

# Configuring Kerberos Realm

[← Certificate Properties File Realm](#)

[Administering security realms](#)

[Database \(SQL\) Realm →](#)

In Geronimo, you can create your own custom realm type when none of the provided ones fit your environment needs. This requires creating your own implementation of a `org.apache.geronimo.security.realm.providers` class, which implements the `javax.security.auth.spi.LoginModule` interface to acquire credentials, then use the customized realm by choosing the **Other** realm from console.

## Kerberos realm

Starting from Geronimo 2.2.1, Kerberos realm is supported by providing a wrapper class named `org.apache.geronimo.security.realm.providers.KerberosLoginModule` leveraging kerberos protocol implementation of underlying Java platforms.

To create a new security realm, click **Add new security realm** from the **Security Realms** portlet. On the next page, enter **Kerberos\_security\_realm** in the name of **Security Realm:** field and select **Other** from the **Realm type:** dropdown box. Then click **Next** and fill in the following parameters:

- The Value of **Login Module Class** should be `org.apache.geronimo.security.realm.providers.KerberosLoginModule`
- The value of **Configuration options** for IBM Java platform should be:

```
addOnPrincipalName=admin
addOnPrincipalClass=org.apache.geronimo.security.realm.providers.GeronimoGroupPrincipal
krb_debug=true
krb5LoginModuleClass=com.ibm.security.auth.module.Krb5LoginModule
```

- Leave the other values as default.

If you are using Sun Java platform, the value of **Configuration Options** for Kerberos realm should be:

```
addOnPrincipalName=admin
addOnPrincipalClass=org.apache.geronimo.security.realm.providers.GeronimoGroupPrincipal
krb_debug=true
krb5LoginModuleClass=com.sun.security.auth.module.Krb5LoginModule
```