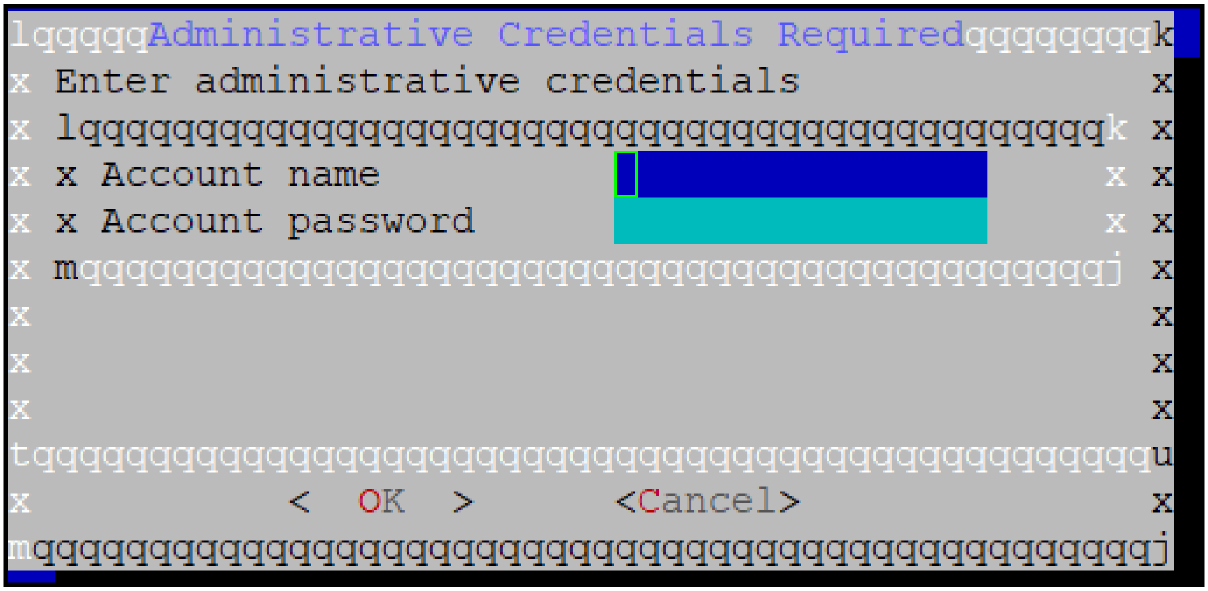
3. Enter the password for the root user and select **OK**.



```
lqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqk
x Enter the root password in order to execute the PLUG -   x
x Forward logs to Audit Store command/module                x
x lqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqk x
x x                                                      x x
tqmqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqjqu
x                  <  OK  >          <Cancel>               x
mqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqj
```

Figure 3-2: Root Password Screen

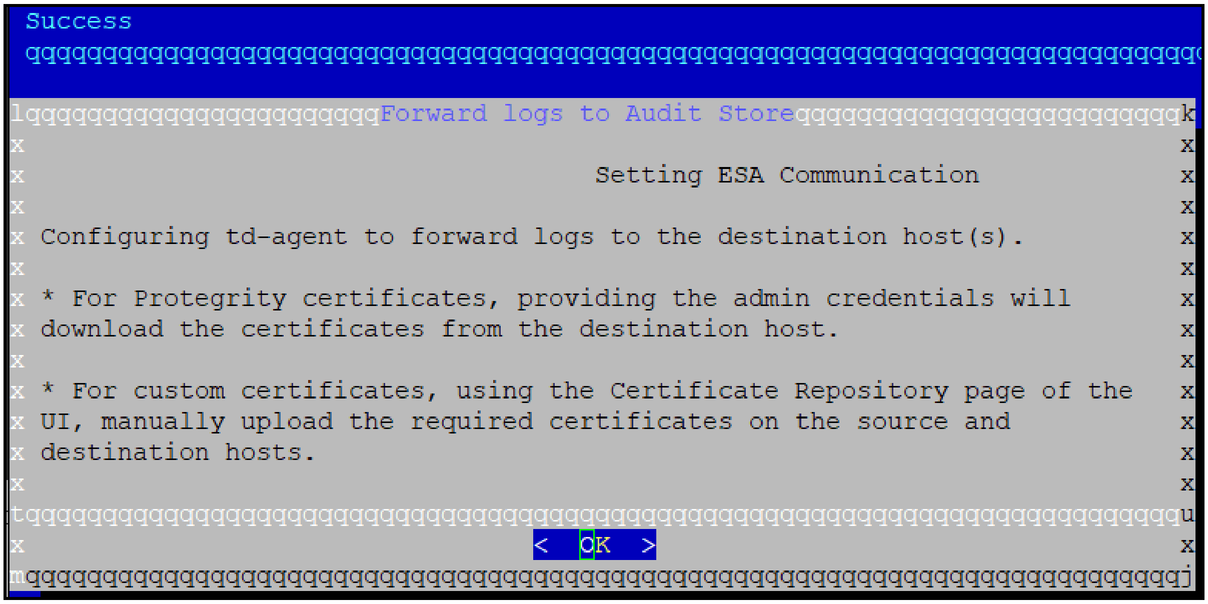
4. Enter the username and password for the admin user and select **OK**.



```
Administrative Credentials Required
 Enter administrative credentials
   Account name
   Account password

            <  OK  >        <Cancel>
```

Figure 3-3: Admin Details Screen

5. Select **OK**.

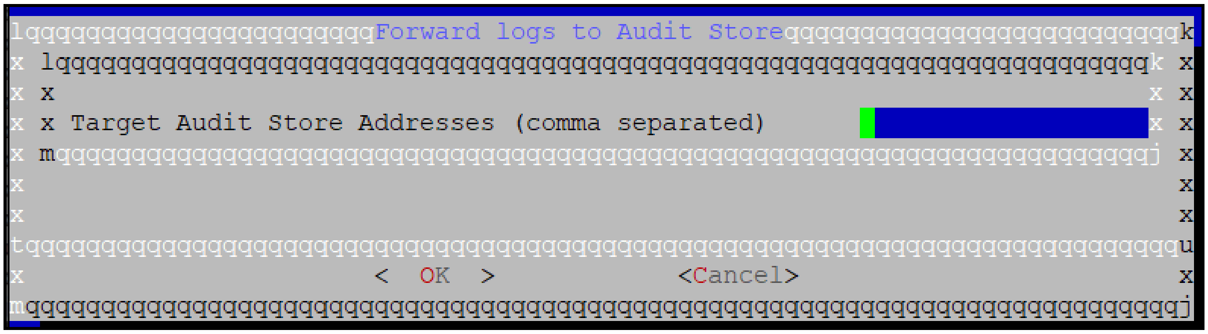


```
Success

Forward logs to Audit Store

                        Setting ESA Communication

Configuring td-agent to forward logs to the destination host(s).

* For Protegrity certificates, providing the admin credentials will
download the certificates from the destination host.

* For custom certificates, using the Certificate Repository page of the
UI, manually upload the required certificates on the source and
destination hosts.

                              <  OK  >
```

Figure 3-4: Certificates Information

6. Enter the IP address of all the nodes in the Audit Store cluster with the *Ingest* role and select **OK**. Specify multiple IP addresses separated by comma.

> **Note:** To identify the node with the *Ingest* roles, login to the ESA Web UI and navigate to **Audit Store Management** > **Nodes**.



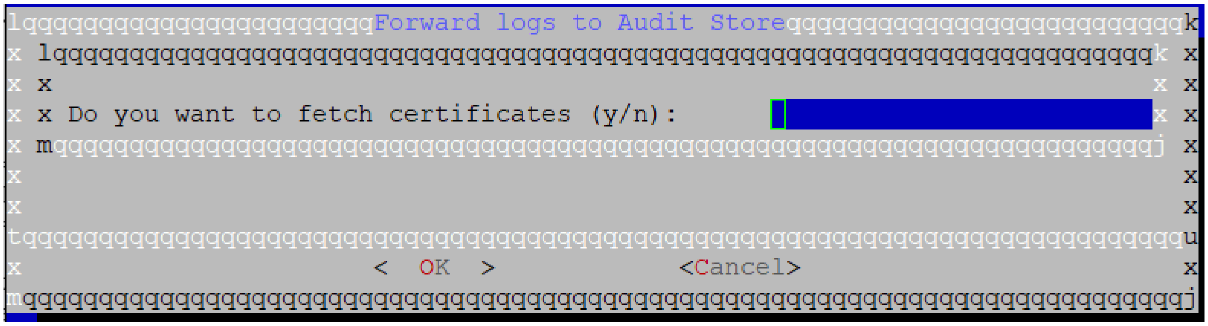
```
Forward logs to Audit Store
Target Audit Store Addresses (comma separated)
< OK >          <Cancel>
```

Figure 3-5: Audit Store Details

7. Enter *y* to fetch certificates and select **OK**.

   Specifying *y* fetches *td-agent* certificates from target node. These certificates can then be used to validate and connect to the target node. They are required to authenticate with the Audit Store while forwarding logs to the target node.

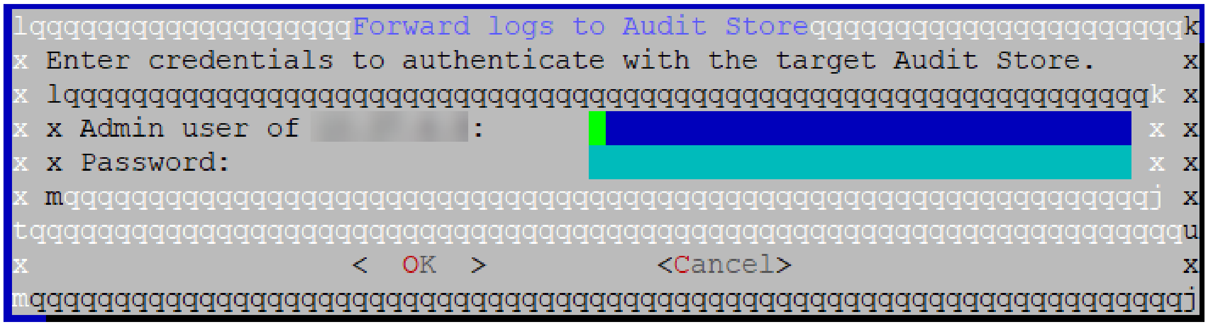
   If the certificates are already available on the system, or you do not want to fetch the certificates, or you want to use custom certificates, then specify *n* in this screen.



```
Forward logs to Audit Store
Do you want to fetch certificates (y/n):
< OK >          <Cancel>
```

Figure 3-6: Audit Store Certificates

8. Enter the credentials for the admin user of the destination machine and select **OK**.



```
Forward logs to Audit Store
Enter credentials to authenticate with the target Audit Store.
Admin user of [redacted]:
Password:
< OK >          <Cancel>
```

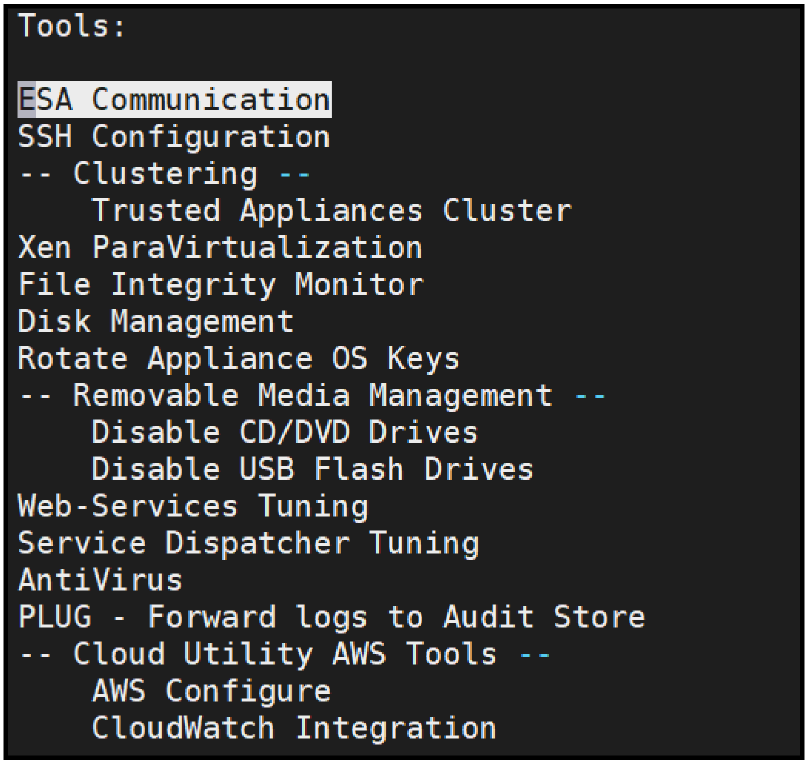
Figure 3-7: Admin User Details

The *td-agent* service is configured to send logs to the Audit Store and the CLI menu appears.

3.2 Forwarding Audit Logs to the Audit Store

The audit logs are the data security operation-related logs, such as protect, unprotect, and reprotect and the PEP server logs. The audit logs from the Appliance, such as, the DSG are forwarded through the *Log Forwarder* service to the Audit Store on the ESA.

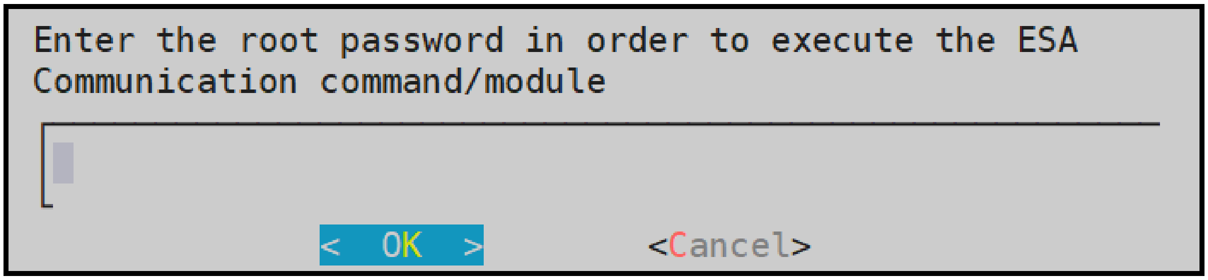
1. Login to the CLI Manager on the Appliance.
2. Navigate to **Tools** > **ESA Communication**.



```
Tools:

ESA Communication
SSH Configuration
-- Clustering --
    Trusted Appliances Cluster
Xen ParaVirtualization
File Integrity Monitor
Disk Management
Rotate Appliance OS Keys
-- Removable Media Management --
    Disable CD/DVD Drives
    Disable USB Flash Drives
Web-Services Tuning
Service Dispatcher Tuning
AntiVirus
PLUG - Forward logs to Audit Store
-- Cloud Utility AWS Tools --
    AWS Configure
    CloudWatch Integration
```

Figure 3-8: Forwarding Audit Logs to Audit Store

3. Enter the password of the *root* user of the Appliance and select **OK**.

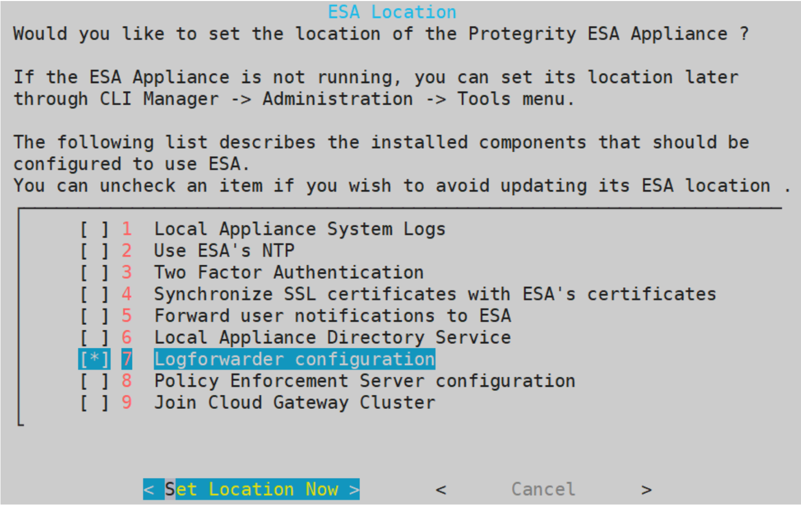


```
Enter the root password in order to execute the ESA
Communication command/module
```

Figure 3-9: Root Password Screen

4. Select the *Logforwarder configuration* option. Press **Tab** to select **Set Location Now** and press **Enter**.

   The *ESA Location* screen appears.

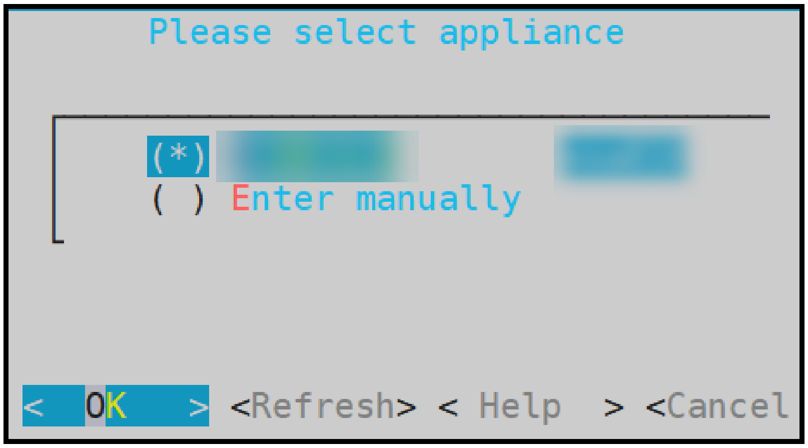


```
                        ESA Location
Would you like to set the location of the Protegrity ESA Appliance ?

If the ESA Appliance is not running, you can set its location later
through CLI Manager -> Administration -> Tools menu.

The following list describes the installed components that should be
configured to use ESA.
You can uncheck an item if you wish to avoid updating its ESA location .

        [ ] 1  Local Appliance System Logs
        [ ] 2  Use ESA's NTP
        [ ] 3  Two Factor Authentication
        [ ] 4  Synchronize SSL certificates with ESA's certificates
        [ ] 5  Forward user notifications to ESA
        [ ] 6  Local Appliance Directory Service
        [*] 7  Logforwarder configuration
        [ ] 8  Policy Enforcement Server configuration
        [ ] 9  Join Cloud Gateway Cluster

          < Set Location Now >       <     Cancel     >
```

Figure 3-10: Setting the ESA location

5. Select the ESA that you want to connect with, and then press **Tab** to select **OK** and press **ENTER**.

   The *ESA selection* screen appears.



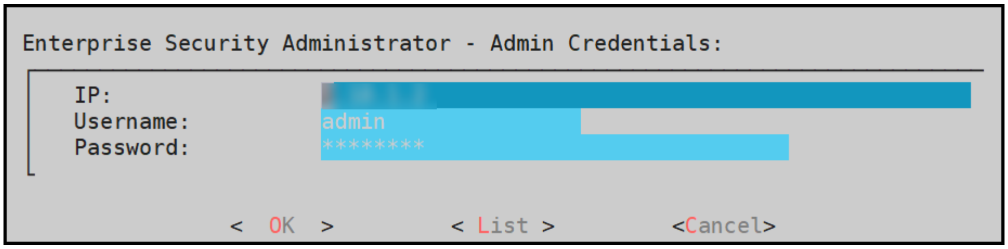
```
      Please select appliance

    (*) [redacted]          [redacted]
    ( ) Enter manually

<   OK   > <Refresh> < Help   > <Cancel
```

Figure 3-11: ESA appliance selection screen

> **Note:** If you want to enter the ESA details manually, then select the **Enter manually** option. You will be asked to enter the ESA IP address or hostname when this option is selected.

6. Enter the ESA administrator username and password to establish communication between the ESA and the Appliance. Press **Tab** to select **OK** and press **Enter**.

   The *Enterprise Security Administrator - Admin Credentials* screen appears.

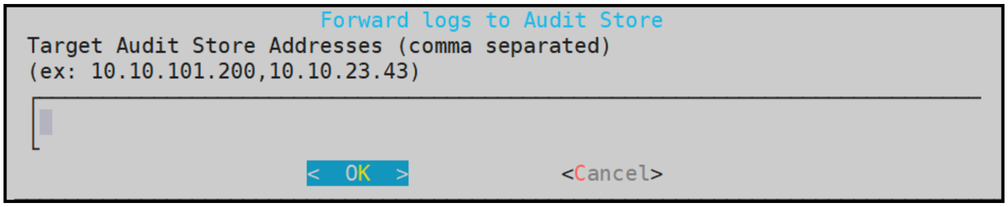


```
Enterprise Security Administrator - Admin Credentials:

   IP:             [redacted]
   Username:       admin
   Password:       ********

          <  OK  >        < List >         <Cancel>
```

Figure 3-12: Enterprise Security Administrator - Admin Credentials screen

7. Enter the IP address or hostname for the ESA. Press **Tab** to select **OK** and press **ENTER**. You can specify multiple IP addresses separated by comma.

   The *Forward Logs to Audit Screen* screen appears.

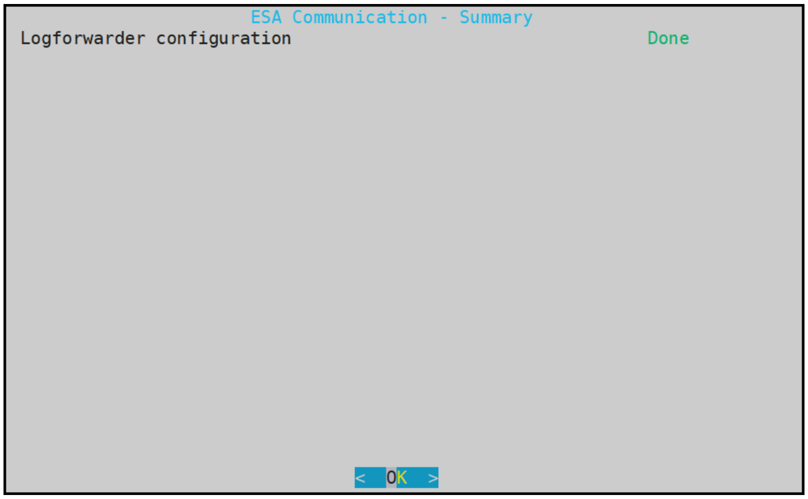


```
Forward logs to Audit Store
Target Audit Store Addresses (comma separated)
(ex: 10.10.101.200,10.10.23.43)
```

Figure 3-13: Forward Logs to Audit Screen

> **Note:** You must specify the IP addresses for all the ESAs that are already configured and then enter the IP for the additional ESA.

8. After successfully establishing the connection with the ESA, the following Summary dialog box appears. Press **Tab** to select **OK** and press **Enter**.



```
ESA Communication - Summary
Logforwarder configuration                                    Done
```

Figure 3-14: ESA Communication - Summary screen

9. Repeat step 1 to step 8 on all the Appliance nodes in the cluster.

3.3 Monitoring the td-agent Buffer Size

When logs are generated, they are sent to the *td-agent* that then forwards it to the Audit Store. If the Audit Store is not reachable due to network issues, then logs start accumulating at the *td-agent* that buffers these logs. There is a possibility that the amount of logs generated increase, when the logs cross the buffer size threshold, which by default is 64 GB, then a buffer overflow takes place.

To avoid this issue, monitor the *td-agent* buffer size, which is the size of all the files in the */opt/protegrity/td-agent/es_buffer* directory. The monitor raises an alert when the *td-agent* buffer size crosses the limit that is configured. Use this to monitor the system and avoid any *Buffer overflow* errors. The example provided here is a sample that you can use to create to raise an alert when the buffer size exceeds 10 GB. Modify the values in this example as per your requirements.

Perform the following steps to update the buffer size:

1. Login to the ESA Web UI.
2. Click **Analytics**.
3. From the Analytics screen, navigate to **Alerting** > **Destinations**.
4. Click **Create Destination**.

5. Specify the following information for the destination and click **Save**:
   - **Name**: SystemDestination
   - **Type**: Email
   - Specify valid email addresses in the required fields.

   > **Note:** Ensure that SMTP is configured on all the nodes that are present in the Audit Store cluster by navigating to **Settings** > **Network** > **SMTP Settings** on each node.
   >
   > For more information about configuring SFTP, refer to the section *Setting Up the Email Server* in the *Protegrity Appliances Overview Guide 9.1.0.5*.

6. From the Analytics screen, navigate to **Alerting** > **Monitors**.
7. Click **Create Monitor**.
8. Specify the following information for the monitor and click **Save Monitor**:
   - **Name**: BufferWatcher
   - Select the **Frequency** as **By interval** for every **1 Hours**.
   - Set the monitor as **Enabled**.
   - **Monitor Query**: Specify the following query.

     ```
     {
         "query": {
             "bool": {{
         "query": {
             "bool": {
                 "must": [
                      {
                         "range": {
                             "returncode": {
                                 "gte": 10485760,
                                 "lte": 104857600
                             }
                         }
                     }            ],
                 "filter": [
                     {
                         "range": {
                             "origin.time_utc": {
                                 "from": "{{period_start}}",
                                 "to": "{{period_end}}",
                                 "include_lower": true,
                                 "include_upper": true,
                                 "format": "epoch_millis",
                                 "boost": 1
                             }
                         }
                     }
                 ],
                 "adjust_pure_negative": true,
                 "boost": 1
             }
         },
         "aggregations": {}
     }
     ```

     In the code, the *returncode* is used to monitor the buffer size in kb. This code monitors and raises an alert if the buffer size is between 10 GB and 100 GB. Modify the values as per your requirement. For example, set the value of *gte* to *1048576* for raising an alert when the buffer crosses the size of 1 GB.

9. Specify the following trigger information and click **Add Trigger**:
   - **Name**: BufferTrigger
   - **Severity**: 1

- **Trigger Query**: Specify the following query.

```
{
                "script":
                {
                    "source": "ctx.results[0].hits.total > 0",
                    "lang": "painless"
                }
 }
```

10. Specify the following action information and click **Add Action**:

- **Name**: BufferEmail
- **Destination**: SystemDestination

> **Note:** Ensure that you use the destination created in step 5.

- **Message Query**: Specify the following query.

```
{
                        "source":
                        {
                            "message": "td-agent buffer size is over the limit
configured for the alert {10gb}",
                            "subject": "td-agent buffer size is over the configured
limit"
                        },
                        "lang": "mustache",
                        "options":
                        {
                            "content_type": "application/json; charset=UTF-8"
                        }
}
```

The monitor is created and raises alerts when the *td-agent* buffer crosses the limit that is configured.

For more information about working with alerts, refer to the section *Working with Alerts* in the *Protegrity Analytics Guide 9.1.0.5*.

3.4 Configuring the Disk Space on the Log Forwarder

The Log Forwarder collects logs from the protectors and forwards them to the Audit Store.

If the Audit Store is not reachable due to network issues, then the Log Forwarder caches the undelivered logs locally on the hard disk.

If the incoming logs being cached is faster than they are being sent to the Audit Store, then the back pressure arises.

The following formula can be used to calculate the disk space on the Log Forwarder, given the estimated audit rate and time to sustain the audit rate, without logs being sent to the Audit Store. Modify the values in this example as per your requirements.

> **Note:**
>
> The default value of the disk space is 256 MB.

Disk Space in Mega bytes = (Audit Rate X Time in Seconds X 5.9) / 1024.

- *Audit Rate* = Number of Policy Audits generated per second

- *Time in Seconds* = Time duration for which the disk can sustain the audit rate without the logs being sent to the Audit Store.

> **Note:**
>
> If the default or the configured value of the *storage.total_limit_size* setting is reached, then all the cache files will be emptied and the Log Forwarder will continue to run.

Perform the following steps to configure the *storage.total_limit_size* setting in the *out_elastic.conf* file on the Protector machine.

1. Login and open a CLI on the Protector machine.
2. Navigate to the *config.d* directory using the following command.

   ```
   cd /opt/protegrity/fluent-bit/data/config.d
   ```

3. Backup the existing *out_elastic.conf* file using the following command.

   ```
   cp out_elastic.conf out_elastic.conf_backup
   ```

   > **Note:** The system uses the *.conf* extension. To avoid configuration conflicts, ensure that you append the backup file name to the file name extension, for example, *.conf_backup* or *.conf_bkup123*.

4. Open the *out_elastic.conf* file using the following command.

   ```
   vi out_elastic.conf
   ```

5. Update the value of *storage.total_limit_size* setting in the output blocks.

   > **Note:**
   >
   > The default value of the *storage.total_limit_size* is 256 MB.

   The following snippet shows the extract of the code.

   ```
   [OUTPUT]
       Name es
       Match logdata
       Retry_Limit False
       Index pty_insight_audit
       Type  _doc
       Time_Key ingest_time_utc
       Upstream C:\Program Files\Protegrity\fluent-bit/data/config.d/upstream_es.cfg
       storage.total_limit_size 256M

   [OUTPUT]
       Name es
       Match flulog
       Retry_Limit 1
       Index pty_insight_audit
       Type  _doc
       Time_Key ingest_time_utc
       Upstream C:\Program Files\Protegrity\fluent-bit/data/config.d/upstream_es.cfg
       storage.total_limit_size 256M

   [OUTPUT]
       Name es
       Match errorlog
       Retry_Limit 1
       Index pty_insight_audit
       Type  _doc
       Time_Key ingest_time_utc
       Upstream C:\Program Files\Protegrity\fluent-bit/data/config.d/upstream_es.cfg
   ```

```
    storage.total_limit_size 256M
```

> **Note:**
>
> Ensure that you update the code with the required disk space size.

6. Save and close the file.
7. Restart the *Log Forwarder* on the Protector using the following commands.

```
/opt/protegrity/fluent-bit/bin/logforwarderctrl stop
/opt/protegrity/fluent-bit/bin/logforwarderctrl start
```

8. If required, then complete the configurations on the remaining Protector machines.

Chapter 4

Sending Logs to an External Security Information and Event Management (SIEM)

The Protegrity infrastructure provides a robust setup for logging and analyzing the logs generated. It might be possible that you already have an existing infrastructure for collating and analyzing logs, and need to analyze the logs generated by the ESA. Use the information provided in this section to forward the logs generated by the ESA to the Audit Store and your own SIEM.

In the default setup, the logs are sent from the Protectors directly to the Audit Store on the ESA using the Log Forwarder on the Protector.

For more information about the default flow, refer to the section *Logging Architecture*.

To forward logs to the Audit Store and the external SIEM, the Protectors send the logs to the *td-agent* on the ESA or Appliance. The *td-agent* is configured to forward the logs to the required locations.

An overview architecture diagram for sending logs to the Audit Store and the external SIEM is shown in the following figure.

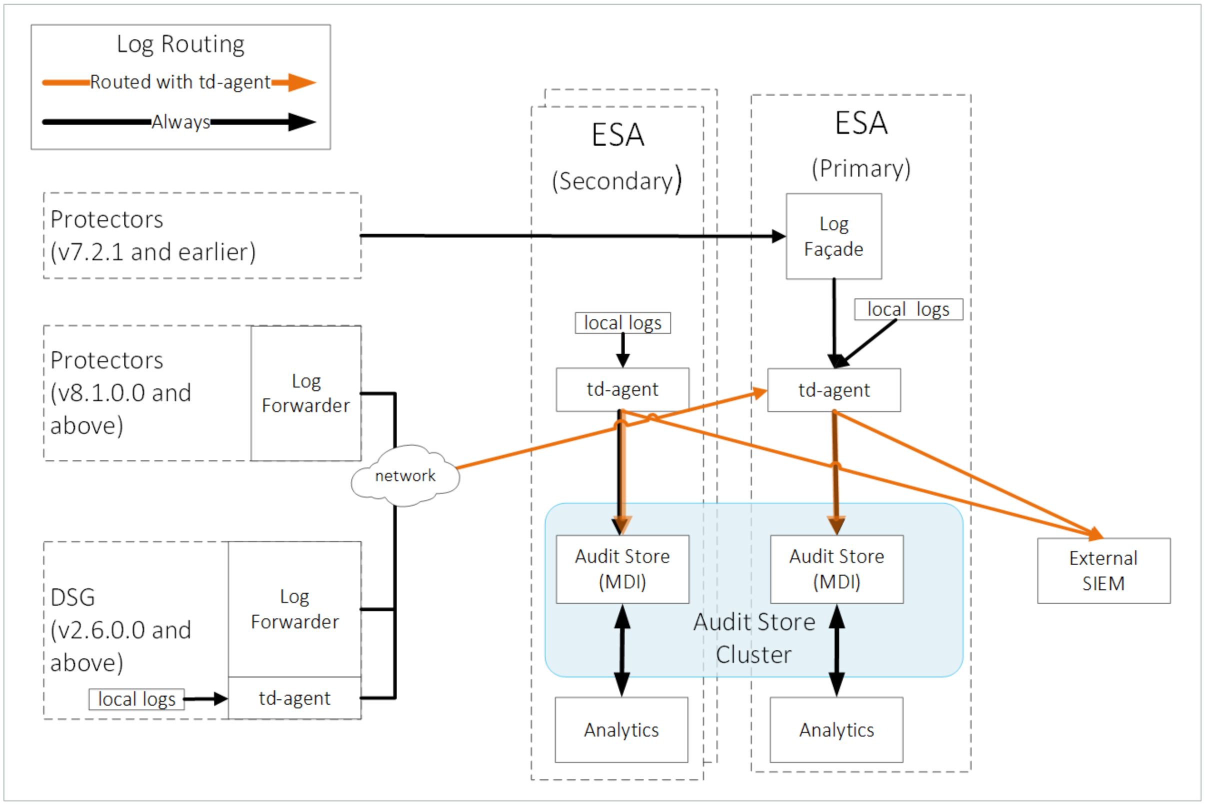


Figure 4-1: Sending Logs to the Audit Store and External SIEM

You can forward the logs generated on the Protector to the Audit Store and the external SIEM using the following steps.

1. *Configuring td-agent to Forward Logs to the Audit Store*
2. *Sending the Protector Logs to the td-agent*
3. *Configuring td-agent to Forward Logs to the External Endpoint*

Note: Ensure that you complete all the steps in the order specified.

4.1 Configuring td-agent to Forward Logs to the Audit Store

Complete the steps provided in this section to configure the *td-agent* to forward the logs received to the Audit Store.

➤ To configure *td-agent*:

1. Add the port *24284* to the rule list on the ESA. This port is configured for the ESA to receives the Protector logs over a secure connection.

   For more information about adding rules, refer to the section *Adding a New Rule with Predefined List of Functionality* in the *Protegrity Appliances Overview Guide 9.1.0.5*.

   a. Login to the CLI Manager of the Primary ESA.
   b. Navigate to **Networking** > **Network Firewall**.
   c. Enter the password for the *root* user.
   d. Select **Add New Rule** and select **Choose**.

e. Select **Accept** and select **Next**.

f. Select **Manually**.

g. Select **TCP** and select **Next**.

h. Specify *24284* for the port and select **Next**.

i. Select **Any** and select **Next**.

j. Select **Any** and select **Next**.

k. Specify a description for the rule and select **Confirm**.

2. Open the OS Console on the Primary ESA.

a. Login to the CLI Manager of the Primary ESA.

b. Navigate to **Administration** > **OS Console**.

c. Enter the root password and select **OK**.

3. Create the *INPUT_forward_external.conf* file for configuring the *td-agent*.

a. Navigate to the *config.d* directory using the following command.

```
cd /opt/protegrity/td-agent/config.d
```

b. Create the *INPUT_forward_external.conf* file using the following command.

```
vi INPUT_forward_external.conf
```

c. Add the following contents to the file.

```
<source>
  @type forward
  bind 0.0.0.0
  port 24284
  <transport tls>
       ca_path           /mnt/ramdisk/certificates/mng/CA.pem
       cert_path         /mnt/ramdisk/certificates/mng/server.pem
       private_key_path  /mnt/ramdisk/certificates/mng/server.key
  </transport>
</source>
```

> **Note:** Ensure that you retain the formatting of the file. Also, ensure that the certificates exist in the directory that is specified.
>
> If you specify an IP address or host name for the *bind*, then ensure that the certificates are updated to match the host name or IP address specified.

d. Save and close the file.

e. Update the permission of the file using the following command.

```
chmod 640 INPUT_forward_external.conf
chown td-agent:plug INPUT_forward_external.conf
```

4. **Optional:** Update the configuration settings to improve the SSL/TLS server configuration on the system.

a. Navigate to the *config.d* directory using the following command.

```
cd /opt/protegrity/td-agent/config.d
```

b. Open the *INPUT_forward_external.conf* file using the following command.

```
vi INPUT_forward_external.conf
```

c. Add the following content in bold to the file. Update and use the ciphers that you require.

```
<source>
  @type forward
  bind 0.0.0.0
  port 24284
  <transport tls>
       ca_path           /mnt/ramdisk/certificates/mng/CA.pem
       cert_path         /mnt/ramdisk/certificates/mng/server.pem
       private_key_path  /mnt/ramdisk/certificates/mng/server.key
       ciphers "ALL:!
aNULL:!eNULL:!SSLv2:!SSLv3:!DHE:!AES256-SHA:!CAMELLIA256-SHA:!AES128-SHA:!CAMELLIA128-
SHA:!TLS_RSA_WITH_RC4_128_MD5:!TLS_RSA_WITH_RC4_128_SHA:!TLS_RSA_WITH_3DES_EDE_CBC_SHA:!
TLS_DHE_RSA_WITH_3DES_EDE_CBC_SHA:!TLS_RSA_WITH_SEED_CBC_SHA:!
TLS_DHE_RSA_WITH_SEED_CBC_SHA:!TLS_ECDHE_RSA_WITH_RC4_128_SHA:!
TLS_ECDHE_RSA_WITH_3DES_EDE_CBC_SHA"
  </transport>
</source>
```

Ensure that you specify the entire line of code on a single line and retain the formatting of the file.

d. Save and close the file.

5. Restart the *td-agent* service.
   a. Login to the ESA Web UI.
   b. Navigate to **System** > **Services** > **Misc** > **td-agent**,
   c. Restart the **td-agent** service.
6. Update the Protector configuration to send logs to the *td-agent*.

   For more information about forwarding logs, refer to the section *Sending the Protector Logs to the td-agent*.

4.2 Sending the Protector Logs to the td-agent

Configure the Protector to send the logs to the *td-agent* on the ESA or Appliance. The *td-agent* forwards the logs received to the Audit Store and the external location.

Before you begin

Ensure that you complete the steps from *Configuring td-agent to Forward Logs to the Audit Store*.

Identify the Fluent Bit version of your Protector using the following steps:

1. Login and open a CLI on the Protector machine
2. Restart the Log Forwarder using the following commands.

```
/opt/protegrity/fluent-bit/bin/logforwarderctrl stop
/opt/protegrity/fluent-bit/bin/logforwarderctrl start
```

3. View the version number in the output displayed.

```
[root@ip-192-168-0-10 bin]# /opt/protegrity/fluent-bit/bin/logforwarderctrl start
Fluent Bit v1.6.10-1.0.0+4.g2f46.master
```

Table 4-1: Configuration Information for Fluent Bit Version

Configuration	v1.6	v1.9 or v2.2
Name of *.conf* file	*out_elastic.conf*	*out.conf*
Name of *.cfg* file	*upstream_es.cfg*	*upstream.cfg*

To configure the Protector:

1. Login and open a CLI on the Protector machine.
2. Back up the existing files.

a. Navigate to the *config.d* directory using the following command.

```
cd /opt/protegrity/fluent-bit/data/config.d
```

b. Backup the existing *out_elastic.conf* file using the following command.

```
mv out_elastic.conf out_elastic.conf_backup
```

> **Note:** For the *.conf* file name, refer to the *Table*.
>
> The system uses the *.conf* extension. To avoid configuration conflicts, ensure that you append the backup file name to the file name extension, for example, *.conf_backup* or *.conf_bkup123*.

c. Backup the existing *upstream_es.cfg* file using the following command.

```
mv upstream_es.cfg upstream_es.cfg_backup
```

> **Note:** For the *.cfg* file name, refer to the *Table*.
>
> The system uses the *.conf* extension. To avoid configuration conflicts, ensure that you append the backup file name to the file name extension, for example, *.conf_backup* or *.conf_bkup123*.

3. Create the *forward_tdagent.conf* file for specifying the logs that must be forwarded to the ESA.

a. Ensure that you are in the */opt/protegrity/fluent-bit/data/config.d* directory.

b. Create the *forward_tdagent.conf* file using the following command.

```
vi forward_tdagent.conf
```

c. Add the following code in the file.

The extract of the code is shown here.

```
[OUTPUT]
    Name           forward
    Match          logdata
    Retry_Limit    1
    Upstream /opt/protegrity/fluent-bit/data/config.d/upstream_forward.cfg
[OUTPUT]
    Name           forward
    Match          errorlog
    Retry_Limit    1
    Upstream /opt/protegrity/fluent-bit/data/config.d/upstream_forward.cfg
[OUTPUT]
    Name           forward
    Match          flulog
    Retry_Limit    1
    Upstream /opt/protegrity/fluent-bit/data/config.d/upstream_forward.cfg
```

> **Note:** Verify that the code in the configuration file is formatted properly. Ensure that there are no additional or trailing spaces, tabs, line breaks, or control characters in the configuration file.

d. Save and close the file.

e. Update the permission of the file using the following command.

```
chmod 640 forward_tdagent.conf
```

4. Create the *upstream_forward.cfg* file for forwarding the logs to the ESA.

a. Ensure that you are in the */opt/protegrity/fluent-bit/data/config.d* directory.

b. Create the *upstream_forward.cfg* file using the following command.

```
vi upstream_forward.cfg
```

c. Add the following code in the file.

The extract of the code is shown here.

```
[UPSTREAM]
    name       forward-balancing
[NODE]
    name       node1
    host       <IP address of the ESA>
    port       24284
    tls        on
    tls.verify off
    tls.ca_file /opt/protegrity/defiance_dps/data/CA.pem
```

> **Note:** Ensure that you update the code with the IP address of the ESA where you updated the *td-agent* settings in *Configuring td-agent to Forward Logs to the Audit Store*.

d. Save and close the file.

e. Update the permission of the file using the following command.

```
chmod 640 upstream_forward.cfg
```

5. Restart *fluent-bit* on the Protector using the following commands.

```
/opt/protegrity/fluent-bit/bin/logforwarderctrl stop
/opt/protegrity/fluent-bit/bin/logforwarderctrl start
```

6. If required, then complete the configurations on the remaining Protector machines.
7. Update the *td-agent* configuration to send logs to the external location.

   For more information about forwarding logs to the external location, refer to the section *Configuring td-agent to Forward Logs to the External Endpoint*.

4.3 Configuring td-agent to Forward Logs to the External Endpoint

Complete the steps in this sections to configure the *td-agent* for sending logs to an external endpoint, such as, an external SIEM.

As per your setup and requirements, the logs forwarded can be formatted using the syslog-related fields and sent over TLS to the SIEM. Alternatively, you can send the logs without any formatting over a non-TLS connection to the SIEM, such as, syslog.

The following options are available, select any *one* of the sections based on your requirements:

- *Forwarding Logs to a Syslog Server*
- *Forwarding Logs to a Syslog Server Over TLS*

4.3.1 Forwarding Logs to a Syslog Server

The ESA has logs generated by the Appliances and the Protectors connected to the ESA. You can forward these logs to your syslog server and use the log data for further analysis as per your requirements. Complete the steps provided in this section to forward the ESA logs directly to an external syslog server.

For a complete list of plugins that you can forward logs to, refer to *https://www.fluentd.org/plugins/all*.

Before you begin

Ensure that you complete the following prerequisites:

- The external syslog server is available and running.

- You have completed the steps from the section *Sending the Protector Logs to the td-agent*.

➤ To forward logs to the external SIEM:

1. Open the CLI Manager on the Primary ESA.
   a. Login to the CLI Manager of the Primary ESA where you configured the *td-agent* in *Configuring td-agent to Forward Logs to the Audit Store*.
   b. Navigate to **Administration** > **OS Console**.
   c. Enter the root password and select **OK**.
2. Navigate to the */products/uploads* directory using the following command.

```
cd /products/uploads
```

3. Obtain the required plugins files using one of the following commands based on your setup.
   - If the Appliance has Internet access, then run the following commands.

```
wget https://rubygems.org/downloads/syslog_protocol-0.9.2.gem
```

```
wget https://rubygems.org/downloads/remote_syslog_sender-1.2.2.gem
```

```
wget https://rubygems.org/downloads/fluent-plugin-remote_syslog-1.0.0.gem
```

   - If the Appliance does not have Internet access, then complete the following steps.
     a. Download the following set up files from a system that has Internet and copy them to the Appliance in the */products/uploads* directory.
        - *syslog_protocol-0.9.2*
        - *remote_syslog_sender-1.2.2*
        - *fluent-plugin-remote_syslog-1.0.0*
     b. Ensure that the files downloaded have the *execute* permission.
4. Prepare the required plugins files using the following commands.
   a. Assign the required ownership permissions to the software using the following command.

```
chown td-agent *.gem
```

   b. Assign the required permissions to the software you installed using the following command.

```
chmod -R 755 /opt/td-agent/embedded/lib/ruby/gems/2.4.0
```

   c. Assign ownership of the *.gem* files to the *td-agent* user using the following command.

```
chown -R td-agent:plug /opt/td-agent/embedded/lib/ruby/gems/2.4.0
```

5. Install the required plugins files using one of the following commands based on your setup.
   - If the Appliance has Internet access, then run the following commands.

```
sudo -u td-agent /opt/td-agent/embedded/bin/fluent-gem install syslog_protocol
```

```
sudo -u td-agent /opt/td-agent/embedded/bin/fluent-gem install remote_syslog_sender
```

```
sudo -u td-agent /opt/td-agent/embedded/bin/fluent-gem install fluent-plugin-
remote_syslog
```

- If the Appliance does not have Internet access, then run the following commands.

  ```
  sudo -u td-agent /opt/td-agent/embedded/bin/fluent-gem install --local /products/
  uploads/syslog_protocol-0.9.2.gem
  ```

  ```
  sudo -u td-agent /opt/td-agent/embedded/bin/fluent-gem install --local /products/
  uploads/remote_syslog_sender-1.2.2.gem
  ```

  ```
  sudo -u td-agent /opt/td-agent/embedded/bin/fluent-gem install --local /products/
  uploads/fluent-plugin-remote_syslog-1.0.0.gem
  ```

6. Update the configuration files using the following steps.

   a. Navigate to the *config.d* directory using the following command.

   ```
   cd /opt/protegrity/td-agent/config.d
   ```

   b. Backup the existing output file using the following command.

   ```
   cp OUTPUT.conf OUTPUT.conf_backup
   ```

   > **Note:** The system uses the *.conf* extension. To avoid configuration conflicts, ensure that you append the backup file name to the file name extension, for example, *.conf_backup* or *.conf_bkup123*.

   c. Open the *OUTPUT.conf* file using the following command.

   ```
   vi OUTPUT.conf
   ```

   d. Update the following contents in the *OUTPUT.conf* file.

   - Update the *match* tag in the file to *<match *.*.* logdata flulog errorlog>*.
   - Add the following code in the *match* tag in the file:

     ```
        <store>
          @type relabel
          @label @syslog
        </store>
     ```

   The final *OUTPUT.conf* file with the updated content is shown here:

   ```
   <filter **>
     @type elasticsearch_genid
     # to avoid duplicate logs
     # https://github.com/uken/fluent-plugin-elasticsearch#generate-hash-id
     hash_id_key _id    # storing generated hash id key (default is _hash)
   </filter>

   <match *.*.* logdata flulog errorlog>
     @type copy

     <store>
         @type elasticsearch
         hosts localhost
         port 9200
         index_name pty_insight_audit
         type_name _doc
         pipeline logs_pipeline
         # adds new data - if the data already exists (based on its id), the op is skipped.
         # https://github.com/uken/fluent-plugin-elasticsearch#write_operation
         write_operation create
         # By default, all records inserted into Elasticsearch get a random _id. This
   option allows to use a field in the record as an identifier.
         # https://github.com/uken/fluent-plugin-elasticsearch#id_key
         id_key _id
         scheme https
         ssl_verify true
         ssl_version TLSv1_2
         ca_file /etc/ksa/certificates/plug/CA.pem
   ```

```
      client_cert /etc/ksa/certificates/plug/client.pem
      client_key /etc/ksa/certificates/plug/client.key
      request_timeout 300s # defaults to 5s https://github.com/uken/fluent-plugin-
elasticsearch#request_timeout
      <buffer>
        @type file
        path /opt/protegrity/td-agent/es_buffer
        retry_forever true           # Set 'true' for infinite retry loops.
        flush_mode interval
        flush_interval 60s
        flush_thread_count 8  # parallelize outputs https://docs.fluentd.org/deployment/
performance-tuning-single-process#use-flush_thread_count-parameter
        retry_type periodic
        retry_wait 10s
      </buffer>
  </store>

  <store>
      @type relabel
      @label @triggering_agent
  </store>

  <store>
    @type relabel
    @label @syslog
  </store>
</match>
```

> **Note:** Verify that the code in the configuration file is formatted properly. Ensure that there are no additional or trailing spaces, tabs, line breaks, or control characters in the configuration file.

e. Save and close the file.

f. Create the *OUTPUT_syslog.conf* file using the following command.

```
vi OUTPUT_syslog.conf
```

g. Perform the steps from one of the following solution as per your requirement.

- **Solution 1: Forward all logs to the external syslog server:**

  Add the following contents to the *OUTPUT_syslog.conf* file.

```
<label @syslog>
    <match *.*.* logdata flulog errorlog>
    @type copy
    <store>
        @type remote_syslog
        host <IP_of_the_syslog_server_host>
        port 514
        <format>
            @type json
        </format>
        protocol udp
        <buffer>
            @type file
            path /opt/protegrity/td-agent/syslog_tags_buffer
            retry_forever true # Set 'true' for infinite retry loops.
            flush_mode interval
            flush_interval 60s
            flush_thread_count 8 # parallelize outputs https://docs.fluentd.org/
deployment/performance-tuning-single-process#use-flush_thread_count-parameter
            retry_type periodic
            retry_wait 10s
        </buffer>
        </store>
    </match>
</label>
```

> **Note:** Verify that the code in the configuration file is formatted properly. Ensure that there are no additional or trailing spaces, tabs, line breaks, or control characters in the configuration file.

- **Solution 2: Forward only the protection logs to the external syslog server:**

  Add the following contents to the *OUTPUT_syslog.conf* file.

```
<label @syslog>
    <match logdata>
    @type copy
    <store>
        @type remote_syslog
        host <IP_of_the_syslog_server_host>
        port 514
        <format>
            @type json
        </format>
        protocol udp
        <buffer>
            @type file
            path /opt/protegrity/td-agent/syslog_tags_buffer
            retry_forever true # Set 'true' for infinite retry loops.
            flush_mode interval
            flush_interval 60s
            flush_thread_count 8 # parallelize outputs https://docs.fluentd.org/
deployment/performance-tuning-single-process#use-flush_thread_count-parameter
            retry_type periodic
            retry_wait 10s
        </buffer>
        </store>
    </match>
</label>
```

> **Note:** Ensure that you specify the *<IP_of_the_syslog_server_host>* in the file. Verify that the code in the configuration file is formatted properly. Ensure that there are no additional or trailing spaces, tabs, line breaks, or control characters in the configuration file.
>
> If you need to use a TCP connection, then update the *protocol* to *tcp*. In addition, specify the *port* that you opened for TCP communication.
>
> For more information about the formatting the output, navigate to *https://docs.fluentd.org/configuration/format-section*.

h. Save and close the file.

i. Update the permissions for the file using the following commands.

```
chown td-agent:td-agent OUTPUT_syslog.conf
chmod 700 OUTPUT_syslog.conf
```

7. Restart the *td-agent* service.

   a. Login to the ESA Web UI.

   b. Navigate to **System** > **Services** > **Misc** > **td-agent**,

   c. Restart the **td-agent** service.

8. Check the status and restart the *rsyslog* server on the remote SIEM system using the following commands.

```
systemctl status rsyslog
systemctl restart rsyslog
```

The logs are now sent to the Audit Store on the ESA and the external SIEM.

4.3.2 Forwarding Logs to a Syslog Server Over TLS

The ESA has logs generated by the Appliances and the Protectors connected to the ESA. You can forward these logs to your syslog server and use the log data for further analysis as per your requirements. Complete the steps provided in this section to forward the ESA logs to an external syslog server using TLS.

For a complete list of plugins that you can forward logs to, refer to *https://www.fluentd.org/plugins/all*.

Before you begin
Ensure that you complete the following prerequisites:

- The external syslog server is available and running.
- You have completed the steps from the section *Sending the Protector Logs to the td-agent*.

➤ To forward logs to the external SIEM:

1. Open the CLI Manager on the Primary ESA.
   a. Login to the CLI Manager of the Primary ESA where you configured the *td-agent* in *Configuring td-agent to Forward Logs to the Audit Store*.
   b. Navigate to **Administration** > **OS Console**.
   c. Enter the root password and select **OK**.
2. Navigate to the */products/uploads* directory using the following command.

   ```
   cd /products/uploads
   ```
3. Obtain the required plugin file using one of the following commands based on your setup.
   - If the Appliance has Internet access, then run the following command.

     ```
     wget https://rubygems.org/downloads/fluent-plugin-syslog-tls-2.0.0.gem
     ```
   - If the Appliance does not have Internet access, then complete the following steps.
     a. Download the *fluent-plugin-syslog-tls-2.0.0.gem* set up file from a system that has Internet and copy it to the Appliance in the */products/uploads* directory.
     b. Ensure that the file downloaded has the *execute* permission.
4. Prepare the required plugin file using the following commands.
   a. Assign the required ownership permissions to the installer using the following command.

   ```
   chown td-agent *.gem
   ```
   b. Assign the required permissions to the software installation directory using the following command.

   ```
   chmod -R 755 /opt/td-agent/embedded/lib/ruby/gems/2.4.0
   ```
   c. Assign ownership of the software installation directory to the required users using the following command.

   ```
   chown -R td-agent:plug /opt/td-agent/embedded/lib/ruby/gems/2.4.0
   ```
5. Install the required plugin file using one of the following commands based on your setup.
   - If the Appliance has Internet access, then run the following command.

     ```
     sudo -u td-agent /opt/td-agent/embedded/bin/fluent-gem install fluent-plugin-syslog-tls
     ```

- If the Appliance does not have Internet access, then run the following command.

  ```
  sudo -u td-agent /opt/td-agent/embedded/bin/fluent-gem install --local /products/
  uploads/fluent-plugin-syslog-tls-2.0.0.gem
  ```

6. Copy the required certificates on the ESA or the Appliance.

   a. Login to the ESA or the Appliance and open the CLI Manager.

   b. Create a directory for the certificates using the following command.

   ```
   mkdir -p /opt/protegrity/td-agent/new_certs
   ```

   c. Update the ownership of the directory using the following command.

   ```
   chown -R td-agent:plug /opt/protegrity/td-agent/new_certs
   ```

   d. Login to the remote SIEM system.

   e. Using a command prompt, navigate to the directory where the certificates are located. For example, ***`cd /etc/pki/tls/certs`***.

   f. Connect to the ESA or Appliance using a file transfer manager. For example, ***`sftp root@ESA_IP`***.

   g. Copy your CA and client certificates to the */opt/Protegrity/td-agent/new_certs* directory using the following command.

   ```
   put CA.pem /opt/protegrity/td-agent/new_certs put client.* /opt/protegrity/td-agent/
   new_certs
   ```

7. Update the configuration files using the following steps.

   a. Navigate to the *config.d* directory using the following command.

   ```
   cd /opt/protegrity/td-agent/config.d
   ```

   b. Backup the existing output file using the following command.

   ```
   cp OUTPUT.conf OUTPUT.conf_backup
   ```

   > **Note:** The system uses the *.conf* extension. To avoid configuration conflicts, ensure that you append the backup file name to the file name extension, for example, *.conf_backup* or *.conf_bkup123*.

   c. Open the *OUTPUT.conf* file using the following command.

   ```
   vi OUTPUT.conf
   ```

   d. Update the following contents in the *OUTPUT.conf* file.

   - Update the *match* tag in the file to *<match *.*.* logdata flulog errorlog>*.
   - Add the following code in the *match* tag in the file:

     ```
        <store>
          @type relabel
          @label @syslogtls
        </store>
     ```

   The final *OUTPUT.conf* file with the updated content is shown here:

   ```
   <filter **>
     @type elasticsearch_genid
     # to avoid duplicate logs
     # https://github.com/uken/fluent-plugin-elasticsearch#generate-hash-id
     hash_id_key _id    # storing generated hash id key (default is _hash)
   </filter>

   <match *.*.* logdata flulog errorlog>
     @type copy

     <store>
         @type elasticsearch
   ```

```
      hosts localhost
      port 9200
      index_name pty_insight_audit
      type_name _doc
      pipeline logs_pipeline
      # adds new data - if the data already exists (based on its id), the op is skipped.
      # https://github.com/uken/fluent-plugin-elasticsearch#write_operation
      write_operation create
      # By default, all records inserted into Elasticsearch get a random _id. This
option allows to use a field in the record as an identifier.
      # https://github.com/uken/fluent-plugin-elasticsearch#id_key
      id_key _id
      scheme https
      ssl_verify true
      ssl_version TLSv1_2
      ca_file /etc/ksa/certificates/plug/CA.pem
      client_cert /etc/ksa/certificates/plug/client.pem
      client_key /etc/ksa/certificates/plug/client.key
      request_timeout 300s # defaults to 5s https://github.com/uken/fluent-plugin-
elasticsearch#request_timeout
      <buffer>
        @type file
        path /opt/protegrity/td-agent/es_buffer
        retry_forever true           # Set 'true' for infinite retry loops.
        flush_mode interval
        flush_interval 60s
        flush_thread_count 8  # parallelize outputs https://docs.fluentd.org/deployment/
performance-tuning-single-process#use-flush_thread_count-parameter
        retry_type periodic
        retry_wait 10s
      </buffer>
  </store>

  <store>
      @type relabel
      @label @triggering_agent
  </store>

  <store>
    @type relabel
    @label @syslogtls
  </store>
</match>
```

> **Note:** Verify that the code in the configuration file is formatted properly. Ensure that there are no additional or trailing spaces, tabs, line breaks, or control characters in the configuration file.

e. Save and close the file.

f. Create the *OUTPUT_syslogTLS.conf* file using the following command.

```
vi OUTPUT_syslogTLS.conf
```

g. Perform the steps from one of the following solution as per your requirement.

- **Solution 1: Forward all logs to the external syslog server:**

  Add the following contents to the *OUTPUT_syslogTLS.conf* file.

```
<label @syslogtls>
  <filter *.*.* logdata errorlog flulog>
    @type record_transformer
    enable_ruby true
    <record>
    severity "${
          case record['level']
          when 'Error'
            'err'
          when 'ERROR'
            'err'
          else
            'info'
```

```
           end
         }"

    #local0 -Protection
    #local1 -Application
    #local2 -System
    #local3 -Kernel
    #local4 -Policy
    #local5 -User Defined
    #local6 -User Defined
    #local7 -Others
    #local5 and local6 can be defined as per the requirement

    facility "${
          case record['logtype']
          when 'Protection'
            'local0'
          when 'Application'
            'local1'
          when 'System'
             'local2'
          when 'Kernel'
             'local3'
          when 'Policy'
             'local4'
          else
             'local7'
          end
               }"

    #noHostName - can be changed by customer
    hostname ${record["origin"] ? (record["origin"]["hostname"] ? record["origin"]
["hostname"] : "noHostName") : "noHostName" }
    </record>
  </filter>

  <match *.*.* logdata errorlog flulog>
    @type copy
    <store>
      @type syslog_tls
      host <IP_of_the_rsyslog_server_host>
      port 601
      client_cert /opt/protegrity/td-agent/new_certs/client.pem
      client_key /opt/protegrity/td-agent/new_certs/client.key
      ca_cert /opt/protegrity/td-agent/new_certs/CA.pem
      verify_cert_name true
      severity_key severity
      facility_key facility
      hostname_key hostname
      <format>
      @type json
      </format>
    </store>
  </match>
</label>
```

> **Note:** Verify that the code in the configuration file is formatted properly. Ensure that there are no additional or trailing spaces, tabs, line breaks, or control characters in the configuration file.

- **Solution 2: Forward only the protection logs to the external syslog server:**

  Add the following contents to the *OUTPUT_syslogTLS.conf* file.

```
<label @syslogtls>
  <filter logdata>
    @type record_transformer
    enable_ruby true
    <record>
    severity "${
          case record['level']
          when 'Error'
            'err'
          when 'ERROR'
            'err'
          else
            'info'
           end
         }"

    #local0 -Protection
    #local1 -Application
    #local2 -System
    #local3 -Kernel
    #local4 -Policy
    #local5 -User Defined
    #local6 -User Defined
    #local7 -Others
    #local5 and local6 can be defined as per the requirement

    facility "${
          case record['logtype']
          when 'Protection'
            'local0'
          when 'Application'
            'local1'
          when 'System'
             'local2'
          when 'Kernel'
             'local3'
          when 'Policy'
             'local4'
          else
             'local7'
          end
               }"

    #noHostName - can be changed by customer
    hostname ${record["origin"] ? (record["origin"]["hostname"] ? record["origin"]
["hostname"] : "noHostName") : "noHostName" }
    </record>
  </filter>

  <match logdata>
    @type copy
    <store>
      @type syslog_tls
      host <IP_of_the_rsyslog_server_host>
      port 601
      client_cert /opt/protegrity/td-agent/new_certs/client.pem
      client_key /opt/protegrity/td-agent/new_certs/client.key
      ca_cert /opt/protegrity/td-agent/new_certs/CA.pem
      verify_cert_name true
      severity_key severity
      facility_key facility
      hostname_key hostname
      <format>
      @type json
      </format>
    </store>
  </match>
</label>
```

Note: Ensure that you specify the *<IP_of_the_rsyslog_server_host>* in the file. Verify that the code in the configuration file is formatted properly. Ensure that there are no additional or trailing spaces, tabs, line breaks, or control characters in the configuration file.

For more information about the formatting the output, navigate to *https://docs.fluentd.org/configuration/format-section*.

The logs are formatted using the *rfc 3164* format that is commonly used.

> For more information about the rfc format, navigate to *https://datatracker.ietf.org/doc/html/rfc3164*.

h. Save and close the file.

i. Update the permissions for the file using the following commands.

```
chown td-agent:td-agent OUTPUT_syslogTLS.conf
chmod 700 OUTPUT_syslogTLS.conf
```

8. Restart the *td-agent* service.

   a. Login to the ESA Web UI.

   b. Navigate to **System** > **Services** > **Misc** > **td-agent**,

   c. Restart the **td-agent** service.

The logs are now sent to the Audit Store on the ESA and the external SIEM over TLS.

Chapter 5

Troubleshooting

This section describes the problems that you might face while working with logging and the solutions or workarounds to resolve those problems.

5.1 Known Issues for the td-agent

A list of known issues with their solution or workaround are provided here. The steps provided to resolve the known issues ensure that your product does not display errors or crash.

- **Known Issue:** The **Buffer overflow** error appears in the */var/log/td-agent/td-agent.log* file.

  Description: When the total size of the files in *td-agent* buffer */opt/protegrity/td-agent/es_buffer* directory reaches the default maximum limit of 64 GB, then the **Buffer overflow** error appears.

  Resolution:

  Add the *total_limit_size* parameter to increase the buffer limit in the *OUTPUT.conf* file using the following steps.

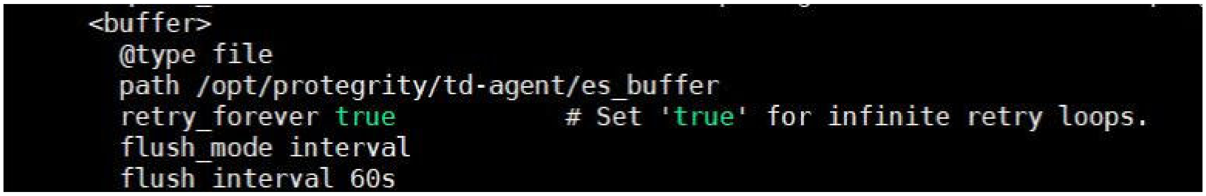
  1. Login to the CLI Manager of the ESA node.
  2. Navigate to **Administration** > **OS Console**.
  3. Stop the *td-agent* service using the following command:

     ```
     /etc/init.d/td-agent stop
     ```

     > **Note:** You can also stop the service by logging into the ESA Web UI, navigating to **System** > **Services**, and stopping the **td-agent** service under **Misc**.

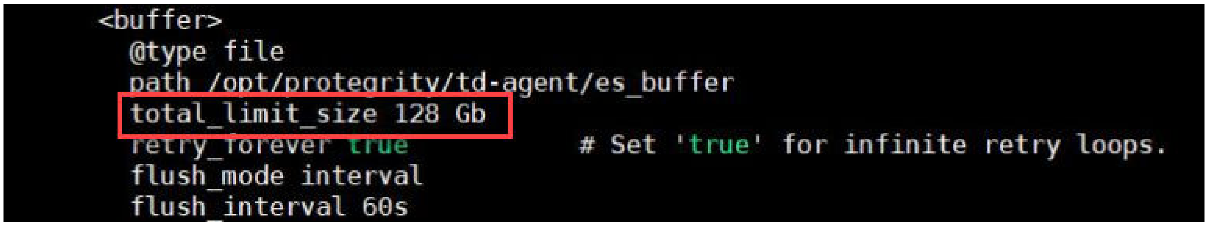
  4. Navigate to the `/opt/protegrity/td-agent/config.d` directory.
  5. Open the `OUTPUT.conf` file.
  6. Add the *total_limit_size* parameter in the buffer section of the `OUTPUT.conf` file.

     In this example, the *total_limit_size* is doubled to **128 GB**.



```
    <buffer>
      @type file
      path /opt/protegrity/td-agent/es_buffer
      retry_forever true            # Set 'true' for infinite retry loops.
      flush_mode interval
      flush_interval 60s
```

Figure 5-1: Before Update



```
   <buffer>
     @type file
     path /opt/protegrity/td-agent/es_buffer
     total_limit_size 128 Gb
     retry_forever true              # Set 'true' for infinite retry loops.
     flush_mode interval
     flush_interval 60s
```

Figure 5-2: Updated `OUTPUT.conf` *File*

7. Save the file.
8. Start the *td-agent* service using the following command:

```
/etc/init.d/td-agent start
```

> **Note:** You can also start the service by logging into the ESA Web UI, navigating to **System** > **Services**, and starting the **td-agent** service under **Misc**.

- **Known Issue:** The **Too many open files** error appears in the */var/log/td-agent/td-agent.log* file.

  Description: When the total number of files in the *td-agent* buffer */opt/protegrity/td-agent/es_buffer* directory reaches the maximum limit, then the **Too many open files** error appears.

  Resolution:

  Change the limit for the maximum number of open files for the *td-agent* service in the */etc/init.d/td-agent* file using the following steps.

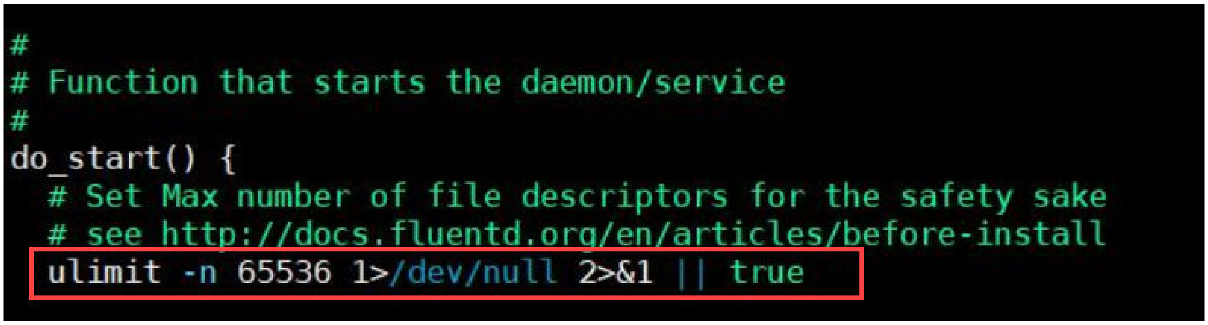
  1. Login to the CLI Manager of the ESA node.
  2. Navigate to **Administration** > **OS Console**.
  3. Stop the *td-agent* service using the following command:

```
/etc/init.d/td-agent stop
```

> **Note:** You can also stop the service by logging into the ESA Web UI, navigating to **System** > **Services**, and stopping the **td-agent** service under **Misc**.

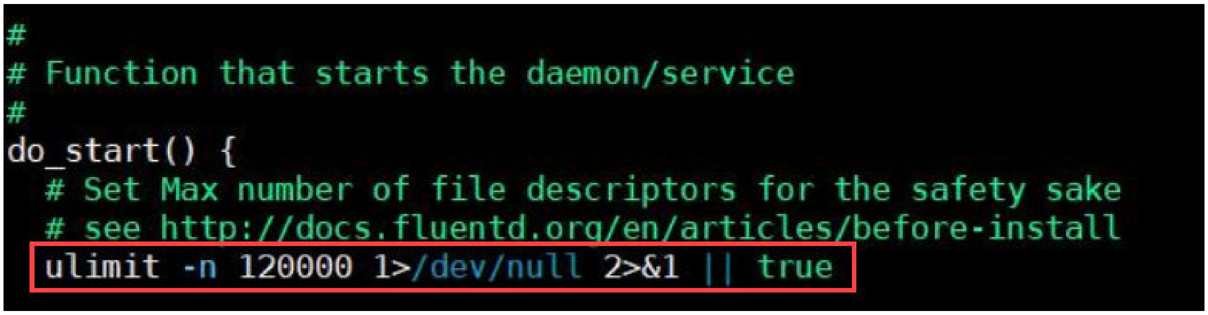
4. Navigate to the `/etc/init.d` directory.
5. Open the `td-agent` file.
6. Change the *ulimit*.

   In this example, the *ulimit* is increased to **120000**.



```
#
# Function that starts the daemon/service
#
do_start() {
  # Set Max number of file descriptors for the safety sake
  # see http://docs.fluentd.org/en/articles/before-install
  ulimit -n 65536 1>/dev/null 2>&1 || true
```

Figure 5-3: Before Update



```
#
# Function that starts the daemon/service
#
do_start() {
  # Set Max number of file descriptors for the safety sake
  # see http://docs.fluentd.org/en/articles/before-install
  ulimit -n 120000 1>/dev/null 2>&1 || true
```

Figure 5-4: Updated `td-agent` *File*

7. Save the file.
8. Start the *td-agent* service using the following command:

```
/etc/init.d/td-agent start
```

> **Note:** You can also start the service by logging into the ESA Web UI, navigating to **System** > **Services**, and starting the **td-agent** service under **Misc**.

5.2 Known Issues for the Log Forwarder

A list of known issues with their solution or workaround are provided here. The steps provided to resolve the known issues ensure that your product does not display errors or stops responding.

Known Issue: The Protector is unable to reconnect to a Log Forwarder after it is restarted

Description: This issue occurs whenever you have a Proxy server between a Protector and a Log Forwarder. When the Log Forwarder is stopped, the connection between the Protector and the Proxy server is still open, even though the connection between the Proxy server and the Log Forwarder is closed. As a result, the Protector continues sending audit files to the Proxy server. This results in loss of the audit files. Whenever the Log Forwarder is restarted, the Protector is unable to reconnect to the Log Forwarder.

This issue is applicable to all the Protectors where the Log Forwarder is not running on the local host machine. For example, this issue is applicable to AIX or z/OS protectors because the Log Forwarder is not running on the same machine where the Protectors have been installed. This issue also occurs if you have a Load Balancer or a Firewall between the Protector and the Log Forwarder, instead of a Proxy server.

Resolution: Remove the Proxy server or ensure that you configure the Proxy server in a way that the connection between the Protector and the Proxy server is stopped as soon as the Log Forwarder is stopped. This ensures that whenever the Log Forwarder is restarted, the Protector reconnects with the Log Forwarder and continues to send the audits to the Log Forwarder without any data loss.

For more information about configuring the Proxy server, contact your IT administrator.