

IRC

IRC Component

The **irc** component implements an [IRC](#) (Internet Relay Chat) transport.

Maven users will need to add the following dependency to their `pom.xml` for this component:

```
<dependency>
  <groupId>org.apache.camel</groupId>
  <artifactId>camel-irc</artifactId>
  <version>x.x.x</version>
  <!-- use the same version as your Camel core version -->
</dependency>
```

URI format

```
irc:nick@host[:port]/#room[?options]
irc:nick@host[:port]?channels=#channel1,#channel2,#channel3[?options]
```

You can append query options to the URI in the following format, `?option=value&option=value&...`

Options

Name	Description	Example	Default Value
channels	Comma separated list of IRC channels to join.	channels=#channel1,#channel2	null
nickname	The nickname used in chat.	irc:MyNick@irc.server.org#channel or irc:irc.server.org#channel?nickname=MyUser	null
username	The IRC server user name.	irc:MyUser@irc.server.org#channel or irc:irc.server.org#channel?username=MyUser	Same as nickname.
password	The IRC server password.	password=somepass	<i>None</i>
realname	The IRC user's actual name.	realname=MyName	<i>None</i>
nickPassword	Camel 2.18: Your IRC server nickname password.	nickPassword=mysecret	<i>None</i>
colors	Whether or not the server supports color codes.	true, false	true
onReply	Whether or not to handle general responses to commands or informational messages.	true, false	false
onNick	Handle nickname change events.	true, false	true
onQuit	Handle user quit events.	true, false	true
onJoin	Handle user join events.	true, false	true
onKick	Handle kick events.	true, false	true
onMode	Handle mode change events.	true, false	true
onPart	Handle user part events.	true, false	true
onTopic	Handle topic change events.	true, false	true
onPrivmsg	Handle message events.	true, false	true
trustManager	The trust manager used to verify the SSL server's certificate.	trustManager=#referenceToTrustManagerBean	The default trust manager, which accepts <i>all</i> certificates, will be used.

keys	Camel 2.2: Comma separated list of IRC channel keys. Important to be listed in same order as channels. When joining multiple channels with only some needing keys just insert an empty value for that channel.	irc:MyNick@irc.server.org /#channel?keys=chankey	null
sslContextParameters	Camel 2.9: Reference to a <code>org.apache.camel.util.jsse.SSLContextParameters</code> in the Registry . This reference overrides any configured <code>SSLContextParameters</code> at the component level. See Using the JSSE Configuration Utility . Note that this setting overrides the <code>trustManager</code> option.	#mySslContextParameters	null

SSL Support

Using the JSSE Configuration Utility

As of Camel 2.9, the IRC component supports SSL/TLS configuration through the [Camel JSSE Configuration Utility](#). This utility greatly decreases the amount of component specific code you need to write and is configurable at the endpoint and component levels. The following examples demonstrate how to use the utility with the IRC component.

Programmatic configuration of the endpoint

```
KeyStoreParameters ksp = new KeyStoreParameters();
ksp.setResource("/users/home/server/truststore.jks");
ksp.setPassword("keystorePassword");

TrustManagersParameters tmp = new TrustManagersParameters();
tmp.setKeyStore(ksp);

SSLContextParameters scp = new SSLContextParameters();
scp.setTrustManagers(tmp);

Registry registry = ...
registry.bind("sslContextParameters", scp);

...

from(...)
    .to("ircs://camel-prd-user@server:6669/#camel-test?nickname=camel-prd&password=password&sslContextParameters=#sslContextParameters");
```

Spring DSL based configuration of endpoint

```
...
<camel:sslContextParameters
    id="sslContextParameters">
    <camel:trustManagers>
        <camel:keyStore
            resource="/users/home/server/truststore.jks"
            password="keystorePassword"/>
        </camel:keyManagers>
    </camel:trustManagers>
</camel:sslContextParameters>...
...
<to uri="ircs://camel-prd-user@server:6669/#camel-test?nickname=camel-prd&password=password&sslContextParameters=#sslContextParameters"/>...
```

Using the legacy basic configuration options

You can also connect to an SSL enabled IRC server, as follows:

```
ircs:host[:port]/#room?username=user&password=pass
```

By default, the IRC transport uses [SSLDefaultTrustManager](#). If you need to provide your own custom trust manager, use the `trustManager` parameter as follows:

```
ircs:host[:port]/#room?username=user&password=pass&trustManager=#referenceToMyTrustManagerBean
```

Using keys

Available as of Camel 2.2

Some irc rooms requires you to provide a key to be able to join that channel. The key is just a secret word.

For example we join 3 channels where as only channel 1 and 3 uses a key.

```
irc:nick@irc.server.org?channels=#chan1,#chan2,#chan3&keys=chan1Key,,chan3key
```

See Also

- [Configuring Camel](#)
- [Component](#)
- [Endpoint](#)
- [Getting Started](#)