the globus alliance
www.globus.org

NCSA

Using the **MyProxy** Online Credential Repository

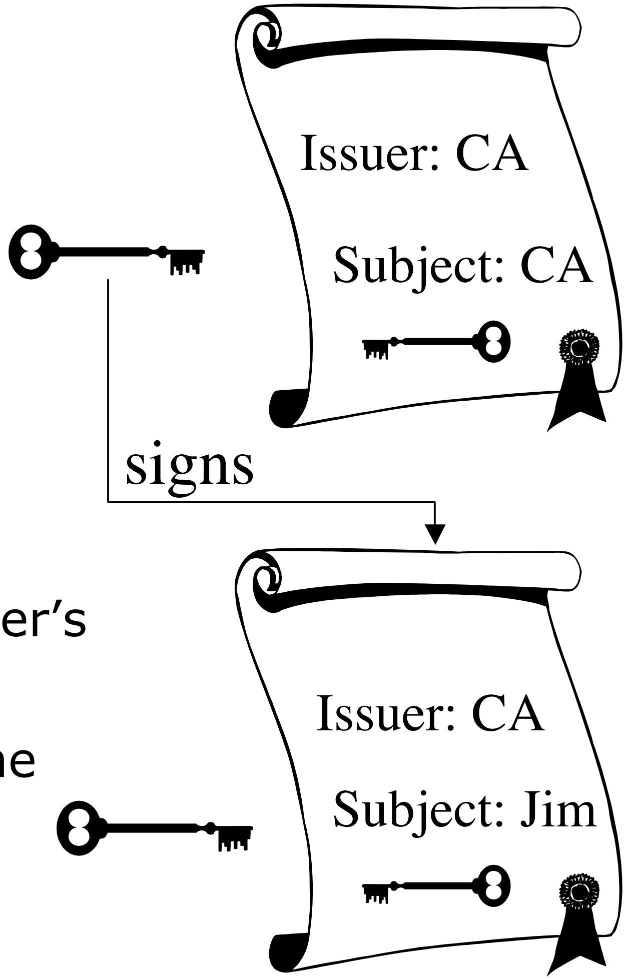
Jim Basney
National Center for Supercomputing Applications
University of Illinois
jbasney@ncsa.uiuc.edu

What is MyProxy?

- Independent Globus Toolkit add-on since 2000
  - 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

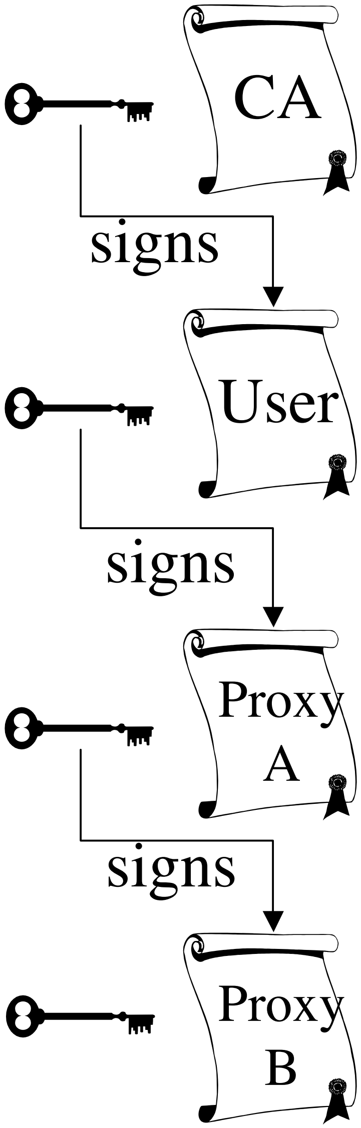
PKI Overview

- 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 (random, …)]

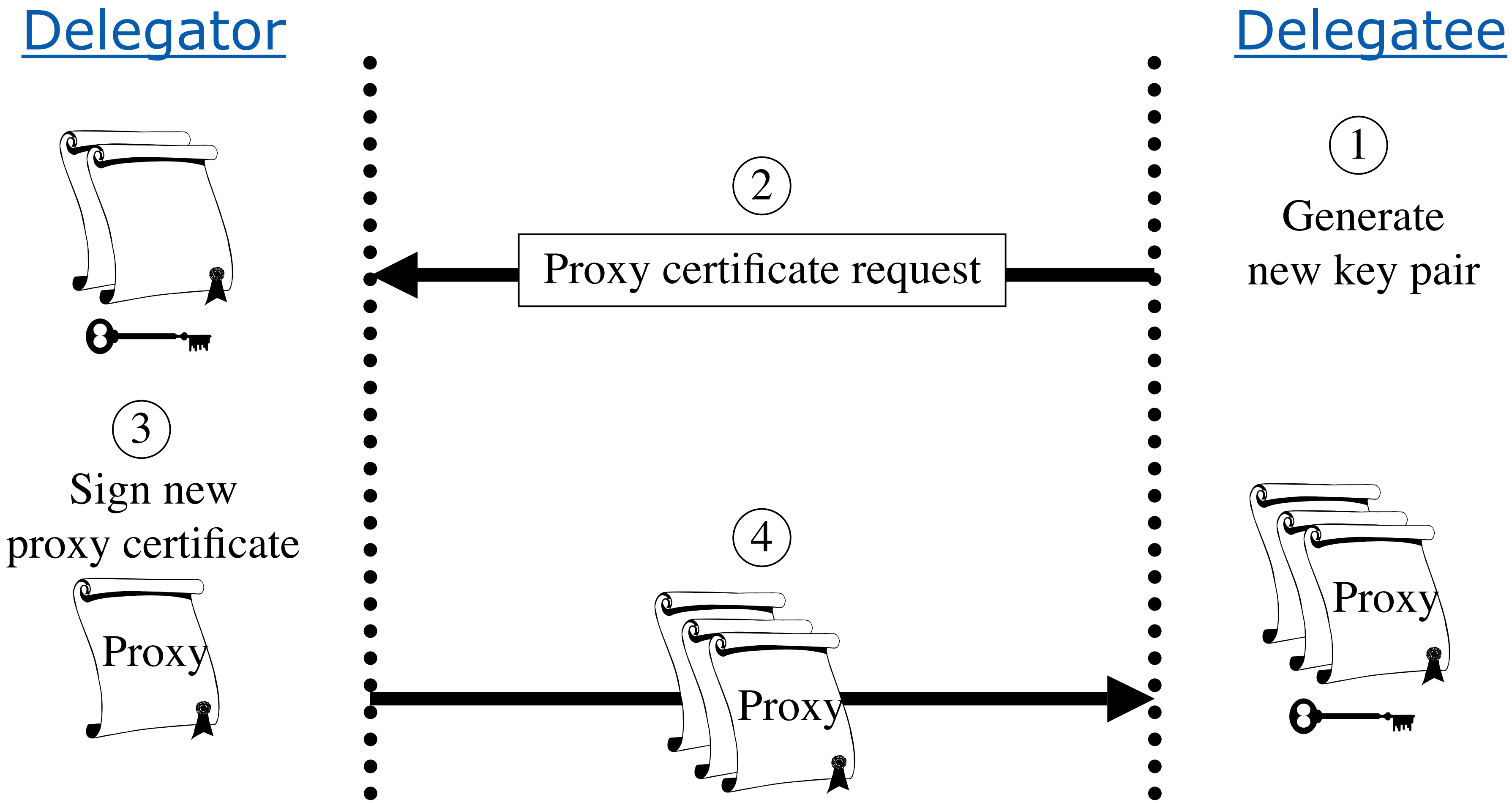


Proxy Credentials

- 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



Proxy Delegation



Delegator

Delegatee

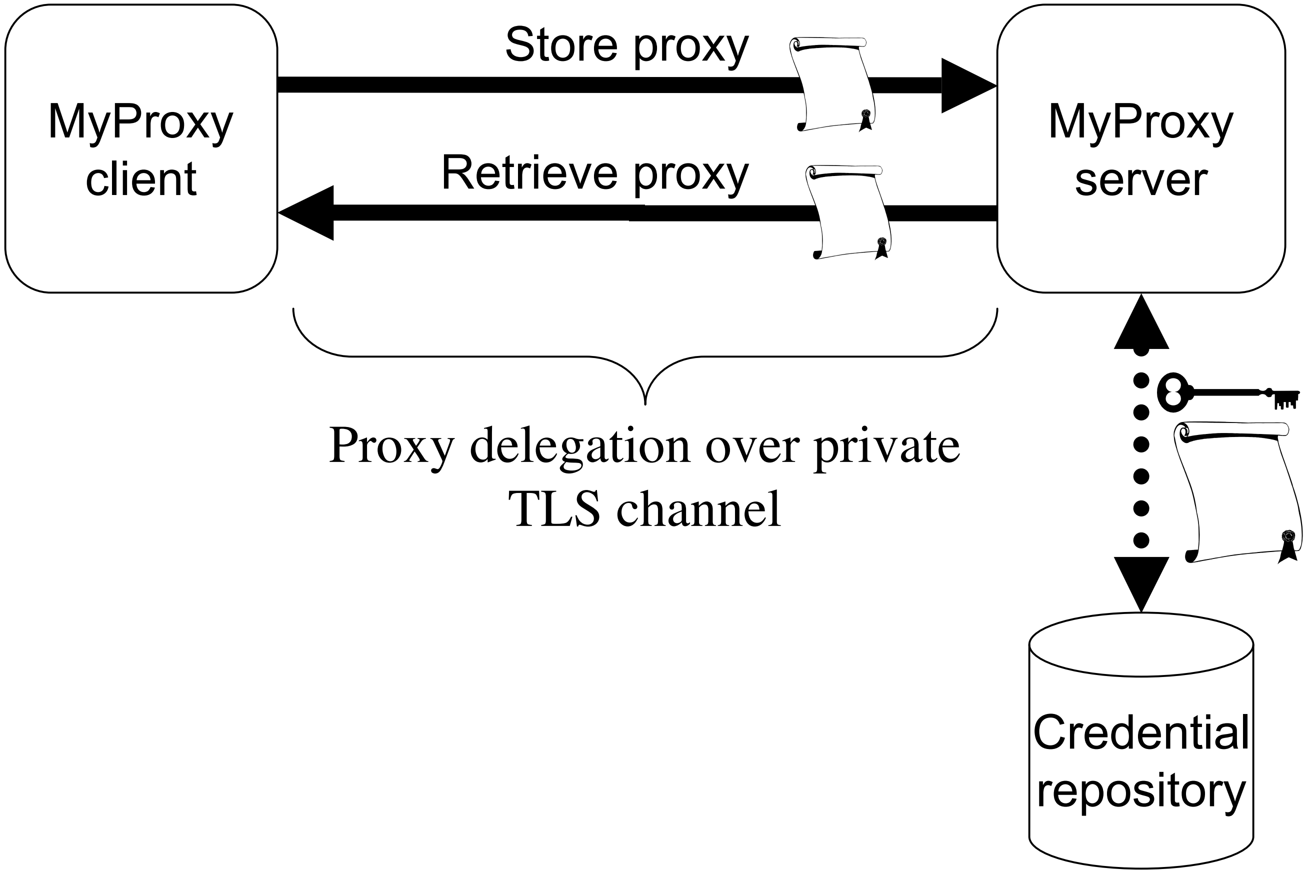
1 Generate new key pair

2 Proxy certificate request

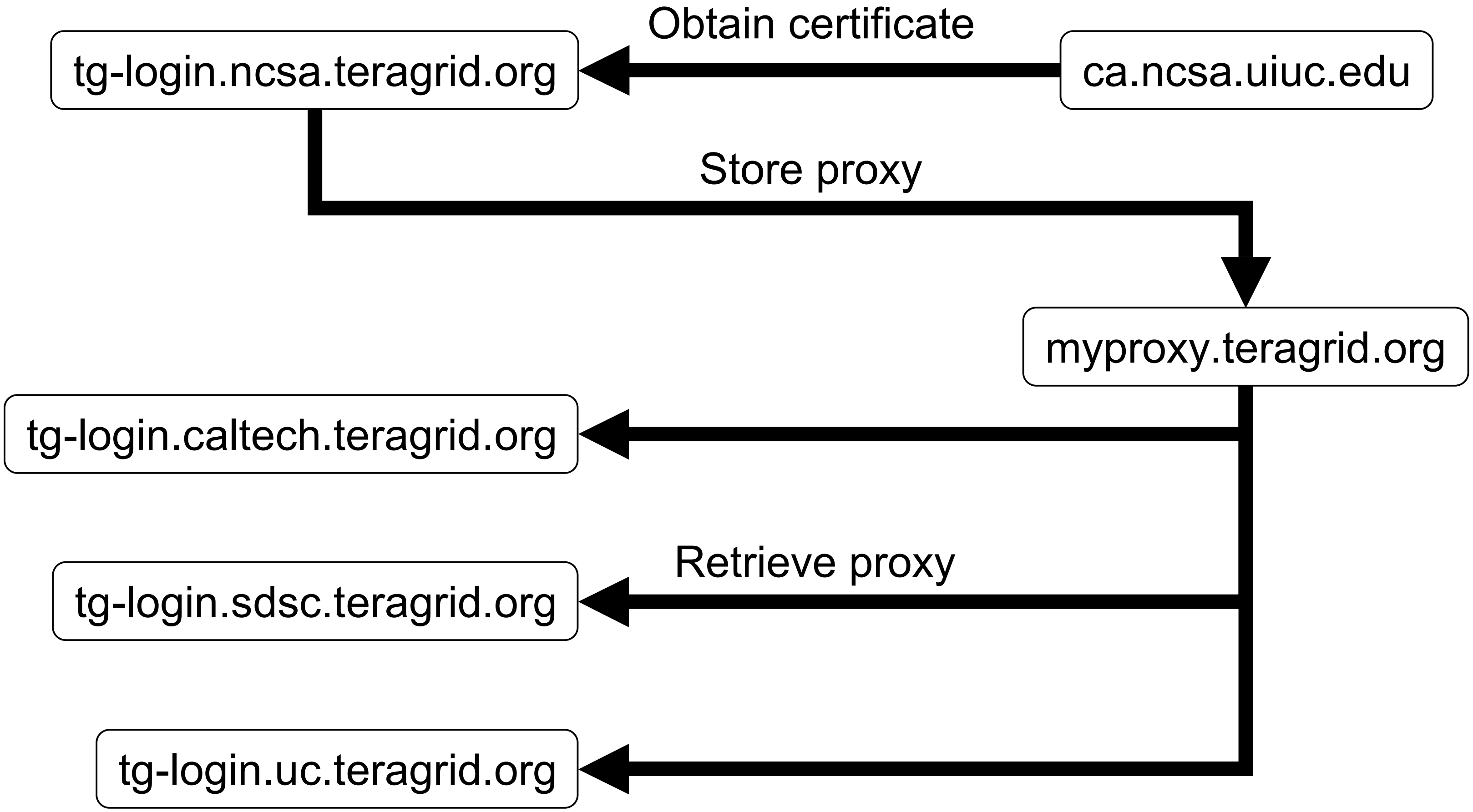
3 Sign new proxy certificate

4

MyProxy System Architecture



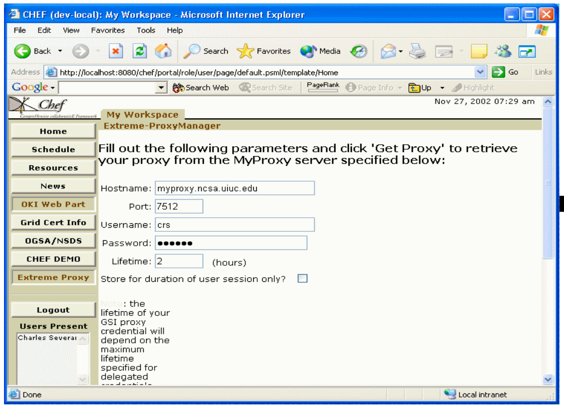
MyProxy: Credential Mobility



the globus alliance
www.globus.org

NCSA

MyProxy and Grid Portals



http://localhost:8080/chef/portal/role/user/page/default.psml/template/Home

Nov 27, 2002 07:29 am

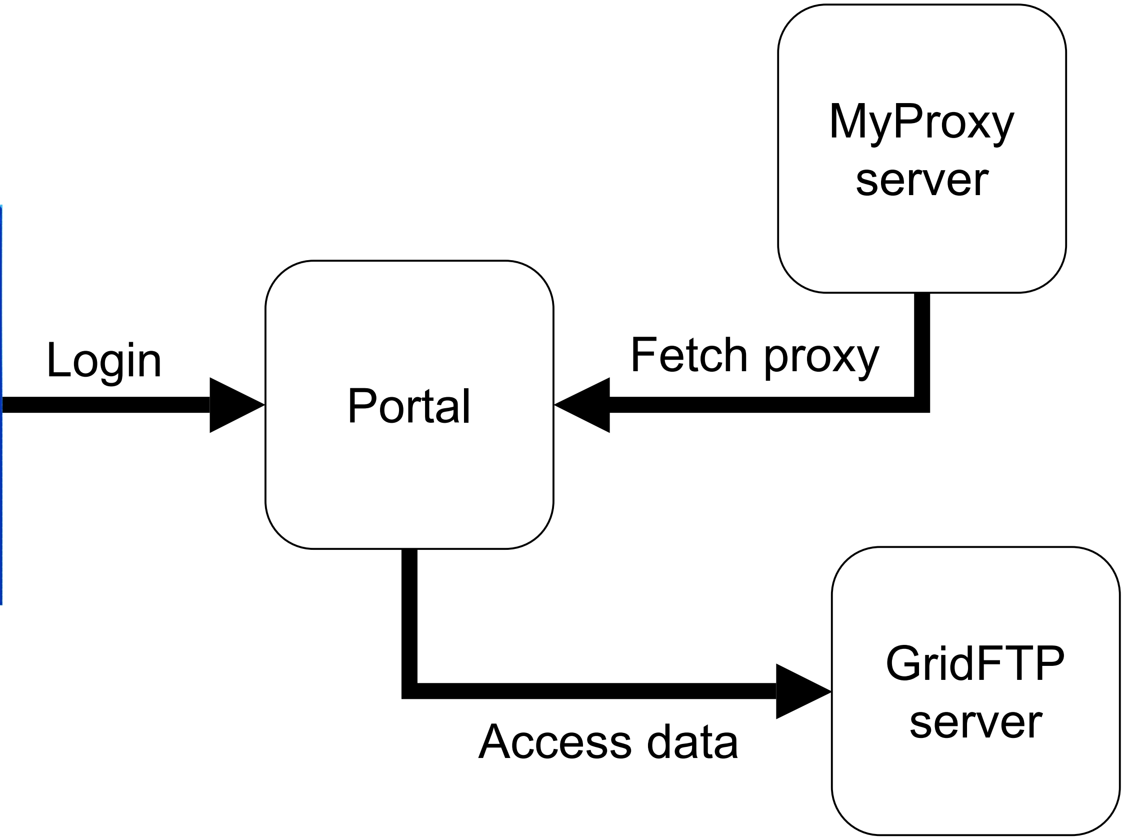
Extreme-ProxyManager

Fill out the following parameters and click 'Get Proxy' to retrieve your proxy from the MyProxy server specified below:

Hostname: myproxy.ncsa.uiuc.edu
Port: 7512
Username: crs
Password: ●●●●●●
Lifetime: 2 (hours)
Store for duration of user session only? □

Note: the lifetime of your GSI proxy credential will depend on the maximum lifetime specified for delegated

Charles Severa

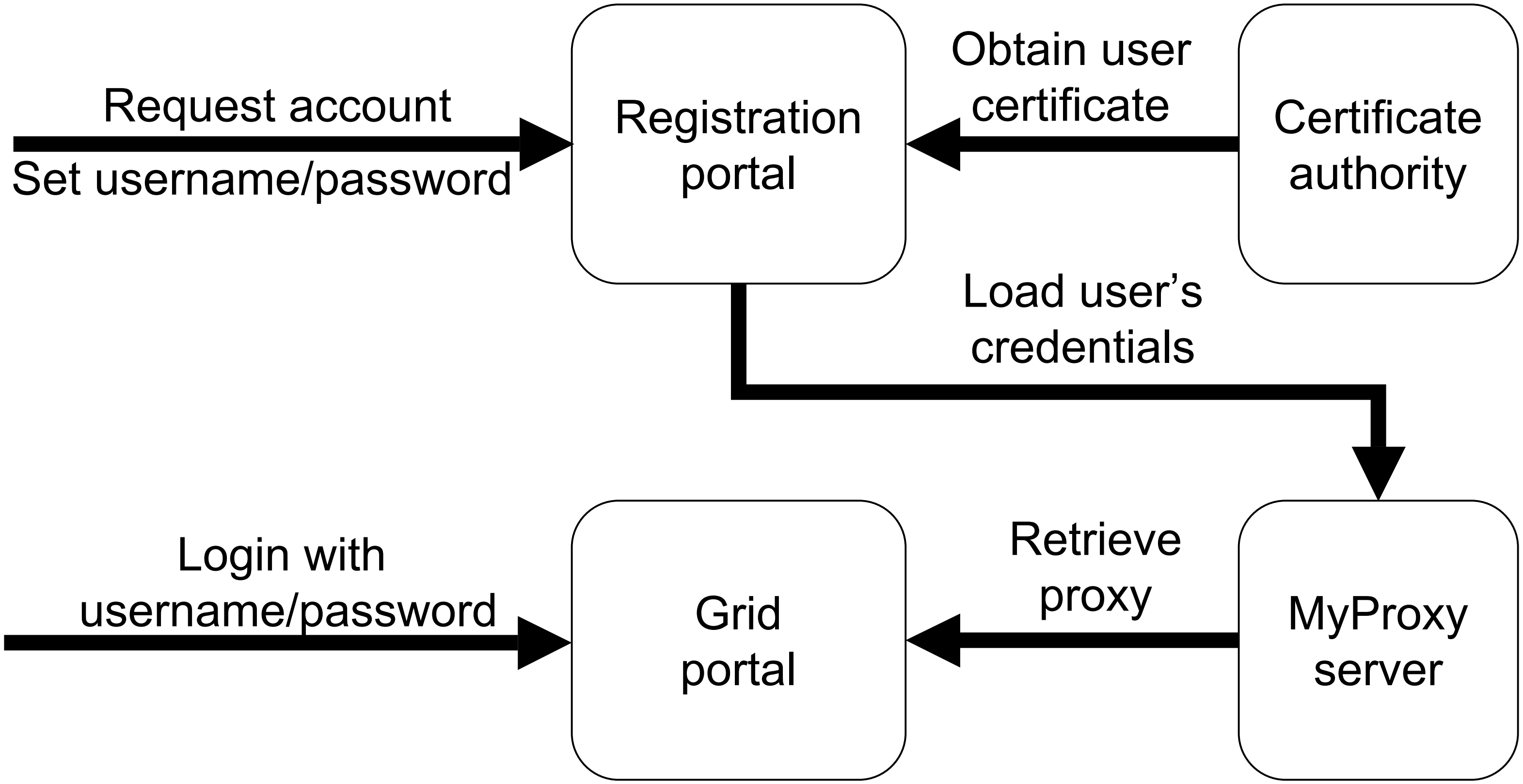


gridsphere portal framework
open-source / portlet jsr168 compliant

GridPort

OGCE
open grid computing environment

MyProxy: User Registration



PURSE: Portal-based User Registration Service

GAMA: Grid Account Management Architecture

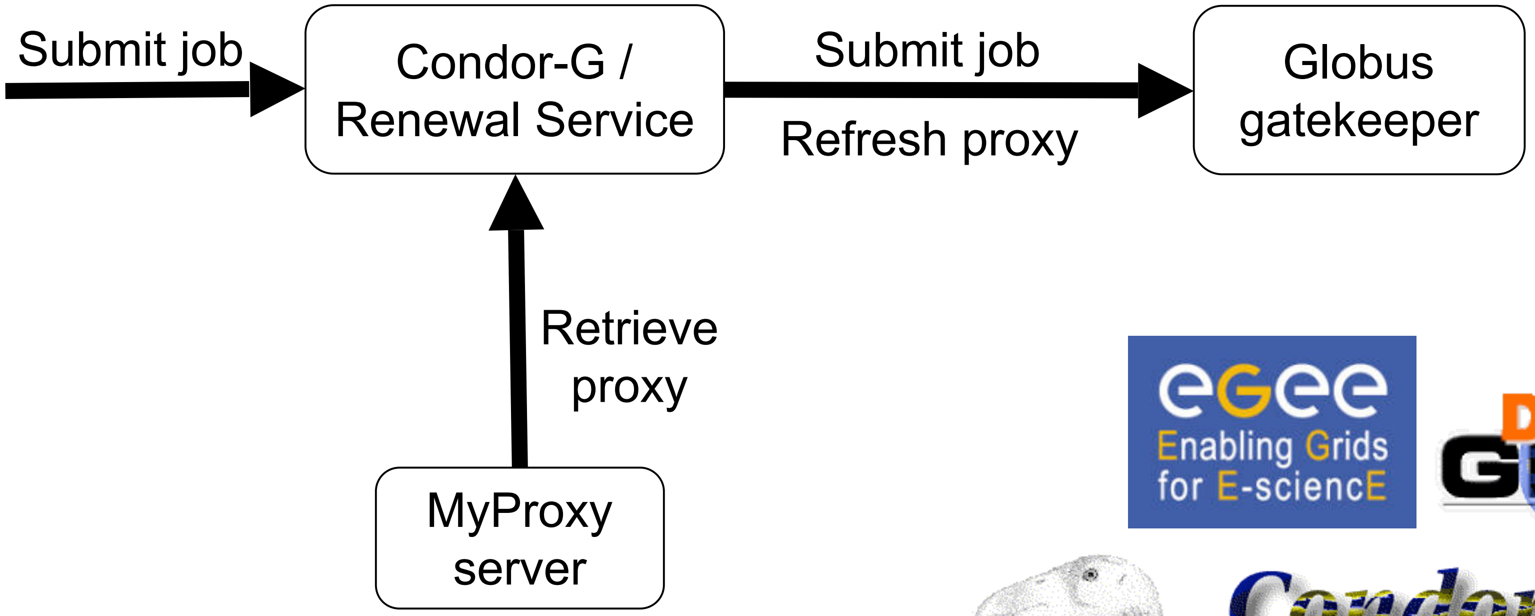
MyProxy: Key Upload/Download

- Provides ability to 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

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 / Renewal Service

Submit job

Refresh proxy

Globus gatekeeper

Retrieve proxy

MyProxy server

Daniel Kouril and Jim Basney, "A Credential Renewal Service for Long-Running Jobs," 6th IEEE/ACM International Workshop on Grid Computing (Grid 2005), Seattle, WA, November 13-14, 2005.

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)

One Time Passwords (OTP)

- Protect against stolen passwords
- Hardware token generates OTP
- Authenticate with OTP alone or combined with key passphrase
- Tested with CryptoCard tokens at NCSA
- Compatible with existing MyProxy clients

Managing Trust Roots

- Address challenge of keeping trust root configuration up-to-date across machines
  - CA certificates and CRLs
- User's trust roots can differ from site's
- myproxy-logon -T
  - Synchronizes contents of ~/.globus/certificates with MyProxy server

MyProxy CA

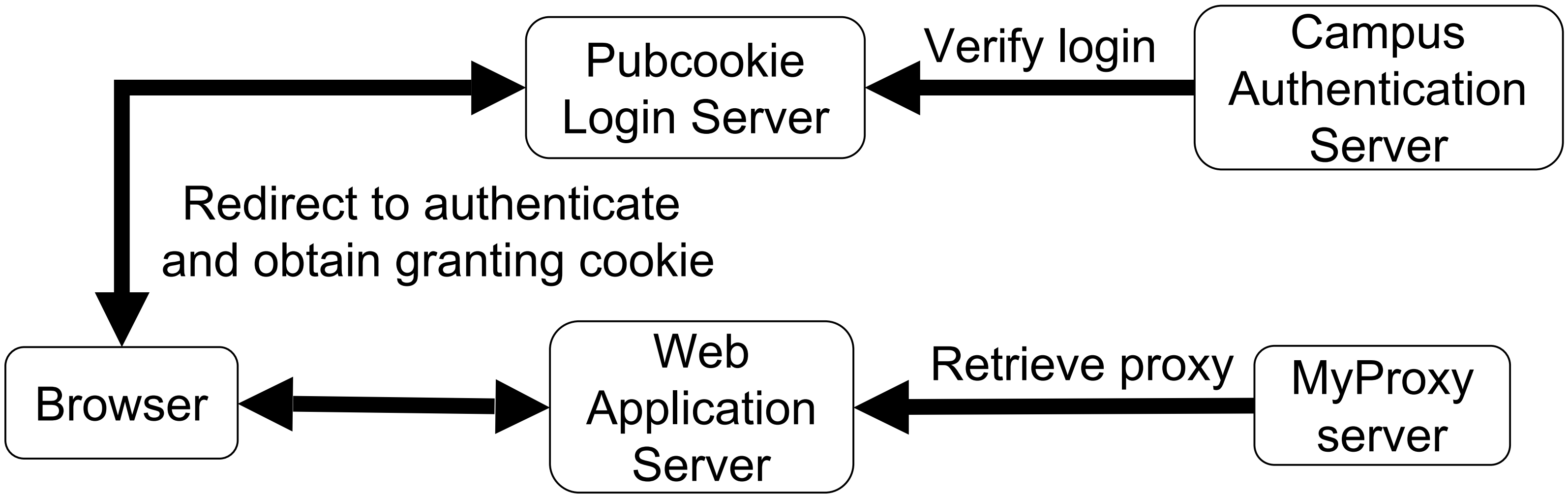
- MyProxy server issues short-lived certificates to authenticated clients
  - Leverage MyProxy authentication mechanisms
  - Compatible with existing MyProxy clients
- Avoid managing long-lived user keys
- Server can function as both CA and repository
  - Issue certificate if no credentials found for user

Coming soon!

MyProxy and Pubcookie

Coming soon!

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



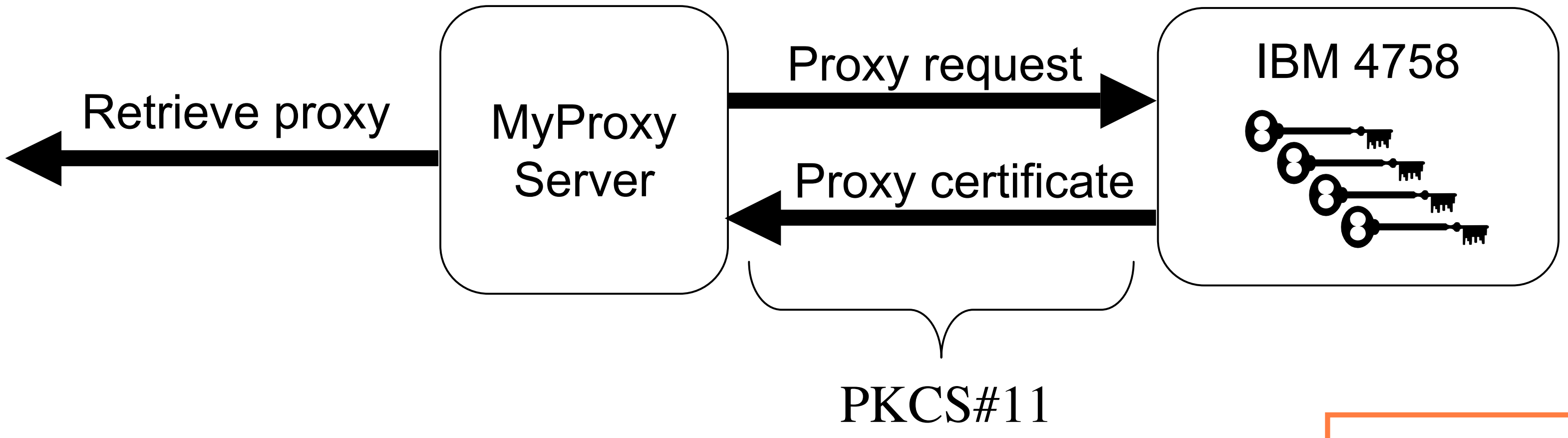
Jonathan Martin, Jim Basney, and Marty Humphrey, "Extending Existing Campus Trust Relationships to the Grid through the Integration of Pubcookie and MyProxy," 2005 International Conference on Computational Science (ICCS 2005), Emory University, Atlanta, GA, May 22-25, 2005.

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



Retrieve proxy

MyProxy Server

Proxy request

Proxy certificate

IBM 4758

PKCS#11

Experimental

M. Lorch, J. Basney, and D. Kafura, "A Hardware-secured Credential Repository for Grid PKIs," 4th IEEE/ACM International Symposium on Cluster Computing and the Grid (CCGrid), April 2004.

MyProxy CoG Clients

- Commodity Grid (CoG) Kits
  - Provide portable (Java, Python, and Perl) MyProxy client tools & APIs
  - Windows support
- For more information:
  - http://www.cogkit.org/

MyProxy Commands

- **myproxy-init**: store proxy
- **myproxy-logon**: retrieve proxy
- **myproxy-info**: query stored credentials
- **myproxy-destroy**: remove credential
- **myproxy-change-pass-phrase**: change password encrypting private key
- **myproxy-store**: store credential
- **myproxy-retrieve**: retrieve credential

MyProxy Installation (Unix)

- Included in GT 4.0

```
$ make gsi-myproxy; make install
```

- 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 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 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

Coming soon!

MyProxy and Standards

- MyProxy protocol specification submitted to GGF recommendations track
  - Currently under steering group review
- MyProxy uses:
  - IETF RFC 2246: Transport Layer Security (TLS) Protocol Version 1.0
  - IETF RFC 3820: Internet X.509 PKI Proxy Certificate Profile
  - DCE RFC 86.0: Pluggable Authentication Modules (PAM)
  - IETF RFC 2222: Simple Authentication and Security Layer (SASL)

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

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.

OSI certified™

Thank you!

Questions/Comments?

Contact:
jbasney@ncsa.uiuc.edu