IBM Disconnected Log Collector
1.7.0

IBM Disconnected Log Collector Guide

Note

Before you use this information and the product that it supports, read the information in “Notices” on page 29.

Contents

Chapter 1. Disconnected Log Collector

IBM® Disconnected Log Collector is free software that accepts events from a limited set of log sources and sends them to an IBM QRadar® deployment. Disconnected Log Collector is compatible with QRadar 7.3.1 or later.

What's new in Disconnected Log Collector

Stay up to date with the new features that are available in IBM Disconnected Log Collector.

1.8.5

Added TLS proxy authentication support

Use the proxy settings if your Disconnected Log Collector is behind a corporate firewall and the only way to access your Disconnected Log Collector is with a proxy server.

The proxy server also needs to access port 32500 on your QRadar instance.

For more information, see "Configuring TLS proxy communication with QRadar" on page 15.

Updated the installation procedure to use a script

The installation method changed to use a .tgz file so you can download and install connector RPMs for protocols directly on your Disconnected Log Collector instance as new ones or updates become available. With this update, you do not have to install a new Disconnected Log Collector version to receive protocol updates.

1.8.4

Fixed issues that caused service disruption

Issues with Java™ and RHEL 9.2 that caused security interruptions were fixed. If you have further issues, contact IBM Support.

Support for TLS 1.3

You can now use TLS 1.3 when you connect your Disconnected Log Collector to QRadar or QRadar on Cloud. Using the most current TLS version increases the security of the connection to your Disconnected Log Collector.

Important: To use TLS 1.3, you must have the most current version of the QRadar Disconnected Log Collector protocol on your QRadar or QRadar on Cloud instance.

Update the TLS version in both the `config.json` file for your Disconnected Log Collector instance, and in the QRadar Log Source Management app for the protocol.

Added backup and restore scripts

Use these scripts to backup and restore your Disconnected Log Collector configuration.

To back up your configuration, run the `/opt/ibm/si/services/dlc/current/script/configBackup.sh` script.

To restore your configuration, run the `/opt/ibm/si/services/dlc/current/script/configRestore.sh` script.

1.8.3

Improved security with protocol updates

The following protocols are updated to improve security compliance with this version of Disconnected Log Collector:

- Log File Protocol
- Microsoft Azure Event Hubs

For more information about the parameters for each log source, see the `readme` files that are provided with the product. The `readme` files are available in the `/opt/ibm/si/services/dlc/conf/template` directory.

1.8.2

Improved security with protocol updates

The following protocols are updated to improve security compliance with this version of Disconnected Log Collector:

- Akamai Kona REST API
- Amazon AWS REST API
- Amazon Web Services
- Apache Kafka
- Ariel REST API
- Blue Coat Web Security Service (WSS) REST API
- Box REST API
- Google G Suite Activity Reports REST API
- JDBC
- Log File Protocol
- Microsoft Azure Event Hubs
- Microsoft Graph Security API
- Microsoft Office 365 REST API
- Microsoft Office 365 Message Trace REST API
- Seculert Protection REST API
- SMB Tail
- SNMP
- Universal Cloud REST API
- Microsoft Defender® for Endpoint REST API

  Important: Due to a change in the Microsoft Defender API suite as of 25 November 2021, Microsoft no longer allows the onboarding of new integrations with their SIEM API.

  To continue to receive data from Microsoft Defender for Endpoint REST API log sources, you must register a new application and create Microsoft Graph Security API log sources to collect the data. For more information, see Migrating Microsoft Defender for Endpoint REST API log sources to Microsoft Graph Security API log sources.

For more information about the parameters for each log source, see the `readme` files that are provided with the product. The `readme` files are available in the `/opt/ibm/si/services/dlc/conf/template` directory.

1.7.0

Support for Disaster Recovery

You can now enable your Disconnected Log Collector device to run on a destination site if your primary site stops working due to a site failure. Disconnected Log Collector works with the IBM QRadar Data Synchronization app to ensure that you do not lose your data.

For more information, see "Disaster Recovery and Disconnected Log Collector" on page 27.

Improved security and accessibility with industry compliance

Disconnected Log Collector is compliant with the Federal Information Processing Standards (FIPS).

1.6.0

Generate requests for server certificate on QRadar

You can use the `generatecertificate.sh` script to generate requests for the server certificate that is used by the Disconnected Log Collector log source protocol on QRadar.

For more information, see "Setting up certificate-based authentication on QRadar" on page 12.

1.5.0

Monitor the health of Disconnected Log Collector

You can enable metrics collection to monitor the health of Disconnected Log Collector. Collect metrics on the event rate and spill file count for the pipeline to QRadar. Send the metrics to QRadar as events.

For more information, see "Sending Disconnected Log Collector health metrics to QRadar" on page 26.

Monitor client certificate expiry

You can monitor the expiry of the client certificate that Disconnected Log Collector uses for secure TLS communication to QRadar. Specify the number of days in advance of the expiry to send a notification event to QRadar.

For more information, see "Sending Disconnected Log Collector health metrics to QRadar" on page 26.

Support for more log source protocols

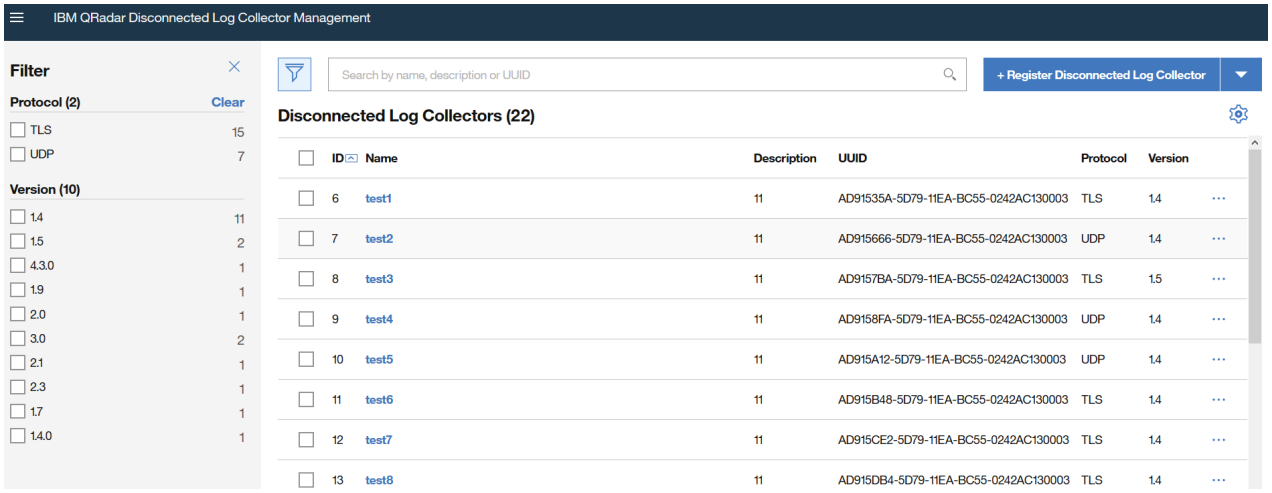
The following log source protocols were added:

- IBM Cloud® Identity Event Service
- Microsoft Graph Security API
- Microsoft Office 365 Message Trace REST API
- Universal Cloud REST API

For more information about the parameters for each log source, see the `readme` files that are provided with the product. The `readme` files are available in the `/opt/ibm/si/services/dlc/conf/template` directory.

1.4.0

In QRadar 7.4.0 or later, use the QRadar Log Source Management app (version 6.0 or later) to register or import Disconnected Log Collector instances that are installed in your environment. You can configure your log sources in the app, which is much faster than by using the Disconnected Log Collector's JSON config file.



In addition, the following log source protocols were added:

- Ariel REST API
- Box REST API
- Centrify Redrock REST API
- Google G Suite Activity Reports REST API
- Netskope Active REST API
- Okta REST API
- Seculert Protection REST API
- VMware vCloud Protocol
- Windows Defender ATP REST API

For more information about the parameters for each log source, see the `readme` files that are provided with the product. The `readme` files are available in the `/opt/ibm/si/services/dlc/conf/template` directory.

1.3.0

The following log source protocols were added:

- SAP ETD REST API
- ObserveIT JDBC
- IBM SIM JDBC
- Windows Security Event Log
- EMC VmWare Protocol

1.2.0

The following log source protocols were added:

- Akamai Kona REST API
- Amazon web Services
- Apache Kafka
- Blue Coat WSS REST API
- Cisco Firepower eStreamer
- Microsoft Azure Event Hubs
- Microsoft Office 365 REST API

- MQJMS
- Oracle Database Listener
- Salesforce REST API
- SMBTail
- SNMPv3
- Windows DHCP Protocol
- Microsoft Exchange Protocol

1.1.0

More protocols were added in Disconnected Log Collector 1.1.0. For a full list of supported protocols, see “Disconnected Log Collector overview” on page 5.

Disconnected Log Collector overview

IBM Disconnected Log Collector sends events to an IBM QRadar deployment by using the User Datagram Protocol (UDP) or the Transport Layer Security over the Transmission Control Protocol (TLS over TCP). When Disconnected Log Collector uses TLS over TCP, it buffers incoming events during times when it is disconnected from QRadar and sends them when the connection is restored. Buffer capacity can be configured, and is limited by the available memory and disk space.

You can use as many Disconnected Log Collector instances as you need in your QRadar environment.

The following image shows an example of Disconnected Log Collector that is deployed in a QRadar environment.

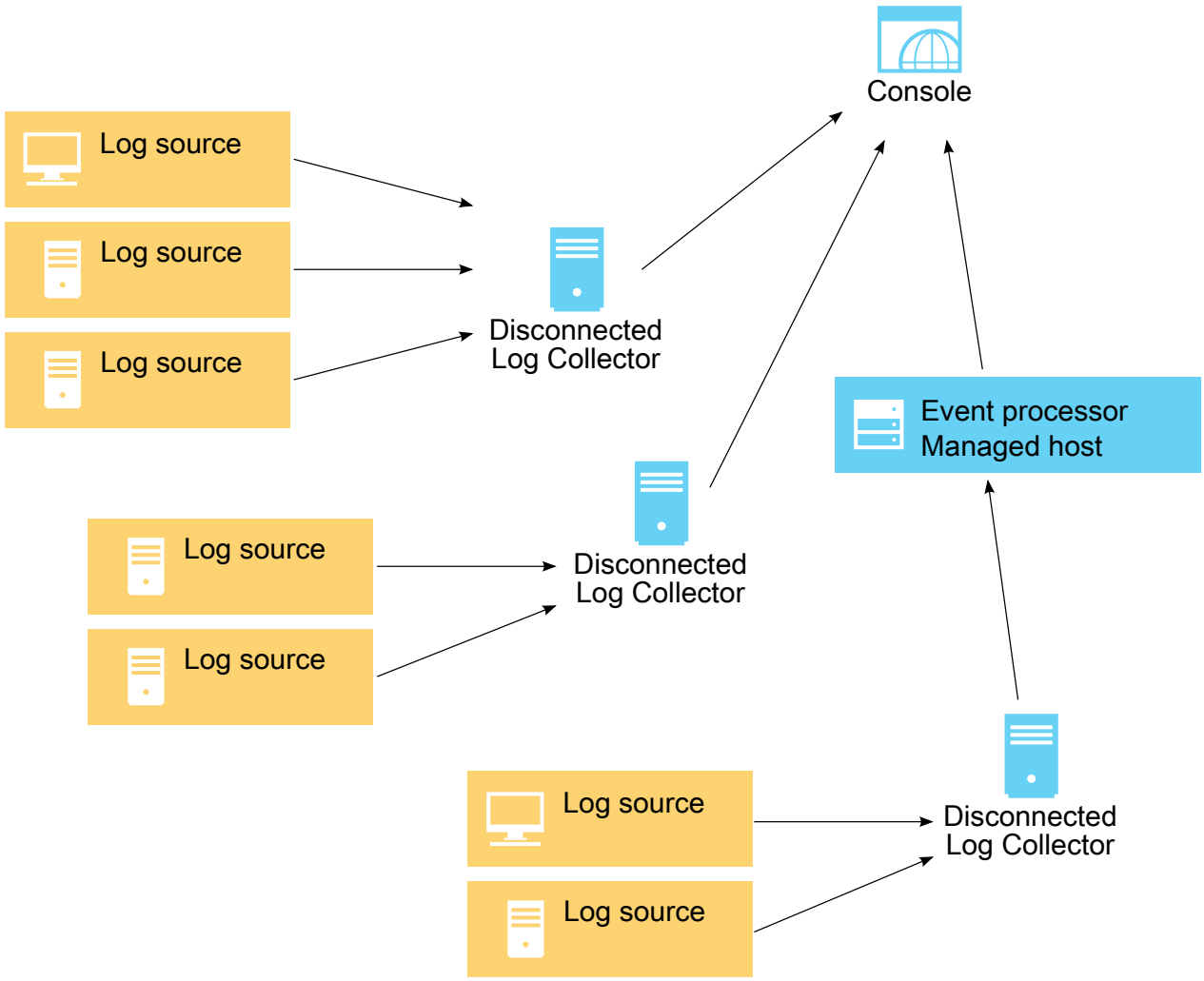


Figure 1. Disconnected Log Collector

Disconnected Log Collector is pre-configured to collect log information from UDP and TCP syslog log sources, and can also be configured for the following log sources:

- Akamai Kona REST API
- Amazon AWS S3 REST API
- Amazon Web Services
- Apache Kafka
- Ariel REST API
- Blue Coat Web Security Service (WSS) REST API
- Box REST API
- Centrify Redrock REST API
- Cisco Firepower eStreamer
- Google G Suite Activity Reports REST API
- IBM Security Verify Event Service (formerly IBM Cloud Identity Event Service)
- JDBC
- JDBC - SiteProtector
- Log File Protocol
- Microsoft Azure Event Hubs

- Microsoft Defender for Endpoint REST API

  Important: Due to a change in the Microsoft Defender API suite as of 25 November 2021, Microsoft no longer allows the onboarding of new integrations with their SIEM API.

  To continue to receive data from Microsoft Defender for Endpoint REST API log sources, you must register a new application and create Microsoft Graph Security API log sources to collect the data. For more information, see Migrating Microsoft Defender for Endpoint REST API log sources to Microsoft Graph Security API log sources.
- Microsoft DHCP Protocol
- Microsoft Exchange
- Microsoft Graph Security API
- Microsoft IIS
- Microsoft Security Event Log over MSRPC
- Microsoft Office 365 REST API
- Microsoft Office 365 Message Trace REST API
- MQ protocol - MQJMS
- Netskope Active REST API
- Okta REST API
- Oracle Database Listener
- Salesforce REST API
- Seculert Protection REST API
- SMB Tail
- SNMPv2
- SNMPv3
- Syslog Redirect
- TCP Multiline Syslog
- TLS Syslog
- UDP Multiline Syslog
- Universal Cloud REST API
- VMware vCloud Director

Business scenarios for using Disconnected Log Collector

Disconnected Log Collector is suitable for a range of business scenarios:

Secured network zones
In high-security unidirectional networks (also known as *data diodes*), Disconnected Log Collector can use the connectionless UDP protocol to send events to QRadar.

Managed security service providers (MSSPs)
Disconnected Log Collector can be installed on small to medium-sized customer sites and doesn't rely on a virtual private network (VPN) to send events to the MSSP. Disconnected Log Collector simplifies administration because each instance clearly belongs to a particular customer domain.

Multi-location businesses
In large retailers and other multi-location businesses, each location typically generates only a few events per second that doesn't justify the cost of a 15xx Event Collector appliance. Disconnected Log Collector can be installed on a cost-effective Linux® computer or virtual machine, where it can collect and send events to the central security infrastructure.

IBM QRadar on Cloud deployments

For businesses that track only events (not flows or vulnerability scans), Disconnected Log Collector is a lightweight alternative to installing a Data Gateway managed host and doesn't rely on a VPN to send events to QRadar on Cloud.

System requirements for Disconnected Log Collector

IBM Disconnected Log Collector is compatible with QRadar 7.3.1 or later.

System hardware

Table 1. System hardware requirements for Disconnected Log Collector

Requirement	Description
Processor	Optimal: 4 CPU cores Minimum: 2 CPU cores
Memory (RAM)	Optimal: 16 GB or more of available RAM. Minimum: 8 GB or more of available RAM.
Disk space	100 GB or more of disk space. **Important:** If you manage your partitions, assign the 100 GB of space to the `/store` partition. The `/store` directory must be either created on the root file system, which must be large enough to accommodate your Disconnected Log Collector instance, or you must create a separate `/store` file system.
Network adapter	One or more network adapters.

Tip: Synchronize the time between the VM and host system to ensure consistent event times.

Operating system

Disconnected Log Collector requires the Red Hat® Enterprise Linux (RHEL) or CentOS Linux V7.x operating system or later. Disconnected Log Collector creates its own user account called **dlc**. It doesn't require any other user accounts on the system.

For more information about installing and configuring RHEL or CentOS Linux, see the RHEL documentation (https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/7/).

Public DNS

If you use a public DNS, and you use a local machine to host your Disconnected Log Collector, add the IP address and hostname of your Disconnected Log Collector in the `/etc/hosts` file on your Linux server. If you do not add the IP address and hostname to `/etc/hosts`, then your local Disconnected Log Collector machine is assigned a dynamic IP address.

Firewall ports

The syslog log source's target port and the destination port must be available and not blocked. By default, the target port is 1514 and the destination port is 32500 for both User Datagram Protocol (UDP) and Transport Layer Security over the Transmission Control Protocol (TLS over TCP).

Installation of Disconnected Log Collector

To install Disconnected Log Collector, you must first install IBM Java and you must open ports on your computer or virtual machine.

Important: If you have a previous version of Disconnected Log Collector installed, upgrade the installation by installing the newer version over your existing installation. Your existing configuration is preserved when you upgrade. For more information about upgrading, see "Installing or upgrading Disconnected Log Collector" on page 9.

Installing Java

IBM Disconnected Log Collector requires IBM SDK, Java Technology Edition, Version 8, 64-bit.

Disconnected Log Collector supports IBM SDK, Java Technology Edition, Version 8.0-6 and later.

1. Go to https://www.ibm.com/support/pages/java-sdk-downloads-version-80, and download the SDK version of the Linux on AMD64/EMT64Tx86 systems 64-bit installation package.
2. Copy the installation package to your computer or VM where you plan to install the Disconnected Log Collector, and then complete the remaining steps on that computer or VM.
3. To install Java, run the following command:

   Tip: Click the **Copy to clipboard** icon at the upper right of your code block, then replace the *<version>* value with the correct version number for the installable package, such as `7.16`. Then, you can run the command.

   ```
   yum -y install ibm-java-x86_64-sdk.<version>.x86_64.rpm
   ```

   Use the default installation location, which is `/opt/ibm/java-x86_64-80/`.

   For more information about installing and configuring IBM SDK, Java Technology Edition, Version 8, see the SDK User Guide (https://www.ibm.com/support/knowledgecenter/en/SSYKE2_8.0.0).

Installing or upgrading Disconnected Log Collector

Install IBM Disconnected Log Collector on a computer or virtual machine (VM) that meets all the system requirements. You can install only one instance of Disconnected Log Collector per computer or VM.

- To use the installation script, you must have Python on your operating system. By default, most operating systems include Python capabilities.
- Ensure that all system requirements are met and that IBM SDK, Java Technology Edition, Version 8, 64-bit is installed. For more information about installing Java, see "Installing Java" on page 9.

If you have a previous version of Disconnected Log Collector that is installed, upgrade the installation by installing the newer version over your existing installation. Your existing configuration is preserved when you upgrade.

1. Download the Disconnected Log Collector installer package from IBM Fix Central (ibm.com/support/fixcentral/).

   Enter `dlc v1.8.5` in the **Search Fix Central** field. The file name for the installer package is `dlc-service-install-1.8.5-1.tgz`.
2. Unpack and run the Disconnected Log Collector installer package by running the following command:

   ```
   tar -zxvf dlc-service-install-1.8.5-1.tgz
   ```

   The Disconnected Log Collector installer package contains the following files:

   - `install.sh`
   - `install.py`
   - `uninstall.sh`
   - `uninstall.py`

- `dlc-service-rpms-1.8.5.tgz`

3. Install the Disconnected Log Collector by running the following script:

```
./install.sh
```

Tip: You can add the version that you want to install after the script name. If you don't add a version, the script prompts you to pick a version. If you only have one version of Disconnected Log Collector in your files, the script installs that version.

4. After the installation is finished, run the following command to restart the Disconnected Log Collector service.

```
systemctl restart dlc
```

5. After the Disconnected Log Collector restarts, run the following command to check the system status.

```
systemctl status dlc
```

An `active (running)` message indicates that the installation was successful and that Disconnected Log Collector is running.

By default, Disconnected Log Collector uses the User Datagram Protocol (UDP) to send log events. Because you still must configure a connection to IBM QRadar, any incoming events are sent only to the local computer.

Opening required ports in the Linux firewall

Some ports must be open in the Linux firewall so that IBM Disconnected Log Collector can receive incoming log sources and communicate with IBM QRadar. Enable port forwarding so that you can use Disconnected Log Collector without needing root privileges.

Ports 1 - 1023 are *privileged* and require a process to be running with root privileges. Because Disconnected Log Collector does not run as root, you must forward any privileged log source listening port to a *non-privileged* port. Non-privileged ports are 1024 or greater.

For example, syslog log sources use port 514. For Disconnected Log Collector to be able to receive the log messages, you must forward port 514 to a non-privileged port, such as port 1514.

1. Log in to the Disconnected Log Collector computer or VM as the root user.
2. Open ports by typing the following commands:

```
firewall-cmd --zone=public --add-port=514/udp --permanent
firewall-cmd --zone=public --add-port=514/tcp --permanent
```

You might have to open other ports for each new log source that you add to your configuration.

3. Forward ports by typing the following commands:

```
firewall-cmd --zone=public --add-forward-port=port=514:proto=tcp:toport=1514 --permanent
firewall-cmd --zone=public --add-forward-port=port=514:proto=udp:toport=1514 --permanent
```

Important: The default syslog log source target port for forwarding is 1514. If you specify a different target port in the `dlc.xml` configuration file, you must substitute it in the port forwarding commands. The target port number must be 1024 or greater.

4. Reload the firewall by typing the following command:

```
firewall-cmd --reload
```

5. Verify that the ports are added by typing the following command:

```
firewall-cmd --list-all
```

Changing the QRadar server destination port

By default, IBM Disconnected Log Collector sends events to IBM QRadar server on port 32500 for both UDP and TLS over TCP. You can change the destination port if 32500 is not available.

The destination port must match the **Listen Port** that is specified in the Disconnected Log Collector log source on the QRadar system.

Important: The following ports are available for Disconnected Log Collector on IBM QRadar on Cloud: 32500, 32501, 32502, and 32503.

1. Log in to the Disconnected Log Collector computer or VM as the root user.
2. Open the `/opt/ibm/si/services/dlc/conf/config.json` file in a text editor.
3. In the **`destination.port`** parameter, enter the listening port for the Event Collector, Event Processor, or QRadar Console that receives events from the Disconnected Log Collector instance. For example:

```
'destination.port':'32501'
```

4. Save and close the `config.json` file.
5. Restart Disconnected Log Collector by typing the following command:

```
systemctl restart dlc
```

Communication between Disconnected Log Collector and QRadar

You can configure IBM Disconnected Log Collector to use one of the following communication protocols to send event data to IBM QRadar: User Datagram Protocol (UDP) or Transport Layer Security over the Transmission Control Protocol (TLS over TCP).

Note: UDP is not supported by QRadar on Cloud

Configuring TLS over TCP communication with QRadar

When IBM Disconnected Log Collector uses TLS over TCP, it buffers incoming events during times when it's disconnected from IBM QRadar and sends them when the connection is restored.

Buffer capacity can be configured, and is limited by the available disk space.

Setting up certificate-based authentication on Disconnected Log Collector

In TLS over TCP communication between IBM Disconnected Log Collector and IBM QRadar, certificate-based communication is used to establish a *chain of trust* in which hardware and software is validated from the end entity to the root certificate.

You must have a root certificate that was issued by a trusted certificate authority (CA). Typically, you use the same root certificate on the Disconnected Log Collector and QRadar systems. Ensure that the root certificate has a meaningful name, such as `root-ca.cer`.

Every certificate has a validity period (a date range) during which it can be used to establish secure communications. After the validity period ends, the certificate expires and must be replaced.

1. Log in to the Disconnected Log Collector computer or VM as the root user.
2. Copy the root certificate to the `/etc/pki/ca-trust/source/anchors` directory and run the following command to update the default truststore:

```
update-ca-trust
```

3. Generate a client certificate signing request (CSR) by typing the following command:

```
/opt/ibm/si/services/dlc/current/script/generateCertificate.sh -csr (-2k | -4k)
```

The `-2k` option represents a 2048-bit key, and `-4k` represents a 4096-bit key. Choose the key size value for the certificate according to the requirements of your organization.

For example:

```
/opt/ibm/si/services/dlc/current/script/generateCertificate.sh -csr -2k
```

4. Enter values for the following parameters.
   a) Enter a two letter code for your country name or leave it blank.
   b) Enter a state or province or leave it blank.
   c) Enter a city name or leave it blank.
   d) Enter your organization name.
   e) Enter your organizational unit.

   The file is saved as `/opt/ibm/si/services/dlc/keystore/<UUID>/dlc-client.key`, where UUID is an identifier that is unique to the Disconnected Log Collector instance.

   Tip: Make note of the UUID identifier that is unique to the Disconnected Log Collector instance. The identifier is the `/opt/ibm/si/services/dlc/keystore/<UUID>` folder name. You'll need the UUID when you configure the Disconnected Log Collector protocol in QRadar.
5. Submit the CSR to your internal or commercial certificate authority for signing, according to their instructions.

   The procedure might involve opening the CSR file and copying a block of encoded text that is contained between the BEGIN and END markers.

   Important: You must have a private certificate authority to sign the certificate for Disconnected Log Collector. If you don't already have one as part of your company infrastructure, you can create one. For example, Easy-RSA is a publicly available tool that you can use to create a certificate authority. For more information, see Easy-RSA (https://github.com/OpenVPN/easy-rsa).
6. Copy the returned client certificate to the `/tmp` directory or your preferred location.
7. Ensure the client certificate is in PEM (Base64 ASCII) format. If the certificate is in DER (binary) format, convert it to PEM format by typing the following command:

   ```
   openssl x509 -inform der -in <certificate_file_name>.der -out <certificate_file_name>.pem
   ```

   The PEM file contains a block of encoded text that is contained between the BEGIN and END markers.
8. Convert the client certificate to PKCS#12 format by typing the following command, and choose a secure password when prompted:

   ```
   /opt/ibm/si/services/dlc/current/script/generateCertificate.sh -p12
   /tmp/<signed_certificate_file_name>
   ```

   A generated personal exchange format (PFX) file is saved as `/opt/ibm/si/services/dlc/keystore/dlc-client.pfx` and the required PFX information is stored in the `/opt/ibm/si/services/dlc/conf/config.json` file.
9. Restart Disconnected Log Collector by typing the following command:

   ```
   systemctl restart dlc
   ```

Setting up certificate-based authentication on QRadar

In TLS over TCP communication between IBM Disconnected Log Collector and IBM QRadar, certificate-based communication is used to establish a *chain of trust* in which hardware and software is validated from the end entity to the root certificate.

You must have a root certificate that was issued by a trusted certificate authority (CA). Typically, you use the same root certificate on the Disconnected Log Collector and QRadar computers. Ensure that the root certificate has a meaningful name, such as `root-ca.cer`. The `client_root_ca.crt` file must be in X.509 format.

If the signer uses an intermediate CA, you must also import the intermediate CA's root certificate into the truststore. In this case, use your own truststore instead of the QRadar server truststore.

Important: If multiple Disconnected Log Collectors exist in the environment, perform the following steps only once on the QRadar system that the Disconnected Log Collector connects to.

1. Use SSH to log in to the Event Collector, Event Processor, or QRadar Console that receives events from the Disconnected Log Collector instance.
2. Copy the root certificate to the `/etc/pki/ca-trust/source/anchors` directory.
3. If you're using your own truststore, type the following commands to add the client certificate's CA and the intermediate CA into your own truststore:

   ```
   keytool -import -alias client_root_ca -file client_root_ca.crt -keystore clientca
   ```

   ```
   keytool -import -alias client_int_ca -file client_int_ca.crt -keystore clientca
   ```

   Important:

   - The `client_root_ca.crt` file must be in X.509 format.
   - Run the second command only if your certificate is signed by an intermediate CA.
4. If you're using the default truststore, type the following command to update the default truststore:

   ```
   update-ca-trust
   ```

5. To configure the server's certificate signing request (CSR), create a text file with this information:

   ```
   [ default ]
   # Change the following line to include the FQDN and IP address of the QRadar console or host
   SAN = DNS:<ec.example.com>,IP:<IP_address>
   [ req ]
   default_bits = 2048                        # RSA key size; change to 4096 if required by
   your
   organization
   encrypt_key = no                           # Protect private key
   default_md = sha256                        # MD to use
   utf8 = yes                                 # Input is UTF-8
   string_mask = utf8only                     # Emit UTF-8 strings
   prompt = no                                # Prompt for DN
   distinguished_name = server_dn             # DN template
   req_extensions = server_reqext             # Desired extensions
   [ server_dn ]
   organizationName = <your_organization_name>
   organizationalUnitName = <your_organizational_unit_name>
   commonName = <common_name>                 # Should match a listed SAN
   [ server_reqext ]
   keyUsage = critical,digitalSignature,keyEncipherment
   extendedKeyUsage = serverAuth,clientAuth
   subjectKeyIdentifier = hash
   subjectAltName = $ENV::SAN
   ```

6. Save the text file as `/tmp/tls-server.conf` or in your preferred location.
7. Generate a server certificate signing request (CSR) by typing the following command:

   ```
   openssl req -new -config /tmp/tls-server.conf -out /tmp/tls-server.csr -keyout /tmp/
   tlsserver.key
   ```

   A server CSR file is saved in `/tmp/tls-server.csr`, and a private key file is saved in `/tmp/tls-server.key`.
8. Submit the CSR to your internal or commercial certificate authority for signing, according to their instructions.

   The procedure might involve opening the CSR file and copying a block of encoded text that is contained between BEGIN and END markers.
9. Copy the returned server certificate to the `/tmp` directory or your preferred location.
10. Ensure that the client certificate is in PEM (Base64 ASCII) format. If the certificate is in DER (binary) format, convert it to PEM format by typing the following command:

```
openssl x509 -inform der -in <certificate_file_name>.der -out <certificate_file_name>.pem
```

Tip: A certificate's file extension does not necessarily indicate the encoding method that is used. For example, a certificate with a `.cer` extension might have Base-64 or DER encoding. Typically, you choose the encoding method during the certificate request procedure. Search the internet for information about OpenSSL commands that convert certificates from one format to another.

The PEM file contains a block of encoded text that is contained between the `BEGIN` and `END` markers.

11. If your CA uses an intermediate CA to sign certificates, ensure that the intermediate CA certificate is in PEM (Base64 ASCII) format. If the certificate is in DER (binary) format, convert it to PEM format (see the previous step). Then, append the intermediate CA certificate to the signed server certificate by typing the following command:

    ```
    cat <intermediate_ca_file_name>.pem >> <signed_server_certificate_file_name>.pem
    ```

12. If the store server certificate that you received is not in PKCS#12 format, such as Distinguished Encoding Rules (DER), convert the client certificate to PKCS#12 format. Type the following command, and choose a secure password when prompted:

    ```
    openssl pkcs12 -inkey /tmp/tlsserver.key -in <signed_server_certificate_file_name>.pem
    -export -out dlc-server.pfx
    ```

    A generated personal exchange format (PFX) file is saved as `/opt/ibm/si/services/dlc/keystore/dlc-client.pfx`. The required PFX information is stored in the `/opt/ibm/si/services/dlc/conf/config.json` file.

13. Choose a secure password when prompted.
14. Copy the server certificate to the QRadar computer in the `/opt/qradar/conf/key_stores` directory. If the `/key_stores` folder doesn't exist, create it.

You can configure the Disconnected Log Collector log source on QRadar by using the `dlc-server.pfx` file that you created.

"Setting up TLS over TCP communication with QRadar" on page 14

Setting up TLS over TCP communication with QRadar

Transport Layer Security over the Transmission Control Protocol (TLS over TCP) provides encrypted and authenticated communication between IBM Disconnected Log Collector and IBM QRadar.

TLS over TCP requires certificate-based authentication between Disconnected Log Collector and QRadar. For more information, see "Setting up certificate-based authentication on Disconnected Log Collector" on page 11 and "Setting up certificate-based authentication on QRadar" on page 12.

Restriction: UDP is not supported by QRadar on Cloud.

1. Log in to the Disconnected Log Collector computer or VM as the root user.
2. Open the `/opt/ibm/si/services/dlc/conf/config.json` file in a text editor.
3. In the **destination.type** parameter, enter TLS (this parameter was set by the certificate-based authentication procedure):

    ```
    'destination.type': 'TLS'
    ```

4. In the **destination.ip** parameter, enter the IP address or the fully qualified domain name (FQDN) for the Event Collector, Event Processor, or QRadar Console that receives events from the Disconnected Log Collector instance. For example:

    ```
    'destination.ip':'192.0.2.0'
    ```

5. Save and close the `config.json` file.
6. Restart Disconnected Log Collector by typing the following command:

    ```
    systemctl restart dlc
    ```

Go to "Add Disconnected Log Collector as a log source in QRadar" on page 16.

Configuring TLS proxy communication with QRadar

Use the TLS proxy for communication between IBM Disconnected Log Collector and IBM QRadar. Disconnected Log Collector supports the basic authentication method for proxy authentication.

In 1.8.5, when you install or upgrade your Disconnected Log Collector, the `config.json` file has a **Proxy** section that you can configure.

Tip: If the proxy server connection is interrupted, the Disconnected Log Collector automatically attempts to re-establish the connection.

1. In the `config.json` file, review the **Proxy** section

   In the following example, the default settings are configured.

```
"Proxy": {
     "proxy.description":"Only applicable to destination types TLS, not applicable to
destination.type: Kafka and UDP",
     "proxy.enabled":"false",
     "proxy.ip":"",
     "proxy.port":"",
     "proxy.username":"",
     "proxy.password":""
}
```

2. To enable the proxy, change the value for the **`proxy.enabled`** parameter to `true`.
3. For the **`proxy.ip`** parameter, enter the IP address of the proxy server.

   The value can be either an IP address or a fully qualified domain name (FQDN).
4. For the **`proxy.port`** parameter, enter the port that the proxy server can receive connections on.
5. Enter the **`proxy.username`** that you configured on the proxy server.
6. Enter the encrypted **`proxy.password`** that you configured on the proxy server.

   To encrypt the proxy password, complete the following steps:

   a) Run the following script:

```
/opt/ibm/si/services/dlc/current/script/encrypt.sh
```

   b) You are prompted to enter and re-enter the proxy password in plain text.

   c) Copy the encrypted password that is displayed.

Note: Connection issues are logged in `/var/log/dlc/dlc.error`.

Configuring UDP communication with QRadar

User Datagram Protocol (UDP) is a connectionless protocol that is suitable for one-way communication, such as in unidirectional networks (also known as *data diodes*). UDP is susceptible to spoofing and should be used only in isolated, secure networks. UDP is the default protocol that IBM Disconnected Log Collector uses to send event logs to an IBM QRadar deployment.

Event log data is buffered only during moments when the incoming events-per-second rate exceeds the computer's ability to relay the information in real time. Event log data is not buffered if the connection is lost between Disconnected Log Collector and QRadar.

1. Log in to the Disconnected Log Collector computer or VM as the root user.
2. Open the `/opt/ibm/si/services/dlc/conf/config.json` file in a text editor.
3. In the **destination.type** parameter, enter UDP (the default):

```
'destination.type': 'UDP'
```

4. In the **destination.ip** parameter, enter the IP address or the fully qualified domain name (FQDN) for the Event Collector, Event Processor, or QRadar Console that receives events from the Disconnected Log Collector instance. For example:

```
'destination.ip':'192.0.2.0'
```

5. Save and close the `config.json` file.
6. Restart Disconnected Log Collector by typing the following command:

```
systemctl restart dlc
```

Go to "Add Disconnected Log Collector as a log source in QRadar" on page 16.

Add Disconnected Log Collector as a log source in QRadar

To collect events from IBM Disconnected Log Collector, you must either use the QRadar Log Source Management app to register Disconnected Log Collector instances with your IBM QRadar deployment or manually install the Disconnected Log Collector protocol and complete configuration steps on your QRadar system.

When you install Disconnected Log Collector, a universally unique identifier (UUID) is created and used for authentication with QRadar. You can use multiple Disconnected Log Collector instances in your environment, and each instance will have a different UUID. To configure multiple Disconnected Log Collector instances, you can do any of the following:

- You can configure multiple Disconnected Log Collector UUIDs to use the same log source in QRadar. That is, each Disconnected Log Collector instance would communicate with QRadar by using the same defined log source.
- You can configure different Disconnected Log Collector instances to use a different listen port in QRadar.
- You can configure each Disconnected Log Collector instance with the same port number but on a different managed host, which is a host with an event collector on it.

Adding the Disconnected Log Collector log source to QRadar

IBM Disconnected Log Collector can communicate with IBM QRadar and forward events to it only by using a Disconnected Log Collector log source. QRadar 7.4.0 and later includes the Disconnected Log Collector protocol.

1. If you are using QRadar version 7.3.3 or earlier, and if your QRadar Console isn't configured to receive automatic updates, download the Disconnected Log Collector protocol from IBM Fix Central (ibm.com/support/fixcentral/).
   a) Log in to the QRadar Console as the root user.
   b) Copy the protocol RPM file to the `/tmp` directory or your preferred location.
   c) Go to the directory, and type the following command:

```
yum -y install <rpm_filename>
```

   d) Log in to QRadar as an administrator.
   e) Go to the **Admin** tab.
   f) Click **Advanced** > **Deploy Full Configuration**.
   QRadar continues to collect events when you deploy the full configuration.
2. In the **Data Sources** section, click **Log Sources**.
3. Click **Add**, and then configure the following protocol-specific parameters for Disconnected Log Collector:

Parameter	Description
Log Source Name	Enter a name for the Disconnected Log Collector log source (for example, `DLC TLS Protocol`).
Log Source Type	Select **Universal DSM**.

Parameter	Description
Protocol Configuration	Select **IBM QRadar DLC Protocol**.
Log Source Identifier	Enter a unique identifier string (for example, the IP address of a computer where Disconnected Log Collector is installed).
Protocol	Select the communication protocol that is used to get events from Disconnected Log Collector. Choose **TLS** (default) or **UDP**. The setting must match the Disconnected Log Collector protocol setting.
Listen Port	Enter the QRadar server port to receive Disconnected Log Collector events. The default port is 32500.
Authentication by Common Name	The Disconnected Log Collector authentication method. If selected, authentication is by the Common Name (UUID) of the client certificate, which is passed by Disconnected Log Collector. If not selected, authentication is by the alias name of the certificate issuer, which is passed by Disconnected Log Collector.
CN/Alias Allowlist	If authentication is by Common Name, enter the UUID of the Disconnected Log Collector instance as the Common Name. If there's more than one instance, enter a comma-separated list of the UUIDs. If authentication is by the alias name, enter the alias name of the root CA that is in the truststore for the Disconnected Log Collector certificate. **Tip:** To see a list of aliases that are in the truststore, run the following command: `keytool -list -v -keystore /etc/pki/ca-trust/extracted/java/cacerts | grep Alias`
Key Store File Name	The file name of the server personal exchange format (PFX) certificate, which is located in the `/opt/qradar/conf/key_stores` directory on the Event Collector, Event Processor, or QRadar Console. This file receives events from the Disconnected Log Collector instance.
Key Store Password	The password for the server PFX certificate.
Check Revocation	Select the checkbox to check whether the certificate is revoked.

Trust Store File Path	By default, the file path of the QRadar server truststore (`/etc/pki/ca-trust/extracted/java/cacerts`).
	If the signer of the Disconnected Log Collector client uses an intermediate CA, it is recommended to use your own truststore instead of the QRadar server truststore. If **Authentication by Common Name** is not selected, the alias for the client certificate's intermediate CA must be included in the **CN/Alias Allowlist**. Use the following commands to add the client certificate's CA and the intermediate CA into your own keystore file:
	`keytool -import -alias client_root_ca -file client_root_ca.crt -keystore clientca`
	`keytool -import -alias client_int_ca -file client_int_ca.crt -keystore clientca`
	Note: The `client_root_ca.crt` file must be in X.509 format.
	To create your own truststore, move the `clientca` keystore file to the `/opt/qradar/conf/key_stores` directory. Then, in **Trust Store File Path**, enter `/opt/qradar/conf/key_stores/clientca`.

Parameter	Description
Trust Store Password	The password for the server trust store (by default, *changeit*).
Target Event Collector	The Event Collector, Event Processor, or QRadar Console that receives events from the Disconnected Log Collector instance.
TLS Protocols	The versions of TLS that can be accepted by the protocol. Select the TLS version supported by your currently installed DLC device. TLSv1.3 is supported with DLC 1.8.4 and later.

4. Click **Save**.
5. In the **Admin** settings, click **Deploy Changes**.

Registering Disconnected Log Collector with QRadar by using the QRadar Log Source Management app

The IBM QRadar Log Source Management app provides an easy-to-use workflow that helps you quickly find, create, edit, and delete log sources. In IBM QRadar 7.4.0 or later, use the QRadar Log Source Management app (version 6.0 or later) to register Disconnected Log Collector instances with your QRadar deployment.

When you register a Disconnected Log Collector with your QRadar deployment, you can use the QRadar Log Source Management app to configure log sources by using protocols that Disconnected Log Collector supports.

You can assign log sources to a registered Disconnected Log Collector, export the configurations, and then import them into your Disconnected Log Collector. You can also use a registered Disconnected Log Collector to define a Domain filter.

1. In the QRadar Log Source Management app, click **Disconnected Log Collectors** > **Register Disconnected Log Collector**.
2. Configure the following parameters:

Field	Description
Name	Enter a name for the Disconnected Log Collector instance (for example, `DLC TLS Protocol`).
Description	Enter a description of the Disconnected Log Collector instance.
UUID	Enter the UUID identifier that is unique to the Disconnected Log Collector instance. The identifier is the `/etc/dlc/instance/<UUID>` folder name.
Protocol	Select the communication protocol that is used to get events from Disconnected Log Collector. Choose **TLS** (default) or **UDP**. The setting must match the Disconnected Log Collector protocol setting.
Version	Enter the Disconnected Log Collector version (for example, `1.7`).

3. Click **Register**.

"Configuring a log source for collection by a Disconnected Log Collector" on page 18

Configuring a log source for collection by a Disconnected Log Collector

When you configure your Disconnected Log Collector as a log source, the events are forwarded to IBM QRadar.

If your log source type is not autodetectable by default, you must do some further configuration to ensure that your forwarded events are detected automatically by QRadar. For more information, see "Adding log sources for Disconnected Log Collector" on page 22 and "Forwarded events" on page 24.

1. In the QRadar Log Source Management app, click **Log Sources**.
2. Click **+ New Log Source** and then click **Single Log Source** or **Multiple Log Sources**.
3. On the **Select a Log Source type** page, select a log source type. Then, type the **Name** and select the Disconnected Log Collector that you registered. Click **Select Protocol Type**.

   Tip: Choosing a Disconnected Log Collector instance removes the **Target Event Collector** log source parameter field because the Disconnected Log Collector instance collects data for the log source.
4. On the **Select a protocol type** page, select a protocol type that your Disconnected Log Collector supports, and then click **Configure Log Source Parameters**.
5. On the **Configure the Log Source parameters** page, choose the log source configuration from which the log source receives events. Then, configure the other parameters that you want to set for the log source.
6. On the **Configure the protocol parameters** page, configure the protocol-specific parameters.

Table 2. Disconnected Log Collector protocol parameters

Parameter	Description
Log Source Identifier	Type a unique name for the log source. The **Log Source Identifier** can be any valid value and does not need to reference a specific server. It can also be the same value as the **Log Source Name**. If you have more than one configured DLC log source, ensure that you give each one a unique name.
Listen Port	Enter the QRadar server port to receive Disconnected Log Collector events. The default port is 32500. 32500 or 32501 are the only ports available for DLC.
Authentication by Common Name	The Disconnected Log Collector authentication method. If selected, authentication is by the Common Name (UUID) of the client certificate, which is passed by Disconnected Log Collector. If not selected, authentication is by the alias name of the certificate issuer, which is passed by Disconnected Log Collector.

Table 2. Disconnected Log Collector protocol parameters (continued)	
Parameter	**Description**
CN/Alias Allowlist	If authentication is by Common Name, enter the UUID of the Disconnected Log Collector instance as the Common Name. If there's more than one instance, enter a comma-separated list of the UUIDs. If authentication is by the alias name, enter the alias name of the root CA that is in the truststore for the Disconnected Log Collector certificate. Each listening port is limited to 50 UUID connections. **Tip:** To see a list of aliases that are in the truststore, run the following command: `keytool -list -v -keystore /etc/pki/ca-trust/extracted/java/cacerts | grep Alias`
Key Store File Name	IBM Support provides the name of the file in the support ticket.
Key Store Password	IBM Support provides the password of the file in the support ticket.
Check Revocation	Select the checkbox to check whether the certificate is revoked.
Trust Store File Path	IBM Support provides the path of the truststore in the support ticket.
Trust Store File Password	IBM Support provides the password of the truststore in the support ticket.

Choose one of the following methods to transfer your log source configuration to the Disconnected Log Collector computer or VM.

- "Transferring the log source configuration when you're connected to the internet" on page 21
- "Transferring the log source configuration when you're not connected to the internet" on page 20

Transferring the log source configuration when you're not connected to the internet

Use the QRadar Log Source Management app to create a configuration file that you copy to your Disconnected Log Collector computer or VM. You can use this method without connecting to the internet. Transferring the log source configuration ensures that you can use the QRadar Log Source Management app to configure the protocols that Disconnected Log Collector collects.

1. In the QRadar Log Source Management app, click **Disconnected Log Collectors**.
2. From the list of your registered Disconnected Log Collector log source configurations, select the Disconnected Log Collector that you are using, and from the menu, click **Export Log Sources**.
3. Enter a password for the export file, and then click **Start**.

   An encrypted configuration file downloads to your computer and is named `dlc-config-<UUID>.json`, where *<UUID>* is the identifier that is unique to the Disconnected Log Collector instance.
4. Log in to the Disconnected Log Collector computer or VM as the root user.

5. Copy the encrypted configuration file to the `/tmp` directory or your preferred location.
6. Generate an import configuration file by running the following command:

```
/opt/ibm/si/services/dlc/current/script/importLogSourceConfig.sh -i dlc-config-<UUID>.json
-o /tmp/logSources.json
```

7. When prompted, enter the password that you specified for the encrypted configuration file.

   The following message appears after the import configuration file is successfully validated:

```
Successfully validate log source file '/tmp/logSources.json'
```

   Tip: If the `logSources.json` file does not validate successfully, review the `/var/log/dlc/logSources.log` file for details. Fix any issues, and then run the validation script again.
8. Copy the validated import configuration file to `/opt/ibm/si/services/dlc/conf/`.

   Tip: Back up the current `logSources.json` file so you have a version of the file that is saved elsewhere.
9. Restart Disconnected Log Collector by typing the following command:

```
systemctl restart dlc
```

Transferring the log source configuration when you're connected to the internet

Use the Disconnected Log Collector computer or VM to create a configuration file that you copy to your Disconnected Log Collector computer or VM. You can use this method only if you are connected to the internet. Transferring the log source configuration ensures that you can use the QRadar Log Source Management app to configure the protocols that Disconnected Log Collector collects.

1. Log in to the Disconnected Log Collector computer or VM as the root user.
2. Generate an import configuration file by running the following command:

```
/opt/ibm/si/services/dlc/current/script/importLogSourceConfig.sh -h <QRadar_IP_address> -u
<QRadar_user_name> -o /tmp/logSources.json
```

   For example, your command might look like this:

```
/opt/ibm/si/services/dlc/current/script/importLogSourceConfig.sh -h 192.0.2.0 -u admin
-o /tmp/logSources.json
```

3. When prompted, enter the QRadar account password.
4. When the import configuration file is successfully validated, the following message appears:

```
Successfully validate log source file '/tmp/logSources.json'
```

   Tip: If the `logSources.json` file does not validate successfully, review the `/var/log/dlc/logSources.log` file for details. Fix any issues, and then run the validation script again.
5. Copy the validated import configuration file to `/opt/ibm/si/services/dlc/conf/`.

   Tip: Back up the current `logSources.json` file so you have a version of the file that is saved elsewhere.
6. Restart Disconnected Log Collector by typing the following command:

```
systemctl restart dlc
```

Adding log sources for Disconnected Log Collector

Disconnected Log Collector is configured to collect log information from UDP and TCP syslog log sources. You can add other log sources by modifying the `logSources.json` configuration file, which defines the log sources.

`readme` files are provided for the additional log source protocols that you can use with Disconnected Log Collector, and a script is provided to validate your log source definitions. The validation script ensures that the `json` file is properly formatted, and also validates the values that you provide for each parameter against the schema definition in the `readme` file.

Important: Define the new log source definitions in a file other than the `logSources.json` file, and then add the definitions into `logSources.json` when the configuration is complete and valid.

Disconnected Log Collector regularly scans `logSources.json` file for changes. If you edit the `logSources.json` file directly, your log source collection might be disrupted if you enter invalid information.

Note: Either when adding log sources with Gateway capabilities and/or when using the **logSourceIdentifierPattern** variable, if the expressions contain backslash such as "\s" or "\d", then you must escape the backslash char in the JSON. The expression then becomes -

```
\\s or \\d
```

1. Log in to the Disconnected Log Collector computer or VM as the root user.
2. In a text editor, create a JSON file with the following structure:

```
{
        "LogSources":[

        ]
}
```

3. From the `/opt/ibm/si/services/dlc/conf/template` directory, open the `readme` file for the log source that you want to add to Disconnected Log Collector.
4. Paste the log source definition (the `readme` file contents) between the square brackets in your JSON file.

   Important: If you're adding multiple log sources, each log source definition must have opening and closing curly braces. Each log source section must be separated by a comma, as in the following example.

   Figure 2. logSources.json formatting example

5. Edit the values as needed for your environment.

A `readme` file is provided for each log source `json` template that contains information about the values for each parameter.

Tip: Refer to the *QRadar DSM Configuration Guide* for more information about the parameters. The DSM documentation refers to the parameters as they are displayed in the IBM QRadar application. The `logSources.json` parameters are named according to the database labels.

Note: Each log source has a unique **DatabaseId** value. If you add log sources, you must ensure that the **DatabaseId** value is unique for the new log sources. If there are duplicate **DatabaseId** values in the `logSources.json` file, only the first log source is recognized by Disconnected Log Collector. The validation script identifies duplicate **DatabaseId** values.

6. Do the following to encrypt a log source password:
   a) Run the following command:

   ```
   /opt/ibm/si/services/dlc/current/script/encrypt.sh
   ```

   b) Enter the password that you want to encrypt, and again to confirm the password.

   The script displays an encrypted password.

   c) Copy the encrypted password into your log source configuration file.
7. Validate the configuration file by running the following command:

   ```
   /opt/ibm/si/services/dlc/current/script/log_source_validate.sh <path_to_file>/
   <file_to_validate>.json
   ```

   Ensure that you include the *<path_to_file>/<file_to_validate>*`.json` part of the command. Otherwise, the script validates the `logSources.json` file.

   The following message appears after the file is successfully validated:

   ```
   Successfully validate log source file '<path_to_file>/<file_to_validate>.json'
   ```

   Trouble: If the file does not validate successfully, review the `/var/log/dlc/`*`<file_to_validate>`*`.log` file for details. Fix any issues, and then run the validation script again.
8. When your file is valid, copy the new log source definitions into the `logSources.json` file.
   a) Go to the `/opt/ibm/si/services/dlc/conf` directory.
   b) Make a backup of the `logSources.json` file.
   c) Copy the new log source definitions into the `logSources.json` file. Ensure that you add the log source definitions between the square brackets in the `logSources.json` file.
   d) Save the `logSources.json` file.

   Note: Disconnected Log Collector regularly scans the `logSources.json` file for changes. Any changed log sources are restarted and new sources are started. Changes are detected within 5 minutes.
9. To validate the `logSources.json` file after you add new protocols, run the following command:

   ```
   /opt/ibm/si/services/dlc/current/script/log_source_validate.sh
   ```

   The following message appears after the file is successfully validated:

   ```
   Successfully validate log source file '/opt/ibm/si/services/dlc/conf/logSources.json'
   ```

   Trouble: If the `logSources.json` file does not validate successfully, review the `/var/log/dlc/logSources.log` file for details. Fix any issues, and then run the validation script again.
10. If you are defining JDBC for MySQL, copy the JDBC driver (for example, `mysql-connector-java-`*`<version>`*`.jar`) to the `/opt/ibm/si/services/dlc/current/lib` directory.

11. If you modified the TLS syslog log source values, restart Disconnected Log Collector by typing the following command:

```
systemctl restart dlc
```

Forwarded events

The **IBM QRadar DLC Protocol** brings forwarded events from one or more IBM Disconnected Log Collector instances into IBM QRadar.

Forwarded events from log source types that are autodetectable are autodetected as if the events were sent directly to QRadar. The protocol type for these forwarded events is **Forwarded**, regardless of which protocol the Disconnected Log Collector instance used to collect them. If events are sent by using Transport Layer Security over the Transmission Control Protocol (TLS over TCP), then the Log Source Identifier of the autodetected log source includes the UUID of the forwarding Disconnected Log Collector instance. For example, *192.0.2.0277f291f-dca9-4c59-978a-9d6deb0223b0*. This format helps to ensure proper separation of event data.

Forwarded events from log source types that are not autodetectable by default require some configuration. You can create log sources for these events, singularly or in bulk, by using the QRadar **Log Sources** window, the Log Source Management app, or the Log Sources REST API. You must set the log sources' **Protocol Configuration** parameter to **Forwarded** for events that are forwarded by a Disconnected Log Collector instance. If the events are sent by using TLS over TCP, then the Log Source Identifier must include the UUID of the forwarding Disconnected Log Collector instance.

Alternatively, in QRadar 7.3.2, you can configure **Log Source Autodetection** for log source types that are not autodetectable by default. You can configure autodetection for any log source type (custom or IBM provided) by using the DSM Editor **Configuration** tab.

For more information about adding log sources singularly, in bulk, or by using **Log Source Autodetection**, see the *DSM Configuration Guide*.

Installing a certificate for a log source protocol

Some log source protocols require a certificate so that IBM Disconnected Log Collector can communicate with the target server. You install the certificate by running a script command.

The following protocols require a certificate:

- Akamai Kona REST API
- Amazon AWS REST API
- Amazon Web Services
- Blue Coat WSS REST API
- Cisco Firepower eStreamer
- Microsoft Azure Event Hubs
- Salesforce REST API

Ensure that you set the **`getCerts`** parameter to no in the `logsources.json` file for these protocols.

1. Log in to the Disconnected Log Collector computer or VM as the root user.
2. Go to the `/opt/ibm/si/services/dlc/1.2.0.master.`*`version`*`/script` directory.
3. Run the following command:

```
./getcert.sh <server.fullyqualifed.domain.name>
```

You do not have to restart Disconnected Log Collector after you have installed the certificate.

Setting the maximum EPS rate

You can set the maximum events per second (EPS) rate that IBM Disconnected Log Collector sends to IBM QRadar.

The default is 5000. The maximum rate is applied to UDP and TLS over TCP connections to QRadar.

1. Log in to the Disconnected Log Collector computer or VM as the root user.
2. Open the `/opt/ibm/si/services/dlc/conf/config.json` file in a text editor.
3. In the **EPS** parameter, enter the maximum EPS rate that Disconnected Log Collector sends to QRadar.
4. Save and close the `config.json` file.
5. Restart Disconnected Log Collector by typing the following command:

```
systemctl restart dlc
```

Changing the spillover memory and disk usage settings

If you are using TLS over TCP to send log messages to IBM QRadar, IBM Disconnected Log Collector uses the configured memory and disk space to buffer log messages. You can change these values to meet your storage requirements for the hardware that you are using.

Disconnected Log Collector buffers events if there are more events than the configured events per second (EPS) rate or if Disconnected Log Collector is offline. Events are buffered in memory and, when the maximum is reached, the events are saved to spillover files on your hard disk.

The `spilloverqueue.properties` file specifies the memory settings in the following properties:

- `ecs-dlc_dlc_TCP_TO_QRADAR.capacity.in.mem`
- `ecs-dlc_dlc_TCP_TO_QRADAR.total.memory.size.mb`

Note: The value that is specified for `ecs-dlc_dlc_TCP_TO_QRADAR.capacity.in.mem` is overridden by the following line in the `/opt/ibm/si/services/dlc/<version>.master.#####/eventgnosis/config` file:

```
<parameter type="Number">50000</parameter>          <!-- qMemCapacity -->
```

The spill file settings are defined by the following properties:

- `ecs-dlc_dlc_TCP_TO_QRADAR.max.files`
- `ecs-dlc_dlc_TCP_TO_QRADAR.max.file.size.mb`

The default values use 477 100 MB files.

1. Log in to the Disconnected Log Collector computer or VM as the root user.
2. Open the `/opt/ibm/si/services/dlc/conf/spilloverqueue.properties` file in a text editor.
3. Change the following values to set the in memory buffering:

```
ecs-dlc_dlc_TCP_TO_QRADAR.capacity.in.mem=50000
ecs-dlc_dlc_TCP_TO_QRADAR.total.memory.size.mb=1000
```

4. Change the following values to set the spill file settings:

```
ecs-dlc_dlc_TCP_TO_QRADAR.max.file.size.mb=100
ecs-dlc_dlc_TCP_TO_QRADAR.max.files=477
```

5. Save and close the file.
6. Restart Disconnected Log Collector by typing the following command:

```
systemctl restart dlc
```

Sending Disconnected Log Collector health metrics to QRadar

If you use TLS over TCP communication, you can enable metrics to track the number of accumulated spillover files and the events per second (EPS) rate that IBM Disconnected Log Collector sends to IBM QRadar. Disconnected Log Collector sends the metrics as events to QRadar, where you can create rules to react to the metrics.

Also, Disconnected Log Collector sends an event to QRadar to notify when the client certificate is about to expire. By default, the event is sent 14 days before the certificate expires.

1. Log in to the Disconnected Log Collector computer or VM as the root user.
2. Open the `/opt/ibm/si/services/dlc/conf/config.json` file in a text editor.
3. Set the **DLCMetricsEventsEnabled** parameter to `true`.
4. In the **tls.keystoreexpirywindow** parameter, enter the number of days' notice to be given before the client certificate expires.
5. Save and close the file.
6. Restart Disconnected Log Collector by typing the following command:

```
systemctl restart dlc
```

Disconnected Log Collector starts sending metrics events to QRadar. The event name is **DLC Metrics**.

Updating cipher suite permissions for Disconnected Log Collector

Harden your IBM Disconnected Log Collector instance by modifying the cipher suite permissions in Java.

A *cipher suite* is a set of algorithms that are used to secure a connection between clients and servers by using the TLS or SSL protocols. During that handshake process, they agree about which cipher suite to use to establish an HTTPS connection. After the cipher suite is agreed upon, the client and server proceed with the key exchange and other connected parts.

1. Open the IBM Java security file on your Disconnected Log Collector instance at `/opt/ibm/java-x86_64-80/jre/lib/security/java.security`.
2. Locate the section that includes `dk.tls.disabledAlgorithms` to find the list of restricted ciphers. For example, this output shows a list of restricted ciphers that are separated by a comma and a backward slash (\):

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

3. Update the list of restricted ciphers by either listing a specific cipher suite or by specifying standard names that correspond to a group of cipher suites. Separate them with a comma, space, and backwards slash (`,  \`).
   The following list includes, but is not limited to, examples of cipher suites that you can restrict:
   - `SSL_RSA_WITH_AES_128_CBC_SHA`
   - `SSL_RSA_WITH_AES_256_CBC_SHA`
   - `SSL_RSA_WITH_AES_128_CBC_SHA256`
   - `SSL_RSA_WITH_AES_256_CBC_SHA256`
   - `SSL_RSA_WITH_AES_128_GCM_SHA256`
   - `SSL_RSA_WITH_AES_256_GCM_SHA384`
4. Save your changes and restart the Disconnected Log Collector instance by using the following command.

```
systemctl restart dlc
```

Disaster Recovery and Disconnected Log Collector

With the IBM QRadar Data Synchronization app, your IBM QRadar configurations and data are automatically mirrored to another identical QRadar system. This app helps implement disaster recovery protocols for your IBM Disconnected Log Collector device.

For more information about the QRadar Data Synchronization app, see Introducing the IBM QRadar Data Synchronization app.

Important: Before you can use disaster recovery with Disconnected Log Collector, you must install jq, which is a lightweight command-line JSON processor. jq ensures that the destination Disconnected Log Collector can parse the information that you export from your main Disconnected Log Collector.

To download jq, type the following command:

```
yum install jq
```

Migrating Disconnected Log Collector data to the destination QRadar site

If you created a disaster recovery QRadar environment, but did not create a disaster recovery Disconnected Log Collector, then your server certificate must include information about both the main and destination sites. This information ensures that your Disconnected Log Collector log source transfers properly between sites.

1. Copy the root certificate that is used for Disconnected Log Collector from the main IBM QRadar site to the destination site.
   a. If you're using the default Java truststore, the root CA certificates are not synchronized between the main and destination QRadar sites. Copy the root certificate from the `/etc/pki/ca-trust/source/anchors` folder on the main site to the same folder on the destination site. Then, run the `update-ca-trust` command on the destination site to import the certificate.
   b. If you're using your own custom Java truststore, the truststore is not synchronized between the main and destination QRadar sites. Copy the truststore file that you use with the Disconnected Log Collector log source to the same folder on the destination site.
2. Copy the server certificate that you use for the Disconnected Log Collector log source from the `/opt/qradar/conf/key_stores` folder on the main site to the same folder on the destination site.

Tip: When you generate the server certificate for the Disconnected Log Collector log source, you can add the IP address of the secondary QRadar box to the SAN of the certificate request.

```
DNS:<ec.example.com>,IP:<Primary IP address>,IP:<Destination IP address>
```

For more information, see Setting up certificate-based authentication on Disconnected Log Collector.

1. Stop the Disconnected Log Collector service on the main QRadar site by typing the following command:

   ```
   systemctl stop dlc
   ```

2. Update the `destination.ip` value in `/opt/ibm/si/services/dlc/conf/config.json` to be the IP address of the host you want to point to on the destination site.
3. Start the Disconnected Log Collector service on the destination QRadar site by typing the following command:

   ```
   systemctl start dlc
   ```

Backing up and restoring Disconnected Log Collector by using scripts

To backup or restore Disconnected Log Collector, run the configBackup and configRestore scripts before you install or upgrade to a new Disconnected Log Collector instance. For instance, a hardware malfunction or an OS update, and where a Disconnected Log Collector rebuild is necessary.

1. Run the `configBackup.sh` script to back up all your configuration information to a file. In the following filepath, *timestamp* represents the current timestamp in the filename: `/store/tmp/dlc_config_backup_`*`timestamp`*`.tar.gz`.
   For example, `/store/tmp/dlc_config_backup_2023-12-06T13:46:39.tar.gz`.

   ```
   configBackup.sh
   ```

2. Run the `configRestore.sh` script to restore all your saved configuration information. You must include an input parameter of the configuration backup. For example, `configRestore.sh /store/tmp/dlc_config_backup_2023-12-06T13:46:39.tar.gz`.

   ```
   configRestore.sh <filepath>
   ```

   Important: If the configuration is to restore a different Disconnected Log Collector instance, you're prompted to shut down the original Disconnected Log Collector.

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.

Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

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

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

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