CDAProxy

De TIDXOSWiki

CDAProxy is a proxy that allows clients linked with cdaclient to obtain credentials from CDA Server or stored in the proxy host. It's designed to surpass mobile client limitations, but has features useful also for conventional clients.

CDAProxy features:

- Proxy generate the private key and cert request instead of the client. This has the advantages that consumes less battery on client, is faster and in some limited devices, may use a more secure random number generator. This requires of course that clients only use proxies that they trust. Typical scenarios are proxy running in user organisation (e.g. an enterprise) or a personal CDA proxy running in user PC as a method to sharing user credential between PC and mobile device.
- Connection between proxy and CDA server is encrypted, but connection between client and proxy may use SSL connections or unencrypted connections. The last is useful when connection between client and proxy is secure (e.g. use a WPA session, or a local link USB network).
- Authentication method can be different between proxy-client that between server-proxy. For example, CDA password may be hard to type and CDAProxy instead ask only a PIN to client. This feature is also useful to integrate cdaclient with local enterprise single sign-on architecture: user organisation normally authenticate its users using its own security infraestructure, with its own policy about passwords. A future CDA server with PAM support, may require the use of a PCKS#11 device not supported by the mobile device.
- A organisation may choose to run its own CDAProxy server for privacy reasons. A organisation may authenticate its users, but use mapped accounts with the Virtual Organisation CDA Server.
- May run in any port; this is useful when the port or IP of CDA server is unreachable by the client (e.g. there is a firewall). Network administrators may prefer to run a CDAProxy than open the CDA server port in firewall, because open the port implies allowing arbitrary SSL connections using that port.
- May serve credentials stored in local disk instead of ask to CDA server. This is useful to obtain a credential only once from CDA server and caching while does not expire. Credential is accessible if CDAProxy is, although CDA server is not available.
- Integrates an optional "showmyip" server. This component is not needed by

CDA, but it's useful with other services when client is behind a NAT.

Tabla de contenidos

running CDAProxy

To lauch CDAProxy, simply run *cdaproxy*. CDAProxy admits two parameters, but both are optional:

- *-c <config_file>*: Use config_file as configuration file. If this option is missed, configuration file is */etc/xos/cdaproxy/proxy.conf*
- *-f*: run in the foreground. But default, cdaproxy runs as a daemon in background. This option is useful to debug.

CDAProxy configuration

CDAProxy configuration file has the sintax of a Gnome/KDE *.desktop* file, that is very similar to Windows *.ini* files.

File example_proxy.conf is a example configuration file, with comments about each possible parameter in the file.

Configuration file has a mandatory section [proxy] with the general configuration of the proxy (information about ports, use or not of SSL, proxy server certificate...).

Configuration file has a section for each user it support. For example, if proxy accepts connections from user "bob", must exist a section [bob] in configuration file. A optional section [all_users_defaults] contains parameters that are inhered for each user section.

Configuration file may have a special user section: [other_users]. If this section is present, the configuration is applied to any user to access proxy that has not its own user section. If this section is missing, proxy denies access to any username that has not a user section.

Configuration of [proxy] section

- proxyport: this parameter is mandatory. It's the TCP port where the proxy listens incoming connections.
- usessl: if this parameter is true (default is false), the proxy server socket use TLS/SSL. If usessl=true, parameter proxy_pem is required.
- proxy_pem: the file path of a PEM file with the private key and certificate to use in TLS/SSL proxy server socket. This file may be generated using *cdaproxy_createproxycert.sh* socket. This parameter is ignored if usessl=false, otherwise is required.
- showmyipport: the TCP port used to listen "show my IP" requests. If this parameter is not present, this optional server (not needed by CDA functionality) is disabled.

Configuration of user sections

For each user, credential may be obtained from one of this ways:

- local: the credential is stored in local filesystem
- cdaserver: the credential is obtained from a CDA server

Each user section must include a parameter "type" indicating if credential source is "local" or "cdaserver".

type=local

This parameters are available when user section is of type "local":

- credential_file: the path of a PEM file containing the RSA private key and X.509 certificate to return user as credential. If RSA key is encrypted, user is authenticated asking him the symmetric key needed to decrypt RSA key. This parameter is required.
- proxy_password: the password used to authenticate user. This parameter is optional.

type=cdaserver

This parameters are available when user section is of type "cdaserver":

- cdahost: the hostname of the CDA server. This parameter is required.
- cdaport: the TCP port of the CDA server. This parameter is required.
- servercert: The path file with the X.509 server cert chain. If this parameter is present, remote server certificate is verified using this file. If this parameter is missing, proxy does not verify the certificate of CDA server and application may suffer a man-in-middle attack.
- cacert: The path file with the X.509 of the CDAroot that signs user credential. This parameter is optional: if present, proxy verifies that user credential is correctly signed.
- proxy_password: The password to authenticate user. If missing, proxy don't authenticate user: instead of it pass password to CDA server to authenticate it.
- cda_user: This parameter is optional: if specified, uses this username to connect with CDA server instead of username used to connect with proxy.
- cda_password: this parameter is optional: if specified, uses this password to authenticate with CDA, instead of password provided by user. This allows to use a different password in CDAProxy to authenticate users that the password in CDA server. This is useful, for example, to use a short, easy to type in a smartphone keyboard, password.

Utilities

- dump_server_chain: This program receives as parameter a TLS/SSL server address in form "<hostname:port>". Connect with the specified server and dump the server chain certificates to standard output. Usefulness of this utility is to obtain the CDA server rootCA certificate to include it in the file referenced by parameter "servercert".
- cdaproxy_createproxycert.sh: This scripts generate a RSA private key and a self-signed certificate valid for 365 days . Content is saved in server.pem. This file may be used as the private key and certificate of the SSL proxy server socket (param proxypem in configuration file).
- cdaproxy_cdaproxy_createtestcred.sh: This scripts generate a test credential, with a RSA private key encrypted.

Future works

- PAM support
- libxos-getcred support (this allows remote applications to use proxy modules and to interact with the local user of the cdaproxy, that may be the enterprise security responsible).
- XMPP (Jabber) support as alternative to HTTP. Users may be authenticated without password, it they are members of a special group in the roster.

Appendix: CDA Proxy configuration file example

```
[proxy]

# proxy server use SSL in connections with clients?
usessl=false

# TCP port where the proxy listen connections
proxyport=8082

# File with the RSA key and certificate of the proxy server. If you want to
# generate a self-signed certificate, run cdaproxy_createproxycert.sh script.
# This parameter is mandatory if proxy server use SSL.
proxypem=server.pem

# Optionally, cdaproxy may integrate also a showmyip server
showmemyipport=8081

## For each username accepted by the proxy, you must include a section named
## with the username. In this example, we included the users "bob" and "mary"
##
## all_users_default section contains values that are inherited by all users
## section. Users sections my overwrite this parameters.
##
## Optionally, you may include a other_users section. If this section is
## present, it apply to any user not listed explicitly. If section is missing
## only users with a section in configuration file are authorised to use proxy.
##
## For each user, it's possible to load the credential from a local file
## (attribute type=local) or request it to a CDA server (attribute
## type=cdaserver). In both cases, attribute "proxy_password" is optional: if
## present the proxy authenticates user. If omited and type=cdaserver,
## only cdaserver authenticates. If omited and type=local, user password is used
## to decrypt the RSA private key.
##
## If type=local, parameter credential_file with the path of a PEM file with the
## RSA key and public certificate is mandatory.
## If type=cdaserver, parameters cdahost and cdaport are mandatory;
## proxy_password, cda_user, cda_password, servercert and cacert are optional.

[all_users_default]

# Certificate of the server or of the CA that sign the server certificate.
# It this parameter is missing, don't check the server certificate (this is
# insecure but useful for debugging).
#servercert=cdacert.pem

# Root certificate of the CA that signs the obtained certificate. If this
# parameter is missing, don't check the obtained certificate.
#cacert=cacert.pem

# The CDA server hostname
cdahost=xtreemos-b.esc.rl.ac.uk

# The CDA server port
cdaport=6730

credential_file=credential.pem

[bob]
# credential is inside credential.pem; the RSA key is crypted with the password
# of "bob". Proxy check password trying decrypting the RSA key.
type=local

[mary]
type=cdaserver
# password to authenticate client in our proxy (it missing, authenticate only
```