How to Configure rsyslog Server to Accept Logs via SSL/TLS

By admin

Purpose of this post is to explain how to configure rsyslog server to transmit logs via SSL/TLS. Logs which were transmitted from client to rsyslog server will be encrypted over n/w so that we have additional level security.

Procedure Summary

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

Requirements

rsyslog server/client with the below 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
```

Test Case

1. Generate CA certifiates in Rsyslog server or any other CA server.

```
# certtool --generate-privkey --outfile ca-key.pem
Generating a 2048 bit RSA private key...
```

```
# certtool --generate-self-signed --load-privkey ca-key.pem --outfile ca.pem     ===>>> generating CA certificate
Generating a self signed certificate...
Please enter the details of the certificate's distinguished name. Just press enter to ignore a field.
Country name (2 chars): ind
Organization name: Myorg
Organizational unit name: myBU
Locality name: BLR
State or province name: KA
Common name: CAcert
UID:
This field should not be used in new certificates.
E-mail:
Enter the certificate's serial number in decimal (default: 1482511911):

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):
Is this a TLS web client certificate? (y/N):
Is this also a TLS web server certificate? (y/N):
Enter the e-mail of the subject of the certificate:
Will the certificate be used to sign other certificates? (y/N): y
Will the certificate be used to sign CRLs? (y/N):
Will the certificate be used to sign code? (y/N):
Will the certificate be used to sign OCSP requests? (y/N):
Will the certificate be used for time stamping? (y/N):
Enter the URI of the CRL distribution point:
X.509 Certificate Information:
Version: 3
Serial Number (hex): 585d5627
Validity:
Not Before: Fri Dec 23 16:51:52 UTC 2016
Not After: Mon Dec 21 16:51:55 UTC 2026
Subject: C=ind,O=Myorg,OU=myBU,L=BLR,ST=KA,CN=CAcert
Subject Public Key Algorithm: RSA
Modulus (bits 2048):
c0:78:d2:ba:a0:93:7d:81:a3:f7:a5:f4:86:a4:c2:2d
c6:1c:c1:d2:95:c9:d5:5b:40:f9:15:a2:06:3e:f2:fa
09:f6:87:fe:36:cf:6f:85:75:ec:a1:f6:98:c7:e1:5d
7a:de:d5:a5:da:34:c7:5a:b5:f3:f2:80:a5:b8:fe:66
f3:b0:25:05:74:d3:7e:f0:45:3d:65:0a:f1:1f:5d:14
01:74:ef:9c:5f:48:b6:4b:b2:62:c5:e5:b0:21:41:92
86:bb:43:0f:2c:4a:ba:ef:1e:69:85:de:ce:42:3e:55
2c:1d:f8:82:d8:77:6a:46:ec:ac:73:b7:b3:e8:53:c4
6e:13:eb:da:27:ba:7d:70:0f:62:d5:04:b7:f7:2e:c9
57:5d:1e:0d:c2:14:8b:81:ff:9f:63:b8:4e:c9:b6:ae
ad:8c:e5:eb:c1:77:70:f6:9c:90:0d:f1:9c:16:85:b3
d8:1d:70:00:82:aa:ea:1b:f4:65:a1:e7:b7:33:4a:07
46:46:e4:45:d7:3f:72:63:43:00:1b:c2:8a:d5:a2:aa
13:7f:28:b7:00:50:1d:9b:28:92:60:a7:b1:ba:3c:7b
58:e7:8b:85:ba:8c:10:da:13:28:56:f2:9c:26:70:7f
cb:fb:81:4d:05:2d:0f:93:21:20:d7:75:5f:27:86:13
Exponent (bits 24):
01:00:01
Extensions:
Basic Constraints (critical):
Certificate Authority (CA): TRUE
Key Usage (critical):
Certificate signing.
Subject Key Identifier (not critical):
b606f5fa9bcd986ec25d2496c7d3a5c9270cc5f7
Other Information:
Public Key Id:
b606f5fa9bcd986ec25d2496c7d3a5c9270cc5f7

Is the above information ok? (Y/N): y
```

```
Signing certificate...
```

```
# certtool --generate-privkey --outfile rslclient-key.pem --bits 2048
Generating a 2048 bit RSA private key...
```

```
# certtool --generate-request --load-privkey rslclient-key.pem --outfile request.pem
Generating a PKCS #10 certificate request...
Country name (2 chars): ind
Organization name: Myorg
Organizational unit name: Mybu
Locality name: blr
State or province name: KA
Common name: dhcp-blr-kmgm-blk2-4fl-6fl-10-178-22-154.in.oracle.com
UID:
Enter a dnsName of the subject of the certificate: dhcp-blr-kmgm-blk2-4fl-6fl-10-178-22-154.in.oracle.com
Enter a dnsName 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 and RSA-EXPORT ciphersuites)? (y/N):
Will the certificate be used for encryption (RSA ciphersuites)? (y/N):
Is this a TLS web client certificate? (y/N): y
Is this also a TLS web server certificate? (y/N): y
```

```
# certtool --generate-certificate --load-request request.pem --outfilerslclient-cert.pem --load-ca-certificate ca.pem --load-ca-privkey ca-key.pem =======>>> generating client key certi

Generating a signed certificate...
Enter the certificate's serial number in decimal (default: 1482512116):

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

Extensions.
Do you want to honour the extensions from the request? (y/N):
Does the certificate belong to an authority? (y/N): n
Is this a TLS web client certificate? (y/N): y
Is this also a TLS web server certificate? (y/N): y
Enter a dnsName of the subject of the certificate: dhcp-blr-kmgm-blk2-4fl-6fl-10-178-22-154.in.oracle.com
Enter a dnsName of the subject of the certificate:
Enter the IP address of the subject of the certificate:
Will the certificate be used for signing (DHE and RSA-EXPORT ciphersuites)? (y/N):
Will the certificate be used for encryption (RSA ciphersuites)? (y/N):
X.509 Certificate Information:
Version: 3
Serial Number (hex): 585d56f4
Validity:
Not Before: Fri Dec 23 16:55:18 UTC 2016
Not After: Mon Dec 21 16:55:20 UTC 2026
Subject: C=ind,O=Myorg,OU=Mybu,L=blr,ST=KA,CN=dhcp-blr-kmgm-blk2-4fl-6fl-10-178-22-154.in.oracle.com
Subject Public Key Algorithm: RSA
Modulus (bits 2048):
9f:07:cd:0b:46:04:cd:60:be:52:43:86:3e:28:61:0e
54:6d:4b:bd:a1:31:7d:b3:4b:33:c0:b1:92:54:5d:b5
b5:67:ba:67:3f:d5:7f:5a:5a:e6:ba:71:dc:c9:4e:a3
f6:60:14:e1:60:cf:df:c1:c2:46:42:05:54:80:c1:a0
98:7e:c1:02:3b:8e:1e:0a:da:87:86:12:51:d6:db:91
3e:df:c5:32:4c:b2:fc:f8:74:fd:f1:91:89:d3:4e:8b
4a:27:bb:13:73:b3:cf:24:b6:c7:73:ad:47:58:d2:04
22:1d:af:d0:e3:be:7c:d4:85:67:ff:fd:61:55:c5:48
9d:0d:ff:aa:f0:78:78:5b:ef:14:12:f0:e4:53:84:cf
b9:62:1d:20:a2:22:40:ae:9e:15:41:9b:a2:55:f3:6a
00:fe:66:8e:01:af:31:52:80:54:37:af:14:91:e8:49
d1:08:2c:24:21:74:cf:11:e3:30:5e:e4:b7:ce:0d:dc
6a:1a:16:76:8a:0f:bc:c1:37:e3:30:0f:af:29:ca:ff
ac:eb:ed:dd:72:28:0a:6f:ea:58:35:67:0d:2a:57:ff
af:54:61:fc:52:8f:53:7c:f9:8d:5d:2c:a9:24:60:2c
c1:13:59:24:da:df:93:9a:0f:fd:74:b0:db:81:d5:17
Exponent (bits 24):
01:00:01
Extensions:
Basic Constraints (critical):
Certificate Authority (CA): FALSE
Key Purpose (not critical):
TLS WWW Client.
TLS WWW Server.
Subject Alternative Name (not critical):
DNSname: dhcp-blr-kmgm-blk2-4fl-6fl-10-178-22-154.in.oracle.com
Subject Key Identifier (not critical):
34e95fc3db224ce9b4ed05f619359a4c4140826e
Authority Key Identifier (not critical):
b606f5fa9bcd986ec25d2496c7d3a5c9270cc5f7
Other Information:
Public Key Id:
34e95fc3db224ce9b4ed05f619359a4c4140826e

Is the above information ok? (Y/N): y

Signing certificate...
```

```
# rm -rf request.pem
```

– Generate certificates for server:

```
# certtool --generate-privkey --outfile rslserver-key.pem --bits 2048
Generating a 2048 bit RSA private key...
```

```
certtool --generate-request --load-privkey rslserver-key.pem --outfile request.pem
Generating a PKCS #10 certificate request...
Country name (2 chars): ind
Organization name: Myorg
Organizational unit name: Mybu
Locality name: blr
State or province name: ka
Common name: dhcp-blr-kmgm-blk2-4fl-6fl-10-178-23-94.in.oracle.com
UID:
Enter a dnsName of the subject of the certificate: dhcp-blr-kmgm-blk2-4fl-6fl-10-178-23-94.in.oracle.com
Enter a dnsName 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 and RSA-EXPORT ciphersuites)? (y/N):
Will the certificate be used for encryption (RSA ciphersuites)? (y/N):
Is this a TLS web client certificate? (y/N): y
Is this also a TLS web server certificate? (y/N): y
Generating server key certificate using request.pem
```

```
# certtool --generate-certificate --load-request request.pem --outfilerslserver-cert.pem --load-ca-certificate ca.pem --load-ca-privkey ca-key.pem
Generating a signed certificate...
Enter the certificate's serial number in decimal (default: 1482512336):

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

Extensions.
Do you want to honour the extensions from the request? (y/N):
Does the certificate belong to an authority? (y/N): n
Is this a TLS web client certificate? (y/N): y
Is this also a TLS web server certificate? (y/N): y
Enter a dnsName of the subject of the certificate: dhcp-blr-kmgm-blk2-4fl-6fl-10-178-23-94.in.oracle.com
Enter a dnsName of the subject of the certificate:
Enter the IP address of the subject of the certificate:
Will the certificate be used for signing (DHE and RSA-EXPORT ciphersuites)? (y/N):
Will the certificate be used for encryption (RSA ciphersuites)? (y/N):
X.509 Certificate Information:
Version: 3
Serial Number (hex): 585d57d0
Validity:
Not Before: Fri Dec 23 16:58:57 UTC 2016
Not After: Mon Dec 21 16:59:00 UTC 2026
Subject: C=ind,O=Myorg,OU=Mybu,L=blr,ST=ka,CN=dhcp-blr-kmgm-blk2-4fl-6fl-10-178-23-94.in.oracle.com
Subject Public Key Algorithm: RSA
Modulus (bits 2048):
ac:2d:46:c2:41:7b:16:a6:80:7f:9f:46:2c:64:02:2d
61:f9:9d:dc:21:c6:fb:97:b0:cc:cb:00:ec:af:20:a3
09:8f:d2:6d:5c:56:46:1d:ff:bf:d6:e1:ce:70:08:04
67:6c:b0:bf:2f:02:c5:b7:03:0d:d5:c6:15:5c:af:5c
b3:1f:98:5b:80:09:60:8c:f2:4f:80:cf:9c:f2:bc:a3
81:46:b0:49:e3:ac:73:79:26:30:b6:41:b5:5a:19:3a
a4:a6:c7:3f:9b:7e:b9:ea:70:ea:21:87:38:68:f1:aa
01:0b:93:73:72:09:cf:7a:96:59:90:37:e5:ea:3b:c8
fa:f0:8b:ab:1a:f9:7b:9c:ee:c4:fc:92:0d:fe:01:ec
5d:3e:a8:dc:35:26:05:8d:d8:f2:94:0d:01:76:2a:64
d1:67:9d:ab:44:4c:a8:24:d6:d7:5a:70:76:f4:da:04
ff:40:0c:1a:5e:49:a2:65:69:94:88:08:71:70:1f:c9
a3:a0:b0:99:61:39:7a:a0:2a:b7:e8:ca:28:fd:52:89
f5:a2:32:ff:b8:38:12:39:2f:9a:2b:0d:16:33:91:1d
4f:49:78:1b:51:43:b0:d7:6d:bd:2e:84:73:d3:33:9a
3a:82:98:38:06:ed:e8:56:c6:41:2a:69:89:9e:26:b3
Exponent (bits 24):
01:00:01
Extensions:
Basic Constraints (critical):
Certificate Authority (CA): FALSE
Key Purpose (not critical):
TLS WWW Client.
TLS WWW Server.
Subject Alternative Name (not critical):
DNSname: dhcp-blr-kmgm-blk2-4fl-6fl-10-178-23-94.in.oracle.com
Subject Key Identifier (not critical):
f7f986ecdd10bf2646cd74f7e20e3d9b0f746765
Authority Key Identifier (not critical):
b606f5fa9bcd986ec25d2496c7d3a5c9270cc5f7
Other Information:
Public Key Id:
f7f986ecdd10bf2646cd74f7e20e3d9b0f746765

Is the above information ok? (Y/N): y

Signing certificate...
```

2. Copy ca.pem, rsl-client* certificate to client and server certificate to rsyslog-server

```
# rsync -aP rslserver-* root@10.178.22.148:/etc/pki/tls/private/
# rsync -aP ca.pem root@10.178.22.148:/etc/pki/tls/private/
```

```
# ls -l /etc/pki/tls/private/
total 12
-rw-r--r-- 1 root root 1233 Dec 23 19:58 ca.pem
-rw-r--r-- 1 root root 1452 Dec 23 20:06 rslclient-cert.pem
-rw------- 1 root root 1679 Dec 23 19:59 rslclient-key.pem
```

```
# ls -l /etc/pki/tls/private/
total 12
-rw-r--r-- 1 root root 1233 Dec 23 20:09 ca.pem
-rw-r--r-- 1 root root 1448 Dec 23 20:09 rslserver-cert.pem
-rw------- 1 root root 1675 Dec 23 20:09 rslserver-key.pem
```

Rsyslog configuration

1. Install both packages in rsyslog-client/rsyslog-server.

```
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
```

2. RSYSLOG server configuration:
– Add the below modules/configuration to /etc/rsyslog.conf.

Note: Modules should be in the below order as drivers should be loaded before imtcp module.

```
$DefaultNetstreamDriver gtls

$DefaultNetstreamDriverCAFile /etc/pki/tls/private/ca.pem
$DefaultNetstreamDriverCertFile /etc/pki/tls/private/rslserver-cert.pem
$DefaultNetstreamDriverKeyFile /etc/pki/tls/private/rslserver-key.pem

$ModLoad imtcp

$InputTCPServerStreamDriverAuthMode anon
$InputTCPServerStreamDriverMode 1 # run driver in TLS-only mode

$ActionSendStreamDriverAuthMode x509/name
$ActionSendStreamDriverPermittedPeer dhcp-blr-kmgm-blk2-4fl-6fl-10-178-22-154.in.oracle.com
$ActionSendStreamDriverMode 1 # run driver in TLS-only mode

$InputTCPServerRun 514

# Increase the amount of open files rsyslog is allowed, which includes open tcp sockets
# This is important if there are many clients.
# http://www.rsyslog.com/doc/rsconf1_maxopenfiles.html
$MaxOpenFiles 2048
```

For Example:

```
# cat /etc/rsyslog.conf
# rsyslog v5 configuration file

# For more information see /usr/share/doc/rsyslog-*/rsyslog_conf.html
# If you experience problems, see http://www.rsyslog.com/doc/troubleshoot.html

#### MODULES ####

$ModLoad imuxsock # provides support for local system logging (e.g. via logger command)
$ModLoad imklog # provides kernel logging support (previously done by rklogd)
#$ModLoad immark # provides --MARK-- message capability

# Provides UDP syslog reception
#$ModLoad imudp
#$UDPServerRun 514

# Provides TCP syslog reception

#### GLOBAL DIRECTIVES ####

# Use default timestamp format
$ActionFileDefaultTemplate RSYSLOG_TraditionalFileFormat

# File syncing capability is disabled by default. This feature is usually not required,
# not useful and an extreme performance hit
#$ActionFileEnableSync on

# Include all config files in /etc/rsyslog.d/
$IncludeConfig /etc/rsyslog.d/*.conf

$DefaultNetstreamDriver gtls

$DefaultNetstreamDriverCAFile /etc/pki/tls/private/ca.pem
$DefaultNetstreamDriverCertFile /etc/pki/tls/private/rslserver-cert.pem
$DefaultNetstreamDriverKeyFile /etc/pki/tls/private/rslserver-key.pem

$ModLoad imtcp

$InputTCPServerStreamDriverAuthMode anon
$InputTCPServerStreamDriverMode 1 # run driver in TLS-only mode

$ActionSendStreamDriverAuthMode x509/name
$ActionSendStreamDriverPermittedPeer dhcp-blr-kmgm-blk2-4fl-6fl-10-178-22-154.in.oracle.com ======>>>> mention your rsyslog peer name or pattern ( *. )
$ActionSendStreamDriverMode 1 # run driver in TLS-only mode

$InputTCPServerRun 514

# Increase the amount of open files rsyslog is allowed, which includes open tcp sockets
# This is important if there are many clients.
# http://www.rsyslog.com/doc/rsconf1_maxopenfiles.html
$MaxOpenFiles 2048

#### RULES ####

# Log all kernel messages to the console.
# Logging much else clutters up the screen.
#kern.* /dev/console

# Log anything (except mail) of level info or higher.
# Don't log private authentication messages!
*.info;mail.none;authpriv.none;cron.none /var/log/messages

# The authpriv file has restricted access.
authpriv.* /var/log/secure

# Log all the mail messages in one place.
mail.* -/var/log/maillog
```

```
# Log cron stuff
cron.* /var/log/cron

# Everybody gets emergency messages
*.emerg *

# Save news errors of level crit and higher in a special file.
uucp,news.crit /var/log/spooler

# Save boot messages also to boot.log
local7.* /var/log/boot.log

# ### begin forwarding rule ###
# The statement between the begin ... end define a SINGLE forwarding
# rule. They belong together, do NOT split them. If you create multiple
# forwarding rules, duplicate the whole block!
# Remote Logging (we use TCP for reliable delivery)
#
# An on-disk queue is created for this action. If the remote host is
# down, messages are spooled to disk and sent when it is up again.
#$WorkDirectory /var/lib/rsyslog # where to place spool files
#$ActionQueueFileName fwdRule1 # unique name prefix for spool files
#$ActionQueueMaxDiskSpace 1g # 1gb space limit (use as much as possible)
#$ActionQueueSaveOnShutdown on # save messages to disk on shutdown
#$ActionQueueType LinkedList # run asynchronously
#$ActionResumeRetryCount -1 # infinite retries if host is down
# remote host is: name/ip:port, e.g. 192.168.0.1:514, port optional
#*.* @@remote-host:514
# ### end of the forwarding rule ###
```

rsyslog client configuration

Make sure we have below directives.

```
$DefaultNetstreamDriver gtls

$DefaultNetstreamDriverCAFile /etc/pki/tls/private/ca.pem
$DefaultNetstreamDriverCertFile /etc/pki/tls/private/rslclient-cert.pem
$DefaultNetstreamDriverKeyFile /etc/pki/tls/private/rslclient-key.pem

$ActionSendStreamDriverPermittedPeer dhcp-blr-kmgm-blk2-4fl-6fl-10-178-23-94.in.oracle.com =======>>> that should be your rsyslog server
$ActionSendStreamDriverMode 1 # run driver in TLS-only mode
$ActionSendStreamDriverAuthMode x509/name
```

For Example:

```
# cat /etc/rsyslog.conf
# rsyslog v5 configuration file

# For more information see /usr/share/doc/rsyslog-*/rsyslog_conf.html
# If you experience problems, see http://www.rsyslog.com/doc/troubleshoot.html

#### MODULES ####

$ModLoad imuxsock # provides support for local system logging (e.g. via logger command)
$ModLoad imklog # provides kernel logging support (previously done by rklogd)
#$ModLoad immark # provides --MARK-- message capability

# Provides UDP syslog reception
#$ModLoad imudp
#$UDPServerRun 514

# Provides TCP syslog reception
#$ModLoad imtcp
#$InputTCPServerRun 514

#### GLOBAL DIRECTIVES ####

# Use default timestamp format
$ActionFileDefaultTemplate RSYSLOG_TraditionalFileFormat

# File syncing capability is disabled by default. This feature is usually not required,
# not useful and an extreme performance hit
#$ActionFileEnableSync on

# Include all config files in /etc/rsyslog.d/
$IncludeConfig /etc/rsyslog.d/*.conf

$DefaultNetstreamDriver gtls

$DefaultNetstreamDriverCAFile /etc/pki/tls/private/ca.pem
$DefaultNetstreamDriverCertFile /etc/pki/tls/private/rslclient-cert.pem
$DefaultNetstreamDriverKeyFile /etc/pki/tls/private/rslclient-key.pem

$ActionSendStreamDriverPermittedPeer dhcp-blr-kmgm-blk2-4fl-6fl-10-178-23-94.in.oracle.com
$ActionSendStreamDriverMode 1 # run driver in TLS-only mode
$ActionSendStreamDriverAuthMode x509/name

#$ActionSendStreamDriverAuthMode x509/name
#$ActionSendStreamDriverPermittedPeer *
#$ActionSendStreamDriverMode 1 # run driver in TLS-only mode

#### RULES ####

# Log all kernel messages to the console.
# Logging much else clutters up the screen.
#kern.* /dev/console

# Log anything (except mail) of level info or higher.
# Don't log private authentication messages!
*.info;mail.none;authpriv.none;cron.none /var/log/messages

# The authpriv file has restricted access.
authpriv.* /var/log/secure

# Log all the mail messages in one place.
mail.* -/var/log/maillog

# Log cron stuff
cron.* /var/log/cron

# Everybody gets emergency messages
*.emerg *
```

```
# Save news errors of level crit and higher in a special file.
uucp,news.crit /var/log/spooler

# Save boot messages also to boot.log
local7.* /var/log/boot.log

# ### begin forwarding rule ###
# The statement between the begin ... end define a SINGLE forwarding
# rule. They belong together, do NOT split them. If you create multiple
# forwarding rules, duplicate the whole block!
# Remote Logging (we use TCP for reliable delivery)
#
# An on-disk queue is created for this action. If the remote host is
# down, messages are spooled to disk and sent when it is up again.
#$WorkDirectory /var/lib/rsyslog # where to place spool files
#$ActionQueueFileName fwdRule1 # unique name prefix for spool files
#$ActionQueueMaxDiskSpace 1g # 1gb space limit (use as much as possible)
#$ActionQueueSaveOnShutdown on # save messages to disk on shutdown
#$ActionQueueType LinkedList # run asynchronously
#$ActionResumeRetryCount -1 # infinite retries if host is down
# remote host is: name/ip:port, e.g. 192.168.0.1:514, port optional
*.* @@10.178.23.94:514 ========>>>>>> server:port
:msg, contains, "kernel" @@10.178.23.94:514
# ### end of the forwarding rule ###
```

Final step: restart the services on both client/server.

```
service rsyslog restart
Shutting down system logger: [ OK ]
Starting system logger: [ OK ]
```

Expected outputs:

```
# tailf /var/log/messages
Dec 23 22:36:20 server2 kernel: IPv6: eth2: IPv6 duplicate address 2606:b400:c11:68:a00:27ff:fedd:bb31 detected! ========>>>> client messages here
Dec 23 22:38:54 server2 kernel: IPv6: eth1: IPv6 duplicate address 2606:b400:c11:68:a00:27ff:feb6:a80e detected!
Dec 23 22:38:54 server1 kernel: IPv6: eth1: IPv6 duplicate address 2606:b400:c11:68:a00:27ff:feb6:a80e detected!
Dec 23 22:38:55 server1 kernel: IPv6: eth2: IPv6 duplicate address 2606:b400:c11:68:a00:27ff:fedd:bb31 detected!
Dec 23 22:41:28 server2 kernel: IPv6: eth1: IPv6 duplicate address 2606:b400:c11:68:a00:27ff:feb6:a80e detected!
Dec 23 22:41:28 server1 kernel: IPv6: eth1: IPv6 duplicate address 2606:b400:c11:68:a00:27ff:feb6:a80e detected!
Dec 23 22:41:29 server1 kernel: IPv6: eth2: IPv6 duplicate address 2606:b400:c11:68:a00:27ff:fedd:bb31 detected!
Dec 23 22:44:44 server2 kernel: IPv6: eth1: IPv6 duplicate address 2606:b400:c11:68:a00:27ff:feb6:a80e detected!
Dec 23 22:44:44 server1 kernel: IPv6: eth1: IPv6 duplicate address 2606:b400:c11:68:a00:27ff:feb6:a80e detected!
Dec 23 22:44:44 server1 kernel: IPv6: eth2: IPv6 duplicate address 2606:b400:c11:68:a00:27ff:fedd:bb31 detected!
Dec 23 22:45:12 server1 kernel: Kernel logging (proc) stopped. =====>>> client messages got logged to the rsyslogserver
Dec 23 22:45:12 server1 rsyslogd: [origin software="rsyslogd" swVersion="5.8.10" x-pid="6340" x-info="http://www.rsyslog.com"] exiting on signal 15.
Dec 23 22:45:12 server1 kernel: imklog 5.8.10, log source = /proc/kmsg started.
Dec 23 22:45:12 server1 rsyslogd: [origin software="rsyslogd" swVersion="5.8.10" x-pid="6679" x-info="http://www.rsyslog.com"] start
```

```
# tcpdump -nnvvvS -s 0 -U -w /tmp/sniff.rsyslog dst 10.178.23.94 and dst port 514
tcpdump: listening on eth1, link-type EN10MB (Ethernet), capture size 65535 bytes
^C133 packets captured
133 packets received by filter
0 packets dropped by kernel
3211 packets dropped by interface
```

Troubleshooting

Based on errors from rsyslog server:

1. If you see the messages as encrypted and not in normal text then we should check modules order, this happened because of gtls driver was loaded after imtcp port.

```
Dec 23 19:34:36 rsyslogd: [origin software="rsyslogd" swVersion="5.8.10" x-pid="3344" x-info="http://www.rsyslog.com"] start
Dec 23 19:34:41 #026#003#002#000V#001#000#000R#003#002X].�i�)�#005t�#037F�{ot �f�\���Y��L�#013�#000#000$#0003#000E#0009#000�#000#026#0002#000D#0008#000�#000#023#000f#000/#000A#0005#000�
Dec 23 19:34:42 #026#003#002#000V#001#000#000R#003#002X].��#0158��9��n���Mz�S���W"co��#015#000#000$#0003#000E#0009#000�#000#026#0002#000D#0008#000�#000#023#000f#000/#000A#0005#000�
```

2. module not found :

```
Dec 23 20:09:39  rsyslogd-2067: could not load module '/lib64/rsyslog/lmnsd_gtls.so', dlsym: x^D: undefined symbol: modInit
```

A) Check the certificates and reissue or copy it to server.

3. Verify this parameter ” $ActionSendStreamDriverAuthMode x509/name”

```
Dec 23 19:53:20 rsyslogd-2088: error: peer name not authorized - not permitted to talk to it. Names: (null) [try http://www.rsyslog.com/e/2088 ]
Dec 23 19:53:20 rsyslogd-2089: netstream session 0x7fb45c01c0e0 will be closed due to error
[try http://www.rsyslog.com/e/2089 ]
```

Basic configuration errors :

```
Dec 23 19:32:50  rsyslogd: [origin software="rsyslogd" swVersion="5.8.10" x-pid="3262" x-info="http://www.rsyslog.com"] start
Dec 23 19:32:50 rsyslogd-3003: invalid or yet-unknown config file command - have you forgotten to load a module? [try http://www.rsyslog.com/e/3003 ]
Dec 23 19:32:50 rsyslogd: the last error occured in /etc/rsyslog.conf, line 36:"$InputTCPServerStreamDriverAuthMode x509/name"
Dec 23 19:32:50  rsyslogd-3003: invalid or yet-unknown config file command - have you forgotten to load a module? [try http://www.rsyslog.com/e/3003 ]
Dec 23 19:32:50  rsyslogd: the last error occured in /etc/rsyslog.conf, line 37:"$InputTCPServerStreamDriverMode 1 # run driver in TLS-only mode"
```