

IBM QRadar Version 7.5

Common Criteria for NIAP Revision 1.0

IBM

Version 1.0, June 20, 2023

Contents

<i>About this guide</i>	4
<i>Document control</i>	5
<i>Chapter 1. Configuration of Common Criteria on a QRadar All-in-one system</i>	6
What product is being evaluated?	6
NDCPP requirements	6
QRadar and Common Criteria acronyms	7
Evaluated capabilities	7
Supported cipher suites	7
<i>Chapter 2. QRadar installations for highly secure environments</i>	8
Installing QRadar in a STIG environment.....	8
Creating a non-root user in a STIG-compliant environment	9
Running the hardening script on the Console	10
Editing scripts to configure QRadar in STIG environments	10
Changing the boot loader configuration.....	12
Logging in to QRadar	14
Logging out of QRadar.....	15
<i>Chapter 3. Secure communication configuration</i>	16
Disabling auto updates for configuration files	16
Configuring SSH to use public key authentication	16
Configuring the SSHD_config file	17
Configuring Java TLS ciphers	17
Limiting the cipher suites in use	18
<i>Chapter 4. Certificate generation and verification</i>	19
Creating a CSR request by using a 2048-bit RSA key	19
Certification Revocation List	20
Verifying the CA certificates	20
<i>Chapter 5. Configure TLS syslog for inbound data</i>	21
Installing the Certificate Management App.....	21
Importing the root CA of the TLS client	21
Importing the TLS server certificates.....	22
Adding a log source	22
<i>Chapter 6. Configure event forwarding for outbound data</i>	24

Importing the root CA of the TLS server	24
Importing the TLS client certificates.....	25
Adding forwarding destinations to QRadar	25
Configuring routing rules for event forwarding	26
Verifying the event forwarding configuration	29
<i>Chapter 7. QRadar system configuration</i>	<i>30</i>
Configuring system time.....	30
Adding or editing a QRadar login message	30
Administrative logins	31
Configuring the password policy.....	32
Configuring the inactivity time period.....	32
Configuring the local session timeout period	33
Accessing QRadar RESTful API	33
x509 Certificate Management and Validation in QRadar	33
QRadar self-test	37
Verifying secure updates.....	37
Viewing the audit logs.....	37
Sample Audits.....	38
<i>Notices.....</i>	<i>45</i>
Trademarks	47
Terms and conditions for product documentation	47
Applicability.....	47
Personal use	47
Commercial use	48
Rights	48
IBM Online Privacy Statement.....	48
General Data Protection Regulation	49

About this guide

This documentation includes the requirements and procedures for configuring Common Criteria on IBM QRadar.

Intended audience

The intended audience for this guide is:

- System administrators or developers who are configuring Common Criteria for IBM QRadar.
- NIAP certification personnel who are configuring and testing Common Criteria for IBM QRadar.

Technical documentation

To find IBM Security QRadar product documentation in the QRadar products library, see [Accessing IBM Security Documentation Technical Note](#) (www.ibm.com/support/docview.wss?rs=0&uid=swg21614644).

Contacting customer support

For information about contacting customer support, see [QRadar Support – Assistance 101](#) (<https://ibm.biz/qradarsupport>).

Statement of good security practices

IT system security involves protecting systems and information through prevention, detection and response to improper access from within and outside your enterprise. Improper access can result in information being altered, destroyed, misappropriated or misused or can result in damage to or misuse of your systems, including for use in attacks on others. No IT system or product should be considered completely secure and no single product, service or security measure can be completely effective in preventing improper use or access. IBM systems, products and services are designed to be part of a lawful comprehensive security approach, which will necessarily involve additional operational procedures, and may require other systems, products or services to be most effective. IBM DOES NOT WARRANT THAT ANY SYSTEMS, PRODUCTS OR SERVICES ARE IMMUNE FROM, OR WILL MAKE YOUR ENTERPRISE IMMUNE FROM, THE MALICIOUS OR ILLEGAL CONDUCT OF ANY PARTY.

Please Note:

Use of this Program may implicate various laws or regulations, including those related to privacy, data protection, employment, and electronic communications and storage. IBM QRadar may be used only for lawful purposes and in a lawful manner. Customer agrees to use this Program pursuant to, and assumes all responsibility for complying with, applicable laws, regulations and policies. Licensee represents that it will obtain or has obtained any consents, permissions, or licenses required to enable its lawful use of IBM QRadar.

Document control

The following table shows updates that were made to the *IBM QRadar V7.5 Common Criteria for NIAP* documentation.

Date	Revision	Updates
March 27, 2023	1.0	Initial Version with document control using “IBM QRadar Version 7.3.2 – Common Criteria for NIAP Revision 1.4” as source
May 4, 2023	1.0	IBM QRadar Version 7.5 Common Criteria for NIAP Revision 1.0

Chapter 1. Configuration of Common Criteria on a QRadar All-in-one system

The common criteria guide for IBM QRadar provides requirements and procedures for configuring Common Criteria by using prescribed NIAP methodology on a QRadar All-in-One system. The QRadar All-in-One system is a network device that detects potential threats through the review of event data and flow data that is collected from network sources.

What product is being evaluated?

The evaluated product is a Dell 3129C (All-in-One) device that runs IBM QRadar SIEM.

The software identification for the evaluated product is IBM QRadar SIEM V7.5 NIAP.

QRadar consolidates log source event data from device endpoints and applications that are distributed throughout a network. QRadar performs normalization and correlation activities on this raw data and can forward data to another network server when event forwarding is configured. Communication with network peers for either inbound or outbound log event data is accomplished by using TLS protected communication channels. QRadar can authenticate inbound peers by using X.509v3 certificates, or by providing an X.509v3 certificate to authenticate itself as part of an outbound TLS connection.

The QRadar All-in-One is the Target of Evaluation (TOE). The TOE can be administered either locally or remotely. The QRadar product consolidates log source event data from multiple devices, endpoints, and applications distributed throughout a network.

NDcPP requirements

- The following features are required for QRadar to satisfy NDcPP (Network Device Collaborative Protection Profile) requirements: Certificate Revocation is required for all certificates that are used by QRadar.
- TLS protection is needed for inbound and outbound audit or event log transmissions.
- QRadar must offer and demand X.509 certificate authentication for TLS protected communications.
- QRadar must be able to configure specific cryptographic cipher suites that are used with all TLS protected communications.
- QRadar must accept TLS connections only by using TLS version 1.2 or higher.
- QRadar must use a strong entropy source such as Jitter or HAVEGED.
- QRadar must have Cryptographic Algorithm Validation Program (CAVP) certificates for all cryptographic algorithms that are claimed in the security target (ST).

QRadar and Common Criteria acronyms

Several acronyms are used in this guide.

Acronyms

The following table lists the acronyms that are used in this guide.

Table 1. QRadar and Common Criteria acronyms

Acronym	Description
CC	Common Criteria
CCEVS	Common Criteria Evaluation and Validation Scheme
EC	Event Collector
EP	Event Processor
NIAP	National Information Assurance Partnership
NTP	Network Time Protocol
TOE	Target of Evaluation
TSF	TOE Security Functionality

Evaluated capabilities

An IBM QRadar All-in-One appliance is a single appliance that includes complete data collection, data processing, storage, monitoring, searching, reporting, and offense management capabilities. The appliance collects event and flow data from log sources in your network, then processes and stores the data, and makes it available for security monitoring and threat analysis.

The Common Criteria configuration adds support for security capabilities, such as protected transport of event audit data and secure communication by using TLS 1.2 or higher.

Supported cipher suites

During the STIG hardening, QRadar is configured to use TLS protocol version 1.2 and to use the supported cipher suites.

Before you evaluate QRadar in a test environment, we recommend that you configure TLS to support only the ciphers that are required to complete the evaluation.

The following TLS cipher suites are supported:

- TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256
- TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256

Chapter 2. QRadar installations for highly secure environments

This document provides guidance for implementing security standards for IBM QRadar deployments in highly secure environments, such as the federal government. These security standards meet the requirements set by the Defense Information Systems Agency (DISA).

Hardening of the operating system and QRadar hosts to implement the Security Technical Implementation Guide (STIG) standards is part of making QRadar deployments more secure. Some of the steps that are required to secure a QRadar deployment are not specified in the Red Hat Enterprise Linux STIG documents.

The steps in Chapter 2 are part of the TOE installation process, to be performed as initial configuration prior to normal operation.

To configure the Target of Evaluation (TOE) machine into the common criteria evaluated configuration, you must complete the following steps:

1. Install the QRadar software in FIPS mode.
The STIG hardening scripts are included in the installation.
2. Create a non-root user.
3. Run the hardening script on the QRadar console.
4. Edit the QRadar configuration.
5. Modify the GRUB2 boot loader configuration.
6. Reboot the appliance and log in.

Installing QRadar in a STIG environment

Set up the Target of Evaluation (TOE) machine by installing QRadar on a Dell 3129C (All-in-One) appliance.

Before you begin

Ensure that the following requirements are met:

- The required hardware is installed.
- A keyboard and monitor are connected by using the VGA connection.
- You have the required license key for your appliance. The temporary license key is good for 5 weeks.

Procedure

1. Install QRadar in FIPS mode by following the steps in the *IBM QRadar Installation Guide*.
Note: Select **Console All-in-One** as the appliance type.
2. Configure the QRadar Console root user timeout by adding the following line in the `etc/profile` file:

```
[ $UID -eq 0 ] && TMOUT=600
```


What to do next

Create a non-root user.

Creating a non-root user in a STIG-compliant environment

You can't log in remotely as the root user in a STIG-compliant environment.

Create a non-root user who has **sudo** access and choose a non-root user name such as *stiguser*.

Procedure

1. To create the non-root user, type the following commands:

```
useradd -c 'Admin User' -d /home/stiguser -m -s /bin/bash stiguser  
passwd stiguser
```

The password must follow these guidelines:

- a. Consist of 15 or more characters.
- b. Not repeat the same character consecutively more than two times.
- c. Not repeat the same character type consecutively more than two times.
- d. Have at least one uppercase character.
- e. Have at least one numerical character.
- f. Have at least one special character.

Tip: These new password requirements are enforced when the STIG script is run. If your root password doesn't meet these requirements, you can change it now.

2. Edit the `/etc/sudoers` file.
 - a. At the end of the file, type the following line:

```
stiguser    ALL=(ALL)    ALL
```

Note: It is conventional to use tabs for white space but it's not a requirement; for example: `stiguser ALL=(ALL) ALL`

- b. Use the `#` symbol to comment out any lines that contain `NOPASSWD`.

Tip: If you use the Vim text editor, type `:/NOPASSWD` in command mode to search for any instances of `NOPASSWD`.

3. Verify that the new user can log in from a remote host and use the `sudo` command to become a root user.
For example, use an SSH client such as PuTTY to log in to IBM QRadar as `stiguser`, and then run a command by using `sudo`.

```
sudo cat /etc/shadow
```

Running the hardening script on the Console

To help secure the system, you must run hardening scripts on the IBM QRadar Console.

Before you begin

Before you run the hardening script, verify that the *stiguser* can log in remotely.

Procedure

1. Go to the STIG directory by typing the following command:

```
cd /opt/qradar/util/stig/bin
```
2. Run the STIG hardening script by typing the following command:

```
./stig_harden.sh -a
```
3. Type *yes* at the following prompt: **Do you want to continue (yes/no)?**
4. Restart the QRadar appliance.
5. Verify that the *stiguser* can log in remotely and can elevate to root with the following command

```
sudo cat /etc/shadow.
```
6. Change the root user's password to meet the password requirements. Ensure that the root authentication works locally.

Editing scripts to configure QRadar in STIG environments

Extra configuration tasks, such as configuring the mail server, disabling the DHCP client, updating iptables, and changing the backup log directory location are required when you configure QRadar in STIG environments.

Procedure

1. To ensure that the mail server on each host is listening on local interfaces.
 - a) Make a backup copy of the `/etc/postfix/main.cf` file.
 - b) Edit the `/etc/postfix/main.cf` file and verify that the `inet_interfaces` line is similar to one of the following examples:
 - `inet_interfaces = localhost.`
 - `inet_interfaces = loopback-only.`
2. Verify that the `BOOTPROTO` parameter is set to `none` or `static` in the configuration files.
 - a) Type the following command:

```
grep -rl BOOTPROTO /etc/sysconfig/network-scripts/ifcfg*
```
 - b) For each interface configuration file that is returned, where `BOOTPROTO` does not equal `none` or `static`, change the `BOOTPROTO` value to `none`.

Example:

```
DEVICE=eth0
ONBOOT=yes
BOOTPROTO=none
IPADDR=192.168.122.254
IPADDR=192.168.122.254
```

3. Change IPTables and set the default INPUT policy to DROP.

- a) Make a backup copy of the `/opt/qradar/bin/iptables_update.pl` file.
- b) Edit the `/opt/qradar/bin/iptables_update.pl` file and change `INPUT ACCEPT [0:0]` to `INPUT DROP [0:0]`.
- c) Run the `/opt/qradar/bin/iptables_update.pl` script.

4. Add the following line to the `/etc/hosts.allow` file on the QRadar Console:

```
time: ALL
```

5. Change the backup log directory.

- a) Search for the `/var/log/backup.log` log file and if it exists, move the file to `/store/LOGS`.

Note: The `/var/log/backup.log` does not exist on a fresh installation.

- b) Make a backup copy of the `/opt/qradar/bin/backup.sh` file.
- c) Edit the `/opt/qradar/bin/backup.sh` file and change this text:

```
InitLog @syslog:local1.info || ErrorExit 'Failed to initialize logging'
```

To this text:

```
InitLog /store/LOGS/backup.log || ErrorExit 'Failed to initialize logging'
```

6. Create an AIDE baseline and schedule integrity checks. Then, after you make the configuration changes, perform aide updates.

- a) As root user, initialize the AIDE database by typing the following command:

```
aide --init
mv -f /var/lib/aide/aide.db.new.gz /var/lib/aide/aide.db.gz
```

b) To reduce the system startup time, schedule the AIDE integration check to run at daily intervals. For example, to schedule the AIDE checks to run at 4:20 AM every day, log in as the root user and edit `/etc/crontab` by adding

```
20 4 * * * root /usr/sbin/aide --check.
```

Scheduling daily interval checking ensures that the checks are run more frequently. The administrator can change the schedule later if a different time period or interval is required.

c) To view the scheduled AIDE output, type the following command:

```
sudo less /var/log/aide/aide.log
```

Important: If there are unapproved changes, the system might be compromised. The administrator should perform a factory installation. For more information, see Chapter 2, “QRadar installations for highly secure environments”.

d) After reviewing the changes, create a new baseline:

```
aide --update
mv -f /var/lib/aide/aide.db.new.gz /var/lib/aide/aide.db.gz
```

e) Run the aide update after you install, uninstall, change the system configuration, or run a QRadar deployment action.

```
aide --update
mv -f /var/lib/aide/aide.db.new.gz /var/lib/aide/aide.db.gz
```

What to do next

Modify the GRUB2 boot loader configuration.

Changing the boot loader configuration

You must update the GRUB2 configuration to configure the non-root user for the STIG environment, and for the changes that were made by the hardening script to be effective. You must update the GRUB2 configuration on the QRadar Console, event processors, and flow processors.

Procedure

1. Enter the following command to back up the GRUB 2 configuration files:

```
tar -cvf /root/grub2backup.tar /etc/grub.d /etc/default/grub
/boot/grub2
```

2. Create a `/boot/grub2/user.cfg` file that uses the GRUB password utility by running the following command that prompts for the password:

```
grub2-setpassword -o /boot/grub2/
```

3. Edit `/etc/grub.d/10_linux` to replace `--unrestricted` with `--users root` on the line beginning with `CLASS=`.

4. Save the changes and exit.

5. Run the command `grub2-mkconfig -o /boot/grub2/grub.cfg`.

If you are completing a software (non-appliance) installation, the procedure is now complete.

Important: If you are completing an appliance installation (there is a `/recovery` partition), then the following steps must also be completed or you cannot boot the system. If the `/recovery` partition is not automatically mounted, you have to mount it manually.

6. If the file `/recovery/grub2/grub.cfg` exists, copy the users file `cp /boot/grub2/user.cfg /recovery/grub2/`.

7. Edit `/recovery/grub2/user.cfg` and find the line `menuentry "Normal System"`.

- a. Insert the content of file `/boot/grub2/user.cfg` on the line before the `menuentry "Normal System"` line. The result appears similar to the following example (all one line):

```
GRUB2_PASSWORD=grub.pbkdf2.sha512.10000.00F025BA99D48B00BCCA5C45F
9F3
0A29AAB2B1B2B6369B3783A948DB117E81CE0A6ADD035CF0C4E2F223455869944
B142F41B265C59E242E8661B2D0B0CC9D37.871FE29A0318BA50F40C103346EC5
DFB5573F141D5D98ABE9B5B985804FF95B2392D5497247F820100212BFF4E3FCA
0525FD28A0C60E4E961AE9A94DB0086B3F
```

- b. On the line after `GRUB2_PASSWORD`, insert the following lines:

```
set superusers="root"
password_pbkdf2 root ${GRUB2_PASSWORD}
```

- c. At the end of each `menuentry` line, and before the `{ add --users root`.

```
menuentry "Normal System" --users root {
    and
    menuentry "Factory re-install [QRadar <version_number>]" --users
root {
```

8. Run the script `grub2-mkconfig -o /recovery/grub2/grub.cfg`.

9. Save and exit and then restart the system.

The bootup user is `root` and the password is the one from the previous step, `grub2-setpassword`.

What to do next

Reboot the appliance and log in.

Logging in to QRadar

Only security administrators can login to the Target of Evaluation (TOE) system via its command line or Web GUI interfaces. This restricts access to all management functions, including management of X.509 certificates. To access the web interface on your IBM QRadar appliance, log in remotely using your web browser.

No features of the QRadar console, aside from login, are available until after successful authentication.

About this task

IBM QRadar is a web-based application. For the features to work properly, you must use a supported web browser.

Table 2. Supported web browsers for QRadar products

Web browser	Supported versions
64-bit Mozilla Firefox	Latest
64-bit Microsoft Internet Explorer with Microsoft Edge mode enabled.	Latest
64-bit Google Chrome	Latest

Procedure

1. In your browser window, type https://<QRadar_IP_Address>.
To log in to QRadar in an IPv6 or mixed environment, wrap the IP address in square brackets:

```
https://[<QRadar_IP_Address>].
```

2. Type the log in credentials:

- User name: admin
- Password: <Password that was created during the installation process>

The default license key provides you access to the system for 5 weeks.

Note: To access the command-line interface on your IBM QRadar appliance, type the following command in a terminal on a remote system:

```
ssh stiguser@<QRadar_IP>
```

Logging out of QRadar

To log out of the QRadar administrative session, follow these steps:

Procedure

1. To log out of the web interface, click the user icon in the top-right corner of the page, and then click Log out.
2. To log out of the remote SSH session, type 'logout' or 'exit'.
3. To log out of the local session, type 'logout' or 'exit'.

Chapter 3. Secure communication configuration

The steps in Chapter 3 are part of the TOE installation process, to be performed as initial configuration prior to normal operation.

After you've hardened the Qradar system by using the STIG scripts, you must prepare for configuring TLS secure communication.

1. "Disabling auto updates for configuration files".
2. Configure SSH to use public key authentication only.
3. Configure the SSHD_config file.
4. "Configuring Java TLS ciphers".
5. Limit the cipher suites in use.

Disabling auto updates for configuration files

Configure Qradar so that it does not automatically apply updates to the configuration files.

Procedure

1. Click the **Admin** tab.
2. On the navigation menu, click **System Configuration > Auto update**.
3. Click **Change settings**.
4. In the **Update types** section, under **Configuration updates**, change the **Update type** to **Disable**.
5. Click **Save**.
6. Click **Save** when prompted for **Application Restart Required**.

Configuring SSH to use public key authentication

Configure SSH authentication on the IBM Qradar 3129 (All-in-One) appliance to accept only public key authentication of users.

Before you begin

On the remote computer that is used to connect to Qradar, locate the SSH public key to be used for a specific Qradar console login user (e.g. stiguser). Each user on the Qradar console will have their own store of public keys to be used for authentication. This store is the `.ssh/authorized_keys` folder within the user's home directory on the IBM Qradar 3129 (All-in-One) appliance (e.g. `/home/stiguser/.ssh/authorized_keys`).

Procedure

1. At the command line on the Qradar 3129 (All-in-One), copy the public key to the target user's `authorized_keys` file. The public key should be appended to the file, on its own line. It will appear similar to, but longer than, the following (all on one line):

```
ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAQCAQDZd1io1rT4x1F1j24hrcHNDQGFUvj9mm+19iZL/ttoVD
ZDJoz+5J+Liz8Hf3EGB ... zGE4WU5ckIr+GV06sRLkGQ== stiguser@host.example.com
```


Note: this step can be repeated with public keys from each remote computer that is permitted to login with a given user.

2. Next, edit the `sshd_config` file by typing the following command:

```
vi /etc/ssh/sshd_config
```

3. Set `PubkeyAuthentication` to `Yes`. This instructs SSHd to attempt Pubkey authentication for new sessions.
4. Optionally, set `PasswordAuthentication` to `No` if Pubkey authentication is to be used exclusively, this will disable password-based SSH logins.
5. Restart SSHD by typing the following command:

```
service sshd restart
```

Configuring the `SSHD_config` file

Edit the `/etc/ssh/sshd_config` file to control the ciphers that are available.

Procedure

1. In the `/etc/ssh/sshd_config` file, uncomment the `HostKey` line that references the RSA key. Leave the other `HostKey` lines commented out.

Note: `HostKeys` will be generated automatically when the `SSHd` service is restarted. To regenerate `Host Keys`, delete `/etc/ssh/ssh_host_*` key files (`rm /etc/ssh/ssh_host_*`) and restart `SSHd` with `systemctl restart sshd`.

The following example shows what the file looks like when you uncomment the `HostKeys` line that references the RSA key.

```
HostKey /etc/ssh/ssh_host_rsa_key  
#HostKey /etc/ssh/ssh_host_dsa_key  
HostKey /etc/ssh/ssh_host_ecdsa_key
```

2. Replace the `Ciphers`, `Macs` and `KexAlgorithms` lines near the bottom of the file with the following three lines:

```
Ciphers aes128-cbc,aes256-cbc  
KexAlgorithms diffie-hellman-group14-sha1,diffie-hellman-group14-sha256  
Macs hmac-sha2-512,hmac-sha2-256
```

3. Type the following command to restart the service:

```
systemctl restart sshd
```

Note: The SSH session rekey is hardcoded to rekey after one hour or following the exchange of one GB of data, whichever comes first.

Configuring Java TLS ciphers

Follow these steps to exclude cipher suites from the Java security policy file.

Procedure

Type this command to edit the configuration file:

```
sudo vim /opt/ibm/java-x86_64-80/jre/lib/security/java.security
```

Replace this line of text:

```
jdk.tls.disabledAlgorithms=SSLv3, TLSv1, TLSv1.1, RC4, DES, MD5withRSA, DH keySize < 1024, DESede, \ EC keySize < 224, 3DES_EDE_CBC, anon, NULL, DES_CBC
```

with this text:

```
jdk.tls.disabledAlgorithms=SSLv3, TLSv1, TLSv1.1, RC4, DES, MD5withRSA, DH keySize < 1024, DESede, EC keySize < 224, 3DES_EDE_CBC, anon, NULL, DES_CBC, SHA1, SHA384, DHE, SSL_RSA_WITH_AES_128_GCM_SHA256, SSL_RSA_WITH_AES_256_GCM_SHA384, TLS_RSA_WITH_AES_128_CBC_SHA, SSL_RSA_WITH_AES_128_CBC_SHA, TLS_RSA_WITH_AES_256_CBC_SHA, SSL_RSA_WITH_AES_256_CBC_SHA, SSL_RSA_WITH_AES_128_CBC_SHA256, SSL_RSA_WITH_AES_256_CBC_SHA256
```

Reboot the appliance for the changes to take effect.

Limiting the cipher suites in use

Configure QRadar to use only those cipher suites that are required for the evaluation.

NOTE: The TOE does not allow administrators to configure the Supported Elliptic Curves/Supported Groups Extension sent during TLS negotiations.

Procedure

1. Type the following command to edit the ssl.conf file:

```
vim /etc/httpd/conf.d/ssl.conf
```

2. Replace the SSLCipherSuite line with the following text:

```
SSLCipherSuite ECDHE-RSA-AES128-GCM-SHA256:ECDHE-RSA-AES128-SHA256
```

3. Type the following command to restart the service:

```
systemctl restart httpd
```

Chapter 4. Certificate generation and verification

Creating a CSR request by using a 2048-bit RSA key

The steps in Chapter 4 are part of the TOE installation process, to be performed as initial configuration prior to normal operation.

Generate an SSL certificate request from your IBM QRadar Console.

Procedure

1. Use an SSH client to log in to QRadar Console.
2. Generate a 2048-bit RSA private key file by typing the following command.

```
openssl genrsa -out qradar.key 2048
```

3. The `qradar.key` file is created in the current directory.
4. Generate the certificate signing request (CSR) file by using the command:

```
openssl req -new -key qradar.key -out qradar.csr
```

5. The `qradar.csr` file is created for the certificate authority (CA) to use.
6. When prompted in the command line, provide the following information:

```
Country Name (2 letter code) [XX]:CA
State or Province Name (full name) []:NB
Locality Name (eg, city) [Default City]:Fredericton
Organization Name (eg, company) [Default Company Ltd]:IBM
Organizational Unit Name (eg, section) []:SI
Common Name (eg, your name or your server's hostname) []:myhostname
Email Address []:username@example.com
```

7. If the command line asks for more properties, leave the fields empty.

Note: If you enter a password for the `Challenge Password` property and you forget the entry, you might not be able to use the CSR. The CA might not support a challenge password.

8. Before you send the CSR, verify the information by typing the following command:

```
openssl req -noout -text -in qradar.csr
```

9. Use Secure File Transfer Protocol or another program to securely copy the CSR file to your computer.
10. Submit the CSR to the certificate authority.
11. Once the signed certificate is returned from the CA, copy it to the QRadar console. Execute `/opt/qradar/bin/install-ssl-cert.sh` and follow the prompts to provide the certificate, key and any intermediate certs.

12. After `install-ssl-cert.sh` runs successfully, delete the original key file created in step 2:

```
rm -f qradar.key
```

Certification Revocation List

Certification revocation activities are an installation task when performed as part of the initial system setup and are an operational task when performed after initial system setup.

IBM QRadar uses the certificate revocation list (CRL) to provide a list of certificates that are revoked.

On the TOE, QRadar supports only the CRL file for certificate revocation checks. The TOE does not support Online Certificate Status Protocol (OCSP).

The CRL distribution point for the server and client certificates must be reachable from the TOE. If the CRLDP is unreachable by the QRadar console, the certificate will be rejected.

To revoke a certificate by using the certificate revocation list (CRL) the certificate must contain one or more "CRL distribution points", which are lists of URL's that point to the location of the certificate revocation lists. The TOE will cache these lists to avoid repeated downloads of large CRL files. By default, the TOE will cache CRL files for one day.

Verifying the CA certificates

Certificate verification is performed automatically by the certificate management app when the app is used to upload the certificate. See Chapter 5 for a detailed procedure on using the certificate management app.

Chapter 5. Configure TLS syslog for inbound data

Configuring TLS activities are an installation task when performed as part of the initial system setup and are an operational task when performed after initial system setup.

Installing the Certificate Management App

Procedure

1. Download the QRadar Certificate Management from the IBM App Exchange in .zip format
2. Log in to the QRadar Console with an admin account
3. Navigate to the admin tab.
4. Select “Extensions Management”
5. In the “Extensions Management” popup window, click “Add” in the upper right corner of the window
6. In the “Add a New Extension” popup window, click “Browse”
7. Select the zip file that was downloaded in step 1.
8. Check the box with the label “Install immediately”
9. click “Add”
10. When the “QRadar Certificate Management” popup window opens, select “Install”
11. Once the install completes, select “OK”
12. In the “Administer Application(s)” popup window, select “Yes”
13. Ensure that “QRadar Certificate Management” appears under the “Installed Extensions” list

Importing the root CA of the TLS client

Note: certificate management, import and deletion steps are described more completely in Chapter 7.

Procedure

1. Log in to the QRadar Console with an admin account
2. Navigate to the admin tab.
3. Select the “QRadar Certificate Management” app under the header “QRadar Certificate Management”.
4. Navigate to the “Root Certificates” tab
5. Click “Upload” in the top right corner
6. Drag and drop the root certificate to be uploaded into the “Upload new root certificate” popup window
7. Select “Confirm”
8. Select “Submit”
9. Close the “IBM QRadar Certificate Management” popup window and return to the admin tab on the main console

10. Select “Deploy Changes” in the upper toolbar

Importing the TLS server certificates

Note: certificate management, import and deletion steps are described more completely in Chapter 7.

Procedure

1. Log in to the QRadar Console with an admin account
2. Navigate to the admin tab.
3. Select the “QRadar Certificate Management” app under the header “QRadar Certificate Management”
4. In the “IBM QRadar Certificate Management” window, navigate to the “Client/Server Certificates” tab
5. Select “Upload” in the top right corner of the window
6. Input a name for the server certificate into the text box under the “Name *” label
7. Select “Server” in the dropdown list under the “Purpose *” label
8. Select “Log Source” in the dropdown list under the “Component” label
9. Drag and drop the server key into the space labelled “Drag and drop the key file here or upload *”
10. Select “Confirm”
11. Drag and drop the server certificate into the space labelled “Drag and drop the certificate file here or upload *”
12. Select “Confirm”
13. Drag and drop the intermediate file into the space labelled “Drag and drop the intermediate file here or upload *”
14. Select “Confirm”
15. Click “Submit” (Note: The space labelled “Drag and drop the root-certificate here or upload” can be left without a file as long as the the root certificate was imported as per the procedure under “Importing the root CA of the TLS client”)
16. Verify that the given friendly name of the uploaded certificate appears in the list on the “Client/Server Certificate” tab of the “IBM QRadar Certificate Management” window.
17. Click on the friendly name of the uploaded certificate to open a “Certificate Details” popup window to get an overview of the certificate.
18. Select “Close” to close the popup window.
19. Close the “IBM QRadar Certificate Management” popup window
20. Navigate to the admin tab on the main console
21. Select “Deploy Changes” in the upper toolbar

Adding a log source

Add a log source to receive events from your network devices or appliances.

Procedure

1. Log in to the QRadar Console with an admin account
2. Navigate to the admin tab.
3. select “Log Sources”
4. In the “IBM QRadar Log Source Management” popup window, select “Log Sources – Manage Log Sources”
5. Select “+New Log Source” in the upper right corner of the “IBM QRadar Log Source Management” window
6. Select “Single Log Source”
7. Select “Universal DSM” as the Log Source Type
8. Select “Step 2: Select Protocol Type” in the bottom right of the window
9. Select “TLS Syslog” as the protocol type
10. Select “Step 3: Configure Log Source Parameters” in the bottom right of the window
11. Provide a name for the log source in the text field labelled “Name **”
12. Select “Step 4: Configure Protocol Parameters” in the bottom right of the window
13. Provide a Log Source Identifier in the text field labelled “Log Source Identifier **”
14. Select “TLS” in the dropdown list labelled “Authentication Mode **”
15. Select “Choose from QRadar Certificate Store” in the dropdown list labelled “Server Certificate Type **”
16. Select the friendly name of the server certificate given as per the procedure for “Importing the TLS server certificates” from the dropdown list labelled “Server Certificate Store Name **”. If a dropdown list doesn’t appear, type the friendly name into the text box.
17. Select “TLS 1.2 and above” in the dropdown list labelled “TLS Protocols **”. If that option is not present, select “TLS 1.1 and above”. TLS 1.2 and above will still be used regardless.
18. Select “Step 5: Test Protocol Parameters” in the bottom right of the window.
19. Select “Start Test”
20. Once the test completes, select “Finish”. The created log source should now be in the log sources list in the “IBM QRadar Log Source Management” window
21. Close the “IBM QRadar Log Source Management” window
22. Navigate to the admin tab on the main console
23. Select “Deploy Changes” in the upper toolbar if prompted

Chapter 6. Configure event forwarding for outbound data

Configuring event forwarding is an installation task when performed as part of the initial system setup and an operational task when performed after initial system setup.

You can configure IBM QRadar systems to forward data to one or more vendor systems, such as ticketing or alerting systems. You can also forward normalized data to other QRadar systems. The target system that receives the data from QRadar is known as a *forwarding destination*.

With the exception of domain tagging, QRadar systems ensure that all forwarded data is unaltered. Domain information is removed from forwarded data. Events and flows that contain domain information are automatically assigned to the default domain on the receiving system.

To avoid compatibility problems when sending event data, ensure that the QRadar deployment that receives the data is the same version or higher than the deployment that is sending the data.

The TLS event forwarding connections in QRadar are capable of using X.509 validation.

Follow these steps to configure QRadar to use TLS and full X.509 validation:

1. Install the certificate management app (see procedure “Installing the Certificate Management App” in Chapter 5)
2. Import the root CA of the TLS server.
3. Import the TLS client certificates.
4. Create a forwarding destination.
5. Configure a routing rule to forward events to the target device.
6. Perform the procedure “Verifying the event forwarding configuration”.

Importing the root CA of the TLS server

1. Log in to the QRadar Console with an admin account
2. Navigate to the admin tab.
3. Select the “QRadar Certificate Management” app under the header “QRadar Certificate Management”.
4. Navigate to the “Root Certificates” tab
5. Select “Upload” in the top right corner
6. Drag and drop the root certificate to be uploaded into the “Upload new root certificate” popup window
7. Select “Confirm”
8. Select “Submit”
9. Close the “IBM QRadar Certificate Management” popup window
10. Navigate to the admin tab on the main console
11. Select “Deploy Changes” in the upper toolbar

Importing the TLS client certificates

1. Log in to the QRadar Console with an admin account
2. Navigate to the admin tab.
3. Select the “QRadar Certificate Management” app under the header “QRadar Certificate Management”.
4. In the “IBM QRadar Certificate Management” window, navigate to the “Client/Server Certificates” tab
5. Select “Upload” in the top right corner of the window
6. Input a name for the client certificate into the text box under the “Name *” label
7. Select “Client” in the dropdown list under the “Purpose *” label
8. Select “Event Forwarding” in the dropdown list under the “Component” label
9. Drag and drop the client key into the space labelled “Drag and drop the key file here or upload **”
10. Select “Confirm”
11. Drag and drop the client certificate into the space labelled “Drag and drop the certificate file here or upload **”
12. Select “Confirm”
13. Drag and drop the intermediate file into the space labelled “Drag and drop the intermediate file here or upload **”
14. Select “Confirm”
15. Click “Submit” (Note: The space labelled “Drag and drop the root-certificate here or upload” can be left without a file as long as the the root certificate was imported as per the procedure under “Importing the root CA of the TLS server”)
16. Verify that the given friendly name of the uploaded certificate appears in the list on the “Client/Server Certificate” tab of the “IBM QRadar Certificate Management” window.
17. Click on the friendly name of the uploaded certificate to open a “Certificate Details” popup window to get an overview of the certificate.
18. Select “Close” to close the popup window.
19. Close the “IBM QRadar Certificate Management” popup window
20. Navigate to the admin tab on the main console
21. Select “Deploy Changes” in the upper toolbar

Adding forwarding destinations to QRadar

Before you can configure bulk or selective data forwarding, you must add forwarding destinations on the QRadar console.

Procedure

1. Log in to the QRadar Console with an admin account
2. Navigate to the admin tab.
3. Select “Forwarding Destinations”
4. In the “Forwarding Destinations” popup window, select “Add” in the upper left corner
5. Provide the name and address for the forwarding destination in the associated text fields labelled “Name:” and “Destination Address:”. This destination should be a remote syslog server or another QRadar host. This destination must match the hostname provided by the certificate presented by the TLS peer. QRadar does not allow IP addresses to be configured as the identifier associated with a TLS connection.
6. In the dropdown list under the label “Protocol”, select “TCP over TLS 1.1. or above. Note that even though this name lists TLS 1.1, only TLS 1.2 or higher will be used.
7. Check the box beside “Enable client authentication”
8. In the dropdown list under the label “Client Certificate:”, select the client certificate that was created.
9. Select “Save”

Configuring routing rules for event forwarding

After you added one or more forwarding destinations, you can create filter-based routing rules to forward event data.

About this task

You can configure routing rules to forward data in either online or offline mode:

- In **Online** mode, your data remains current because forwarding is performed in real time. If the forwarding destination becomes unreachable, data can potentially be lost.
- In **Offline** mode, all data is stored in the database and then sent to the forwarding destination. This assures that no data is lost, however, there might be delays in data forwarding.

In either mode the QRadar console will reconnect TLS pathways after an unintentional disconnect.

The following table describes some of the **Routing Rules** parameters

Table 3. Routing Rules window parameters

Parameter	Description
Forwarding Event Collector	This option is displayed when you select the Online option. Specifies the Event Collector that you want this routing rule to process data from.
Forwarding Event Processor	This option is displayed when you select the Offline option. Specifies the Event Processor that you want this routing rule process data from.

	Restriction: This option is not available if Drop is selected from the Routing Options pane.
--	---

Table 3. **Routing Rules** window parameters (continued)

Routing Options	<ul style="list-style-type: none"> • The Forward option specifies that data is forwarded to the specified forwarding destination. Data is also stored in the database and processed by the Custom Rules Engine (CRE). • The Drop option specifies that data is dropped. The data is not stored in the database and is not processed by the CRE. This option is not available if you select the Offline option. • The Bypass Correlation option specifies that data bypasses CRE, but it is stored in the database. This option is not available if you select the Offline option. <p>You can combine two options:</p> <ul style="list-style-type: none"> • Forward and Drop <p style="padding-left: 40px;">Data is forwarded to the specified forwarding destination. Data is not stored in the database and is not processed by the CRE.</p> <ul style="list-style-type: none"> • Forward and Bypass Correlation <p style="padding-left: 40px;">Data is forwarded to the specified forwarding destination. Data is also stored in the database, but it is not processed by the CRE. The CRE at the forwarded destination processes the data.</p> <p style="padding-left: 40px;">If data matches multiple rules, the safest routing option is applied. For example, if data that matches a rule that is configured to drop and a rule to bypass CRE processing, the data is not dropped. Instead, the data bypasses the CRE and is stored in the database.</p> <p style="padding-left: 40px;">All events are counted against the EPS license.</p>
-----------------	---

Procedure

1. On the Admin tab, click System Configuration.
2. Click the Routing Rules icon, and click Add.
3. Type a name and description for your routing rule.
4. In the Mode field, select either Online or Offline.

QRadar uses an Event Collector for **Online** mode, and an Event Processor for **Offline** mode.

5. In the **Forwarding Event Collector** or **Forwarding Event Processor** list box, select the appliance from which you want to forward data.

6. In the **Data Source** field, select **Events**.
7. In the **Event Filters** box, specify the filter criteria:
 - a) To forward all incoming data, select the Match All Incoming Events check box. Restriction: If you select this check box, you cannot add a filter.
 - b) To add a filter, specify the filter criteria and click Add Filter.
 - c) Repeat the steps for each filter that you want to add.
8. In the Routing Options box, specify the routing options to apply to the forwarded data:
 - a) If you want to edit, add, or delete a forwarding destination, click the Manage Destinations link.
 - b) To forward log data that matches the specified filters, select the Forward check box and then select the check box for each forwarding destination.

Restriction: If you select the **Forward** check box, you can select only one of these check boxes: **Drop**, **Bypass Correlation**, or **Log Only**.

Learn more about routing options:

- The **Forward** option specifies that data is forwarded to the specified forwarding destination. Data is also stored in the database and processed by the Custom Rules Engine (CRE).
- The **Drop** option specifies that data is dropped. The data is not stored in the database and is not processed by the CRE. This option is not available if you select the **Offline** option. Any events that are dropped are credited back 100% to the license.
- The **Bypass Correlation** option specifies that data bypasses CRE, but it is stored in the database. This option is not available if you select the **Offline** option.
- The **Log Only (Exclude Analytics)** option specifies that events are stored and flagged in the database as Log Only and bypass CRE. These events are not available for historical correlation, and are credited back 100% to the license. This option is not available for flows.
- **Note:** The **Log Only** option requires a license for QRadar Data Store. After the license is applied and the **Log Only** option is selected, events that match the routing rule will be stored to disk and will be available to view and for searches. The events bypass the custom rule engine and no real-time correlation or analytics occur. The events can't contribute to offenses and are ignored when historical correlation runs. Some apps will also ignore Log Only events (<https://www.ibm.com/support/docview.wss?uid=swg22009471>).
- You can combine three options:
 - **Forward** and **Drop**
 - Data is forwarded to the specified forwarding destination. Data is not stored in the database and is not processed by the CRE.

- **Forward and Bypass Correlation**
Data is forwarded to the specified forwarding destination. Data is also stored in the database, but
 - it is not processed by the CRE. The CRE at the forwarded destination processes the data.
- **Forward and Log Only (Exclude Analytics)**
 - Events are forwarded to the specified forwarding destination in online mode. Events are stored and flagged in the database as Log Only and bypass CRE. These events are not available for historical correlation, and are credited back 100% to the license. This option is not available in offline mode.
 - If data matches multiple rules, the safest routing option is applied. For example, if data that matches a rule that is configured to drop and a rule to bypass CRE processing, the data is not dropped. Instead, the data bypasses the CRE and is stored in the database.

9. Click Save.

Verifying the event forwarding configuration

Follow these steps to verify the event forwarding configuration.

Before you begin

On a remote system, gather the server certificate, the intermediate certificate, and the root CA that issued the server certificate.

If the intermediate certificate and the root CA for the client are different from those of the server, you will also need to gather the client intermediate certificate and root CA.

Combine all the certificate files into a single CA file.

Procedure

On a remote host, run the following command to start the TLS server:

```
openssl s_server -cert server.crt -key server.key -CAfile <CA_file>
-verify 3 -accept <Port_Defined_In_Forwarding_Dest> -Verify on -
verify_return_error
```

Results

The TLS server starts and opens a session. If there are events that meet the criteria that is defined in the routing rule, QRadar forwards those events to this TLS server and they are printed out in the opened session.

Note: When a connection to the remote TLS server is disconnected unintentionally, QRadar tries to re-establish the connection automatically.

Chapter 7. QRadar system configuration

System configuration is an installation task when performed as part of the initial system setup and an operational task when performed after initial system setup. Unless otherwise noted below.


You need to configure some system settings and follow the guidelines that are provided Complete the following tasks:

- Configure time settings.
- Create a login banner.
- Login using the different system access methods.
- Follow the guidelines for password creation.
- Complete a QRadar self-test.

Configuring system time

Set the *system time* on your IBM QRadar Console manually. Time can be set by an authenticated administrator through the WebUI or at the Command Line Interface using the procedures shown below.

WebUI Procedure

1. On the navigation menu () , click Admin.
2. In the System Configuration section, click System and License Management.
3. In the Display list, select Systems.
4. Select the host for which you want to configure the system time settings.
5. From the Actions menu, click View and Manage System.
6. Click the System Time tab.
7. To configure time on the QRadar Console, follow these steps:
 - a) In the Time Zone list, select the time zone that applies to the QRadar Console.
 - b) To manually configure the time, click Set time manually:, and then set the date and time for the console.

Note: If you set the system time to a future date that is affected by Daylight Saving Time (DST) changes, the time you set is adjusted by 1 hour. For example, on 4 July 2016 in the U.S.A, you set the date to December 16, 2016 and the time to 8:00 PM. The time that you set ignores the DST change and is adjusted to 7:00 PM.

8. Click Save.
9. Click OK to accept that services are restarted, or Cancel to cancel the changes.

Command Line Interface Procedure

1. Time can be displayed using the command: `date`
2. Time can be set to a new value using the command: `date -s '2022-12-22 12:22:23'`


Note: Setting the time using the command line interface has the exact same effect and limitations as setting the time through the WebUI.

Adding or editing a QRadar login message

Create a new login message or edit an existing login message on your IBM QRadar Console.

Procedure

To create the login message for the UI:

1. On the navigation menu () , click Admin.
2. In the User Management section, click Authentication.
3. Click General Authentication Settings.
4. In the Login Page section, enable Login Message
 - a) Type your message in the Edit Login Message window.
 - b) To force users to consent to the login message before they can log in, enable Require Explicit consent of this message for Login in the Login Page section.
 - c) Click Save Settings.
5. To see your changes, log out of QRadar.

To create the login message for the CLI (SSH or console)


1. Use SSH to login to the QRadar console.
2. Edit the /etc/issue file, replace the content with the login banner text.
3. Log out and log back in to see the banner.

Administrative logins

When you initially configure IBM QRadar, you must create user accounts for all users that require access to the system. You can use the **User Management** feature on the **Admin** tab to define user roles, security profiles, and user accounts to control who has access to the system, which tasks they can perform, and which data they have access to.

You can configure QRadar to block remote login requests from a user account for a configurable period of time after a configurable number of failed remote login attempts. Admin accounts are never locked out from using the local console.

Procedure

1. On the navigation menu () , click **Admin**.
2. In the **User Management** section, click **Authentication**.
3. Click **General Authentication settings** and configure the following settings:
 - a) In the **Maximum Account Login Failures** field, specify the number of times a login attempt can fail.
 - b) In the **Account Login Failure Attempt Window (in minutes)** field, specify the length of time during which a maximum number of login failures can occur before the system is locked.

- c) In the Account **Login Failure Block Time (in minutes)** field, specify the length of time that the system is locked if the maximum login failures value is exceeded.
 - d) When the configured **Maximum Account Login Failures** per account is reached within the number of minutes set in the **Account Login Failure Attempt Window (in minutes)**, the user account will be locked out.
4. Click **Save Settings**.
 5. On the **Admin** tab menu, select **Advanced > Deploy Full Configuration**.

Note: QRadar continues to collect events when you deploy the full configuration. When the event collection service must restart, QRadar does not restart it automatically. A message displays that gives you the option to cancel the deployment and restart the service at a more convenient time.

Configuring the password policy

You can configure the password policy settings that apply to local user accounts. When the policy is updated, users are prompted to change their password if they log in with a password that does not meet the new requirements.

About this task

Users should be encouraged to follow these guidelines when setting their password:

- Use a password that is significantly different from previous passwords.
Do not append a symbol or character to a previously used password because this change is not sufficiently different.
- Use a minimum of 15 characters. The minimum length in QRadar may be configured between 15 and 255 characters.
- Do not use complete words that are listed in a dictionary.
- Use a mixture of uppercase letters, lowercase letters, digits, and symbols.
- Do not use personal information that is known about you, for example, pets names, your name, kids names, or any information that is available in the public domain.

Procedure

1. Click the **Admin** tab.
2. On the navigation menu, click **System Configuration > User Management > Authentication**.
3. Click **Local Password Policy Configuration**.
4. Specify the password policy configuration that you want to enforce.
5. Click **Update Password Policy**.
6. On the **Admin** tab menu, select **Advanced > Deploy Full Configuration**.

Configuring the inactivity time period

Configure the number of minutes of inactivity after which IBM QRadar logs the user out and terminates the session.

Procedure

1. On the navigation menu (☰), click **Admin**.

2. In the **User Management** section, click **Authentication**.
3. Click **General Authentication settings**.
4. In the **Inactivity Timeout (in minutes)** field, specify the number of minutes of inactivity before QRadar logs the user out.
5. Click **Save Settings**.
6. On the **Admin** tab, select **Advanced > Deploy Full Configuration**.

Note: QRadar continues to collect events when you deploy the full configuration. When the event collection service must restart, QRadar does not restart it automatically. A message displays that gives you the option to cancel the deployment and restart the service at a more convenient time.

Configuring the local session timeout period

You can configure the TMOU environment variable to automatically log users out after a specified period of inactivity.

Procedure

1. Use SSH to log in to your QRadar Console.
2. Type `sed -i s/TMOU=.* /TMOU=XX/ /etc/profile` to edit the configuration file, replace XX with the number of seconds to wait before automatic logout.

Accessing QRadar RESTful API

Use the representational state transfer (REST) application programming interface (API) to make HTTPS queries and integrate IBM QRadar with other solutions. REST API communication is protected by using TLS via the Apache server, where each request is authenticated individually.

The QRadar API doc page can be used to call most QRadar APIs, but you cannot use it to upload files such as the root CA certificate. You must use a third-party API tool, such as Insomnia, to do import the certificates.

Procedure

1. Enter the following URL in your web browser to access the technical documentation interface:
`https://QRadar_All_in_one_IPaddress/api_doc`
`https://<QRadar_IP_address>/api_doc`
2. Click the header for the API that you want to access; for example, /ariel.
3. Click the subhead for the endpoint that you want to access, for example, /databases.
4. Click **Try it out** to receive properly formatted HTTPS responses.
5. Review and gather the information that you need to implement in your third-party solution.

x509 Certificate Management and Validation in QRadar

The Certificate Management App for QRadar, described in Chapter 5, provides a user interface for the certificate management API that is part of the RESTful API (above). It provides for importing and removing Certificate Authority (CA) root, or anchor, certificates as well as endpoint certificates and their

chains. During import, verification is performed to confirm the certificate's compliance with QRadar's expectation for validity, usage and revocation status.

Importing a CA root/anchor certificate

1. Log in to the QRadar Console with an admin account
2. Navigate to the admin tab.
3. Select the "QRadar Certificate Management" app under the header "QRadar Certificate Management".
4. Navigate to the "Root Certificates" tab
5. Select "Upload" in the top right corner
6. Drag and drop the root certificate to be uploaded into the "Upload new root certificate" popup window
7. Select "Confirm"
8. Select "Submit"

Upon submit, QRadar will confirm:

- That the Validity dates are valid for the current time
- That the KeyUsage (KU) extension includes Certificate Signing and CRL Signing uses
- That the Basic Constraints extension specifies CA:TRUE

Failure of any of the above tests will prevent the import of the certificate to the system trust. Upon success, the certificate will be added to QRadar's system truststore allowing it to be used as a root of trust for any TLS trusted communication.

Deleting a CA root/anchor certificate

1. Log in to the QRadar Console with an admin account
2. Navigate to the admin tab.
3. Select the "QRadar Certificate Management" app under the header "QRadar Certificate Management".
4. Navigate to the "Root Certificates" tab
5. Check the box next to the certificate to be deleted
6. Click "delete" on the upper right of the page
7. Select "Delete" in the dialog to confirm.

The certificate will be removed from the system truststore and will no longer serve as a root of trust for TLS certificate validation.

If the root certificate is referenced by any currently imported endpoint certificates, deletion will be prevented.

NOTE: Deletion of endpoint certificates is performed using the instructions above for deleting a CA root/anchor certificate, however, the admin would Navigate to the "Client/Server Certificates" tab in step 4 above and follow all other steps as stated.

Importing an endpoint certificate

1. Log in to the QRadar Console with an admin account
2. Navigate to the admin tab.
3. Select the “QRadar Certificate Management” app under the header “QRadar Certificate Management”.
4. In the “IBM QRadar Certificate Management” window, navigate to the “Client/Server Certificates” tab
5. Select “Upload” in the top right corner of the window
6. Input a name for the client certificate into the text box under the “Name **” label
7. Select “Client”, “Server” or “Server Client” in the dropdown list under the “Purpose **” label to indicate the intended use of the certificate.
8. Select an appropriate component in the dropdown list under the “Component” label based on the functional area where the certificate will be used.
9. Drag and drop the client key into the space labelled “Drag and drop the key file here or upload **”
10. Select “Confirm”
11. Drag and drop the client certificate into the space labelled “Drag and drop the certificate file here or upload **”
12. Select “Confirm”
13. Drag and drop the intermediate file into the space labelled “Drag and drop the intermediate file here or upload **”
14. Select “Confirm”
15. Click “Submit” (Note: The space labelled “Drag and drop the root-certificate here or upload” can be left without a file as long as the the root certificate was imported as per the procedure under “Importing a CA root/anchor certificate”)
16. Close the “IBM QRadar Certificate Management” popup window
17. Navigate to the admin tab on the main console
18. Select “Deploy Changes” in the upper toolbar

Upon submit, QRadar will perform multiple verification steps to assess the validity of the certificate for its intended use:

- That the Validity dates are valid for the current time
- That the KeyUsage (KU) extension includes
 - Digital Signature, Key Encipherment for certificates intended for server use
 - Digital Signature for certificates intended for client use
 - Other uses are permitted in addition to the above
- That the Extended Key Usage (EKU), if present, includes
 - TLS Web Server Authentication is required for certificates intended for server use
 - TLS Web Client Authentication is required for certificates intended for client use

- Other uses are permitted in addition to the above
- EKU is not required but will be tested if present
- That the Basic Constraints extension specifies CA:FALSE
- If the CRL Distribution Points (CRLDP) extension is present, revocation status will be checked
 - If the certificate has been revoked, the certificate will be rejected (not imported)
 - If none of the CRLDP URL(s) are reachable by QRadar, the certificate will be rejected
- The certificate signature will be verified by examining the intermediate certificates in order
 - The certificate signature will be verified based on the issuers public key
 - The issuer certificate will be checked for correct key usage
 - KU includes Certificate Sign, CRL Sign
 - Basic Constraints is CA:TRUE
 - If the Issuer certificate includes the CRLDP extension, revocation status will be checked with the same constraints as for the endpoint certificate.
 - Each certificate in the chain will be validated in this manner up to the root.
 - The root certificate must be in the system truststore and must meet the same constraints as laid out in “Importing a CA root/anchor certificate”
 - Root certificates will not be checked for revocation

Certificate validation during establishment of a trusted TLS channel

When QRadar acts as either a TLS Server or Client, it will validate the peer certificate using the same validation procedure as in “Importing an endpoint certificate” with the addition of the following verification tests:

- If QRadar is the TLS client (as in Chapter 6), the peer (server) certificate will be authenticated based upon the hostname configured in the TLS client. The certificate will be accepted if:
 - The Subject Alternative Name (SAN) is present, then the requested hostname must be present in the list of alternative names. The Common Name (CN) field of the Subject will not be considered.
 - If the SAN is not present, then the CN must match the requested hostname.
- If QRadar is the TLS server and it requires the client peer to provide a certificate for authentication, the certificate will optionally (see configuration in Chapter 5) examine the CN and/or the Issuer.
 - If the CN is in the CN Allow List configured in the server, the certificate will be accepted. This is a string comparison.
 - If the Issuer of the peer certificate matches the Issuer PEM configured in the server, the certificate will be accepted. This is an addition signature verification.
- The above two tests are independent, if both are present, both must pass.

Note: The TOE does not allow IP addresses to be configured as the identifiers associated with a TLS connection

QRadar self-test

QRadar runs cryptographic "known answer" self-tests during startup.

With exception of the Digital Signature Algorithm (DSA), each cryptographic algorithm provided by the Linux Kernel Crypto API is tested using a "known answer" test to verify the correct operation of the algorithm.

A failure of any of the power-up self-tests panics the module. The only recovery is to reboot. If a reboot cannot solve the problem, the system may be compromised, and QRadar must be reinstalled on the system.

Additionally, the TOE performs TOE software integrity checks using the Advanced Intrusion Detection Environment (AIDE) tool. The AIDE tool compares the current TOE software to a known good baseline. The AIDE tool generates a report indicating the status of TOE software each time it executes, this report indicates if a TOE software integrity check occurred.

Verifying secure updates

Applying updates is an installation task when performed as part of the initial system setup. An operational system must be taken offline during system update and returned to operational state afterward.

Administrators need to manually apply QRadar secure updates when they are released. Secure updates are signed by QRadar Security Keys.

If the signature verification fails, or initialization of the update installer fails for any reason and error message is presented and the update terminates. TOE version will not be updated.

Procedure

1. Download the QRadar V7.5.0 Update Pack 4 and the accompanying signature file from <https://www-945.ibm.com/support/fixcentral>. If the update pack is accompanied by patch-installer.zip then download that as well. (if patch-installer.zip is not present, follow the guidance here: <https://ibm.biz/qradarcodesigning>)

2. Copy the update file to the system by typing the following command:

```
scp 750_QRadAr_UpdatePackage_2021.* patch-installer.zip  
stiguser@qradar_IP:/home/ stiguser
```

3. Type the following command, that uses the file name of the update that you want to install:

```
unzip patch-installer.zip  
sudo sh ./patch-installer.sh 750_QRadAr_UpdatePackage_2021.*.sfs
```

4. This command verifies the update's signature and signing certificate. If successful, it then launches the update.

5. Type the following command to verify the QRadar patch version:

```
sudo /opt/qradar/bin/myver -v
```

Viewing the audit logs

Changes that are made by IBM QRadar users are recorded in the audit logs. You can use Secure Shell (SSH) to review the audit logs and monitor changes to your system.

About this task

All audit logs are stored in plain text and are archived and compressed when the audit log file reaches 200 MB. The current log file is named `audit.log`. When the file reaches 200 MB, the file is compressed and renamed to `audit.log.1.gz`. The file number increments each time that a log file is archived. QRadar retains up to 50 archived log files.

The Target of Evaluation (TOE) system displays the following warning message before the local storage space for the audit log is full:

```
Audit log rotation event.
```

```
The audit.log file has reached maximum capacity and will be overwritten.
```

Procedure

1. Using SSH, log in to QRadar as the root user:

- **User Name:** root
- **Password:** *password*

2. Review the following audit logs:

- `/var/log/audit/audit.log`
- `/var/log/qradar.log`
- `/var/log/secure`
- `/var/log/qradar.error`
- `/var/log/<patch_reference>/patches.log`

Sample Audits

Failure to establish a HTTPS session

```
<YYYY Mmm DD HH:MM:SS> ::ffff:127.0.0.1 admin@192.0.2.1 (1434) /console/login | [Authentication] [User] [LoginFailed] Local authentication failed. UserName = admin
```

Failure to establish an SSH session

```
<YYYY Mmm DD HH:MM:SS> qradar-niap sshd[30900]: Failed password for root from 192.0.2.1 port 54449 ssh2
```

```
<YYYY Mmm DD HH:MM:SS> qradar-niap sshd[30667]: Unable to negotiate with 192.0.2.1 port 47888: no matching cipher found. Their offer: aes256-cbc [preauth]
```

```
<YYYY Mmm DD HH:MM:SS> QRadar sshd[25912]: Unable to negotiate with 192.0.2.1 port 31296: no matching host key type found. Their offer: ssh-dss [preauth]
```

```
<YYYY Mmm DD HH:MM:SS> QRadar sshd[26341]: Unable to negotiate with 192.0.2.1 port 41963: no matching MAC found. Their offer: hmac-sha2-512,hmac-sha2-256
```

```
<YYYY Mmm DD HH:MM:SS> QRadar sshd[163584]: Unable to negotiate with 192.0.2.1 port 43691: no matching key exchange method found. Their offer: diffie-hellman-group14-sha1,diffie-hellman-group-exchange-sha256,diffie-hellman-group-exchange-sha1,diffie-hellman-group1-sha1,rsa2048-sha256,rsa1024-sha1 [preauth]
```

<YYYY Mmm DD HH:MM:SS> QRadar sshd[26972]: channel 0: rcvd big packet 262127, maxpack 32768

<YYYY Mmm DD HH:MM:SS> QRadar sshd[13968]: debug1: SSH2_MSG_KEXINIT sent [preauth]

Failure to establish a TLS session (client)

<YYYY Mmm DD HH:MM:SS> ::ffff:127.0.0.1 [ecs-ec.ecs-ec] [SFCT_140]
com.q1labs.sem.selectiveforwarding.SelectiveForwardingCommunicatorThread: [WARN]
[NOT:0000004000][192.0.2.2/- -] [-/- -]2023-04-12 14:55:03.0464 [qradar-niap.test.local:6514]
Exceeded maximum number of retries, dropping event[1].

<YYYY Mmm DD HH:MM:SS> ::ffff:192.0.2.2 [offline_forwarder.offline_forwarder]
[OfflineForwarderThread_1_1] com.ibm.si.ep.offlineforwarding.OfflineForwarderWorkerThread: [INFO]
[NOT:0000006000][192.0.2.2/- -] [-/- -]Lost connection to IP[192.0.2.2:9514]

<YYYY Mmm DD HH:MM:SS> ::ffff:127.0.0.1 OfflineForwarderThread_1_1 | [Q1X509TrustManager]
[Validation] [ValidationFailed] (offline_forwarder) Server Certificate Validation failed.
chain:[0]X509Certificate : { SubjectDN : CN=qradar-niap.test.local, OU=Crash Test Dummy Division,
O=Example Corp, L=Fredericton, ST=New Brunswick, C=CA, IssuerDN : CN=Crash Test Intermediate CA,
OU=Crash Test Dummy Division, O=Example Corp, C=CA},
exception:com.q1labs.frameworks.crypto.trustmanager.exceptions.Q1CertificatePathValidatorExceptio
n: PKIX certificate chain validation failed because Path does not chain with any of the trust anchors

<YYYY Mmm DD HH:MM:SS> ::ffff:127.0.0.1 OfflineForwarderThread_1_1 | [Q1X509TrustManager]
[Validation] [ValidationFailed] (offline_forwarder) Server Certificate Validation failed.
chain:[0]X509Certificate : { SubjectDN : CN=qradar-niap.test.local, OU=Crash Test Dummy Division,
O=Example Corp, L=Fredericton, ST=New Brunswick, C=CA, IssuerDN : CN=Crash Test Intermediate CA,
OU=Crash Test Dummy Division, O=Example Corp, C=CA},[1]X509Certificate : { SubjectDN : CN=Crash
Test Intermediate CA, OU=Crash Test Dummy Division, O=Example Corp, C=CA, IssuerDN : CN=Crash
Test Root CA, OU=Crash Test Dummy Division, O=Example Corp, C=CA},
exception:java.security.cert.CertificateException: No subject alternative names matching IP address
192.0.2.2 found

<YYYY Mmm DD HH:MM:SS> 127.0.0.1 SFCT_116 | [Q1X509TrustManager] [Validation]
[ValidationFailed] (ecs-ec) Server Certificate Validation failed. chain:[0]X509Certificate : { SubjectDN :
CN=qradar-niap.test.local, OU=Crash Test Dummy Division, O=Example Corp, L=Fredericton, ST=New
Brunswick, C=CA, IssuerDN : CN=Crash Test Intermediate CA, OU=Crash Test Dummy Division,
O=Example Corp, C=CA},[1]X509Certificate : { SubjectDN : CN=Crash Test Intermediate CA, OU=Crash
Test Dummy Division, O=Example Corp, C=CA, IssuerDN : CN=Crash Test Root CA, OU=Crash Test
Dummy Division, O=Example Corp, C=CA}, exception:com.ibm.jsse2.util.j: Extended key usage does not
permit use for TLS server authentication

<YYYY Mmm DD HH:MM:SS> ::ffff:192.0.2.2 [offline_forwarder.offline_forwarder]
[OfflineForwarderThread_1_1] com.q1labs.sem.forwarding.network.ForwardingTCPOverSSLConnector:
[WARN] [NOT:0000004000][192.0.2.2/- -] [-/- -][qradar-niap.test.local:192.0.2.2:9514] Unable to
connect over SSL

<YYYY Mmm DD HH:MM:SS>::ffff:127.0.0.1 [ecs-ec.ecs-ec] [SFCT_140]
javax.net.ssl.SSLHandshakeException: Received fatal alert: handshake_failure

Failure to establish a TLS session (server)

<YYYY Mmm DD HH:MM:SS> ::ffff:192.0.2.2 [ecs-ec-ingress.ecs-ec-ingress] [Thread-293825]
com.q1labs.semsources.sources.tlssyslog.TLSConnectionProcessor: [ERROR]
[NOT:0000003000][192.0.2.2/- -] [-/- -]An IOException occurred during SSL Socket Handshake with
/0:0:0:0:0:0:1:37672 Closing socket.

<YYYY Mmm DD HH:MM:SS> ::ffff:192.0.2.2 [ecs-ec-ingress.ecs-ec-ingress] [Thread-294398]
javax.net.ssl.SSLHandshakeException: Empty server certificate chain

<YYYY Mmm DD HH:MM:SS> ::ffff:127.0.0.1 Thread-294719 | [Q1X509TrustManager] [Validation]
[ValidationFailed] (ecs-ec-ingress) Client Certificate Validation failed. chain:[0]X509Certificate : {
SubjectDN : CN=qradar-niap.test.local, OU=Crash Test Dummy Division, O=Example Corp,
L=Fredericton, ST=New Brunswick, C=CA, IssuerDN : CN=Crash Test Intermediate CA, OU=Crash Test
Dummy Division, O=Example Corp, C=CA},
exception:com.q1labs.frameworks.crypto.trustmanager.exceptions.Q1CertificatePathValidatorExceptio
n: PKIX certificate chain validation failed because Path does not chain with any of the trust anchors

<YYYY Mmm DD HH:MM:SS> ::ffff:192.0.2.2 [ecs-ec-ingress.ecs-ec-ingress] [Thread-304]
com.q1labs.semsources.sources.tlssyslog.TLSConnectionProcessor: [ERROR]
[NOT:0000003000][192.0.2.2/- -] [-/- -]Client authentication failed. The client certificate's common
name CN=[qradar-niap.test.local] does not match any patterns in the provided CN Allowlist: [localhost]

Login failure limit reached

<YYYY Mmm DD HH:MM:SS> qradar-niap sshd[29326]: pam_faillock(sshd:auth): Consecutive login
failures for user stiguser account temporarily locked

All use of identification mechanisms (UI)

<YYYY Mmm DD HH:MM:SS> ::ffff:127.0.0.1 admin@192.0.2.1 (2366) /console/login | [Authentication]
[User] [LoginAttempt] Local authentication successful. UserName = admin

<YYYY Mmm DD HH:MM:SS> ::ffff:127.0.0.1 admin@192.0.2.1 (2366) /console/login | [Authentication]
[Session] [AdminSessionCreated] UserName=admin

<YYYY Mmm DD HH:MM:SS> ::ffff:127.0.0.1 admin@192.0.2.1 (2366) /console/login | [Authentication]
[User] [UserLogin] admin

<YYYY Mmm DD HH:MM:SS> ::ffff:127.0.0.1 admin@192.0.2.1 (1434) /console/login | [Authentication]
[User] [LoginFailed] Local authentication failed. UserName = admin

<YYYY Mmm DD HH:MM:SS> ::ffff:127.0.0.1 admin@192.0.2.1 (1404) /console/JSON-
RPC/QRadar.logout QRadar.logout | [Authentication] [Session] [AdminSessionDestroyed]
UserName=admin

<YYYY Mmm DD HH:MM:SS> ::ffff:127.0.0.1 admin@192.0.2.1 (1404) /console/JSON-
RPC/QRadar.logout QRadar.logout | [Authentication] [User] [UserLogout] admin

All use of identification mechanism (CLI)

<YYYY Mmm DD HH:MM:SS> qradar-niap sshd[14909]: Accepted password for root from 192.0.2.1 port 55448 ssh2

<YYYY Mmm DD HH:MM:SS> qradar-niap sshd[30073]: Accepted publickey for root from 192.0.2.2 port 51944 ssh2: RSA SHA256:oZAP7HaWQRN/CBcgU+M20dPwFLP79oLLjCTcl+pNdrQ

<YYYY Mmm DD HH:MM:SS> qradar-niap sshd[30073]: pam_unix(sshd:session): session opened for user root by (uid=0)

<YYYY Mmm DD HH:MM:SS> qradar-niap sshd[30073]: Received disconnect from 192.0.2.2 port 51944:11: disconnected by user

<YYYY Mmm DD HH:MM:SS> qradar-niap sshd[14909]: pam_unix(sshd:session): session closed for user root

<YYYY Mmm DD HH:MM:SS> qradar-niap sshd[30900]: Failed password for root from 192.0.2.1 port 54449 ssh2

<YYYY Mmm DD HH:MM:SS> qradar-niap sshd[14909]: Timeout, client not responding.

Unsuccessful certificate validation attempt

<YYYY Mmm DD HH:MM:SS> ::ffff:127.0.0.1 OfflineForwarderThread_1_1 | [Q1X509TrustManager] [Validation] [ValidationFailed] (offline_forwarder) Server Certificate Validation failed. chain:[0]X509Certificate : { SubjectDN : CN=qradar-niap.test.local, OU=Crash Test Dummy Division, O=Example Corp, L=Fredericton, ST=New Brunswick, C=CA, IssuerDN : CN=Crash Test Intermediate CA, OU=Crash Test Dummy Division, O=Example Corp, C=CA}, exception:com.q1labs.frameworks.crypto.trustmanager.exceptions.Q1CertificatePathValidatorException: PKIX certificate chain validation failed because Path does not chain with any of the trust anchors

<YYYY Mmm DD HH:MM:SS> ::ffff:127.0.0.1 admin@169.254.3.7 (2719) /console/restapi/api/staged_config/certificates/end_certificates | [Q1X509TrustStore] [Config] [CertificateAdded] (tomcat) Added X.509 Trusted Certificates to whitelist: subject=55:F7:6C:EC:65:67:8D:97:8B:E7:36:26:88:B4:3C:47:A7:98:90:BB

<YYYY Mmm DD HH:MM:SS> ::ffff:127.0.0.1 [tomcat.tomcat] [Trust Manager Watch Service] com.q1labs.frameworks.crypto.trustmanager.Q1X509TrustStore: [INFO] [NOT:0000006000][169.254.3.7/- -] [-/- -]Audit logging msg:(tomcat) Added X.509 Trusted Certificates to whitelist: subject= QRadar.crt

<YYYY Mmm DD HH:MM:SS> 127.0.0.1 SFCT_8317 | [Q1X509TrustManager] [Validation] [ValidationFailed] (ecs-ec) Server Certificate Validation failed. chain:[0]X509Certificate : { SubjectDN : CN=qradar-niap.test.local, OU=Crash Test Dummy Division, O=Example Corp, L=Fredericton, ST=New Brunswick, C=CA, IssuerDN : CN=Crash Test Intermediate CA, OU=Crash Test Dummy Division, O=Example Corp, C=CA}, exception:com.q1labs.frameworks.crypto.trustmanager.exceptions.Q1CertificateInvalidUsageException: Invalid Certificate Usage

<YYYY Mmm DD HH:MM:SS> 127.0.0.1 SFCT_8317 | [Q1X509TrustManager] [Validation] [ValidationFailed] (ecs-ec) Validating certificate chain failed. chain:[0]X509Certificate : { SubjectDN : CN=qradar-niap.test.local, OU=Crash Test Dummy Division, O=Example Corp, L=Fredericton, ST=New Brunswick, C=CA, IssuerDN : CN=Crash Test Intermediate CA, OU=Crash Test Dummy Division, O=Example Corp, C=CA}, params:CertValidatorParameters [enableLegacySupport :false,checkPinning :true,checkRevocation :true,checkSelfsigned :true,checkUsage :true,checkCaIssuersInAuthInfoAccess :false,trustStores :/etc/pki/ca-trust/extracted/java/cacerts,/opt/ibm/si/services/ecs-ec/current/frameworks_conf//cached_crls,], exception:com.q1labs.frameworks.crypto.trustmanager.exceptions.Q1CertificateInvalidUsageException: Invalid Certificate Usage

<YYYY Mmm DD HH:MM:SS> ::ffff:127.0.0.1 [ecs-ec.ecs-ec] [SFCT_140] com.q1labs.frameworks.crypto.trustmanager.Q1X509TrustManager: [INFO] [NOT:0000006000][169.254.3.7/- -] [-/- -]Audit logging msg:(ecs-ec) Server Certificate Validation failed. chain:[0]X509Certificate : { SubjectDN : CN=qradar-niap.test.local, OU=Crash Test Dummy Division, O=Example Corp, L=Fredericton, ST=New Brunswick, C=CA, IssuerDN : CN=Crash Test Intermediate CA, OU=Crash Test Dummy Division, O=Example Corp, C=CA}, exception:com.q1labs.frameworks.crypto.trustmanager.exceptions.Q1CertificatePathValidatorException: PKIX certificate chain validation failed because signature check failed

<YYYY Mmm DD HH:MM:SS> ::ffff:127.0.0.1 [ecs-ec.ecs-ec] [SFCT_140] com.q1labs.frameworks.crypto.trustmanager.Q1X509TrustManager: [INFO] [NOT:0000006000][169.254.3.7/- -] [-/- -]Audit logging msg:(ecs-ec) Server Certificate Validation failed. chain:[0]X509Certificate : { SubjectDN : CN=qradar-niap.test.local, OU=Crash Test Dummy Division, O=Example Corp, L=Fredericton, ST=New Brunswick, C=CA, IssuerDN : CN=Crash Test Intermediate CA, OU=Crash Test Dummy Division, O=Example Corp, C=CA}, exception:com.q1labs.frameworks.crypto.trustmanager.exceptions.Q1CertificatePathValidatorException: PKIX certificate chain validation failed because Certificate has been revoked, reason: UNSPECIFIED, revocation date: Tue Aug 11 13:04:48 EDT 2020, authority: CN=qradar-niap.test.local, O=Example Corp, L=Fredericton, ST=New Brunswick, C=CA, extension OIDs: []

<YYYY Mmm DD HH:MM:SS> 127.0.0.1 [SFCT_8317] com.q1labs.frameworks.crypto.trustmanager.Q1X509TrustManager: [INFO] [NOT:0000006000][169.254.3.7/- -] [-/- -]Audit logging msg:(ecs-ec) Server Certificate Validation failed. chain:[0]X509Certificate : { SubjectDN : CN=qradar-niap.test.local, OU=Crash Test Dummy Division, O=Example Corp, L=Fredericton, ST=New Brunswick, C=CA, IssuerDN : CN=Crash Test Intermediate CA, OU=Crash Test Dummy Division, O=Example Corp, C=CA}, exception:java.security.cert.CertificateExpiredException: NotAfter: Tue Aug 11 13:09:00 EDT 2020

<YYYY Mmm DD HH:MM:SS> ::ffff:127.0.0.1 admin@169.254.3.7 (3025) /console/restapi/api/staged_config/certificates/root_certificates/55F76CEC65678D978BE7362688B43C47A79890BB | [Action] [RestAPI] [APISuccess] [admin] [e32a18e8-fccf-43c6-bf01-6038522f1227] [SECURE] | ContextPath=/console | Headers=[host: qradar-niap.test.local] | Method=DELETE | PathInfo=/staged_config/certificates/root_certificates/55F76CEC65678D978BE7362688B43C47A79890BB | Protocol=HTTP/1.1 | QueryString=null | RemoteAddr=169.254.3.7 | RemotePort=54382

Any Attempt to Initiate a manual update

<YYYY Mmm DD HH:MM:SS> console01 127.0.0.1 root@192.0.2.1 53730 22 | [Patch] [Signature,] [VerifySucceeded,] : Signature verify succeeded, valid signature from an IBM signer
Mon Feb 6 14:36:05 AST 2023: Running "/media/cdrom/installer --exec" from /media/cdrom using Patch /root/750_QRadar_UpdatePackage_2021.6.4.20221129155237.sfs
Wed Feb 22 17:07:00 AST 2023: Upgraded QRadar to version 2021.6.4.20221129155237 from version 2021.6.3.20220829221022

<YYYY Mmm DD HH:MM:SS> console01 127.0.0.1 root@192.0.2.1 53730 22 | [Patch] [Signature,] [VerifyFailed,] : Signing certificate verify failed, untrusted certificate

<YYYY Mmm DD HH:MM:SS> console01 127.0.0.1 root@192.0.2.1 53730 22 | [Patch] [Signature,] [VerifyFailed,] : Sig7nature verify failed, bad signature

<YYYY Mmm DD HH:MM:SS> console01 127.0.0.1 root@192.0.2.1 53730 22 | [Patch] [Signature,] [VerifyFailed,] : Signature verify failed, signed by an unknown entity

<YYYY Mmm DD HH:MM:SS> console01 127.0.0.1 root@192.0.2.1 53730 22 | [Patch] [Installer,] [VerifyFailed,] : Failed to install patch, mount failure

Management activity

<YYYY Mmm DD HH:MM:SS> qradar-niap 127.0.0.1 root@192.0.2.1 54433 22 | [Backend] [Command] [CommandExecuted] : systemctl restart ecs-ec-ingress

<YYYY Mmm DD HH:MM:SS> qradar-niap 127.0.0.1 root@ | [Backend] [Command] [CommandExecuted] : /opt/qradar/bin/myver -v

Discontinuous changes to time

<YYYY Mmm DD HH:MM:SS> ::ffff:127.0.0.1 pool-2-thread-3 | [Action] [SystemServerSettings] [SetSystemTimeSettings] Changed the system time and timezone settings on server 51 (new settings="{\"timezoneId\":\"America/Moncton\",\"currentTime\":\"1682446980000\",\"currentDate\":\"2023-04-25 15:23:00\",\"isSyncWithNtpServer\":false}”, initiating-user=“admin”)

Initiation of update

<YYYY Mmm DD HH:MM:SS> console01 127.0.0.1 root@192.0.2.1 53730 22 | [Patch] [Signature,] [VerifySucceeded,] : Signature verify succeeded, valid signature from an IBM signer
Mon Feb 6 14:36:05 AST 2023: Running "/media/cdrom/installer --exec" from /media/cdrom using Patch /root/750_QRadar_UpdatePackage_2021.6.4.20221129155237.sfs
Wed Feb 22 17:07:00 AST 2023: Upgraded QRadar to version 2021.6.4.20221129155237 from version 2021.6.3.20220829221022

<YYYY Mmm DD HH:MM:SS> console01 127.0.0.1 root@192.0.2.1 53730 22 | [Patch] [Signature,] [VerifyFailed,] : Signing certificate verify failed, untrusted certificate

<YYYY Mmm DD HH:MM:SS> console01 127.0.0.1 root@192.0.2.1 53730 22 | [Patch] [Signature,] [VerifyFailed,] : Signature verify failed, bad signature

<YYYY Mmm DD HH:MM:SS> console01 127.0.0.1 root@192.0.2.1 53730 22 | [Patch] [Signature,] [VerifyFailed,] : Signature verify failed, signed by an unknown entity

<YYYY Mmm DD HH:MM:SS> console01 127.0.0.1 root@192.0.2.1 53730 22 | [Patch] [Installer,] [VerifyFailed,] : Failed to install patch, mount failure

Trusted Path Initiation, failure

<YYYY Mmm DD HH:MM:SS> ::ffff:127.0.0.1 [ecs-ec.ecs-ec] [SFCT_140] com.q1labs.sem.selectiveforwarding.SelectiveForwardingCommunicatorThread: [INFO] [NOT:0000006000][192.0.2.2/- -] [-/- -][qradar-niap.test.local:6514] Established connection

<YYYY Mmm DD HH:MM:SS> ::ffff:127.0.0.1 [ecs-ec.ecs-ec] [SFPT] com.q1labs.sem.selectiveforwarding.SelectiveForwardingCommunicatorThread: [WARN] [NOT:0000004000][192.0.2.2/- -] [-/- -][qradar-niap.test.local:6514] has been disabled. Flushing queue, dropped[391] event(s).

<YYYY Mmm DD HH:MM:SS> ::ffff:127.0.0.1 admin@192.0.2.1 (2366) /console/login | [Authentication] [User] [LoginAttempt] Local authentication successful. UserName = admin

<YYYY Mmm DD HH:MM:SS> ::ffff:127.0.0.1 admin@192.0.2.1 (2366) /console/login | [Authentication] [Session] [AdminSessionCreated] UserName=admin

<YYYY Mmm DD HH:MM:SS> ::ffff:127.0.0.1 admin@192.0.2.1 (2366) /console/login | [Authentication] [User] [UserLogin] admin

<YYYY Mmm DD HH:MM:SS> ::ffff:127.0.0.1 admin@192.0.2.1 (1434) /console/login | [Authentication] [User] [LoginFailed] Local authentication failed. UserName = admin

<YYYY Mmm DD HH:MM:SS> ::ffff:127.0.0.1 admin@192.0.2.1 (1404) /console/JSON-RPC/QRadar.logout QRadar.logout | [Authentication] [Session] [AdminSessionDestroyed] UserName=admin

<YYYY Mmm DD HH:MM:SS> ::ffff:127.0.0.1 admin@192.0.2.1 (1404) /console/JSON-RPC/QRadar.logout QRadar.logout | [Authentication] [User] [UserLogout] admin

<YYYY Mmm DD HH:MM:SS> qradar-niap sshd[14909]: Accepted password for root from 192.0.2.1 port 55448 ssh2

<YYYY Mmm DD HH:MM:SS> qradar-niap sshd[30073]: Accepted publickey for root from 192.0.2.2 port 51944 ssh2: RSA SHA256:oZAP7HaWQRN/CBcgU+M20dPwFLP79oLLjCTcl+pNdrQ
Apr 24 14:58:25 qradar-niap sshd[30073]: pam_unix(sshd:session): session opened for user root by (uid=0)

<YYYY Mmm DD HH:MM:SS> qradar-niap sshd[30073]: Received disconnect from 192.0.2.2 port 51944:11: disconnected by user

<YYYY Mmm DD HH:MM:SS> qradar-niap sshd[14909]: pam_unix(sshd:session): session closed for user root

<YYYY Mmm DD HH:MM:SS> qradar-niap sshd[30900]: Failed password for root from 192.0.2.1 port 54449 ssh2

Admin Actions

```
<YYYY Mmm DD HH:MM:SS> ::ffff:127.0.0.1 admin@192.0.2.1 (1150) /console/JSON-RPC/QRadar.updateForwardingDestination QRadar.updateForwardingDestination | [Configuration] [SelectiveForwardingDestination] [SelectiveForwardingDestinationAdded] ... ( Object Data="com.q1labs.core.dao.selectiveforwarding.light.SelectiveForwardingDestinationObject { ID: 23, Name: qradar-syslog-forwarding, IPAddress: 192.0.2.1, Port: 6514, Creation Time (ms): 1680888261229, Modified Time (ms): 1680888261229, Enabled: true, Event Protocol: TLS11ABOVE, Event Format: PAYLOAD, Profile ID: 0, Enable hostname verification: true, Enable mutual certificates authentication: true, Friendly name of the certificate for client authentication: 28", Id="23" )
```

```
<YYYY Mmm DD HH:MM:SS> ::ffff:127.0.0.1 admin@192.0.2.1 (4558) /console/restapi/api/system/authorization/settings | [Configuration] [AuthenticationConfiguration] [AuthenticationConfigurationUpdated] Inactivity Timeout has been enabled with a timeout of 600000
```

```
<YYYY Mmm DD HH:MM:SS> ::ffff:127.0.0.1 admin@192.0.2.1 (7083) /console/restapi/api/system/authorization/settings | [Configuration] [AuthenticationConfiguration] [AuthenticationConfigurationUpdated] Logon message has been enabled with a value of 'QRadar Test Banner'
```

```
<YYYY Mmm DD HH:MM:SS> ::ffff:127.0.0.1 admin@192.0.2.1 (4220) /console/restapi/api/system/authorization/password_policies/1 | [Action] [RestAPI] [APISuccess] [admin] [1d6020ae-d726-492e-8f50-e7501898a4bc] [SECURE] | ContextPath=/console | Headers=[host: 169.254.3.7][accept: application/json][user-agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/112.0.0.0 Safari/537.36] | Method=POST | PathInfo=/system/authorization/password_policies/1 | Protocol=HTTP/1.1 | QueryString=null | RemoteAddr=192.0.2.1 | RemotePort=52284
```

Resetting Passwords

```
<YYYY Mmm DD HH:MM:SS> ::ffff:127.0.0.1 admin@192.0.2.1 (2364) /console/restapi/api/staged_config/access/users/7 | [Authentication] [User] [PasswordChange] Updated password for admin
```

Stop/Start Auditing service

```
<YYYY Mmm DD HH:MM:SS> QRadar audispd: node=qradar-niap.test.local type=SERVICE_START msg=audit(1680535802.195:6): pid=1 uid=0 auid=4294967295 ses=4294967295 msg='unit=auditd comm="systemd" exe="/usr/lib/systemd/systemd" hostname=? addr=? terminal=? res=success'
```

```
<YYYY Mmm DD HH:MM:SS> QRadar audispd: node=qradar-niap.test.local type=SERVICE_STOP msg=audit(1682104484.165:3410): pid=1 uid=0 auid=4294967295 ses=4294967295 msg='unit=auditd comm="systemd" exe="/usr/lib/systemd/systemd" hostname=? addr=? terminal=? res=success'
```

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785 U.S.A.

For license inquiries regarding double-byte character set (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some jurisdictions do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you provide in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Director of Licensing
IBM Corporation
North Castle Drive, MD-NC119

Armonk, NY 10504-1785
US

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

The performance data and client examples cited are presented for illustrative purposes only. Actual performance results may vary depending on specific configurations and operating conditions..

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

Statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

All IBM prices shown are IBM's suggested retail prices, are current and are subject to change without notice. Dealer prices may vary.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to actual people or business enterprises is entirely coincidental.

Trademarks

IBM, the IBM logo, and ibm.com[®] are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at www.ibm.com/legal/copytrade.shtml.

Terms and conditions for product documentation

Permissions for the use of these publications are granted subject to the following terms and conditions.

Applicability

These terms and conditions are in addition to any terms of use for the IBM website.

Personal use

You may reproduce these publications for your personal, noncommercial use provided that all proprietary notices are preserved. You may not distribute, display or make derivative work of these publications, or any portion thereof, without the express consent of IBM.

Commercial use

You may reproduce, distribute and display these publications solely within your enterprise provided that all proprietary notices are preserved. You may not make derivative works of these publications, or reproduce, distribute or display these publications or any portion thereof outside your enterprise, without the express consent of IBM.

Rights

Except as expressly granted in this permission, no other permissions, licenses or rights are granted, either express or implied, to the publications or any information, data, software or other intellectual property contained therein.

IBM reserves the right to withdraw the permissions granted herein whenever, in its discretion, the use of the publications is detrimental to its interest or, as determined by IBM, the above instructions are not being properly followed.

You may not download, export or re-export this information except in full compliance with all applicable laws and regulations, including all United States export laws and regulations.

IBM MAKES NO GUARANTEE ABOUT THE CONTENT OF THESE PUBLICATIONS. THE PUBLICATIONS ARE PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, NON- INFRINGEMENT, AND FITNESS FOR A PARTICULAR PURPOSE.

IBM Online Privacy Statement

IBM Software products, including software as a service solutions, ("Software Offerings") may use cookies or other technologies to collect product usage information, to help improve the end user experience, to tailor interactions with the end user or for other purposes. In many cases no personally identifiable information is collected by the Software Offerings. Some of our Software Offerings can help enable you to collect personally identifiable information. If this Software Offering uses cookies to collect personally identifiable information, specific information about this offering's use of cookies is set forth below.

Depending upon the configurations deployed, this Software Offering may use session cookies that collect each user's session id for purposes of session management and authentication. These cookies can be disabled, but disabling them will also eliminate the functionality they enable.

If the configurations deployed for this Software Offering provide you as customer the ability to collect personally identifiable information from end users via cookies and other technologies, you should seek your own legal advice about any laws applicable to such data collection, including any requirements for notice and consent.

For more information about the use of various technologies, including cookies, for these purposes, See IBM's Privacy Policy at <http://www.ibm.com/privacy> and IBM's Online Privacy Statement at <http://www.ibm.com/privacy/details> the section entitled "Cookies, Web Beacons and Other Technologies" and the "IBM Software Products and Software-as-a-Service Privacy Statement" at <http://www.ibm.com/software/info/product-privacy>.

General Data Protection Regulation

Clients are responsible for ensuring their own compliance with various laws and regulations, including the European Union General Data Protection Regulation. Clients are solely responsible for obtaining advice of competent legal counsel as to the identification and interpretation of any relevant laws and regulations that may affect the clients' business and any actions the clients may need to take to comply with such laws and regulations. The products, services, and other capabilities described herein are not suitable for all client situations and may have restricted availability. IBM does not provide legal, accounting or auditing advice or represent or warrant that its services or products will ensure that clients are in compliance with any law or regulation.

Learn more about the IBM GDPR readiness journey and our GDPR capabilities and Offerings here: <https://ibm.com/gdpr>

IBM®