RSyslog TLS Configuration and Setup

Last updated by | Naser Mohammed | Mar 19, 2023 at 7:18 AM CDT

Contents

How to set up TLS from one Syslog Client to a Syslog Server

Summary of the process

- As we need to establish trust between client/server we would need to generate the CA certificates for each of the server/client.
- We will copy the respective client certificate to client node and server certificate to rsyslog server.
- Certificate Authority server can be rsyslog server or an another server.
- To accept the logs over tls we will add some more modules to rsyslog server configuration file.
- To send the logs over tls we will add some more modules to rsyslog client configuration file.
- Make sure order of the modules are correct in both server/client configuration files.

Prerequisites:

```
sudo apt-get install rsyslog-gnutls
```

Required packages

```
rsyslog-gnutls-5.8.10-10.0.1.el6_6.x86_64
rsyslog-5.8.10-10.0.1.el6_6.x86_64
gnutls-utils-2.8.5-19.el6_7.x86_64
gnutls-2.8.5-19.el6_7.x86_64
```

(STEP ONE) Generate a Self-Signing Certificate Authority Certificate

Build the CA-Key file

```
certtool --generate-privkey --outfile ca-key.pem
```

This takes a short while. Be sure to do some work on your workstation, it waits for random input. Switching between windows is sufficient 😉

Now create the (self-signed) CA certificate:

```
certtool --generate-self-signed --load-privkey ca-key.pem --outfile ca.pem
```

This generates the CA certificate. This command queries you for a number of things. Use appropriate responses. When it comes to certificate validity, keep in mind that you need to recreate all certificates when this one expires. So it may be a good idea to use a long period, eg. 3650 days (roughly 10 years). You need to specify that the certificates belongs to an authority. The certificate is used to sign other certificates.

Certificate Authority Certificate Demo

```
Common name: CACert
UID:
Organizational unit name: CACertOU
Organization name: CACertOrg
Locality name: Grapevine
State or province name: TX
Country name (2 chars): Us
Enter the subject's domain component (DC): CAAuthority
Enter an additional domain component (DC):
This field should not be used in new certificates.
E-mail: someone@somewhere.com
Enter the certificate's serial number in decimal (123) or hex (0xabcd)
(default is 0x324ccfaaaf99550f907d882af6a3da1ccf0f26dd)
value:


Activation/Expiration time.
The certificate will expire in (days): 3650


Extensions.
Does the certificate belong to an authority? (y/N): y
Path length constraint (decimal, -1 for no constraint): -1
Is this a TLS web client certificate? (y/N): n
Will the certificate be used for IPsec IKE operations? (y/N):
Is this a TLS web server certificate? (y/N):
Enter a dnsName of the subject of the certificate:
Enter a URI of the subject of the certificate:
Enter the IP address of the subject of the certificate:
Enter the e-mail of the subject of the certificate:
Will the certificate be used for signing (required for TLS)? (Y/n):
Will the certificate be used for data encryption? (y/N):
Will the certificate be used to sign OCSP requests? (y/N):
Will the certificate be used to sign code? (y/N):
Will the certificate be used for time stamping? (y/N):
Will the certificate be used for email protection? (y/N):
Will the certificate be used to sign other certificates? (Y/n): Y
Will the certificate be used to sign CRLs? (y/N):
Enter the URI of the CRL distribution point:
```

(STEP TWO) Syslog Server Machine cert

Build the Server machine private key

```
certtool --generate-privkey --outfile server-key.pem --bits 2048
```

Build the Server machine request for public key from CA Certificate

```
certtool --generate-request --load-privkey server-key.pem --outfile request.pem
```

Demo

```
  Common name: ServerCert
  Organizational unit name: ServerCertOU
  Organization name: ServerCertOrg
  Locality name: Grapevine
  State or province name: TX
  Country name (2 chars): US
  Enter the subject's domain component (DC): 192.168.1.179
  Enter an additional domain component (DC):
  UID:
  Enter a dnsName of the subject of the certificate: 192.168.1.179 <<-- the Syslog Server's IP
  Enter an additional dnsName of the subject of the certificate:
  Enter a URI of the subject of the certificate:
  Enter the IP address of the subject of the certificate:
  Enter the e-mail of the subject of the certificate:
  Enter a challenge password:
  Does the certificate belong to an authority? (y/N): n
  Will the certificate be used for signing (DHE ciphersuites)? (Y/n):
  Will the certificate be used for encryption (RSA ciphersuites)? (Y/n):
  Will the certificate be used to sign code? (y/N):
  Will the certificate be used for time stamping? (y/N):
  Will the certificate be used for email protection? (y/N):
  Will the certificate be used for IPsec IKE operations? (y/N):
  Will the certificate be used to sign OCSP requests? (y/N):
  Is this a TLS web client certificate? (y/N): Y
  Is this a TLS web server certificate? (y/N): Y
```

Generate the Server Machine Certificate (this file will be used on both the client and server machines)

```
 certtool --generate-certificate --load-request request.pem --outfile server-cert.pem --load-ca-certificate
ca.pem --load-ca-privkey ca-key.pem
```

Demo

```
  Generating a signed certificate...
  Enter the certificate's serial number in decimal (123) or hex (0xabcd)
  (default is 0x1e5b71b3b6ee00f0530f3526460d715dcaca4762)
  value:


  Activation/Expiration time.
  The certificate will expire in (days): 3649


  Extensions.
  Do you want to honour all the extensions from the request? (y/N): Y
  Does the certificate belong to an authority? (y/N):
  Is this a TLS web client certificate? (y/N): Y
  Will the certificate be used for IPsec IKE operations? (y/N):
  Is this a TLS web server certificate? (y/N): Y
  Enter a dnsName of the subject of the certificate: 192.168.1.179 <<-- the Syslog Server's IP Address
  Enter an additional dnsName of the subject of the certificate:
  Enter a URI of the subject of the certificate:
  Enter the IP address of the subject of the certificate:
  Will the certificate be used for signing (DHE ciphersuites)? (Y/n):
  Will the certificate be used for encryption (RSA ciphersuites)? (Y/n):
  Will the certificate be used for data encryption? (y/N):
  Will the certificate be used to sign OCSP requests? (y/N):
  Will the certificate be used to sign code? (y/N):
  Will the certificate be used for time stamping? (y/N):
  Will the certificate be used for email protection? (y/N):
```

```
  rm request.pem < delete the last request >
```

(STEP THREE) Client Server Machine certificate

Build the Client machine private key

```
  certtool --generate-privkey --outfile client-key.pem --bits 2048
```

Build the Client machine request for public key from CA Certificate

```
  certtool --generate-request --load-privkey client-key.pem --outfile request.pem
```

Demo

```
  Common name: ClientCert
  Organizational unit name: ClientCertOU
  Organization name: ClientCertOrg
  Locality name: Grapevine
  State or province name: TX
  Country name (2 chars): US
  Enter the subject's domain component (DC):
  UID:
  Enter a dnsName of the subject of the certificate: 192.168.1.166 <<-- the Syslog Client's IP Address
  Enter an additional dnsName of the subject of the certificate:
  Enter a URI of the subject of the certificate:
  Enter the IP address of the subject of the certificate:
  Enter the e-mail of the subject of the certificate:
  Enter a challenge password:
  Does the certificate belong to an authority? (y/N): n
  Will the certificate be used for signing (DHE ciphersuites)? (Y/n):
  Will the certificate be used for encryption (RSA ciphersuites)? (Y/n):
  Will the certificate be used to sign code? (y/N):
  Will the certificate be used for time stamping? (y/N):
  Will the certificate be used for email protection? (y/N):
  Will the certificate be used for IPsec IKE operations? (y/N):
  Will the certificate be used to sign OCSP requests? (y/N):
  Is this a TLS web client certificate? (y/N): Y
  Is this a TLS web server certificate? (y/N): Y
```

Generate the Client Machine Certificate

```
 certtool --generate-certificate --load-request request.pem --outfile client-cert.pem --load-ca-certificate
ca.pem --load-ca-privkey ca-key.pem
```

Demo

```
  Generating a signed certificate...
  Enter the certificate's serial number in decimal (123) or hex (0xabcd)
  (default is 0x4816b1ddd3320dea219895d706aba965f7abe23f)
  value:


  Activation/Expiration time.
  The certificate will expire in (days): 3649


  Extensions.
  Do you want to honour all the extensions from the request? (y/N): Y
  Does the certificate belong to an authority? (y/N): n
  Is this a TLS web client certificate? (y/N): Y
  Will the certificate be used for IPsec IKE operations? (y/N):
  Is this a TLS web server certificate? (y/N): Y
  Enter a dnsName of the subject of the certificate: 192.168.1.166 <<-- the Syslog Client's IP Address
  Enter an additional dnsName of the subject of the certificate:
  Enter a URI of the subject of the certificate:
  Enter the IP address of the subject of the certificate:
  Will the certificate be used for signing (DHE ciphersuites)? (Y/n):
  Will the certificate be used for encryption (RSA ciphersuites)? (Y/n):
  Will the certificate be used for data encryption? (y/N):
  Will the certificate be used to sign OCSP requests? (y/N):
  Will the certificate be used to sign code? (y/N):
  Will the certificate be used for time stamping? (y/N):
  Will the certificate be used for email protection? (y/N):
```

(STEP FOUR) Copy client files to client machine and Server files to Rsyslog Collector

Client Files

```
  ca.pem

  client-cert.pem

  client-key.pem
```

Server Files (If you ran the files on this server you will not need to move them, just point to the directory)

```
  ca.pem

  server-cert.pem

  server-key.pem
```

use WinSCP to copy files over in Binary mode

Convert to other formats (P12 PKS)

```
 certtool --load-certificate client-cert.pem --load-privkey client-key.pem --to-p12 --outder --outfile client-key.p12
```

Check Point configuration

Demo

```
 cp log export set name ubuntu20 ca-cert /var/log/syslogcert/ca.pem client-cert /var/log/syslogcert/client-key.p12 client-secret R4nger*10
```

(STEP FIVE) Rsyslog.conf Configurations

rsyslog.conf - Client Setup

```
# make gtls driver the default and set certificate files
global(
DefaultNetstreamDriver="gtls"
DefaultNetstreamDriverCAFile="/var/log/syslogcert/ca.pem" <- this should be the file location of the file
DefaultNetstreamDriverCertFile="/var/log/syslogcert/client-cert.pem" <- this should be the file location of th
DefaultNetstreamDriverKeyFile="/var/log/syslogcert/client-key.pem" <- this should be the file location of the
)
#
# set up the action for all messages
action(
type="omfwd"
target="192.168.1.179"
protocol="tcp"
port="6514"
StreamDriver="gtls"
StreamDriverMode="1" # run driver in TLS-only mode
StreamDriverAuthMode="x509/name"
StreamDriverPermittedPeers="192.168.1.179"
)
```

rsyslog.conf - Central Rsyslog Server

` global(DefaultNetstreamDriver="gtls" DefaultNetstreamDriverCAFile="/var/log/syslogcert/ca.pem" <- this should be the file location of the file DefaultNetstreamDriverCertFile="/var/log/syslogcert/server-cert.pem" <- this should be the file location of the file DefaultNetstreamDriverKeyFile="/var/log/syslogcert/server-key.pem" <- this should be the file location of the file)

load TCP listener

module(load="imtcp" StreamDriver.Name="gtls" StreamDriver.Mode="1" StreamDriver.Authmode="anon")

start up listener at port 6514

input(type="imtcp" port="6514")

provides UDP syslog reception

module(load="imudp") input(type="imudp" port="514")

provides TCP syslog reception

module(load="imtcp")

input(type="imtcp" port="514") `

(STEP SIX) Restart the Rsyslog Server on both the client and server

```
systemctl restart rsyslog
```

Debugging

They will need to run tail -f /var/log/messages and see if their is TLS errors, if there is they will need to redo their certificates. `tail -f /var/log/messages`