Managing Credentials with MyProxy

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

http://myproxy.ncsa.uiuc.edu/
What is MyProxy?

- A service for managing X.509 PKI credentials
  - A credential repository and certificate authority
- An Online Credential Repository
  - Issues short-lived X.509 Proxy Certificates
  - Long-lived private keys never leave the server
- An Online Certificate Authority
  - Issues short-lived X.509 End Entity Certificates
- Supporting multiple authentication methods
  - Passphrase, Certificate, PAM, SASL, Kerberos
- Open Source Software
  - Included in Globus Toolkit 4.0 and CoG Kits
  - C, Java, Python, and Perl clients available

http://myproxy.ncsa.uiuc.edu/
MyProxy Logon

- Authenticate to retrieve PKI credentials
  - End Entity or Proxy Certificate
  - Trusted CA Certificates
  - Certificate Revocation Lists (CRLs)
- MyProxy maintains the user’s PKI context
  - Users don’t need to manage long-lived credentials
  - Enables server-side monitoring and policy enforcement (ex. passphrase quality checks)
  - CA certificates & CRLs updated automatically at login

http://myproxy.ncsa.uiuc.edu/
MyProxy Authentication

- Key Passphrase
- X.509 Certificate
  - Used for credential renewal
- Pluggable Authentication Modules (PAM)
  - Kerberos password
  - One Time Password (OTP)
  - Lightweight Directory Access Protocol (LDAP) password
- Simple Authentication and Security Layer (SASL)
  - Kerberos ticket (SASL GSSAPI)

http://myproxy.ncsa.uiuc.edu/
MyProxy Online Credential Repository

- **Stores X.509 End Entity and Proxy credentials**
  - Private keys encrypted with user-chosen passphrases
  - Credentials may be stored directly or via proxy delegation
  - Users can store multiple credentials from different CAs

- **Access to credentials controlled by user and administrator policies**
  - Set authentication requirements
  - Control whether credentials can be retrieved directly or if only proxy delegation is allowed
  - Restrict lifetime of retrieved proxy credentials

- **Can be deployed for a single user, a site, a virtual organization, a resource provider, a CA, etc.**

http://myproxy.ncsa.uiuc.edu/
MyProxy Online Certificate Authority

- Issues short-lived X.509 End Entity Certificates
  - Leverages MyProxy authentication mechanisms
  - Compatible with existing MyProxy clients
- Ties in to site authentication and accounting
  - Using PAM and/or Kerberos authentication
  - “Gridmap” file maps username to certificate subject
    - LDAP support under development
- Avoid need for long-lived user keys
- Server can function as both CA and repository
  - Issues certificate if no credentials for user are stored

http://myproxy.ncsa.uiuc.edu/
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 $[h(\text{random}, \ldots)]$
PKI Enrollment

1. Generate new key pair
2. Certificate request
3. Sign new end entity certificate
4. User
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

http://myproxy.ncsa.uiuc.edu/
Proxy Delegation

1. **Delegator**
   - Generate new key pair
   - Sign new proxy certificate

2. **Proxy certificate request**

3. **Delegatee**
   - Generate new key pair
   - Proxy

4. Proxy

http://myproxy.ncsa.uiuc.edu/
MyProxy Repository

MyProxy client

Store proxy

Retrieve proxy

MyProxy server

Proxy delegation over private TLS channel

Credential repository

http://myproxy.ncsa.uiuc.edu/
MyProxy Certificate Authority

MyProxy client

Retrieve certificate

MyProxy server

Private TLS channel

PAM

Site Authentication Service

http://myproxy.ncsa.uiuc.edu/
MyProxy: Credential Mobility

Obtain certificate

ca.ncsa.uiuc.edu

Store proxy

myproxy.teragrid.org

Retrieval proxy

http://myproxy.ncsa.uiuc.edu/
MyProxy and Grid Portals

http://myproxy.ncsa.uiuc.edu/
User Registration Portals

**PURSE:**
Portal-based User Registration Service

**GAMA:**
Grid Account Management Architecture

http://myproxy.ncsa.uiuc.edu/
MyProxy: Key Upload/Download

- Store and retrieve keys and certificates directly over the network
  - Encrypted keys transferred over SSL/TLS encrypted channel
  - In contrast to using proxy delegation
- Allows storing end-entity credentials
- Key retrieval must be explicitly enabled by server administrator and key owner

http://myproxy.ncsa.uiuc.edu/
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


http://myproxy.ncsa.uiuc.edu/
MyProxy and Pubcookie

- Combine web and grid single sign-on
  - Authenticate to MyProxy with Pubcookie granting cookie


http://myproxy.ncsa.uiuc.edu/
Example: TeraGrid User Portal

- Use TeraGrid-wide Kerberos username and password for portal authentication
  - Obtain PKI credentials for resource access across TeraGrid sites via portal & externally
- Plan to use MyProxy CA with Kerberos PAM authentication
  - Leverage existing NCSA Online CA

http://myproxy.ncsa.uiuc.edu/
Example: LTER Grid Pilot Study

- Build a portal for environmental acoustics analysis
- Leverage existing LDAP usernames and passwords for portal authentication
  - Obtain PKI credentials for job submission and data transfer
  - Using MyProxy PAM LDAP authentication

http://myproxy.ncsa.uiuc.edu/
Example: NERSC OTP PKI

- Address usability issues for One Time Passwords
  - Obtain session credentials using OTP authentication
- Prototyping MyProxy CA with PAM Radius authentication
  - ESnet Radius Authentication Fabric federates OTP authentication across sites

National Energy Research Scientific Computing Center

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


http://myproxy.ncsa.uiuc.edu/
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

http://myproxy.ncsa.uiuc.edu/
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

http://myproxy.ncsa.uiuc.edu/
MyProxy Server Replication

- **Primary/Secondary model (like Kerberos)**
  - If primary is down, fail-over to secondary for credential retrieval
  - Store, delete, and change passphrase on primary only
  - Client-side fail-over under development

- **Simple configuration**
  - Run myproxy-replicate via cron
  - Alternatively, use rsync over ssh

http://myproxy.ncsa.uiuc.edu/
Related Work

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

- **UVA CredEx**
  - WS-Trust credential exchange service

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

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

- **Kerberos PKINIT**
  - PKI -> Kerberos

[http://myproxy.ncsa.uiuc.edu/](http://myproxy.ncsa.uiuc.edu/)
MyProxy Community

- MyProxy is an open source, community project
  - Many contributions from outside NCSA
- 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.
  - Please report your experiences

http://myproxy.ncsa.uiuc.edu/
Thank you!

Questions/Comments?

Contact:
jbasney@ncsa.uiuc.edu