

```
jar xf okvclient.jar ssl
```

11. Copy the Oracle Key Vault support files (the `okvclient.ora` file and the `ssl` directory) into the configuration directory.
12. (Optional) If you wish to password-protect the key file, use the instructions in [Password-Protecting the keyring_okv Key File](#).

After completing the preceding procedure, restart the MySQL server. It loads the `keyring_okv` plugin and `keyring_okv` uses the files in its configuration directory to communicate with Oracle Key Vault.

Configuring keyring_okv for Gemalto SafeNet KeySecure Appliance

Gemalto SafeNet KeySecure Appliance uses the KMIP protocol (version 1.1 or 1.2). The `keyring_okv` keyring plugin (which supports KMIP 1.1) can use KeySecure as its KMIP back end for keyring storage.

Use the following procedure to configure `keyring_okv` and KeySecure to work together. The description only summarizes how to interact with KeySecure. For details, consult the section named Add a KMIP Server in the [KeySecure User Guide](#).

1. Create the configuration directory that contains the KeySecure support files, and make sure that the `keyring_okv_conf_dir` system variable is set to name that directory (for details, see [General keyring_okv Configuration](#)).
2. In the configuration directory, create a subdirectory named `ssl` to use for storing the required SSL certificate and key files.
3. In the configuration directory, create a file named `okvclient.ora`. It should have following format:

```
SERVER=host_ip:port_num  
STANDBY_SERVER=host_ip:port_num
```

For example, if KeySecure is running on host 198.51.100.20 and listening on port 9002, the `okvclient.ora` file looks like this:

```
SERVER=198.51.100.20:9002  
STANDBY_SERVER=198.51.100.20:9002
```

4. Connect to the KeySecure Management Console as an administrator with credentials for Certificate Authorities access.
5. Navigate to Security >> Local CAs and create a local certificate authority (CA).
6. Go to Trusted CA Lists. Select Default and click on Properties. Then select Edit for Trusted Certificate Authority List and add the CA just created.
7. Download the CA and save it in the `ssl` directory as a file named `CA.pem`.
8. Navigate to Security >> Certificate Requests and create a certificate. Then you can download a compressed `tar` file containing certificate PEM files.
9. Extract the PEM files from in the downloaded file. For example, if the file name is `csr_w_pk_pkcs8.gz`, decompress and unpack it using this command:

```
tar zxvf csr_w_pk_pkcs8.gz
```

Two files result from the extraction operation: `certificate_request.pem` and `private_key_pkcs8.pem`.