

Configuring MapR Security (MapR-SASL) for Drill

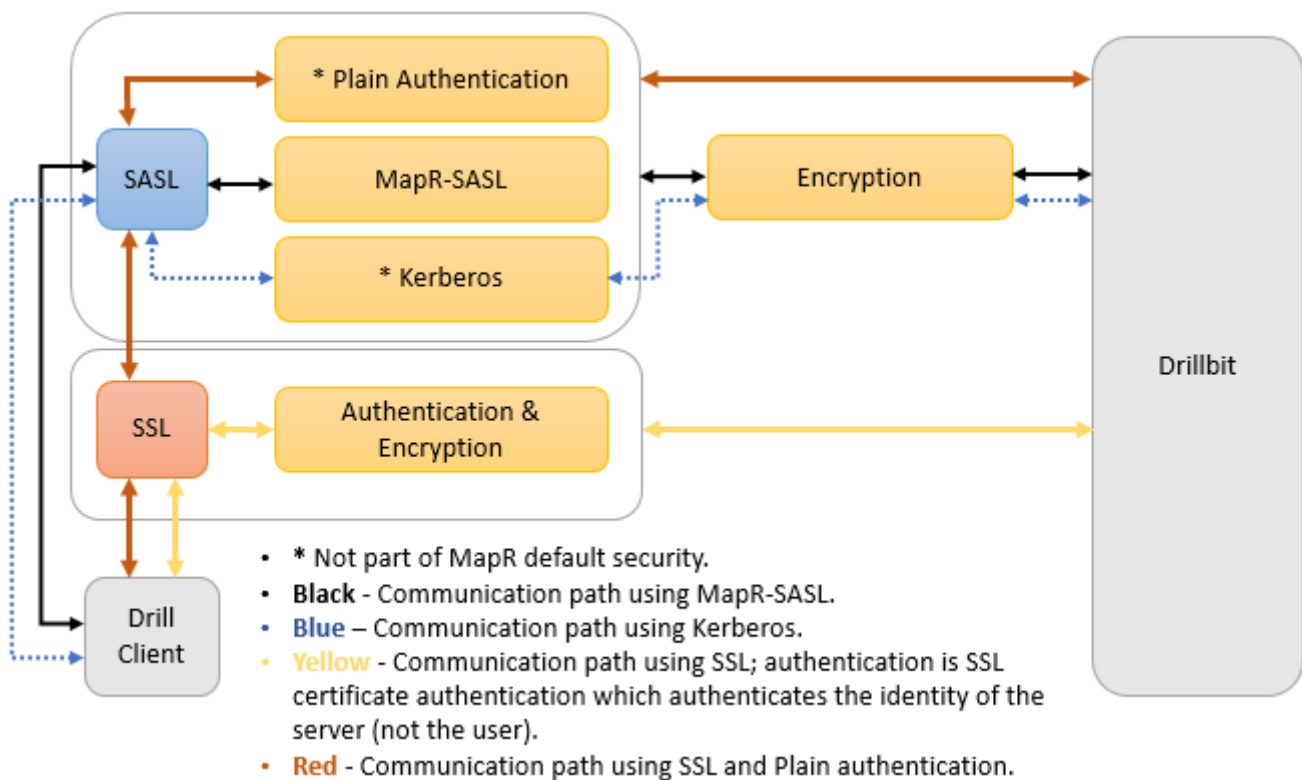
MapR Drill supports authentication through the MapR-SASL mechanism and also supports encryption. Authentication is the process of establishing confidence of authenticity. Encryption is the process of converting information or data in plain text into ciphertext to prevent unauthorized access.

Drill 1.10 supports authentication mechanisms between the Drill client and Drillbit through which identities can be proven before accessing secure cluster data. As of MapR 6.0 and Drill 1.11, Drill supports network encryption and authentication using the MapR-SASL mechanism to encrypt the communication channel between the Drill client, such as JDBC or ODBC, and Drillbit, and between Drillbits. The Drill Web Console supports encryption through HTTPS.

If the default security feature is enabled on the MapR cluster, Drill is automatically secured with MapR-SASL authentication and encryption, by default. If you upgraded to MapR 6.0 and are configuring custom security in the cluster, you can manually enable encryption for MapR-SASL. See [Securing Drill](#) (securing_drill.html#securing_drill) for more information.

Note: For encryption and authentication to work together, the Drill client and Drillbits (servers) must all be running Drill 1.11 or later. Drill clients running earlier versions of Drill cannot connect to a Drillbit when encryption is enabled.

The following diagram depicts the possible SASL and SSL communication paths between the Drill client and server, including the scenario where Plain authentication is used, when Drill 1.11 and later is running in a MapR 6.0 cluster:




Note: Plain authentication does not support encryption using the SASL layer. You must enable SSL to encrypt the communication channels when Plain authentication is used. See [Using SSL/TLS for Encryption](#) (config-ssl-encrypt.html#config-ssl-encrypt).

✓ Enabling Authentication and Encryption

This section describes how to configure Drillbits in a secure MapR cluster to use the MapR-SASL authentication mechanism with and without encryption. When MapR-SASL is enabled, all Drill clients, such as JDBC and ODBC, must connect to Drill servers using SASL.

An administrator can enable or disable authentication and encryption for Drill through configuration parameters in the Drill startup configuration file, `drill-override.conf`, located in `/opt/mapr/drill/drill-<version>/conf`.

 **Note:** A Drill client is authenticated when a drillbit process running in a secure Drill cluster confirms the identity of the client. The data exchanged between the Drill client and Drillbit is encrypted when Drill client authentication and encryption is available through JDBC and ODBC interfaces, as described in the [Drill Drivers](#) (`drill_connectors.html#drill_connectors`) section.

Pre-requisites

- Ensure that your MapR cluster is secure. To manually configure secure clusters with MapR security, see [Enable Wire-Level Security](#) (`../SecurityGuide/Enable-wire-level-security.html#enable-wire-level-security`).
- To use encryption, authentication must be configured and enabled with the encryption-specific configurations.

Post-requisite

You must restart the Drillbit process on each node after you enable security and/or modify the configuration options, as shown:


```
$ maprcli node services -name drill-bits -action restart -nodes <node host names separated by a space>
```

▼ Enabling Authentication

You can enable MapR-SASL as the only authentication mechanism, or you can enable multiple types of authentication. To enable authentication for a Drillbit using MapR-SASL, set the `auth.mechanism` and `user.auth.enabled` configuration options in the `drill-override.conf` file located in `$DRILL_HOME/conf`, as shown in the following two examples:


Example 1: Drill Client to Drillbit Authentication using MapR-SASL Only

```
drill.exec:{
    security: {
        user.auth.enabled: true,
        auth.mechanism : ["MAPRSASL"]
    }
}
```

 **Note:** All the queries are executed as a service or process user when impersonation is disabled.

Example 2: Drill Client to Drillbit Authentication with User Impersonation using MapR-SASL

```
drill.exec:{
  security: {
    user.auth.enabled: true,
    auth.mechanism : ["MAPRSASL"],
  }
  impersonation: {
    enabled: true,
    max_chained_user_hops: 3
  }
}
```

 **Note:** All the queries are executed as the authenticated (ticket) user when impersonation is enabled.

Example 3: Drill Client to Drillbit using Multiple Authentication Mechanisms

```
drill.exec:{
  security: {
    user.auth.enabled: true,
    user.auth.impl: "pam4j",
    security.user.auth.packages += "org.apache.drill.exec.rpc.user.security"
    user.auth.pam_profile: ["sudo", "login", "mapr-admin"],
    auth.mechanism : ["MAPRSASL", "KERBEROS", "PLAIN"],
    auth.principal : "mapr/_host@REALM.COM",
    auth.keytab : "/opt/mapr/conf/mapr.keytab"
  }
}
```

Example 4: Drillbit to Drillbit Authentication using MapR-SASL


```
drill.exec:{
  security: {
    bit.auth.enabled : true
    bit.auth.mechanism : "MAPRSASL"
  }
}
```

Example 5: Drill Client to Drillbit and Drillbit to Drillbit Authentication using MapR-SASL

```
drill.exec {
  security: {
    user.auth.enabled: true,
    auth.mechanisms : ["MAPRSASL"],
    bit.auth.enabled : true,
    bit.auth.mechanism : "MAPRSASL"
  }
}
```

For additional information, see:


- Drill JDBC Driver (../attachments/JDBC_ODBC_drivers/DrillJDBCInstallandConfigurationGuide.pdf) for JDBC configuration options.
- [Configuring Kerberos Authentication](https://drill.apache.org/docs/configuring-kerberos-authentication/) (https://drill.apache.org/docs/configuring-kerberos-authentication/)

 **Note:** Proper setup, configuration, administration, and usage of a Kerberos environment is beyond the scope of this documentation. See the [MIT Kerberos](http://web.mit.edu/kerberos/) (http://web.mit.edu/kerberos/) documentation for support.

- [Configuring Plain Authentication](https://drill.apache.org/docs/configuring-plain-authentication/) (https://drill.apache.org/docs/configuring-plain-authentication/) (For Plain (or basic authentication) using a username and password.)


▼ Enabling Encryption

You can enable encryption when the MapR-SASL and/or Kerberos mechanisms are enabled. To enable encryption for Drill, set the `user.encryption.sasl.enabled` configuration option to “true” in the `drill-override.conf` file located in `$DRILL_HOME/conf`, as shown in the *Examples* section below:

 **Note:** A Drill client, such as JDBC or ODBC, can require a server with encryption capabilities when the `sasl_encrypt` connection parameter is set to “true.” If the client tries connecting to a Drillbit with encryption disabled, the connection fails.

Configuring Encryption

You can customize encryption settings on a Drillbit through options in the Drill startup configuration file, `drill-override.conf`, located in `/opt/mapr/drill/drill-<version>/conf`.

 **Note:** Authentication must be enabled with encryption.

The following table lists the encryption configuration options with their descriptions and default values:


Option	Description	Default
<code>drill.exec.security.user.encryption.sasl.enabled</code>	Determines if encryption on the server is enabled for negotiating privacy with the Drill client.	false
<code>drill.exec.security.bit.encryption.sasl.enabled</code>	Determines if the server is enabled for negotiating privacy with another Drillbit	false


Configuration Examples

The following sections provide example configurations:

Configuration 1: Drill Client to Drillbit Channel Configured with MapR-SASL Authentication and Encryption

With the following server configuration settings, a Drill client running Drill 1.11 with encryption enabled can connect to the Drillbit (server), if the server is configured with the MapR-SASL and/or Kerberos mechanisms.

 **Note:** In this case Drill clients running pre-1.11 versions of Drill cannot connect to the Drillbit through MapR-SASL since encryption is enabled.

 **Note:** Plain authentication not supported.

```
drill.exec {  
    security: {  
        user.auth.enabled: true,  
        auth.mechanisms : ["MAPRSASL"]  
        user.encryption.sasl.enabled : true  
    }  
}
```

Configuration 2: Drillbit to Drillbit Connection with MapR-SASL Authentication and Encryption

The following configuration authenticates and encrypts the connection between Drillbits using the MapR security mechanism.

```
drill.exec {
  security: {
    bit.auth.enabled : true
    bit.auth.mechanism : "MAPRSASL"
    bit.encryption.sasl.enabled : true
  }
}
```

Configuration 3: Drillbit Client to Drillbit and Drillbit to Drillbit Connection with MapR-SASL Authentication and Encryption

The following configuration authenticates and encrypts the connection between the Drill client and Drillbit, and between Drillbits using the MapR security mechanism.

```
drill.exec {
  security: {
    user.auth.enabled: true,
    auth.mechanisms : ["MAPRSASL"],
    user.encryption.sasl.enabled : true

    bit.auth.enabled : true
    bit.auth.mechanism : "MAPRSASL"
    bit.encryption.sasl.enabled : true
  }
}
```

Configuration 4: Drill Client to Drillbit Authentication and Encryption Enabled using Multiple Mechanisms and Drillbit to Drillbit using MapR-SASL

The following configuration authenticates and encrypts the connection between the Drill client and Drillbit using multiple authentication mechanisms, and also authenticates and encrypts the connection between Drillbits using the MapR security mechanism.

```
drill.exec {
  security: {
    user.auth.enabled: true,
    auth.mechanisms : ["MAPRSASL", "KERBEROS"],
    auth.principal : "mapr/_host@REALM.COM",
    auth.keytab : "/opt/mapr/conf/mapr.keytab",
    user.encryption.sasl.enabled : true,
    bit.auth.enabled : true,
    bit.auth.mechanism : "MAPRSASL",
    bit.encryption.sasl.enabled : true
  }
}
```


Client Encryption


For encryption, there is one optional connection parameter, `sasl_encrypt`, for both the JDBC and ODBC clients. The default value for this parameter is `"false."`

If the server is configured with encryption, both the client and the server negotiate for an encrypted connection. If the server-side is not configured with encryption, but is configured with authentication, both the client and the server negotiate for an authenticated connection.

To ensure that the client always connects to a Drillbit with encryption enabled, use the connection parameter as shown:

```
sasl_encrypt=true
```


 **Note:** When `sasl_encrypt=true` is set in the connection URL, the Drill client cannot connect to Drillbits unless they have encryption enabled.

 **Note:** When using the MapR provided ODBC driver, note that the name of the connection parameter is `EnforceSaslEncrypt`. See the [ODBC](#) ([drill_odbc_connector.html#drill_odbc_connector](#)) and [JDBC](#) ([drill_jdbc_connector.html#drill_jdbc_connector](#)) documentation for more information.

Client Connection URL Examples


Example 1: Connection URL where the Drill client negotiates for authentication and encryption with the Drillbit using the MapR-SASL mechanism.

```
jdbc:drill:zk=<zk name>[:<port>][,<zk name2>[:<port>]]...  
<directory>/<cluster ID>;auth=maprsasl
```

 **Note:** If the Drillbit to which the client is connecting has encryption enabled, the only negotiation is for encryption; otherwise negotiation is only for authentication.

Example 2: Connection URL for the Drill client to strictly negotiate for encryption with the Drillbit using the MapR-SASL mechanism.

```
jdbc:drill:zk=<zk name>[:<port>][,<zk name2>[:<port>]]...  
<directory>/<cluster ID>;auth=maprsasl;sasl_encrypt=true
```

 **Note:** If the Drillbit to which the client is connecting has encryption disabled, the connection fails. The connection is only successful if the target Drillbit has encryption enabled.

Monitoring Encryption

You can monitor encryption status from the Drill Web Console at `https://<hostname>:8047</>`.

When you log in to the Web Console, you can see if the client/server side encryption is enabled or disabled for the Drillbit to which the Web client connects. You can also see the number of encrypted and unencrypted client connections to the Drillbit, as well as the number of control and data connections to and from the Drillbit.