Credential Management in the Grid Security Infrastructure

GlobusWorld Security Workshop
January 16, 2003

Jim Basney
jbasney@ncsa.uiuc.edu
http://www.ncsa.uiuc.edu/~jbasney/
Credential Management

- **Enrollment**: Initially obtaining credentials
- **Retrieval**: Getting credentials when and where they’re needed
- **Renewal**: Handling credential expiration
- **Translation**: Using existing credentials to obtain credentials for a new mechanism or realm
- **Delegation**: Granting specific rights to others
- **Control**: Monitoring and auditing credential use
- **Revocation**: Handling credential compromise

We need tools to cope with the complexity of credential management on the Grid.
Grid Credentials

• Identity credentials
  – Different mechanisms (X.509, Kerberos, .NET)
  – Different authorities (CAs, KDCs)
  – Different purposes (authentication, signing, encryption)
  – Different roles (project-based, security levels)

• Authorization credentials
  – X.509 attribute certificates
  – SAML/XACML/XrML assertions

• Trusted credentials
  – CA certificates and policies
  – Other certificates and public keys (SSH, PGP)
Accessing Credentials

• Ubiquitous access to the Grid
  – Initiate secure Grid sessions from desktops, kiosks, PDAs, cell phones, etc.
  – Requires access to needed credentials, including trusted credentials (CA certificates, etc.)
  – Bootstrap from password

• Delegating credentials to transient services
  – May need to retrieve additional credentials and/or renew existing credentials at run-time
  – Need access to trusted credentials and policy information
Traditional PKI Enrollment

1. End entity generates public/private key pair & submits certificate request to CA
2. CA approves/denies certificate request & signs certificate if request is approved
3. End entity retrieves signed certificate from the CA
Traditional PKI Enrollment

• Can be cumbersome for users and CA operators
  – May require a trip to a Registration Authority or some other out-of-band identity verification
  – CA operators must examine each request and sometimes investigate further before deciding to approve or deny
  – Process may take hours or days to complete
End Entity Key Management

- Typical practice in GSI is to store private keys in files encrypted by a passphrase
  - Security depends on well-chosen passphrases and well-secured filesystems
- Users copy private keys to the different systems they use to access the Grid
- Not all Grid users are PKI experts
  - Just want to do their computing securely
  - Can we improve usability and security of end entity key management on the Grid?
- Alternatives: Smart Cards, Online CAs, Online Credential Repositories
Smart Cards

- User-managed, portable credential storage
- Security analogous to car keys or credit cards
- Private keys stay in hardware
- Card standards are mature
- Costs are decreasing but still significant
  - $20 readers, $2 cards
  - Government ID card deployments
- Can pre-load credentials on the card before distributing it
- Some support already in GSI libraries
Online CA

• User authenticates to CA to obtain credentials immediately

• Leverage existing authentication mechanisms (password, Kerberos, etc.)

• Identity mapping:
  – Simple transformation (i.e., include Kerberos principle name in X.509 certificate subject) or administrator-managed mapping

• Signs long-term and/or short-term credentials
  – If long-term, then credentials are user-managed
  – If short-term, credentials retrieved on demand, without need for user key management
Online CA Security

- CA machine must be well-secured
- Signing key must be well-protected (i.e., stored in hardware crypto module)
- Key compromise allows attacker to create arbitrary credentials
- CA compromise may allow attacker to manipulate user authentication or identity mapping info
- If compromised, must revoke CA certificate and change CA signing key
- Short-term credentials don’t need to be revoked
Online Credential Repository

- Store encrypted credentials and access policy in an online repository
  - Repository may be mechanism-aware or may simply hold opaque credentials
- Authenticate to repository to retrieve opaque or delegated credentials
- Separates credential creation from credential management
- Can be deployed by individuals, small groups, VO managers, or CA operators
- Credentials can be pre-loaded to leverage existing authentication mechanisms
Credential Repository Security

- Credentials individually encrypted with user’s passphrase
- Compromise requires offline attack on each credential
- Centralized storage of credentials may violate policies (CA CP/CPS)
- If compromised, credentials in repository must be revoked
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 tasks 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 task’s lifetime
  - Renewal service provides a single point of monitoring and control
  - Renewal policy can be modified at any time
  - For example, disable renewals if compromise is detected or suspected
Multiple Credentials

• Will a single identity credential per user suffice?
  – A lot of work is being done to vet and/or cross-certify Grid CAs
  – How is that different from Kerberos cross-realm authentication?

• Alternative: Provide tools to manage multiple credentials
  – Single sign-on unlocks all credentials
  – Grid protocols negotiate for required credentials (WS-SecurityPolicy)
  – Authorization decision between individual and resource provider, rather than between realms
Credential Wallet

• Consolidated view of my credentials
• Credential management interface
  – Add, remove, or modify credentials
  – Associate policies with credentials
  – Create authorization credentials
• One-stop credential access point
  – Single sign-on unlocks credentials for a session
  – Contains pointers to available credential services
• Manage credentials on my behalf
  – Example: renew credentials as needed
• Notify when events occur or action is required