

# Chapter 16: The Kerberos Configuration File:

## krb5.conf

In this chapter we describe the Kerberos configuration file `krb5.conf`.

A `krb5.conf` file must exist in the `/etc` directory on each UNIX node that is running Kerberos. We provide a template for this file in the **krb5conf** product in KITS (under `ftp://ftp.fnal.gov:8021/products/krb5conf/`).

If you install Fermi **kerberos** from KITS using UPS/UPD or RPM for Linux, the **krb5conf** product (and file) gets installed automatically for you. If you obtain Kerberos from another source, you must obtain this file yourself, edit it as necessary, and copy it into the `/etc` directory of your machine.

You may need to update your `krb5.conf` file from time to time as the template in KITS gets updated. New versions are announced on the *kerberos-users@fnal.gov* mailing list.

If you need to change a setting in `krb5.conf` but cannot or don't want to change the file in `/etc`, you can copy `/etc/krb5.conf` to a new file and edit this copy. Then set the environment variable `$KRB5_CONFIG` to the full name of your copy. Your copy will be honored by client programs such as **kinit** or **rlogin**, but not by programs that need a trusted configuration file, e.g., **ksu** and the service daemons.

## 16.1 What does `krb5.conf` Control?

The file consists of several stanzas, each of which controls certain aspects of the installation:

- `[libdefaults]` sets defaults for Kerberos on your system, e.g., default realm, default ticket lifetime
- `[realms]` tells where to find the KDCs for each realm
- `[instancemapping]` maps client principal properly (for things like cron jobs which require a special principal)
- `[domain_realm]` maps domains to realms
- `[logging]` tells Kerberos where and how to log errors

- [appdefaults] lists default settings for outgoing Kerberized network connection applications and for incoming portal mode connections

In section 16.4 *krb5.conf.template* we list the template `krb5.conf` file (current as of November '01) with annotations.

## 16.2 Reinstall krb5conf Using UPD

To reinstall **krb5conf** and thus update your `/etc/krb5.conf` file using UPS/UPD, log in as *root* (or any login id with permissions to write in `/etc`), and run:

```
% upd install krb5conf -G -c
```

Then on all nodes in the cluster (including the original node), run the command:

```
% ups installAsRoot krb5conf
```

Or instead of the **ups installAsRoot** command, after running **upd install**, you may manually set the `SOURCE_FILE` environmental variable to point to the `krb5.conf.template` script:

```
% SOURCE_FILE=/path/to/krb5/ups/krb5.conf.template
```

and then invoke the `install` script

```
% /path/to/krb5/ups/install
```

## 16.3 Obtain krb5conf without Using UPD

If you're not running UPS/UPD, go to

`ftp://ftp.fnal.gov:8021/products/krb5conf/vx_y/NULL/krb5conf_vx_y_NULL.tar` (where `x_y` is `1_9` as of January 2005). Download and untar the file. Look at the top of the `installAsRoot` script for instructions on how to install it without UPS. If you're not running AFS, check to be sure that the `installAsRoot` script changes the following line in `/etc/krb5.conf` to "false":

```
krb5_run_aklog = false
```

## 16.4 krb5.conf.template

For reference, we provide the `krb5.conf.template` file contents for version `v1_9`, with some explanations inserted. If you install the **krb5conf** product using UPD, the necessary name substitutions will be made as part of the installation; otherwise, you need to edit this file manually.

```
[libdefaults]
```

This section sets defaults for Kerberos on your system.

```
ticket_lifetime = 1560
```

There are some implementations of Kerberos that read the above number as seconds, and is equivalent to 26 hours. In MIT-derived code (which Fermi's is), it's read as minutes.

```
default_realm = xMYREALMx
```

The UPD installation process changes `xMYREALMx` to `FNAL.GOV`. (In Kerberos transactions, this `default_realm` is assumed when you mention any principal without its “@REALM” part.)

```
checksum_type = 1
```

```
ccache_type = 3
```

```
default_tgs_etypes = des-cbc-crc
```

```
default_tkt_etypes = des-cbc-crc
```

```
[realms]
```

This section lists the realms, and for each the KDCs, admin server (master KDC), the `default_domain` for converting between Kerberos v4 and Kerberos v5 service names, and principal-to-account name matching info.

If and when we cross-authenticate with some other site, each host that wants to initiate connections *to* the other site will have to list that site's realm information here. (We think it won't be necessary for accepting connections *from* that site.)

```
FNAL.GOV = {
```

```
    kdc = krb-fnal-1.fnal.gov:88
```

```
    kdc = krb-fnal-2.fnal.gov:88
```

```
    kdc = krb-fnal-3.fnal.gov:88
```

```
    kdc = krb-fnal-4.fnal.gov:88
```

```
    kdc = krb-fnal-5.fnal.gov:88
```

```
    kdc = krb-fnal-6.fnal.gov:88
```

```
    admin_server = krb-fnal-admin.fnal.gov
```

```
    master_kdc = krb-fnal-admin.fnal.gov:88
```

```
    default_domain = fnal.gov
```

```
WIN.FNAL.GOV = {
```

```
    kdc = littlebird.win.fnal.gov:88
```

```
    kdc = bigbird.win.fnal.gov:88
```

```
    default_domain = fnal.gov
```

```
    }  
...  
[instancemapping]
```

This deals with the instance portion of a principal (see *principal* in the *Glossary*). The lines that follow instruct Kerberos to strip a trailing `/cron/*` or `/cms/*` portion of the client principal when generating a Kerberos v4 ticket for the service called `afs`.

```
    afs = {  
        cron/* = ""  
        cms/* = ""  
        afs/* = ""  
    }
```

```
[logging]
```

This section tells Kerberos where and how to log errors; through syslog or directly to file.

```
    kdc = SYSLOG:info:local1  
    admin-server = SYSLOG:info:local2  
    default = SYSLOG:err:auth
```

```
[domain_realm]
```

In this section the domains get mapped to the realms. (This determines the realm in which you need to get a service ticket to log into a Kerberized host in a particular domain.) For individual machines in a domain that need to be mapped to a different realm than the domain as a whole, list each machine separately, mapped to the correct realm. Make your changes in the lower part of this section as noted below.

```
    .fnal.gov = FNAL.GOV  
    .cdms-soudan.org = FNAL.GOV  
    .minos-soudan.org = FNAL.GOV  
    .dhcp.fnal.gov = FNAL.GOV  
    ...  
    .cs.ttu.edu = FNAL.GOV  
    ...  
    .harvard.edu = FNAL.GOV  
    ... (other "friends and family" by request)  
# The whole "top half" is replaced during "ups installAsRoot  
krb5conf", so:  
# It would probably be a bad idea to change anything on or above  
this line  
  
# If you need to add any .domains or hosts, put them here  
[domain_realm]  
    mojo.lunet.edu = FNAL.GOV
```

[appdefaults]

This section lists default application settings (ticket attributes, login parameters, etc.). Each of the applications listed may have additional attributes set (e.g., ticket lifetime, and so on). All of these defaults (or nearly all) can be overridden by a command-line flag. The `krb5.conf` file just sets the defaults for the host. (The ftp client does not look for defaults here; the ftpd ticket lifetime set in the file is invoked for CRYPTOCard FTP access and kcron.)

```
default_lifetime = 7d
retain_ccache = false
```

`retain_ccache` determines whether tickets in a user's ticket cache on a particular host get saved (`true`) or destroyed (`false`) when the user closes his session on that host.

```
autologin = true
forward = true
forwardable = true
```

`forward` should in most cases be set to `true`, in order to forward tickets obtained as “forwardable” to remote hosts by default.

```
renewable = true
encrypt = true
krb5_aklog_path = /usr/krb5/bin/aklog
```

The initial list is for common settings. These values are used by all the applications except when an overriding value is listed for a particular application; see below.

```
telnet = {
}
```

Telnet uses the common settings; no overrides.

```
rcp = {
    forward = false
    encrypt = false
    allow_fallback = true
}
```

Whereas `rcp` sets two overrides (the first of which is unnecessary) and one additional parameter.

```
rsh = {
    allow_fallback = true
}
rlogin = {
    allow_fallback = false
}
```

```
login = {
    forwardable = true
    krb5_run_aklog = true
    krb5_get_tickets = true
    krb4_get_tickets = false
    krb4_convert = false
}
```

`login` is invoked by `telnetd` (not `telnet`) and `sshd` (not `ssh`), and may be invoked by the OS for a local (console) login. CRYPTOCard logins use these settings.

```
kinit = {
    forwardable = true
    krb5_run_aklog = true
}

pam = {
    forwardable = true
}

rshd = {
    krb5_run_aklog = true
}

ftpd = {
    krb5_run_aklog = true
    default_lifetime = 10h
}
```