

How to build a client for renewing user's proxy

- Be sure that the host certificate DN is in the list of trusted hosts in the myproxy server. Not all Grid Services are allowed to renew user proxies. In order to check whether your renewing agent is trusted you can use the Information System like in this example: `ldapssearch -p 2170 -h myproxy.cern.ch -x -LLL -b "mdu-vo-name=resource,o=grid" | grep -i volhcb`
- The user has to register the proxy into myproxy server using the command: `myproxy-init -s myproxy.cern.ch -d -n` The `-n` option means no password required to the rewer service and `-d` option informs that the DN can be used for unequivocally identify the proxy to be renewed among the ones registered in myproxy server.
- Server-renewer side the special user that does the renewal (let's refer to it as *edguser*) has to dress the host credentials: this is the usual trick
 - ◆ `export RENEWER_DIR=/tmp/proxyrenewal`
 - ◆ As root run: `$GLOBUS_LOCATION/bin/grid-proxy-init -cert $X509_USER_CERT -key $X509_USER_KEY -valid 500:00 -out $RENEWER_DIR/renewd-proxy.pem`
 - ◆ As root: `chown dirac:dirac $RENEWER_DIR/renewd-proxy.pem`
 - ◆ As root: `su -l dirac`
- As *dirac*
 - ◆ `export RENEWER_DIR=/tmp/proxyrenewal`
 - ◆ `export X509_USER_PROXY=$RENEWER_DIR/renewd-proxy.pem`
 - ◆ `/home/dirac/renew_proxy dirac`
where the script that must be run looks like

```
> cat /home/edguser/renew_proxy
```

```
#!/bin/bash
```

```
VOADMIN="edguser"
```

```
VOGROUP=`id -gn ${VOADMIN}`
```

```
GLOBUS_LOCATION=${GLOBUS_LOCATION:-/opt/globus}
```

```
export MYPROXY_SERVER=${MYPROXY_SERVER:-myproxy.cern.ch}
```

```
LCG_LOCATION=${LCG_LOCATION:-/opt/lcg}
```

```
RENEWER_DIR=/tmp/proxyrenewal
```

```
RENEWER_PROXY_REPOSITORY=$RENEWER_DIR/repository #directory where all proxies to be renewed are stored
```

```
RENEWER_LOGDIR=$RENEWER_DIR/log #directory where all logs of operations are booked
```

```
export X509_USER_PROXY=$RENEWER_DIR/renewd-proxy.pem #location of the host-signed proxy at the point c.
```

```
if [ ! -d ${RENEWER_LOGDIR} ] ; then
```

```
mkdir -p ${RENEWER_LOGDIR} || exit -1 chown ${VOADMIN}.${VOGROUP} ${RENEWER_LOGDIR}
```

```
chmod 0700 ${RENEWER_LOGDIR}
```

```
fi
```

ConfigureAProxyRenewalClient < LCG < TWiki

```
if [ ! -d ${RENEWER_PROXY_REPOSITORY} ]; then

mkdir -p ${RENEWER_PROXY_REPOSITORY} || do_failure "Cannot create repository dir
${RENEWER_PROXY_REPOSITORY}"

chown ${VOADMIN}.${VOGROUP} ${RENEWER_PROXY_REPOSITORY} chmod 0700
${RENEWER_PROXY_REPOSITORY}

fi

for CERT in `ls $RENEWER_PROXY_REPOSITORY`; do

#it parses indistinguishably all proxies in the repository. Smarter conditions (like expiration time of
each could be aplied at this level)

PROXY_DN=`${GLOBUS_LOCATION}/bin/grid-proxy-info -f
${RENEWER_PROXY_REPOSITORY}/${CERT} -subject` echo $PROXY_DN

if [ $? -ne 0 ]; then

echo "`date +%D %H:%M:%S` : Error. Impossible to get proxy information." >>
${RENEWER_LOGDIR}/events.log

echo "`date +%D %H:%M:%S` : ... $PROXY_DN" >> ${RENEWER_LOGDIR}/events.log

continue

fi

TMP_PROXY=`mktemp`

${GLOBUS_LOCATION}/bin/myproxy-get-delegation -a ${RENEWER_PROXY_REPOSITORY}/${CERT} -d
-o $TMP_PROXY #this is the core.....

if [ $? -eq 0 ]; then

mv $TMP_PROXY ${RENEWER_PROXY_REPOSITORY}/${CERT} #in case of success (why not?) it
overwrites old proxy with the renewed one. It will be valid for 12 hours

echo "`date +%D %H:%M:%S` : Proxy for DN\"${PROXY_DN}\" successfully renewed " >>
${RENEWER_LOGDIR}/events.log

else

echo "`date +%D %H:%M:%S` : ERROR. Unable to renew proxy\"${PROXY_DN}\"" >>
${RENEWER_LOGDIR}/events.log

fi

_done_
```

This topic: LCG > ConfigureAProxyRenewalClient

Topic revision: r3 - 2008-06-05 - RobertoSantinel



Copyright &© 2008-2020 by the contributing authors. All material on this collaboration platform is the property of the contributing authors.
or Ideas, requests, problems regarding TWiki? use [Discourse](#) or [Send feedback](#)