Security in Federated e-Infrastructure and Identity Management

Boris Parák²  Slávek Licehammer¹,²

¹Masaryk University
²CESNET

May 18, 2015
Security Implications
Single-user vs. Federated Users
Security Incident

- as it relates to federation’s user management approach
- compromised appliances (images) out of scope here
- distinct possibilities:
  - compromised user account (credentials) $\rightarrow$ malicious user
  - compromised virtual machine(s) $\rightarrow$ malicious instance
  - often combined, malicious user(s) $\rightarrow$ malicious instance(s)
- need for traceability user(s) $\leftrightarrow$ instance(s)
- need for dynamic and fine-grained access restrictions
User Management

Single-user:
- incredibly simple to set up and maintain, little effort required
- can be created statically/manually by local administrators
- does not require synchronization, aside from the occasional change of credentials

Federated Users:
- requires considerable effort/development or an existing solution
- must be handled globally at the federation level
- frequent synchronization and consistency checks
- identity consolidation is a tricky business
User Isolation

**Single-user:**
- high-level centralized tracking mechanism is required
- references between used resources and identities must be kept by the federation platform
- difficult to track a user with multiple personal identities

**Federated Users:**
- each site can use its own native tools
- no effort required at the federation platform level
- users with multiple identities are tracked by the user management platform
Incident Handling

Single-user:

• difficult to trace an incident locally (owner? identity?)
• fine-grained/localized access restrictions cumbersome
• one compromised set of credentials can affect a lot of resources

Federated Users:

• easy to trace an incident to a particular user
• easy to restrict access just for the user in question
• compromised credentials affect only one “small” account
Resource Allocation

Single-user:
- high-level centralized allocation & usage tracking mechanism is required
- enforcing quotas is difficult, existing local mechanisms are useless
- reservations/quotas “inside” the site nearly impossible

Federated Users:
- each site can use its own native tools
- quota enforcement usually already built-in
- relatively easy local per-group or per-user reservations/quotas
Summary

- fully federated identity management is difficult to deploy
- in most cases, benefits outweigh the drawbacks in long-term
- especially when scaling the infrastructure
- offers ways to delegate responsibility (users, site admins, CMFs)
Identity and Access Management System (IAM)
Perun manages

- Virtual organizations
- Users
- Groups
- Resources
- Attributes
VO and Group Management

Built-in support for virtual organizations

- Configurable enrollment form
- Delegation of rights to manage VO to the end users
- Access management for the VO resources

Group management

- Configurable application form
- Group manager role
- Automatic synchronization with external systems
User Management

Identities
- User can have several existing identities
  - X.509 certificates, SAML, social identities, SSH keys, Kerberos principals, ...
- Identity consolidation
- Perun doesn’t store user’s password, private keys, ...

Enrollments
- Pre-filled information from external authN system

Service users
- Represents services
Access Management

- Resources are assigned to the VOs
- Configuration of the access to the services
  - E.g. UNIX accounts, mailing lists, ACLs for web applications, OpenNebula

Push mechanism

- Omit online queries
- Push only on change (ideal for cloud platforms)

LDAP interface

- For LDAP compatible service
Enrollment and propagation
Attributes Management

- Every entity and also relationship can have assigned the attributes
- Different value types: string, number, list, array
- Access rights on attribute values

Attributes modules

- check proper value of the attribute
- fill default values
- check value of dependent attributes
Perun

- Identity and Access Management System (IAM)
- developed by CESNET and Masaryk University
- open source, available on github
- provided as virtual appliance

http://perun.cesnet.cz
Thank you!

slavek@ics.muni.cz
parak@cesnet.cz