Account Management in QWG Templates

Michel Jouvin
LAL, Orsay
jouvin@lal.in2p3.fr
http://grif.fr

March 12, 2009
Quattor Workshop, London

- VO accounts are created as local accounts on every machine
  - Every VO has a pre-defined UID range
    - Default provided, customizable by sites in template vos_base_uids
  - Every VO has a pre-defined number of accounts
    - Default: 200, customizable by sites in a template voname.tpl in vo
  - Accounts for a VO have a common prefix
    - Default provided, customizable by sites in template vos_account_prefix.tpl
- Pool accounts supported for specific FQANs but not enabled by default
  - VOMS_ROLE_CONFIG_SITE
- When using NFS for home dirs, ability to rewrite /he into something different for VO accounts
  - Allow to keep home dir for other accounts local
  - Variable VO_HOMES_NFS_ROOT
  - @VONAME@ can be used to add a directory level under

- Lack of easy enabler for activating pool accounts fo all specific FQANs
  - May be implemented by extending VOMS_ROLE_CONFIG_SITE to handle a default entry
- Potential exhaustion of UID range (65K ?) if supporting a high number of VOs
  - At least default value will be meaningless
  - Default range per VO is 100O UIDs
- Very difficult to add accounts to a VO without changing existing ones
  - Requires a downtime
  - Management burden as this may imply changing permissions on some dirs, difficult to automatize

- Risk of inconsensistency if something wrong happen on one node as every node has its local view
- Performance : need to check thousands of accounts hundred of groups on every machine

- Use a LDAP server as the central authentication server
  - No account configuration required on other machines, just the LDAP configuration for authentication
- Suppress per-VO range of accounts
  - Just create as many accounts as needed to handle all th
    configured VOs
- Write a new component for managing LDAP-based VO accounts
  - Schema based on VOs or FQANs
  - For each VO or FQAN, give the number of accounts to « create », the group to associate with, base account parameters
  - At each run, check if new accounts must be created as a
    result of the config change and for every account to add
    allocate the next uid available and put it in the right groups

- Example
  - First configure a server with 150 accounts for Atlas and 100 for CMS
    - Atlas UIDs will be 100-249, Atlas SW mgr will be 250, CMS 251-350, CMS SW mgr will be 351
  - Add 50 accounts for Atlas and move SW mgr to pool accounts (10 accounts)
    - New normal accounts for Atlas will be 352-401
    - New SW mgr accounts (9 if we use the existing one as one pool account) will be 402-411
  - Transparent for the clients
- Need to assess stability of LDAP for authentication of a large amount of process (large CE)
  - Anybody with already some good experience ?