Micro Focus Security ArcSight SmartConnectors

Format Preserving Encryption Environment Setup Guide

October 19, 2017

Legal Notices

Warranty

The only warranties for products and services of Micro Focus and its affiliates and licensors ("Micro Focus") are set forth in the express warranty statements accompanying such products and services.
Nothing herein should be construed as constituting an additional warranty. Micro Focus shall not be liable for technical or editorial errors or omissions contained herein.
The information contained herein is subject to change without notice.

Restricted Rights Legend

Confidential computer software. Except as specifically indicated otherwise, a valid license from Micro Focus is required for possession, use or copying. Consistent with FAR 12.211 and 12.212, Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items are licensed to the U.S. Government under vendor's standard commercial license.

Copyright Notice

Trademark Notices

Adobe™ is a trademark of Adobe Systems Incorporated.
Microsoft® and Windows® are U.S. registered trademarks of Microsoft Corporation.
UNIX® is a registered trademark of The Open Group.

Support Contact Information

Phone	A list of phone numbers is available on the Micro Focus Security ArcSight Technical Support Page: https://softwaresupport.softwaregrp.com/support-contact-information
Support Web Site	https://softwaresupport.softwaregrp.com/
Protect 724 Community	https://community.softwaregrp.com/t5/ArcSight/ct-p/arcsight

Revision History

Date	Description
10/19/2017	First release of this technical note.

Contents

SmartConnectors with Format Preserving Encryption

Running a SmartConnector with Format Preserving Encryption enabled requires correct environment setup for the machine hosting the SmartConnector.

The setup primarily involves addressing two areas (1) ensuring proper connectivity between the connector machine and the SecureData appliance server and (2) ensuring the availability of an appliance certificate in all locations needed by the SmartConnector.

In addition, if you are not using Instant Connector Deployment with ArcSight Management Center, or you are upgrading a connector to 7.7.0 from 7.6 or prior version (via ArcSight Management Center or standalone), you will need to manually install the client for each connector host. Consult your SecureData Appliance documentation for instructions for your platform. Follow the instructions to verify that the connectivity is established from the SecureData Client to the SecureData Server and the test program runs as expected before proceeding with SmartConnector configuration for data encryption. Data encryption parameters are described in the configuration guide for each SmartConnector.

Network Connectivity

During connector configuration, during connector initialization at runtime, and also occasionally (even though not frequently) during events processing, the SmartConnector must connect to the SecureData appliance through the HTTPS protocol. Therefore, the HTTPS connectivity between the connector machine and the appliance server must be good.

The following sections describe steps to be followed to ensure connectivity for Linux and Windows operating systems.

Linux

- Edit the **`/etc/hosts`** file of the connector machine to include a line for the IP address and the hostname for the SecureData appliance. This step is required only if the DNS does not recognize the SecureData appliance FQDN.
- If your connector machine does not require a proxy to make an HTTPS connection to the SecureData appliance server, you do not need to set up a proxy. Verify that no global proxy is set for your machine. For a given user, you can open a fresh terminal, log in as the user, and enter **`env | grep –i proxy`** to determine whether a proxy has been inadvertently set by someone else.
- If your connector machine requires a proxy to make an HTTPS connection to the SecureData appliance server, set up a proper proxy. There is more than one way to set up a proxy. The System Administrator can choose the proper way to do it for the case in hand. However, keep in mind the following three important items:
    - If there are machines that do not require a proxy to be reached from the connector machine (such as ESM or Logger, for example), then the **`no_proxy/NO_PROXY`** environment variables must be set to bypass proxies for those hosts.
    - Make sure that the proxy-related variables are set globally, that is, for all users.
    - Verify that whatever proxy-related variables you have set are not transient (valid for one shell/terminal), and also verify they are actually in effect. You may need to reboot your machine to verify. Finally, on a fresh terminal, enter **`env | grep –i proxy`** to see all proxy-related environment variables. For a machine that has a number of hosts that require a proxy and a number of hosts that do not require a proxy, you should see something like:

```
no_proxy=localhost,.xxx.com,.aaa.bbb.com
NO_PROXY=localhost,.xxx.com,.aaa.bbb.com
https_proxy=dddd.ccccc.hh.com:8080
HTTPS_PROXY=dddd.ccccc.hh.com:8080
```

Windows

- Edit the **C:\Windows\System32\Drivers\etc\hosts** file of the connector machine to include a line for the IP address and the hostname for the SecureData appliance. This step is required only if the DNS does not recognize the SecureData appliance FQDN.
- If your connector machine does not require a proxy to make an https connection to the SecureData appliance server, do not set up a proxy. Verify that no global proxy is set for your machine. For a given user, you can open a fresh terminal, log in as the user, and enter **env | grep –i proxy** to determine whether a proxy has been inadvertently set by someone else.
- If your connector machine requires a proxy to reach certain hosts and does not require a proxy to reach some hosts, you can set these parameters from your browser's **Internet Options > Connections > LAN Settings >Advanced** tab. After setting the correct proxies and bypass list properly, open a command prompt as Administrator and enter the following command to import the Internet Options into the HTTP protocol connection:

  ```
  netsh winhttp import proxy source=ie
  ```

Secure Data Appliance Certificates

To be able to make a successful connection and encrypt data, the server certificate must be present in all needed locations. These locations are highlighted in the following sections.

Windows

For SmartConnectors running in Windows, the server certificate must be in the following three locations. Perform these steps after installing the connector core software, but prior to selecting connector and adding parameter information.

- Connector certificate store - **../current/jre/lib/security/cacerts**. When using FIPS for the connector, the certificate must be placed into **..\current\user\agent\fips\bcfips_ks**

  You can import any certificate to this store by using the **../current/jre/bin/keytool** utility. For example, from the **$ARCSIGHT_HOME/current/bin** directory, execute the following command to import the certificate:

  ```
  arcsight keytoolgui
  ```

  Open the keystore in **$ARCSIGHT_HOME/jre/lib/security/cacerts** (the password will be **changeit**).

  From the **Menu** bar, select **Tools** and **Import Certificate**. Upload the certificate file.

  Trust the certificate.

- Trusted Root Certification Authority for the local computer. Use the Windows certificate import wizard to import the certificate.
- Trusted Root Certification Authority for the current user. Use the Windows certificate import wizard to import the certificate.

Finally, it is always advisable to have server certificates that do not have chains. Windows certificates that are put in trusted authority are subject to CRL verification that may involve verifying the entire chain. This is controlled by the Windows system and can be very annoying because Windows may need to connect to outside hosts to verify the chain of trust. If any component cannot be verified, Windows will not trust it and the connection to the SecureData appliance will fail.

Linux

For SmartConnectors running on Linux, the server certificate must be in the following two locations. Perform these steps after installing the connector core software, but prior to selecting connector and adding parameter information.

- Connector certificate store - **`../current/jre/lib/security/cacerts`**. When using FIPS for the connector, the certificate must be placed into **`..\current\user\agent\fips\bcfips_ks`**

  You can import any certificate to this store using the **`../current/jre/bin/keytool`** utility. For example, from the **`$ARCSIGHT_HOME/current/bin`** directory, execute the following command to import the certificate:

  ```
  arcsight keytoolgui
  ```

  Open the keystore in **`$ARCSIGHT_HOME/jre/lib/security/cacerts`** (the password will be **`changeit`**).

  From the **Menu** bar, select **Tools** and **Import Certificate**. Upload the certificate file.

  Trust the certificate.

- The store used by the SecureData client. Assuming the client directory is **`/opt/client`**, the store would be **`/opt/client/trustStore`**. The certificate must be Base64 encoded, not DER; otherwise, the ./c_rehash command will fail. Copy the server certificate (in **`*.pem`** format) in the directory **`/opt/client/trustStore`**, and run the following command:

  `/opt/client/trustStore/c_rehash .`

  Note: Do not forget to type the **`'.'`**