Agenda

- Introduction
- RBAC Overview
- Solution
- Implementation with Samba native methods
- Implementation with custom Samba VFS
- Conclusion
**Introduction**

“Within an organization, roles are relatively stable, while users and permissions are both numerous and may change rapidly.”

<table>
<thead>
<tr>
<th>Controlling all access through roles simplifies</th>
</tr>
</thead>
<tbody>
<tr>
<td>Management</td>
</tr>
</tbody>
</table>
A general-purpose role based access control model was proposed in 1992 by Ferraiolo and Kuhn.

- A key feature of this model is:
  - All access is through roles
  - A role is essentially a collection of permissions
  - All users receive permissions only through the roles to which they are assigned, or through roles they inherit through the role hierarchy
Overview: Model

- Users → Roles
- Roles → Operations
- Roles → Objects
- Users → Sessions
- Sessions → Roles
- User Assignment → Role Hierarchy → Permission Assignment
- user_sessions → session_roles
- Permissions
Overview: RBAC vs ACL

a) Role-Based Access Control

b) Discretionary Access Control Using Access Control Lists
**Introduction: Windows**

**Application**
- namespace for roles, tasks, and operations.

**Scope**
- collection of resources in which each resource of similar type has the same authorization policy.

**Role**
- Usually corresponds to a job category or responsibility (for example, purchaser or hiring manager) and is a collection of tasks that a user must have to do that job.

**Task**
- collection of operations and sometimes other tasks.

**Operation**
- set of permissions that are associated with system-level or API-level security procedures.

**Business Rule**
- script that is associated with a task. A rule allows access decisions to be based on any run-time condition that a script can test.
Implementation Steps

Steps

• Role analysis
• Identify Resources
• RBAC object map
• Samba implementation
A subject can have multiple roles.

A role can have many permissions.

A role can have multiple subjects.

A permission can be assigned to many roles.

Implementation Core revision 0

Implementation
Samba Implementations

Usage Native methods
- Share
- Groups

VFS Module
- Custom Module
- Directory Server
Native Methods : Mapping

Samba User

Group
- With specific prefix Role_[Role name]

Group
- RO Group prefix OBJECTNAME_RO with read only permission
- RW Group prefix OBJECTNAME_RW with read write permission
- OWNER Group prefix OBJECTNAME_OWNER with complete permission

Shares
- Minimal element Windows share
- Windows share name = operation group name
Conf
- Winbind nested groups
- Local group as permission
- Domain group as role
- Force group creation

Share
- valid user ..

ACL
- ACL support on File system
- ACL inherits
Cons
- Minimal entity based on share
- Group number
- Group is a collection of permissions, rather than a collection of users

Pro
- No additional software
- Simple administration
- AD integration
VFS Module: introduction

RBAC Module

VFS Modules
extended capabilities

Directory
manage the identities and relationships
VFS Module : Mapping

- **Samba User**
- **New entity**
  - New branch in Directory server
  - New objectclass
- **VFS layer**
  - Operation handle by vfs ()
  - Attribute in object level
- **New entity**
  - New branch in Directory server
  - New objectclass
Role Object
- userMemberID
- roleName
- adminUser

Resources Object
- objectName
- URL
- Operations
- Role
VFS Module: implementation

Check Role/Permission on: Write/Read/Administration

Disk operations
Directory operations
File operations
Permission mod
**Cons**
- Unstable module
- Performance
- Directory Server
- No GUI

**Pro**
- Flexible
- Centralization Role definition
Wrong Usage

- Unstable organization
- Many cross function user
- Usage RBAC as ACL
- Security Enforcement
Benefit

Multi-user systems

Multi-application systems

Intuitive

- competency, authority and responsibility
Result

Deploy
- 400 people
- 20 roles
- 100 shares
- 7 servers

Enhancement
- Reduction provisioning time 70%
- Simplify security audit
VFS Module: next

- **Release**
  - Alpha GPL Version

- **Cache**
  - Cache on Permissions
  - Cache on Roles

- **Windows Compatibility**
  - Schema Extension
  - Compatibility with AzMan

- **Session**
For Further Questions:

• Fabrizio Manfredi
  • fabrizio.manfredi@gmail.com
  • manfred@freemails.ch

• http://www.beolink.org

Reference