Using the **MyProxy**
Online Credential Repository

Jim Basney
National Center for Supercomputing Applications
University of Illinois
jbasney@ncsa.uiuc.edu
What is MyProxy?

- Independent Globus Toolkit add-on since 2000
  - To be included in Globus Toolkit 4.0
- A service for securing private keys
  - Keys stored encrypted with user-chosen password
  - Keys never leave the MyProxy server
- A service for retrieving proxy credentials
- A commonly-used service for grid portal security
  - Integrated with OGCE, GridSphere, and GridPort
PKI Overview

- **Public Key Cryptography**
  - Sign with private key, verify signature with public key
  - Encrypt with public key, decrypt with private key

- **Key Distribution**
  - Who does a public key belong to?
  - Certification Authority (CA) verifies user’s identity and signs certificate
  - Certificate is a document that binds the user’s identity to a public key

- **Authentication**
  - Signature \[ h \ ( \text{random}, \ldots \) \]
Proxy Credentials

- RFC 3820: Proxy Certificate Profile
- Associate a new private key and certificate with existing credentials
- Short-lived, unencrypted credentials for multiple authentications in a session
  - Restricted lifetime in certificate limits vulnerability of unencrypted key
- Credential delegation (forwarding) without transferring private keys

Diagram:

- CA signs
- User signs
- Proxy A signs
- Proxy B
Proxy Delegation

Delegator

1. Generate new key pair
2. Proxy certificate request
3. Sign new proxy certificate
4. Proxy

Delegatee

1. Generate new key pair

Proxy

http://myproxy.ncsa.uiuc.edu/
MyProxy System Architecture

MyProxy client

Store proxy

Retrieve proxy

MyProxy server

Proxy delegation over private TLS channel

Credential repository

GlobusWORLD 2005
MyProxy: Credential Mobility

Obtain certificate

tg-login.ncsa.teragrid.org

ca.ncsa.uiuc.edu

Store proxy

myproxy.teragrid.org

Retrieve proxy

tg-login.caltech.teragrid.org

tg-login.sdsc.teragrid.org

tg-login.uc.teragrid.org

Obtain certificate

tg-login.ncsa.teragrid.org

ca.ncsa.uiuc.edu

Store proxy

myproxy.teragrid.org

Retrieve proxy

tg-login.caltech.teragrid.org

tg-login.sdsc.teragrid.org

tg-login.uc.teragrid.org
MyProxy and Grid Portals
MyProxy: User Registration

1. Request account
   - Set username/password

2. Obtain user certificate
   - Certificate authority
   - Load user’s credentials

3. Retrieve proxy
   - MyProxy server

4. Login with username/password
   - Grid portal
   - MyProxy server

PURSE: Portal-based User Registration Service

GlobusWORLD 2005  http://myproxy.ncsa.uiuc.edu/
MyProxy Security

- Keys encrypted with user-chosen passwords
  - Server enforces password quality
  - Passwords are not stored
- Dedicated server less vulnerable than desktop and general-purpose systems
  - Professionally managed, monitored, locked down
- Users retrieve short-lived credentials
  - Generating new proxy keys for every session
- All server operations logged to syslog
- Caveat: Private key database is an attack target
  - Compare with status quo
Hardware-Secured MyProxy

- Protect keys in tamper-resistant cryptographic hardware

GlobusWORLD 2003 Flashback

Who Holds The Keys?

- **Viewpoint #1**: End entities should have sole possession of their long-term keys
  - Administrator access to end entity keys voids non-repudiation
- **Viewpoint #2**: End entities can’t be trusted to secure their long-term keys
  - Centralized key distribution enforces password policies and credential lifetime limits
- **Will this issue hinder cross-site collaboration?**
Credential Renewal

- Long-lived jobs or services need credentials
  - Task lifetime is difficult to predict
- Don’t want to delegate long-lived credentials
  - Fear of compromise
- Instead, renew credentials as needed during the job’s lifetime
  - Renewal service provides a single point of monitoring and control
- Renewal policy can be modified at any time
  - Disable renewals if compromise is detected or suspected
  - Disable renewals when jobs complete
MyProxy: Credential Renewal

Submit job
Condor-G
Submit job
Refresh proxy
Globus gatekeeper
Submit job
Fetch proxy
MyProxy server
MyProxy Installation (Unix)

- Included in GT 4.0
- As an add-on component to GT 3.x
  
  
  $ gpt-build myproxy*.tar.gz <flavor>

- Set $MYPROXY_SERVER environment variable to myproxy-server hostname
  
  
  $ export MYPROXY_SERVER=myproxy.ncsa.uiuc.edu

- Set Globus Toolkit environment
  
  
  $ . $GLOBUS_LOCATION/etc/globus-user-env.sh

- Client installation/configuration complete!
MyProxy CoG Clients

- Commodity Grid (CoG) Kits
  - Provide portable (Java and Python) MyProxy client tools & APIs
  - Windows support

- For more information:
  - http://www.cogkit.org/
MyProxy Commands

- **myproxy-init**: store proxy
- **myproxy-get-delegation**: retrieve proxy
- **myproxy-info**: query stored credentials
- **myproxy-destroy**: remove credential
- **myproxy-change-pass-phrase**: change password encrypting private key
MyProxy Server Administration

- Install server certificate and CA certificate(s)
- Configure `/etc/myproxy-server.config` policy
  - Template provided with examples
- Optionally:
  - Configure password quality enforcement
  - Install cron script to delete expired credentials
- Install boot script and start server
  - Example boot script provided
- Use `myproxy-admin` commands to manage server
  - Reset passwords, query repository, lock credentials
MyProxy Server Policies

- Who can store credentials?
  - Restrict to specific users or CAs
  - Restrict to administrator only
- Who can retrieve credentials?
  - Allow anyone with correct password
  - Allow only trusted services / portals
- Maximum lifetime of retrieved credentials

server-wide and per-credential
MyProxy and SASL

- MyProxy supports additional authentication mechanisms via SASL (RFC 2222)
- One Time Passwords (SASL PLAIN with PAM)
  - Protect against stolen passwords
  - Hardware token generates OTP
  - Authenticate with OTP plus MyProxy password
  - Tested with CryptoCard tokens
- Kerberos (SASL GSSAPI)
  - Authenticate with Kerberos ticket plus MyProxy password
Related Work

- **GT4 Delegation Service**
  - Protocol based on WS-Trust and WSRF

- **SACRED (RFC 3767) Credential Repository**
  - http://sacred.sf.net/

- **Kerberized Online CA (KX.509/KCA)**
  - Kerberos -> PKI

- **PKINIT for Heimdal Kerberos**
  - PKI -> Kerberos
GridLogon

- Work in progress
- Inspired by Peter Gutmann’s PKIBoot
  - “Plug-and-Play PKI: A PKI your Mother can Use”
- Password-based authentication to initialize user’s security environment
  - Install identity/attribute/authorization credentials
  - Install CA certificates and CRLs
  - Install additional security configurations
MyProxy Community

- myproxy-users@ncsa.uiuc.edu mailing list
- Bug tracking: http://bugzilla.ncsa.uiuc.edu/
- Anonymous CVS access: pserver:anonymous@cvs.ncsa.uiuc.edu:/CVS/myproxy
- Contributions welcome!
  - Feature requests, bug reports, patches, etc.
Thank you!

Questions/Comments?

Contact:
jbasney@ncsa.uiuc.edu