



Certificate Management Guide 9.1.0.5

Created on: Nov 19, 2024

Notice

Copyright

Copyright © 2004-2024 Protegrity Corporation. All rights reserved.

Protegrity products are protected by and subject to patent protections;

Patent: <https://www.protegrity.com/patents>.

The Protegrity logo is the trademark of Protegrity Corporation.

NOTICE TO ALL PERSONS RECEIVING THIS DOCUMENT

Some of the product names mentioned herein are used for identification purposes only and may be trademarks and/or registered trademarks of their respective owners.

Windows, Azure, MS-SQL Server, Internet Explorer and Internet Explorer logo, Active Directory, and Hyper-V are registered trademarks of Microsoft Corporation in the United States and/or other countries.

Linux is a registered trademark of Linus Torvalds in the United States and other countries.

UNIX is a registered trademark of The Open Group in the United States and other countries.

SCO and SCO UnixWare are registered trademarks of The SCO Group.

Sun, Oracle, Java, and Solaris are the registered trademarks of Oracle Corporation and/or its affiliates in the United States and other countries.

Teradata and the Teradata logo are the trademarks or registered trademarks of Teradata Corporation or its affiliates in the United States and other countries.

Hadoop or Apache Hadoop, Hadoop elephant logo, Hive, and Pig are trademarks of Apache Software Foundation.

Cloudera and the Cloudera logo are trademarks of Cloudera and its suppliers or licensors.

Hortonworks and the Hortonworks logo are the trademarks of Hortonworks, Inc. in the United States and other countries.

Greenplum Database is the registered trademark of VMware Corporation in the U.S. and other countries.

Pivotal HD is the registered trademark of Pivotal, Inc. in the U.S. and other countries.

PostgreSQL or Postgres is the copyright of The PostgreSQL Global Development Group and The Regents of the University of California.

AIX, DB2, IBM and the IBM logo, and z/OS are registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide.

Utimaco Safeware AG is a member of the Sophos Group.

Xen, XenServer, and Xen Source are trademarks or registered trademarks of Citrix Systems, Inc. and/or one or more of its subsidiaries, and may be registered in the United States Patent and Trademark Office and in other countries.

VMware, the VMware “boxes” logo and design, Virtual SMP and VMotion are registered trademarks or trademarks of VMware, Inc. in the United States and/or other jurisdictions.

Amazon Web Services (AWS) and AWS Marks are the registered trademarks of Amazon.com, Inc. in the United States and other countries.

HP is a registered trademark of the Hewlett-Packard Company.

HPE Ezmeral Data Fabric is the trademark of Hewlett Packard Enterprise in the United States and other countries.

Dell is a registered trademark of Dell Inc.

Novell is a registered trademark of Novell, Inc. in the United States and other countries.

POSIX is a registered trademark of the Institute of Electrical and Electronics Engineers, Inc.

Mozilla and Firefox are registered trademarks of Mozilla foundation.

Chrome and Google Cloud Platform (GCP) are registered trademarks of Google Inc.

Swagger Specification and all public tools under the swagger-api GitHub account are trademarks of Apache Software Foundation and licensed under the Apache 2.0 License.

Table of Contents

Copyright.....	2
Chapter 1 Introduction to This Guide.....	5
1.1 Sections contained in this Guide.....	5
1.2 Accessing the Protegrity documentation suite.....	5
Chapter 2 Certificates in ESA.....	6
Chapter 3 Certificate Management in ESA.....	8
3.1 Certificate Repository.....	9
3.2 Uploading Certificates.....	10
3.3 Uploading Certificate Revocation List (CRL).....	12
3.4 Manage Certificates.....	13
3.5 To change certificates.....	14
3.6 To change CRL.....	15
Chapter 4 Certificate Management for Protectors.....	17
Chapter 5 Certificates in DSG.....	18
5.1 Inbound Certificates.....	19
5.1.1 HTTPS using default certificates.....	19
5.1.2 TLS Mutual Authentication.....	20
5.1.3 SFTP.....	20
5.2 Outbound Certificates.....	20
Chapter 6 Replicating Certificates in a Trusted Appliance Cluster (TAC).....	21
Chapter 7 Audit Store Certificates.....	23
7.1 Using Custom Certificates in the Audit Store.....	26
Chapter 8 Validating Certificates.....	28
Appendix 9 Appendix A: Certificates-related Terminology.....	31

Chapter 1

Introduction to This Guide

1.1 Sections contained in this Guide

1.2 Accessing the Protegrity documentation suite

This document provides information on the certificate management details for ESA, Protectors and Data Security Gateway (DSG). It also explains the replication of certificates in a Trusted Appliance Cluster (TAC) and the method to validate the certificates using SSL commands.

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 *Certificates in ESA* provides an overview about the certificates in ESA.
- Section *Certificate Management in ESA* provides information about handling multiple certificates in ESA.
- Section *Certificate Management for Protectors* provides information about the certificate-based authentication between ESA and the Protectors.
- Section *Certificates in DSG* provides information on how you can enable secure communication between Data Security Gateway (DSG) and server or client.
- Section *Replicating Certificates in a Trusted Appliance Cluster (TAC)* explains the certificate replication mechanism for appliances in a Trusted Appliance Cluster (TAC).
- Section *Validating Certificates* explains how you can validate the certificates using SSL commands.
- *Appendix A: Certificates-related Terminology* introduces terminology related to certificates that can help you understand Protegrity Certificate Management.

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

Certificates in ESA

Digital certificates are used to encrypt online communication and authentication between two entities. For two entities exchanging sensitive information, the one that initiates the request for exchange can be called the client and the one that receives the request and constitutes the other entity can be called the server.

The authentication of both the client and the server involves the use of digital certificates issued by the trusted Certificate Authorities (CAs). The client authenticates itself to a server using its client certificate. Similarly, the server also authenticates itself to the client using the server certificate. Thus, certificate-based communication and authentication involves a client certificate, server certificate, and a certifying authority that authenticates the client and server certificates.

Protegrity client and server certificates are self-signed by Protegrity. However, you can replace them by certificates signed by a trusted and commercial CA. These certificates are used for communication between various components in ESA.

The certificate support in Protegrity involves the following:

- The ability to replace the self-signed Protegrity certificates with the CA based certificates.
- The retrieval of username from client certificates for authentication of user information during policy enforcement.
- The ability to download the server's CA certificate and upload it to a certificate trust store to trust the server certificate for communication with ESA.

Points to remember when uploading the certificates:

- ESA supports the upload of certificates with strength equal to 4096 bits. You can upload a certificate with strength less than 4096 bits but the system will show you a warning message.

Caution: Custom certificates for the Audit Store must be generated using a 4096 bit key.

- When uploading, make sure the certificate version is v3. Uploading certificates with version lower than v3 is not supported.

The various components within the Protegrity Data Security Platform that communicate with and authenticate each other through digital certificates are:

- ESA Web UI and ESA
- ESA and Protectors
- Protegrity Appliances and external REST clients

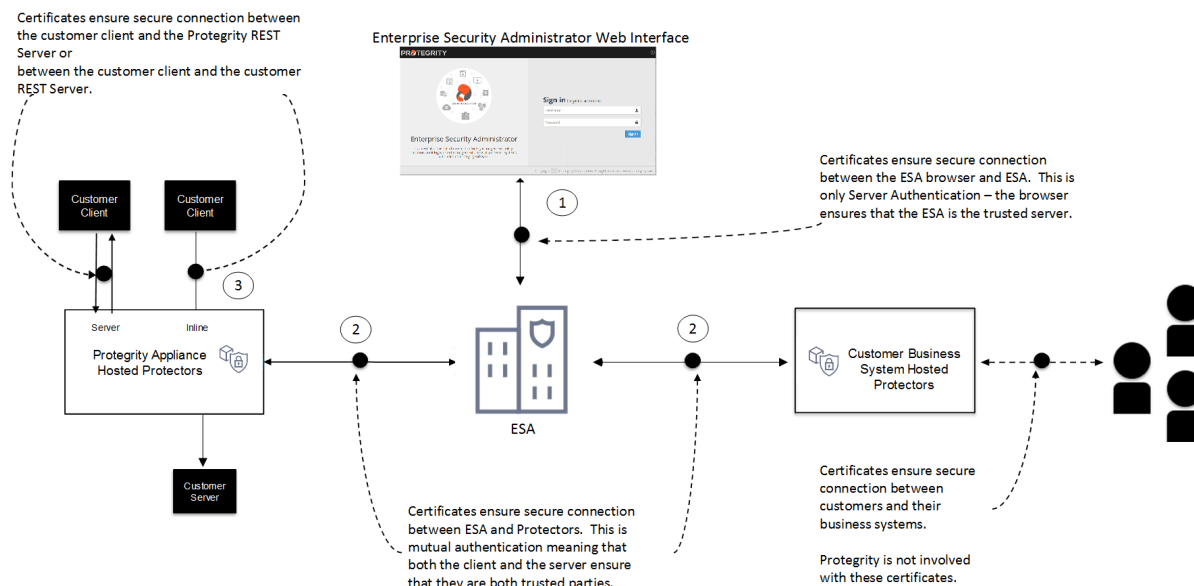


Figure 2-1: Using Certificates with Protegrity

As illustrated in the figure, the use of certificates within the Protegrity systems involves the following:

1. Communication between ESA Web UI and ESA

In case of a communication between the ESA Web UI and ESA, ESA provides its server certificate to the browser. In this case, it is only server authentication that takes place in which the browser ensures that ESA is the trusted server.

2. Communication between ESA and Protectors

In case of a communication between ESA and Protectors, certificates are used to mutually authenticate both the entities. The server and the client i.e. ESA and the Protector respectively ensure that both are trusted entities. The Protectors could be hosted on customer business systems or it could be a Protegrity Appliance.

3. Communication between Protegrity Appliances and external REST clients

Certificates ensure the secure communication between the customer client and Protegrity REST server or between the customer client and the customer REST server.

Chapter 3

Certificate Management in ESA

3.1 Certificate Repository

3.2 Uploading Certificates

3.3 Uploading Certificate Revocation List (CRL)

3.4 Manage Certificates

3.5 To change certificates

3.6 To change CRL

This section provides information about how the certificates are managed in ESA.

When ESA is installed, it generates default self-signed certificates in X.509 v3 PEM format. These certificates are:

- CA Certificate – This consists of the *CA.pem* and *CA.key* file.
- Server Certificate - This consists of the *server.pem* and *server.key* file.
- Client Certificate - This consists of the *client.pem* and *client.key* file.

The services that use and manage these certificates are:

- Management – It is the service which manages certificate based communication and authentication between ESA and its internal components like LDAP, Appliance queue, protectors, etc.
- Web Services – It is the service which manages certificate based communication and authentication between ESA and external clients (REST).
- Consul – It is the service that manages certificates between the Consul server and the Consul client.

Note: If you are working with the certificates, then due to the security enhancement for the latest browser versions, the *KeyUsage* parameter must be configured. Ensure that *KeyUsage= Digital Signature* parameter is *Enabled* for the certificates.

ESA provides a certificate manager where you can manage the default certificates and also upload your own CA-signed certificates. This manager comprises of two components which are as follows:

- Certificate Repository
- Manage Certificates

Note: When creating a CA-signed client certificate which you want use in ESA, it is mandatory that you keep the CN attribute of the client certificate to be “*Protegrity Client*”. Ensure that the “*Protegrity Client*” is appended at the end of the subject line.

Note:

If there are CA cross-sign certificates with the AddTrust legacy, then you must upload the active intermediate certificates from the **Manage Certificates** page. If the expired certificates are present in the certificate chain, then it might lead to failures.

For more information about upload the updated certificates, refer to [To change certificates](#).

For more information about the CA cross-sign certificates with the AddTrust legacy, refer to the following link:

<https://support.sectigo.com/articles/Knowledge/Sectigo-AddTrust-External-CA-Root-Expiring-May-30-2020>

If other attributes, such as email address or name, are appended to the CN attribute, then you perform the following steps to set the *CN* attribute to *Protegrity Client*.

For example, if the CN attribute is set as *Protegrity Client/emailAddress=user@abc.com*, then you must remove the other attributes appended after the / delimiter.

1. In the ESA CLI Manager, navigate to **Administration > OS Console**
2. Run the following command:

```
vi /etc/ksa/pty_get_username_from_certificate.py
```

3. Comment the line containing the CN attribute and enter the following regular expression:

```
REG_EX_GET_VAL_AFTER_CN = "CN=((?:,|[\^,])*)(?:,|$)"
```

4. Save the changes.
5. Navigate to **Administration > Services**
6. Restart the *Service Dispatcher* service.

3.1 Certificate Repository

The certificate repository is a store / repository where ESA stores all the certificates. It gives you the capability to upload certificates to the repository. It also allows you to upload Certificate Revocation List(CRL).

A Certificate Revocation List(CRL) is a list containing entries of digital certificates that are no longer trusted as they are revoked by the issuing Certificate Authority(CA) for any of the following possible reasons:

- The certificate is expired.
- The certificate is compromised.
- The certificate is lost.
- The certificate is breached.

CRLs are used to avoid the usage of certificates that are revoked and are used at various endpoints including the web browsers. When a browser makes a connection to a site, the identity of the site owner is checked using the server's digital certificate. Also, the validity of the digital certificate is verified by checking whether the digital certificate is not listed in the Certificate Revocation List. If the certificate entry is present in this list, then the authentication for that revoked certificate fails.

The Certificate Repository screen is accessible from the **ESA Web UI**, navigate to **Settings > Network > Certificate Repository**. The following figure and table provides the details about the Certificate Repository screen.

Certificate Repository					
ID	Type	Archive Time	Status	Description	Action
78773ee5.0	certificate, key	February 01 2022, 16:13:59	In use	Audit Store Client Cert and Key for plug	
59a91e26.0	certificate, key	February 01 2022, 15:49:21	In use	System initial server certificate and key.	
f7320cdf.0	certificate, key	February 01 2022, 15:49:21	In use	System initial client certificate and key.	
6f2e6d24.0	certificate, key	February 01 2022, 16:10:11	In use	Audit Store Client Cert and Key for es_cluster	
c043178a.0	certificate, key	February 01 2022, 16:16:51	In use	Audit Store Client Cert and Key for ian	
<input type="checkbox"/> 6c78352b.1	certificate, key	February 02 2022, 10:38:10	Expired	Audit Cert & Key	
<input type="checkbox"/> 6c78352b.0	certificate, key	February 02 2022, 10:37:14	Expired	Client Key	

Show entries

Figure 3-1: Certificate Repository Screen

Table 3-1: Certificate Repository Details

Callout	Action	Description
1	ID	ESA generated ID for the certificate and key file.
2	Type	Specifies the type of the file i.e. certificate, key, or CRL.
3	Archive time	It is the timestamp when the certificate was uploaded to the certificate repository.
4	Status	<p>This column shows the status of the certificate in the Certificate Repository, which can be:</p> <ul style="list-style-type: none"> In use - When you hover over this status, it displays the function or the system services (Management or Web Services) using the certificates expired - The certificate is expired. expires in -- days - The number of days left for the certificate to expire.
5	Description	Displays the description given by the user when the certificate is uploaded to Certificate Repository. It is recommended to provide a meaningful description while uploading a certificate.
6	Delete	<p>Allows you to delete multiple selected certificates or CRLs from the Certificate Repository.</p> <p>Note: Only <i>expired</i> certificates or CRLs can be deleted.</p>
7	Information	Provides additional information or details about a certificate and its private key (if uploaded).
8	Delete	<p>Allows you to delete the certificate or CRL from the Certificate Repository.</p> <p>Note: Only <i>expired</i> certificates or CRLs can be deleted.</p>

3.2 Uploading Certificates

This section allows you to upload certificates or CRL through the Certificate Repository screen.

► To upload certificates or CRL:

1. On the ESA Web UI, navigate to **Settings > Network > Certificate Repository**.

Certificate Repository						Upload New Files	
ID	Type	Archive Time	Status	Description	Action		
78773ee5.0	certificate, key	February 01 2022, 16:13:59	In use	Audit Store Client Cert and Key for plug			
59a91e26.0	certificate, key	February 01 2022, 15:49:21	In use	System initial server certificate and key.			
f7320cdf.0	certificate, key	February 01 2022, 15:49:21	In use	System initial client certificate and key.			
c043178a.0	certificate, key	February 01 2022, 16:16:51	In use	Audit Store Client Cert and Key for lan			
<input type="checkbox"/> 6c78352b.1	certificate, key	February 02 2022, 10:38:10	Expired	Audit Cert & Key			
<input type="checkbox"/> 6c78352b.0	certificate, key	February 02 2022, 10:37:14	Expired	Client Key			
Show 40 entries							

Figure 3-2: Certificate Repository screen

2. Click **Upload New Files**.

The *Upload new file to repository* dialog box appears.

Upload new file to repository

* File Type

☒ Certificate/Key
 ☐ Certificate Revocation List

* Certificate file

Server.crt

Key file ⓘ

Server.key

Description

Client Certificate and Key.

Figure 3-3: Upload Certificate screen

3. Click **Certificate/Key** to upload a certificate file and a private key file.

Caution: Certificates have a public and private key. The public key is mentioned in the certificate and as a best practice the private key is maintained as a separate file. In ESA, you can upload either the certificate file or both certificate and private key file together. In ESA Certificate Repository, it is mandatory to upload the certificate file.

Caution: If the private key file is inside the certificate, then you have the option to upload just the Certificate file. The DSKs are identified using the UID column that displays the Key Id.

Note:

It is recommended to use private key with a length of 4096-bit.

- Click **Choose File** to select both certificate and key files.
- Enter the required description in the **Description** text box.
- Click **Upload**.

Caution: If the private key is encrypted, a prompt to enter the passphrase will be displayed.

The certificate and the key file is saved in repository and the *Certificate Repository* screen is updated with the details.

Note:

When you upload a private key that is protected with a passphrase, the key and the passphrase are stored in the hard disk. The passphrase is stored in an encrypted form using a secure algorithm. The passphrase and the private key are also stored in the system memory. The services, such as Apache, RabbitMQ, or LDAP, access this system memory to load the certificates.

If you upload a private key that does not have a passphrase, the key is stored in the system memory. The services, such as Apache, RabbitMQ, or LDAP access the system memory to load the certificates.

Note:

If your organization is using a proxy server to connect to the Internet, ensure that the required custom certificates of that server are uploaded in ESA from the **Certificate Repository** screen.

3.3 Uploading Certificate Revocation List (CRL)

This section explains the steps to upload the Certificate Revocation List (CRL) through the Certificate Repository screen.

► To upload CRL:

- On the ESA Web UI, navigate to **Settings > Network > Certificate Repository**.

Certificate Repository						Upload New Files	
ID	Type	Archive Time	Status	Description	Action		
78773ee5.0	certificate, key	February 01 2022, 16:13:59	In use	Audit Store Client Cert and Key for plug			
59a91e26.0	certificate, key	February 01 2022, 15:49:21	In use	System initial server certificate and key.			
f7320cdf.0	certificate, key	February 01 2022, 15:49:21	In use	System initial client certificate and key.			
c043178a.0	certificate, key	February 01 2022, 16:16:51	In use	Audit Store Client Cert and Key for ian			
<input type="checkbox"/> 6c78352b.1	certificate, key	February 02 2022, 10:38:10	Expired	Audit Cert & Key			
<input type="checkbox"/> 6c78352b.0	certificate, key	February 02 2022, 10:37:14	Expired	Client Key			
Show <input type="text" value="40"/> entries							

Figure 3-4: Certificate Repository screen

- Click **Upload New Files**.
The *Upload new file to repository* dialog box appears.
- Click **Certificate Revocation List** to upload a CRL file.

Upload new file to repository

* File Type

☐ Certificate/Key

☒ Certificate Revocation List

* CRL file

Choose File

No file chosen

Description

Upload

Cancel

Figure 3-5: Upload CRL screen

4. Click **Choose File** to select a CRL file.
5. Enter the required description in the **Description** text box.
6. Click **Upload**.

A confirmation message is displayed and the CRL is uploaded to ESA.

3.4 Manage Certificates

The Manage Certificates module is used to select the certificates that you want to make active and have ESA use them for its communication with various internal components. It allows you to select the certificate revocation list that you want activated. The following figure and table provides the actions available from the Manage Certificates screen.

Manage Certificates

Management

Revocation List

Change Certificates

Client Certificate

Version: V3
Valid To: Sun Jan 25 07:18:55 2032
Subject: O: Protegrity Inc., C: US, CN: Protegrity Client

Bits: 4096
Type: RSA

CA Certificate

Version: V3
Valid To: Sun Jan 25 07:18:54 2032
Subject: O: Protegrity Inc., C: US, CN: Protegrity Root CA - cre...

Bits: 4096
Type: RSA

Server Certificate

Version: V3
Valid To: Sun Jan 25 07:18:54 2032
Subject: O: Protegrity Inc., C: US, CN: protegrity-framework259

Bits: 4096
Type: RSA




1

2

3

Figure 3-6: Additional actions: Manage Certificates screen

Table 3-2: Manage Certificates Action Items

Callout	Action	Description
1	 Hover over the Help icon	Gives information about Management and Web Services groups.
2	 Download server's CA certificate	Download the server's CA certificate. You can download only the server's CA certificate and upload it to another certificate trust store to trust the server certificate for communication with ESA.
3	 Hover over the icon	Gives additional information or details about a certificate.

3.5 To change certificates

This section allows you to change certificates through the Manage Certificates screen.

Before you begin

Note: If you are working with the certificates, then due to the security enhancement for the latest browser versions, the *KeyUsage* parameter must be configured. Ensure that *KeyUsage= Digital Signature* parameter is *Enabled* for the certificates.

► To change certificates:

1. On the ESA Web UI, navigate to **Settings > Network > Manage Certificates**.

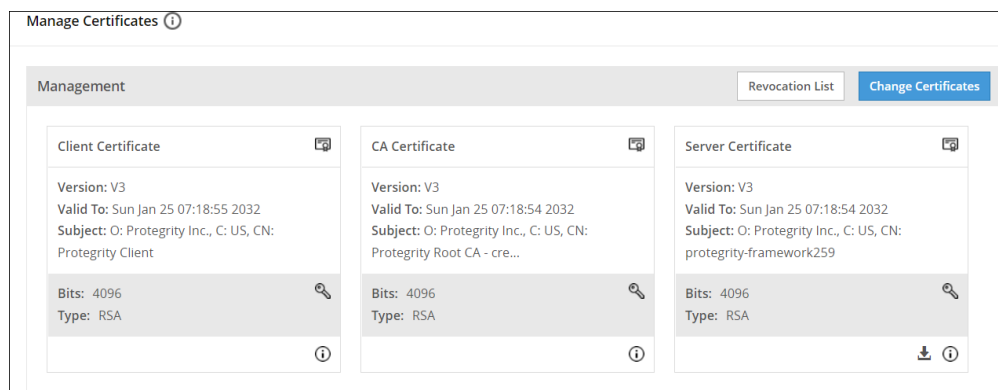


Figure 3-7: Manage Certificates screen

2. Click **Change Certificates**.
3. Select the check box next to the CA Certificate that you want to set as active.

Caution: This section shows server, client, and CA certificates together. However, ensure that you select only the required certificates in their respective screens. You can select multiple CA certificates for *ESA Management* and *Web Services* section. ESA allows you to have only one server and one client active at any given time.

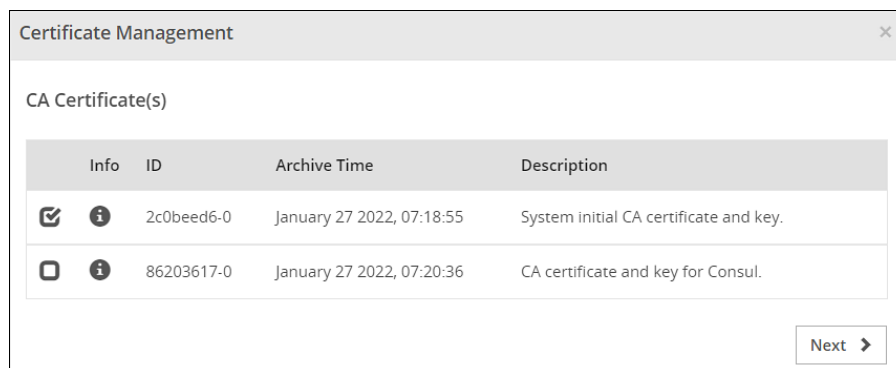


Figure 3-8: Change CA Certificates screen

4. Click **Next**.
5. Select the check box next to the Server Certificate that you want to set as active.

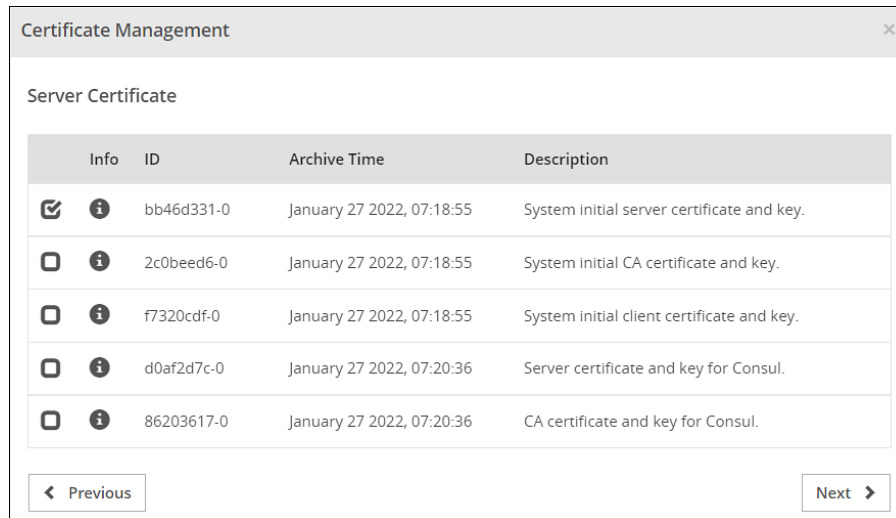


Figure 3-9: Change Server Certificates screen

6. Click **Next**.
- The **Client Certificate** section appears
7. Select the check box next to the Client Certificate that you want to set as active.

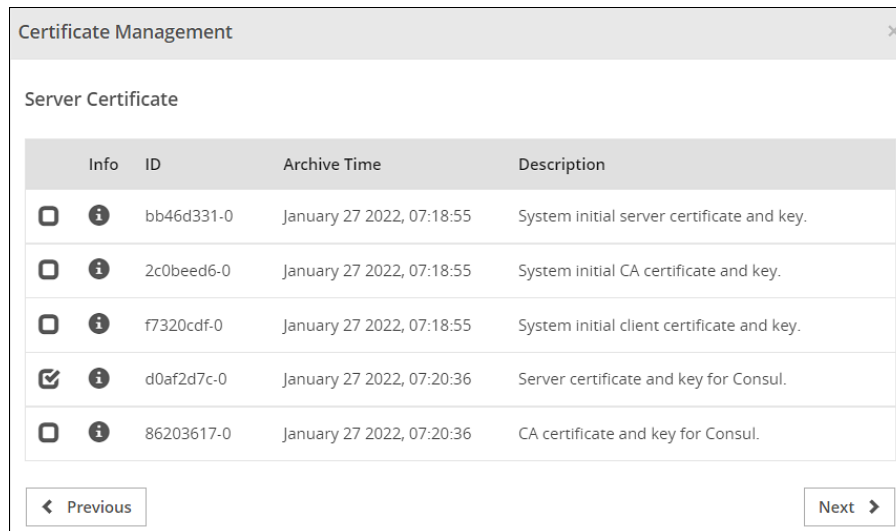


Figure 3-10: Change Client Certificates screen

8. Click **Apply**.
- The following message appears:
The system Management Certificates will be changed and a re-login maybe required. Do you want to continue?
9. Click **Yes**.
- A confirmation message appears and the active certificates are displayed on the screen.

Caution: When you upload a server certificate to ESA and make it active, you are logged out of ESA Web UI as the browser does not trust the new CA signed server certificate. You must login again for the browser to get the new server certificate and to use it for all further communications.

3.6 To change CRL

This section allows you to change the CRL through the Manage Certificates screen.

Before you begin

► To change CRL:

1. On the ESA Web UI, navigate to **Settings > Network > Manage Certificates**.

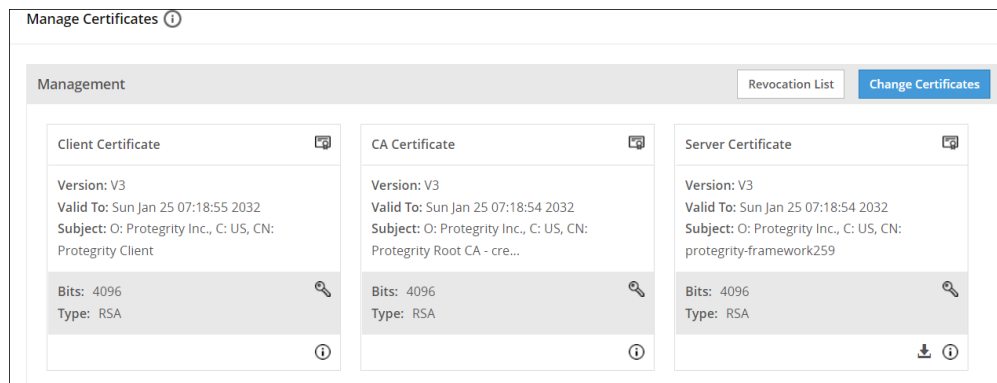


Figure 3-11: Manage Certificates screen

2. Click **Revocation List**.
The *Certificate Revocation List* dialog box appears.
3. Select the **Enable Certificate Revocation List** check box.
4. Select the check box next to the CRL file that you want to set as active.

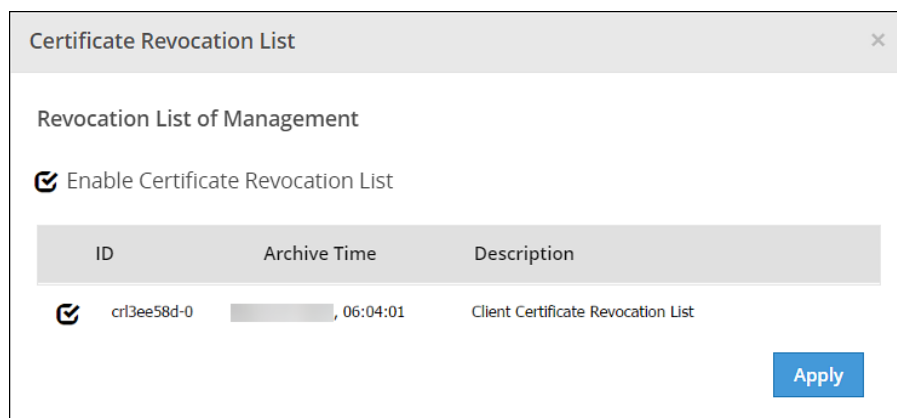


Figure 3-12: Change CRL screen

5. Click **Apply**.
A confirmation message appears.

Chapter 4

Certificate Management for Protectors

As part of the installation process of the v7.x.x PEP Server on the protection endpoint, a list of certificate files are automatically downloaded.

The following files are available in the /opt/protegrity/defiance_dps/data directory.

- CA.pem
- authesa.plm
- cert.key
- cert.pem
- certkeyup.bin

If the active set of certificates change in ESA, then you must re-download the certificates using the GetCertificates utility. The following *GetCertificates* utility must be executed to download the active set of certificates from ESA:

```
GetCertificates -u username [-h hostname] [-p portno] [-d directory]
```

Chapter 5

Certificates in DSG

5.1 Inbound Certificates

5.2 Outbound Certificates

The Data Security Gateway (DSG) acts as an intermediary between the server and clients. DSG is equipped with a set of certificates to enable secure communication between DSG and server or client.

During the install process of DSG, a series of self-signed SSL Certificates are generated. You may use it in a non-production environment. It is recommended to use your own certificate for production use.

When you install a DSG node, the following types of certificates and keys are generated:

- CA certificate –This consists of *CA.pem*, *CA.crt*, and *CA.key* file.
- Server Certificate - This consists of *service.pem*, *service.crt*, and *service.key* file.
- Client Certificate - This consists of *client.pem*, *client.crt*, and *client.key* file.
- Admin Certificate – This consists of *admin.pem*, *admin.crt* and *admin.key*.
- Admin Client Certificate- This consists of *admin_client.crt* and *admin_client.key*.

The certificates in DSG are classified as Inbound Certificates and Outbound Certificates. You must use Inbound certificates for secure communication between client and DSG. In setups, such as Software as a Service (SaaS), where DSG communicates with a SaaS that is not part of the on-premise setup or governed by an enterprise networking perimeter, Outbound certificates are employed.

The following image illustrates the flow of certificates in DSG.

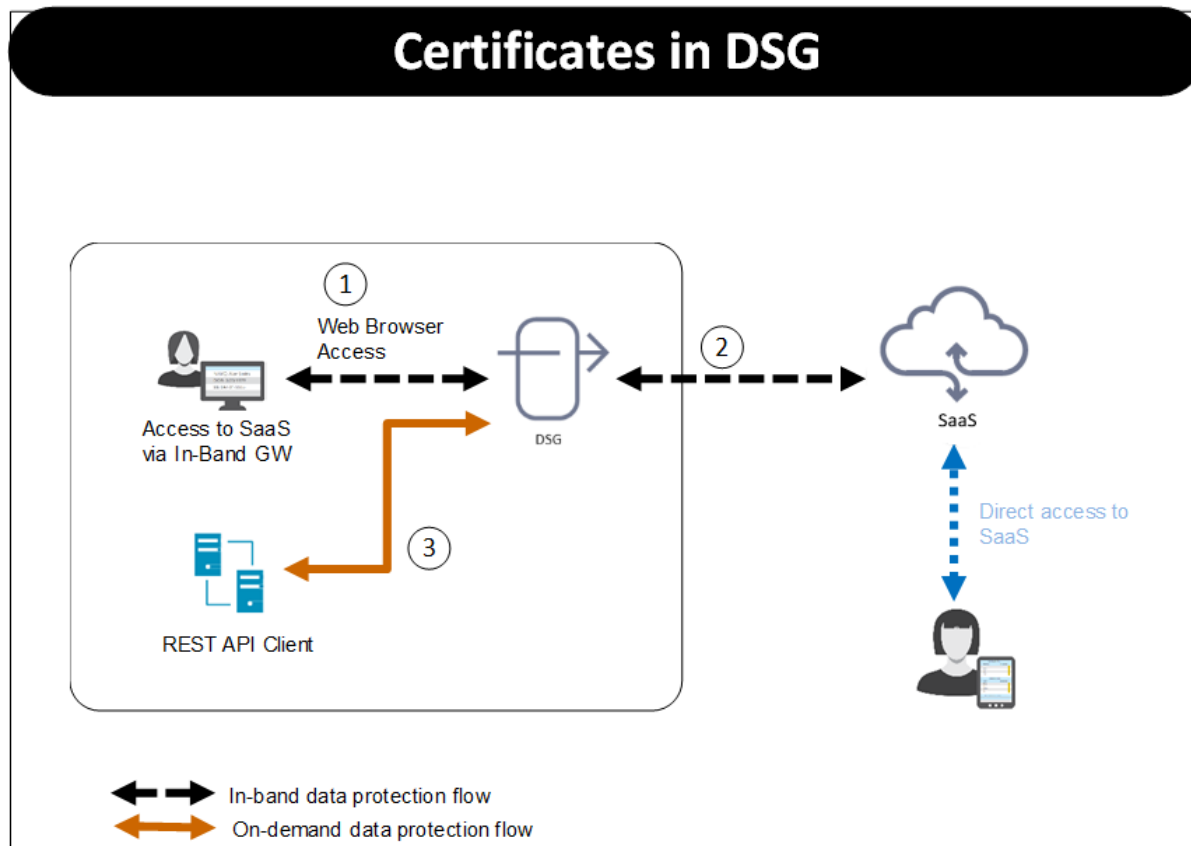


Figure 5-1: Certificates in DSG

Based on the protocol used the certificates that client must present to DSG and DSG must present to destination differ. For the Figure 1, consider HTTPS protocol is used.

Step 1:

When a client tries to access the SaaS through the DSG, DSG uses the certificate configured as part of tunnel configuration to communicate with the client. The client must trust the certificate to initiate the communication between client and DSG.

Step 2:

The step 2 involves DSG forwarding the request to the destination. In the TLS-based outbound communication in DSG, it is expected that the destination uses a certificate that is signed by a trusted certification authority. For example, in case of SaaS, it might use self-signed certificates. In this case, DSG must trust the server's certificate to initiate TLS-based outbound communication.

Step 3:

When the REST API client tries to communicate with the DSG, DSG uses the certificate configured as part of tunnel configuration to communicate with the client. The client browser must accept and trust the certificate to initiate the communication.

5.1 Inbound Certificates

The inbound certificate differs based on the protocol that is used to communicate with the DSG. This section covers certificates involved when using HTTPS using default certificates, TLS mutual authentication, and SFTP protocols.

5.1.1 HTTPS using default certificates

Consider a setup where a client is accessing the destination with DSG in between using the HTTPS protocol. In this case, DSG uses the certificate configured as part of tunnel configuration to communicate with the client.

In non-production environment, you can continue to use the default certificates that are generated when DSG is installed. In case of production deployment, it is recommended that you use your own certificates that are signed by a trusted certification authority.

In case you are using own certificates and keys, ensure that you replace the default CA certificates /keys and other certificates/keys with the signed certificates/keys.

5.1.2 TLS Mutual Authentication

DSG can be configured with trusted root CAs and/or the individual client machine certificates for the machines that will be allowed to connect to DSG. The client presents a client certificate to DSG, DSG verifies it against the CA certificate, and once validated, lets the client machine communicate with destination where DSG is in between.

Ensure that you replace the default CA certificates /keys and other certificates/keys with the signed certificates/keys.

Along with these certificates, every time a request is made to the DSG node, the client machine will present client certificate that was generated using the CA certificate. DSG validates the client certificate so that the client machine can communicate with DSG. The clients that fail to present a valid client certificate will not be able to connect to the destination.

Apart from presenting the certificate, at Tunnel level, ensure that the TLS Mutual Authentication is set to **CERT_OPTIONAL** or **CERT_MANDATORY**. Also, in the Extract rule at Ruleset level, ensure that the **Require Client Certificate** check box is selected if you want to perform this check at service level.

For more information about enabling TLS mutual authentication, refer to section *Mutual Authentication* in the *Data Security Gateway User Guide*.

5.1.3 SFTP

DSG can be configured to work as an intermediary between an SFTP client and server when accessing files using SFTP protocol. With SFTP, credentials are never transmitted in clear and information flows over an SSH tunnel.

If you are using SFTP, ensure that the SFTP server key is uploaded using the Certificates screen on the DSG node. At tunnel level, for an SFTP tunnel, you must specify this server key.

At the rule level, you can add a layer of security using the authentication method option. Using DSG, an SFTP client can communicate with the destination using either Password or SSH keys. Ensure that the SSH keys are trusted.

If you select Password as the authentication method, client must provide the password when prompted. While, if you are using Publickey as the authentication method, the SFTP client must trust DSG publickey and DSG must trust SFTP client publickey.

For more information about SFTP rule level settings and enabling password-less authentication, refer to section *SFTP Gateway* in the *Data Security Gateway User Guide*.

5.2 Outbound Certificates

The DSG can be used as an intermediary between client and destination. For example, in case of SaaS as destination, it is important that the self-signed certificates that a destination uses are trusted by DSG.

It might happen that the SaaS certificates are in DER format. DSG accepts certificates in PEM or CRT format, and hence you must convert the DER format to an acceptable PEM format.

For more information about trusting the self-signed certificates and converting the DER format to PEM format, refer to section *Creating a Service under Ruleset* in the *Data Security Gateway User Guide*.

Chapter 6

Replicating Certificates in a Trusted Appliance Cluster (TAC)

In a Trusted Appliance Cluster (TAC), the certificates are replicated between ESAs. The protectors can communicate with any of the ESAs that are part of the TAC.

The following figure illustrates the replication of certificates between two ESAs in a TAC.

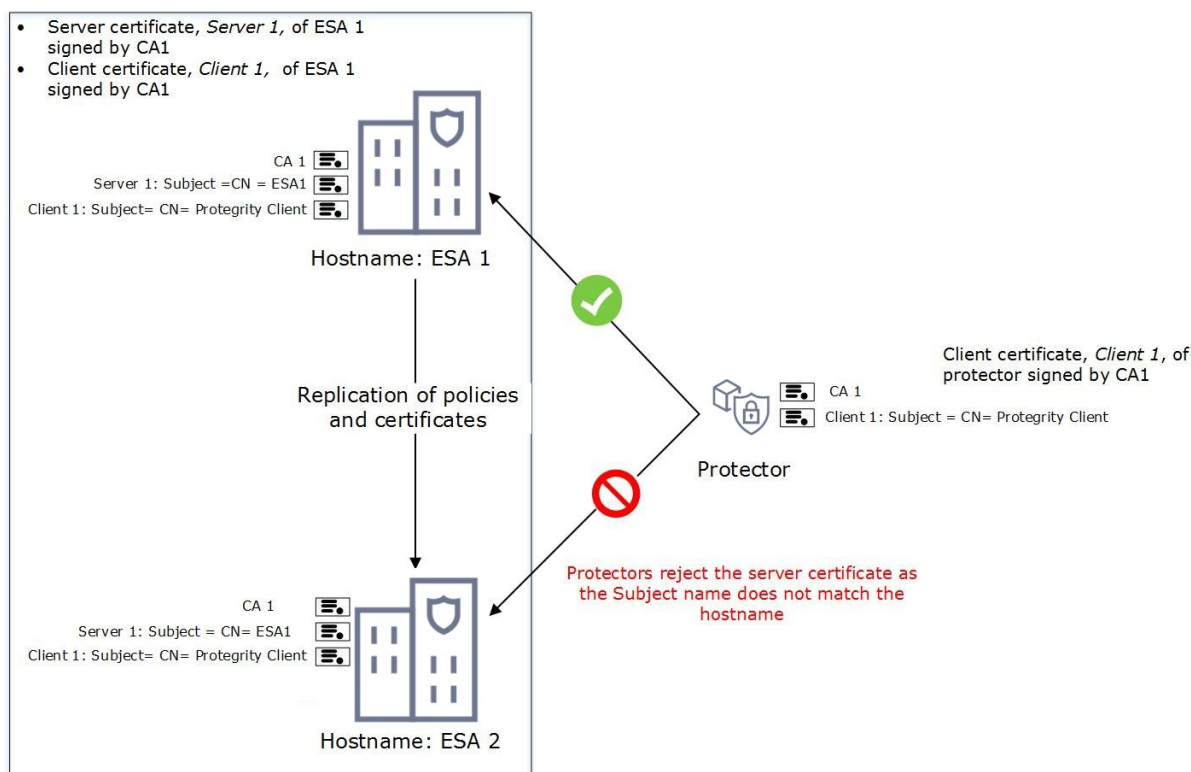


Figure 6-1: Replicating Certificates in TAC

The figure depicts two ESAs in a TAC. The ESA1 contains the server and the client certificates. The certificates in ESA1 are signed by CA1. The Protectors communicate with ESA1 to retrieve the client certificate.

Note: The *Subject* attribute for the server certificates is *CN=<hostname>* and that of the client certificate is *CN= Protegrity Client*.

In a TAC, when replication between ESA1 and ESA2 happens, the CA, server, and client certificates from ESA1 are copied to ESA2. However, when the certificates are replicated from ESA1 to ESA2, the Subject attribute is not updated to the hostname of ESA2. Due to this mismatch, the protectors are not able to communicate with ESA2.

Solution:

To ensure the communication of protectors with the ESA, perform one the following methods:

- Use a Subject Alternative Name (SAN) certificate to add additional hostnames. You can configure multiple ESA domains using a SAN certificate.
- Use wildcard for domain names in certificates to add multiple domains.

Chapter 7

Audit Store Certificates

7.1 Using Custom Certificates in the Audit Store

Certificates are used for secure communication with the Audit Store. These are used for communication between the Audit Store cluster nodes and its clients, such as, Log Forwarder and Analytics.

Note:

The default certificates provided are signed using the system-generated Protegrity-CA certificate. However, after installation you can use custom certificates. Also ensure that all the certificates are signed by the same CA as shown in the following diagram.

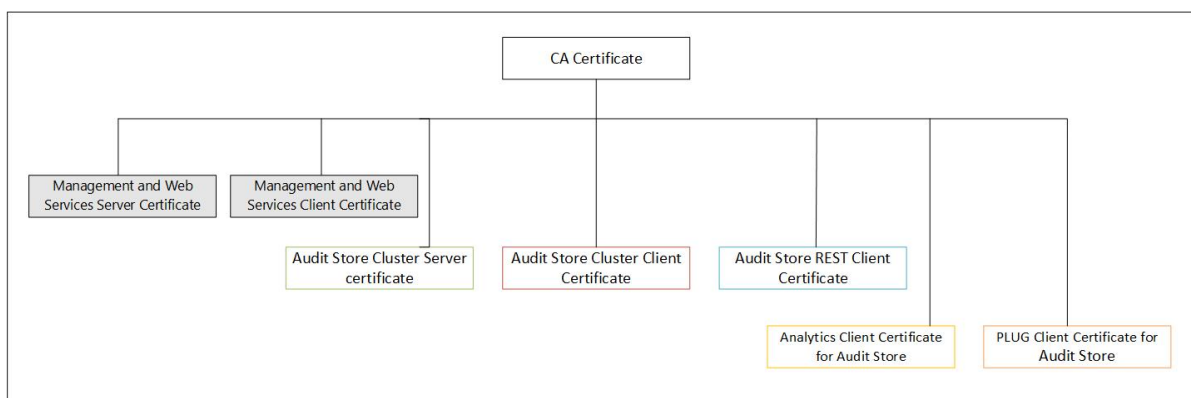


Figure 7-1: Audit Store Certificates

Note:

When you are updating certificates, ensure that the certificates are updated in the following order:

1. Audit Store Cluster certificate
2. Audit Store REST certificate
3. PLUG client certificate for Audit Store
4. Analytics client certificate for Audit Store

The various certificates used for communication between the nodes with their descriptions are provided here.

- **Management & Web Services:** These services manages certificate-based communication and authentication between the ESA and its internal components and between ESA and external clients (REST).

For more information about Management & Web Services certificates, refer to [Certificate Management in ESA](#).

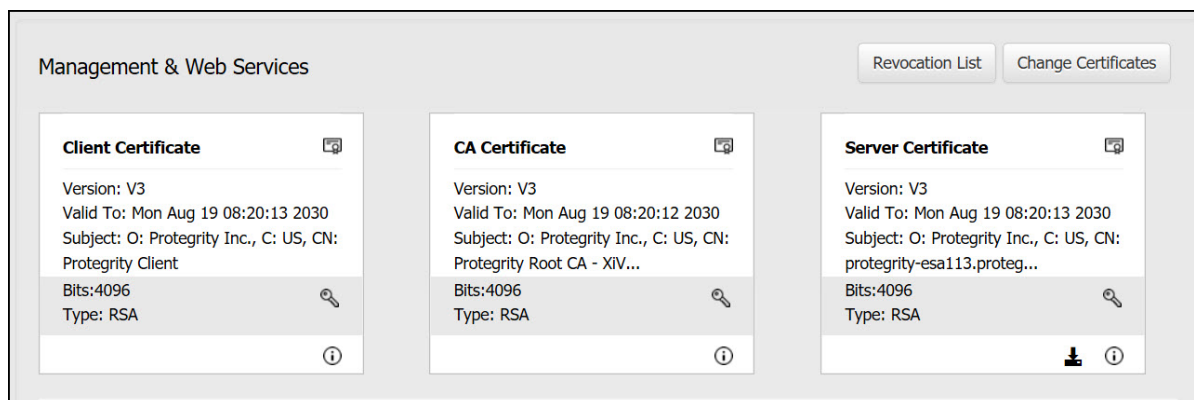


Figure 7-2: Management & Web Services

- **Audit Store Cluster:** This is used for the Audit Store inter-node communication that takes place over the port *9300*.

- **Server certificate:**

The server certificate is used for for inter-node communication. The nodes identify each other using this certificate.

Note:

The Audit Store Cluster and Audit Store REST server certificate must be the same.

- **Client certificate:**

The client certificate is used for applying and maintaining security configurations for the Audit Store cluster.

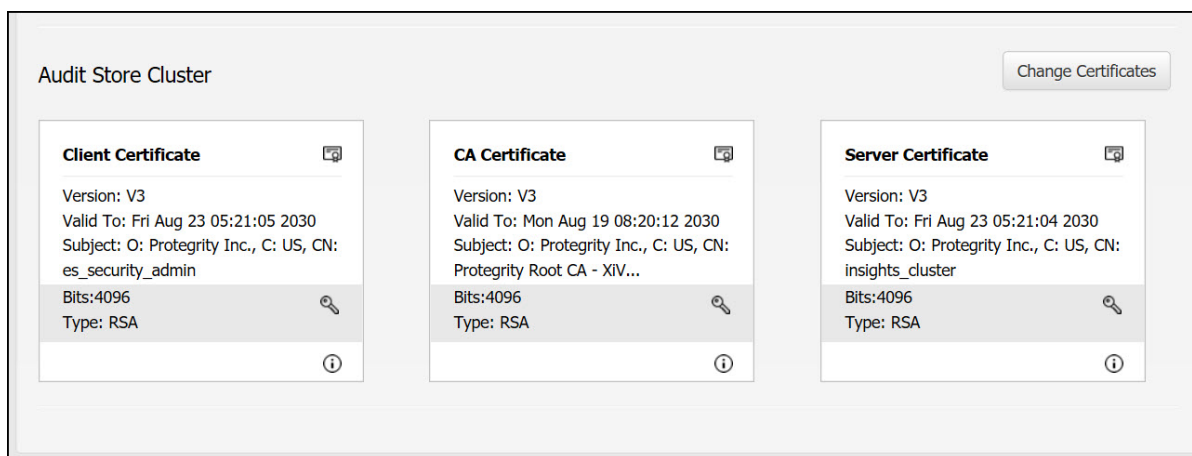


Figure 7-3: Audit Store Cluster

- **Audit Store REST:** This is used for the Audit Store REST API communication over the port *9200*.

- **Server certificate:**

The server certificate is used for mutual authentication with the client.

Note:

The Audit Store Cluster and Audit Store REST server certificate must be the same.

- **Client certificate:**

The client certificate is used by the Audit Store nodes to authenticate and communicate with the Audit Store.

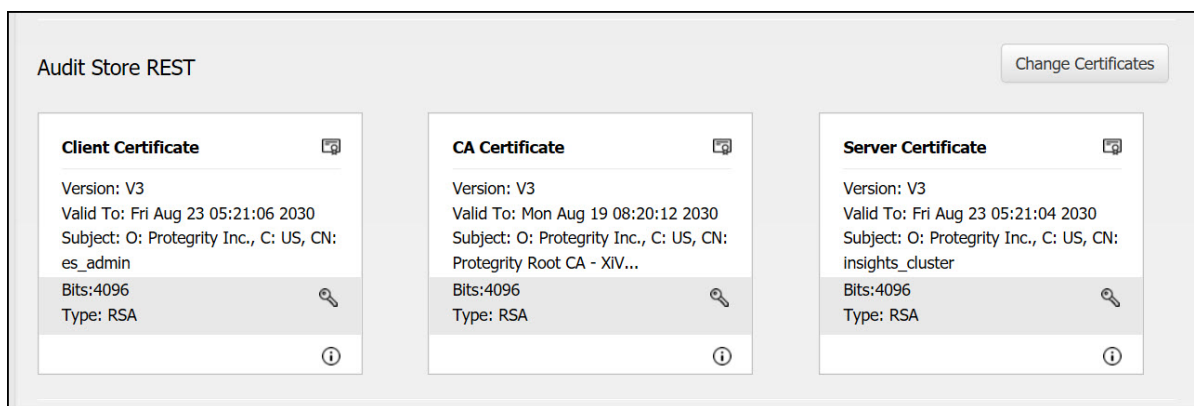


Figure 7-4: Audit Store REST

- **Analytics Client for Audit Store:** This is used for communication between Analytics and the Audit Store.

- **Client certificate:**

The client certificate is used by Analytics to authenticate and communicate with the Audit Store.

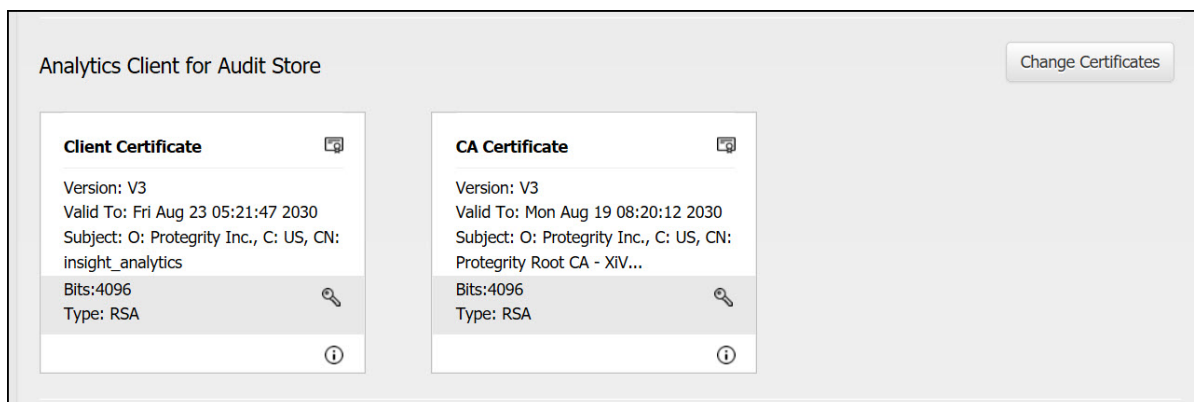


Figure 7-5: Analytics Client for Audit Store

- **PLUG Client for Audit Store:** This is used for communication between logging components and the Audit Store.

- **Client certificate:**

The client certificate is used by the Log Forwarder to authenticate and communicate with the Audit Store.

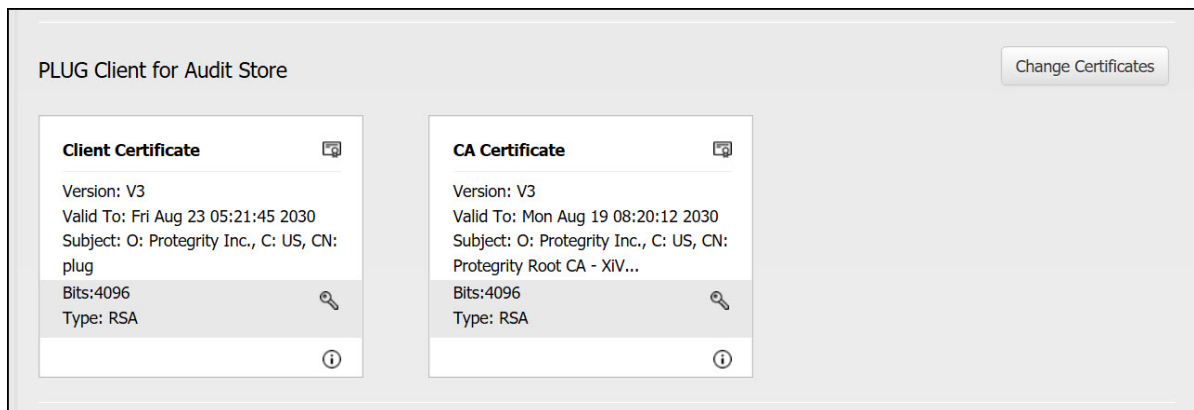


Figure 7-6: PLUG Client for Audit Store

7.1 Using Custom Certificates in the Audit Store

The certificates used for the PLUG component are system-generated Protegrity certificates. If required, you can upload and use your custom CA, Server, and Client certificates for the PLUG components.

When you use custom certificates, ensure that they meet the following prerequisites:

- Ensure that all certificates share a common CA.

Note:

For more information about the Audit Store certificates, refer to the diagram in [Audit Store Certificates](#).

- Ensure that the following requirements are met when creating the certificates:
 - The CN attribute of the Audit Store Server certificate is set to *insights_cluster*.
 - The CN attribute of the Audit Store Cluster Client certificate is set to *es_security_admin*.
 - The CN attribute of the Audit Store REST Client certificate is set to *es_admin*.
 - The CN attribute of the PLUG client certificate for the Audit Store is set to *plug*.
 - The CN attribute of the Analytics client certificate for the Audit Store is set to *insight_analytics*.
 - The Audit Store Server certificates' must contain the following in the Subject Alternative Name (SAN) field:
 - localhost
 - 127.0.0.1
 - FQDN of all the Audit Store nodes in the cluster
 - IP addresses of all the Audit Store nodes in the cluster
 - Hostname of all the Audit Store nodes in the cluster

Note:

If you are using a DNS server, then also include the hostname and FQDN details from the DNS sever in the certificate.

For example, an SSL certificate with the SAN extension of servers ES1, ES2, and ES3 in a cluster will have the following entries:

- localhost
- 127.0.0.1
- ES1

- ES2
- ES3
- ES1.protegrity.com
- ES2.protegrity.com
- ES3.protegrity.com
- IP address of ES1
- IP address of ES2
- IP address of ES3

Note:

If you are upgrading from an earlier version to ESA 8.1.0.0 and use custom certificates, then run the following step after the upgrade is complete and custom certificates are applied for td-agent, Audit Store, and Analytics, if installed.

1. From the ESA Web UI, navigate to **System > Services > Audit Store**.
2. Ensure that the **Audit Store Repository** service is not running. If the service is running, then stop the service.
3. Configure the custom certificates and upload it to the Certificate Repository.
4. Set the custom certificates for the PLUG components as *Active*.
5. From the ESA Web UI, navigate to **System > Services > Audit Store**.
6. Start the **Audit Store Repository** service.
7. Open the ESA CLI.
8. Navigate to **Tools**.
9. Run **Apply Audit Store Security Configs**.
10. Continue the installation to create an Audit Store cluster or join an existing Audit Store cluster.

For more information, refer the *Connecting to the Audit Store* topic in the *Protegrity Analytics Guide 8.1.0.0*.

Chapter 8

Validating Certificates

This section lists the various SSL commands to validate the certificates.

Verifying the validity of a certificate:

You can verify a client or a server certificate using the following commands:

```
openssl verify -CAfile /etc/ksa/certificates/CA.pem /etc/ksa/certificates/client.pem
openssl verify -CAfile /etc/ksa/certificates/CA.pem /etc/ksa/certificates/ws/server.pem
```

If the client or server certificate is signed by the provided CA certificate, then the certificate is valid. The message *OK* appears.

Verifying the purpose of a certificate:

You can verify if the certificate is a client, a server, or a CA certificate using the following command:

```
openssl x509 -in <Certificate name> -noout -purpose
```

For example, run the following command to verify the purpose of the client certificate:

```
openssl x509 -in /etc/ksa/certificates/client.pem -noout -purpose
```

Extracting the CN of a certificate:

To extract the username of a certificate, you must pass the *DN* value to the *pty_get_username_from_certificate* function. The following steps explain how to extract the CN of a certificate.

1. In the CLI Manager, navigate to the OS Console.
2. Run the following command to extract the value that is in the *Subject* attribute of the certificate.

```
openssl x509 -noout -subject -nameopt compat -in /etc/ksa/certificates/client.pem
```

3. Run the following command to extract the username from the *Subject* attribute of the client certificate.

```
etc/ksa/pty_get_username_from_certificate.py "<Value in the Subject attribute of the client certificate>"
```

For example,

```
etc/ksa/pty_get_username_from_certificate.py "/O=Acme Inc./C=US/CN=Protegrity Client"
```

Note: The CN attribute in a certificate can contain the Fully Qualified Domain Name (FQDN) of the client or server. If the length of the FQDN is greater than 64 characters, the hostname is considered as CN to generate a certificate.

Working with intermediate CA certificates

A root certificate is a public key certificate that identifies the root CA. The chain of certificates that exist between the root certificate and the certificate issued to you are known as intermediate CA certificates. You can use an intermediate CA certificate to sign the client and server certificates.

If you have multiple intermediate CA certificates, then you must link all the intermediate certificates and the root CA certificates into a single chain before you upload to the Certificate repository.

The following figure illustrates an example of two intermediate certificates and a root certificate.

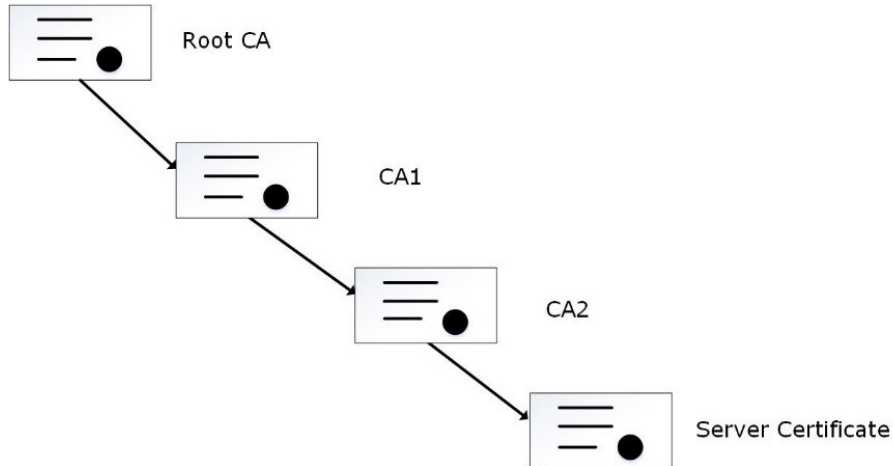


Figure 8-1: Intermediate Certificates and Root Certificate

In the figure, the server certificate is signed by an intermediate certificate CA2. The intermediate certificate CA2 is signed by CA1, which is signed by the root CA.

You can merge the CA certificates using the following command in the OS Console:

```
cat ./CA2.pem ./CA1.pem ./rootCA.pem > ./newCA.pem
```

You must then upload the *newCA.pem* certificate to the Certificate Repository.

Note: Ensure that you link the CA certificates in the appropriate hierarchy.

Increasing the Log Level to view errors for certificates:

If you want to view the errors and warnings generated for certificates, then you can increase the *LogLevel* attribute.

1. In the CLI Manager, navigate to the OS Console.
2. Run the following command to view the *apache.mng.conf* file.

```
vi /etc/ksa/service_dispatcher/servers/apache.mng.conf
```

3. Update the value of the *LogLevel* parameter from *warn* to *debug* and exit the editor.
4. Run the following command to view the *apache.ws.conf* file.

```
vi /etc/ksa/service_dispatcher/servers/apache.ws.conf
```

5. Update the value of the *LogLevel* parameter from *warn* to *debug*.
6. Restart the *Service Dispatcher* service.
7. Navigate to the */var/log/apache2-service_dispatcher* directory.

8. Open the *error.log* file to view the required logs.

Note: After debugging the errors, ensure that you revert the value of the *LogLevel* parameter to *warn* and restart the *Service Dispatcher* service.

Appendix

A

Appendix A: Certificates-related Terminology

The following table introduces terminology related to certificates that can help you understand Protegrity Certificate Management.

Table A-1: Certificates Terminology

Term	Definition
Cipher Suites	A cipher suite is a complete set of methods (technically known as algorithms) needed to secure a network connection through SSL (Secure Sockets Layer) / TLS (Transport Layer Security).
LDAP	Lightweight Directory Access Protocol (LDAP) is an Internet protocol that email and other programs use to look up information from a server.
Certificate Signing Request	The Certificate Signing Request (CSR or certification request) is a request sent from certificate applicant subject to a Certificate Authority (CA) to obtain digital certificate.
Truststore	A truststore is keystore that contains certificates which only contain public keys.