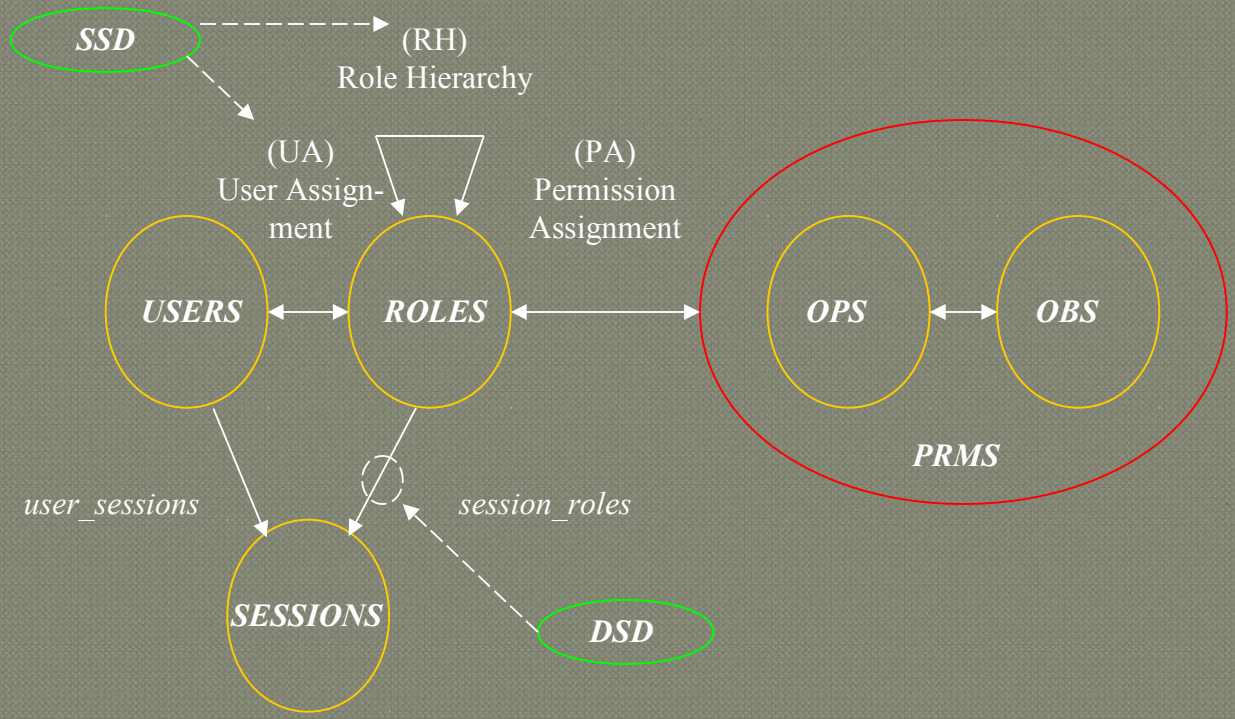
Role-Based Access Control



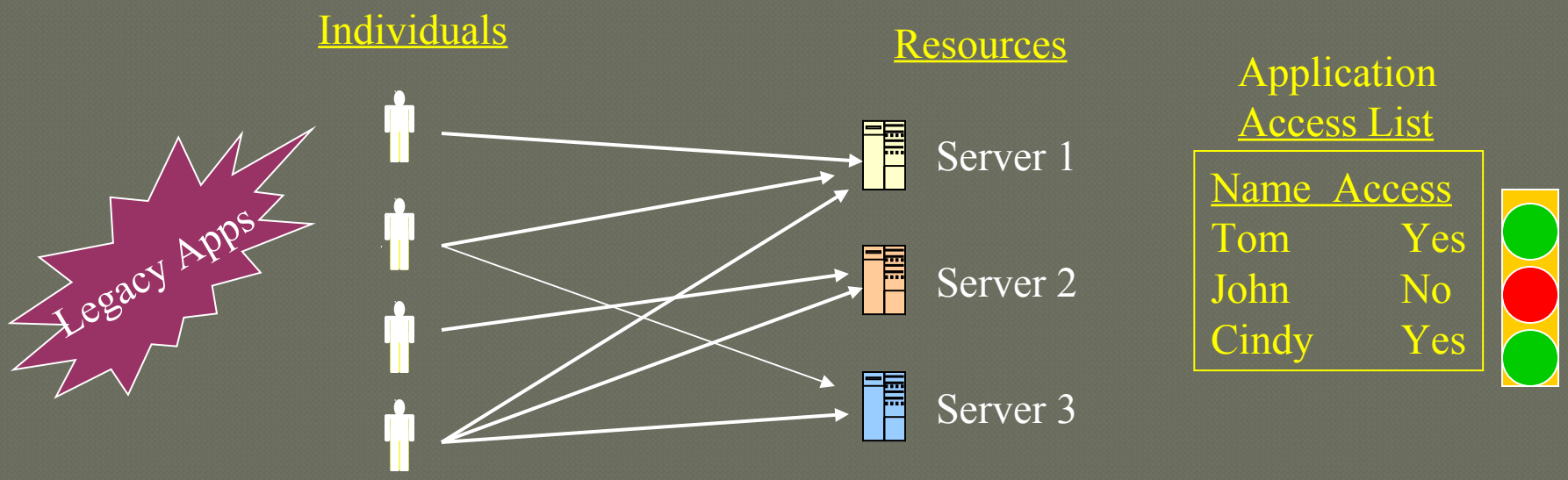
Access Controls - Basics

- Access Control a system to control, monitor and restrict the movement of people, assets or vehicles around a building or site

- Access Control types
  - Discretionary Access Control
  - Mandatory Access Control
  - Role-Based Access Control

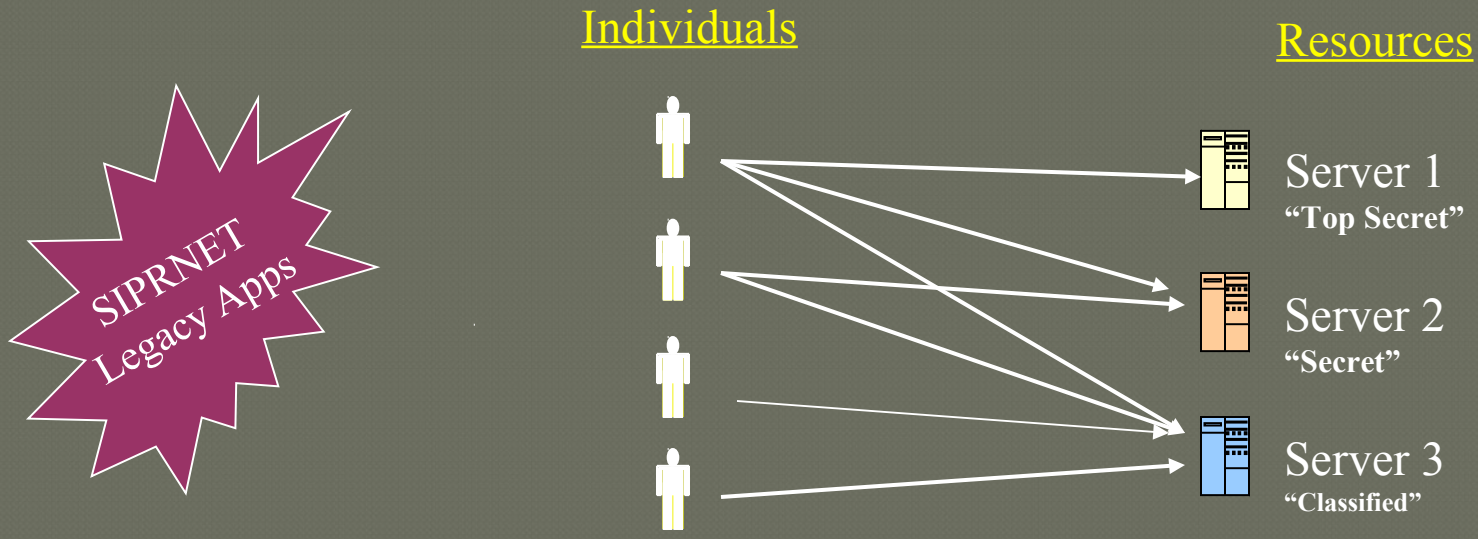
Discretionary AC

- Restricts access to objects based solely on the identity of users who are trying to access them.



Mandatory AC

- MAC mechanisms assign a security level to all information, assign a security clearance to each user, and ensure that all users only have **access** to that data for which they have a clearance.



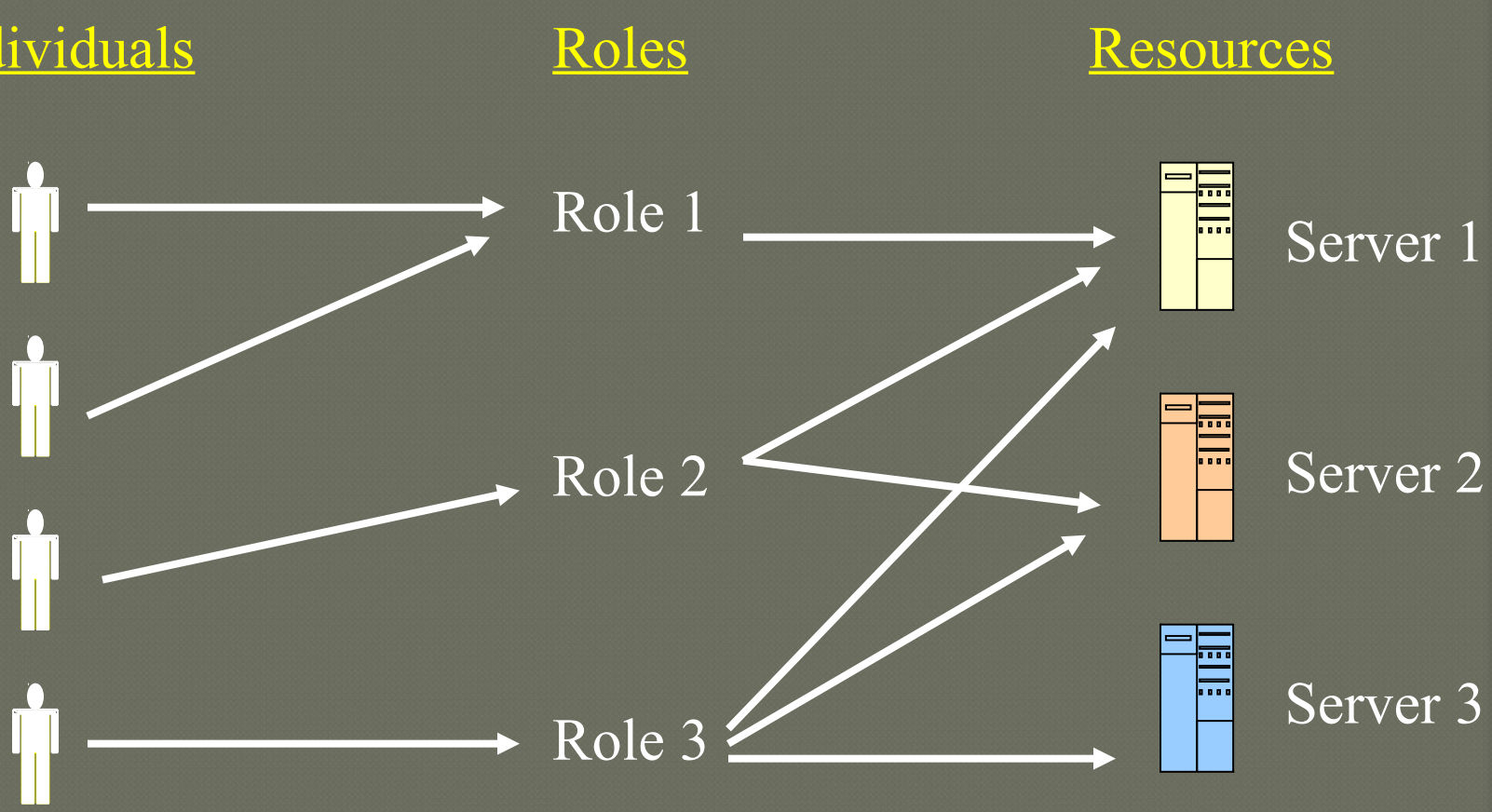
Better security than DAC

Role-Based AC

"Ideally, the [RBAC] system is clearly defined and agile, making the addition of new applications, roles, and employees as efficient as possible"

- A user has access to an object based on the assigned role.
- Roles are defined based on job functions.
- Permissions are defined based on job authority and responsibilities within a job function.
- Operations on an object are invocated based on the permissions.
- The object is concerned with the user's role and not the user.

Role-Based AC



User's change frequently, Roles don't

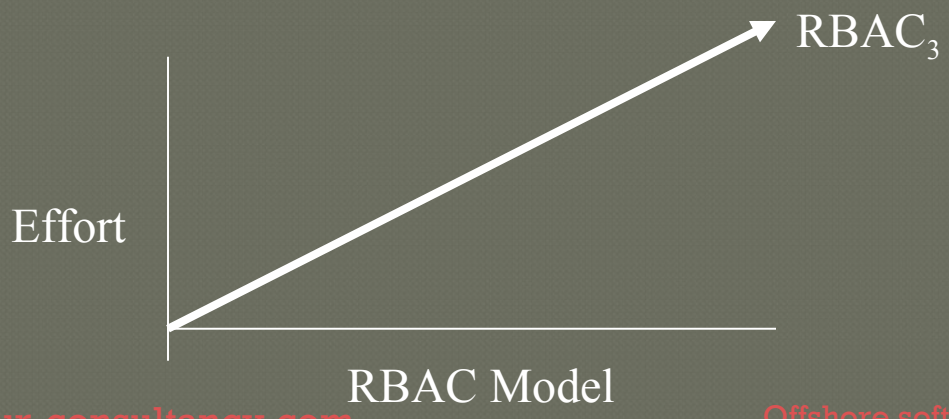
Rules of RBAC

- Three primary rules are defined for RBAC:
  - Role assignment
  - Role authorization
  - Permission authorization

RBAC Variance

A family of RBAC with four models

1. RBAC0: min functionality
2. RBAC1: RBAC0 plus role inheritance
3. RBAC2: RBAC0 plus constraints (restrictions on RBAC configuration)
4. RBAC3: RBAC0 plus all of the above



Role-Based AC Framework

- Core Components
- Constraining Components
  - Hierarchical RBAC
    - General
    - Limited
  - Separation of Duty Relations
    - Static
    - Dynamic

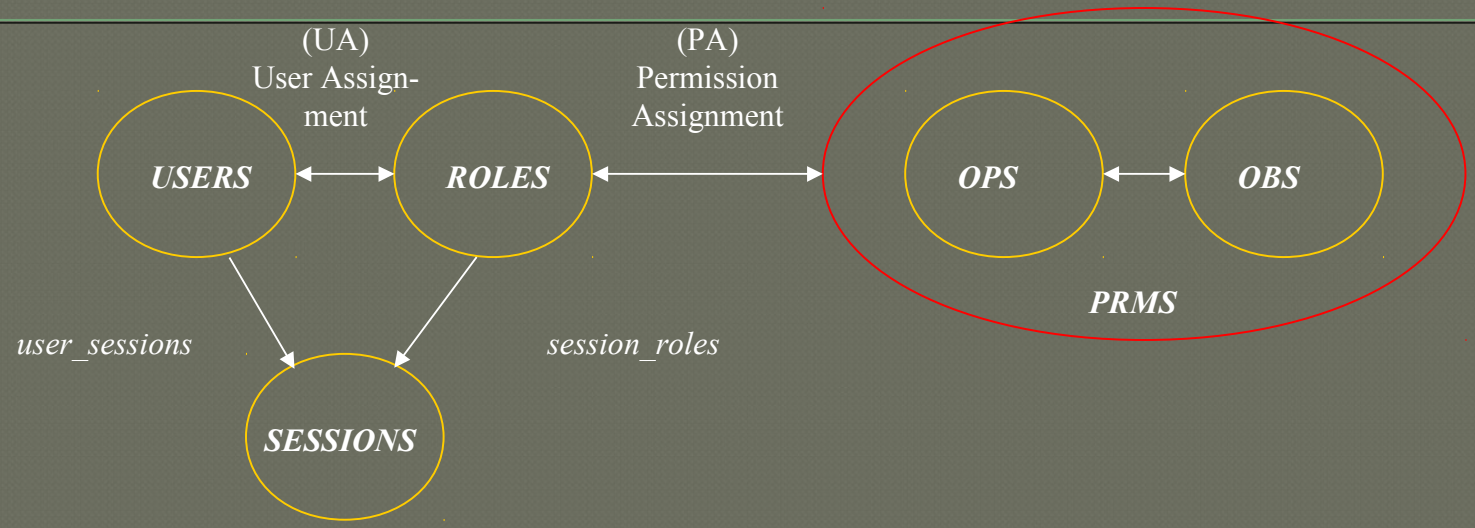
Core Components

- Defines:
  - USERS
  - ROLES
  - OPERATIONS (*ops*)
  - OBJECTS (*obs*)
  - User Assignments (*ua*)
    - assigned_users
- Permissions (*prms*)
  - Assigned Permissions
  - Object Permissions
  - Operation Permissions
- Sessions
  - User Sessions
  - Available Session Permissions
  - Session Roles

Constraint Components

- Role Hierarchies (*rh*)
  - General
  - Limited
- Separation of Duties
  - Static
  - Dynamic

Core RBAC



- Many-to-many relationship among individual users and privileges
- Session is a mapping between a user and an activated subset of assigned roles
- User/role relations can be defined independent of role/privilege relations
- Privileges are system/application dependent
- Accommodates traditional but robust group-based access control

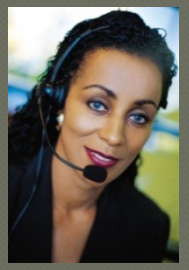
UA (user assignment)

USERS set

A user can be assigned to one or more roles

ROLES set

Developer



Help Desk Rep

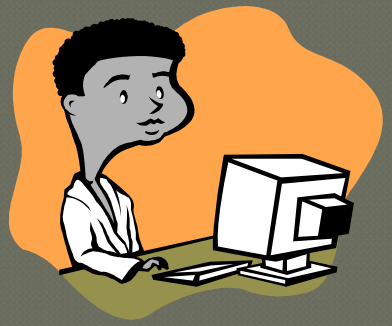
A role can be assigned to one or more users

PA (prms assignment)

PRMS set

A prms can be assigned to one or more roles

ROLES set



Admin.DB1

Create
Delete
Drop

View
Update
Append

A role can be assigned to one or more prms

User.DB1

SESSIONS Assignment

The mapping of user *u* onto a set of sessions.

USERS

SESSION

USER1

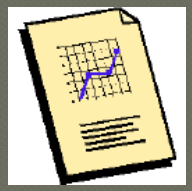
USER2

invokes

guest

admin

user



User2.FIN1.report1.session

User2.DB1.table1.session

User2.APP1.desktop.session

SESSIONS Assignment

The mapping of session *s* onto a set of roles

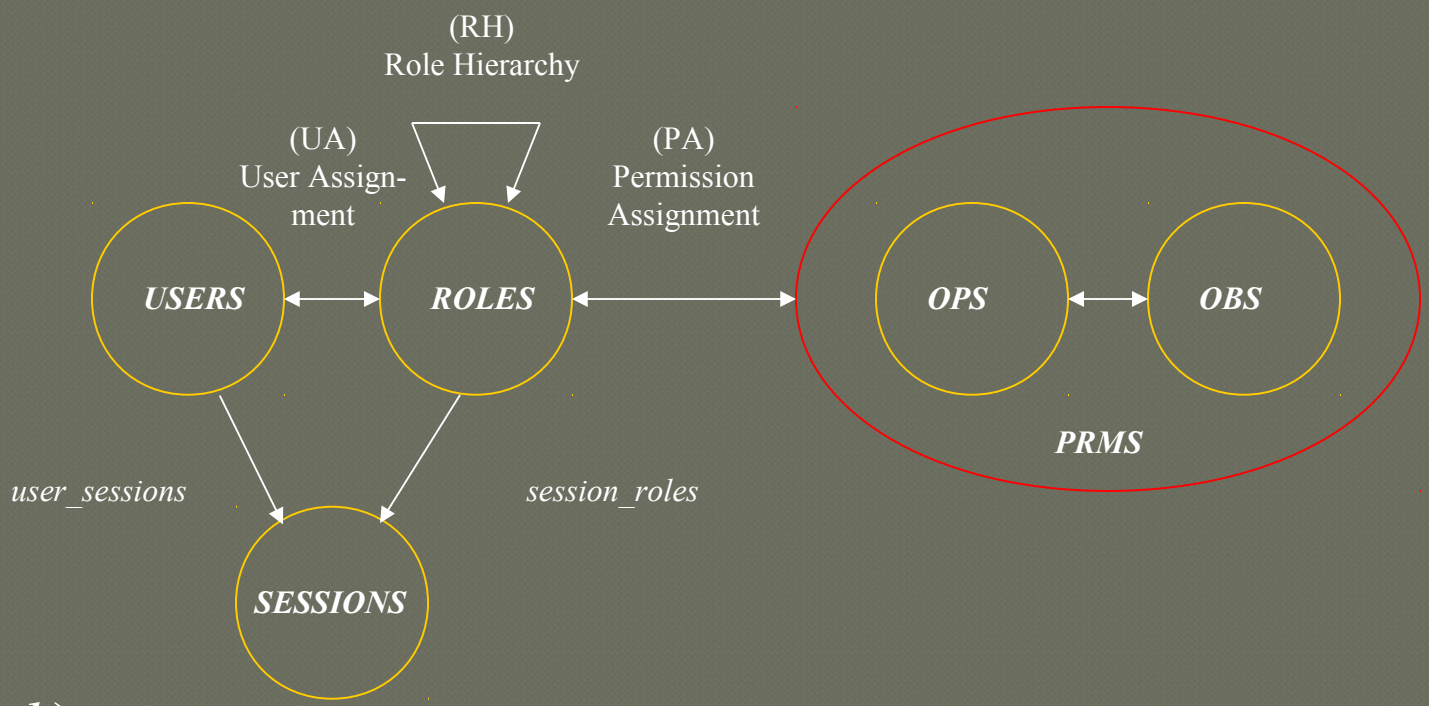
SESSION

ROLES

- Admin
- User
- Guest

DB1.table1.session

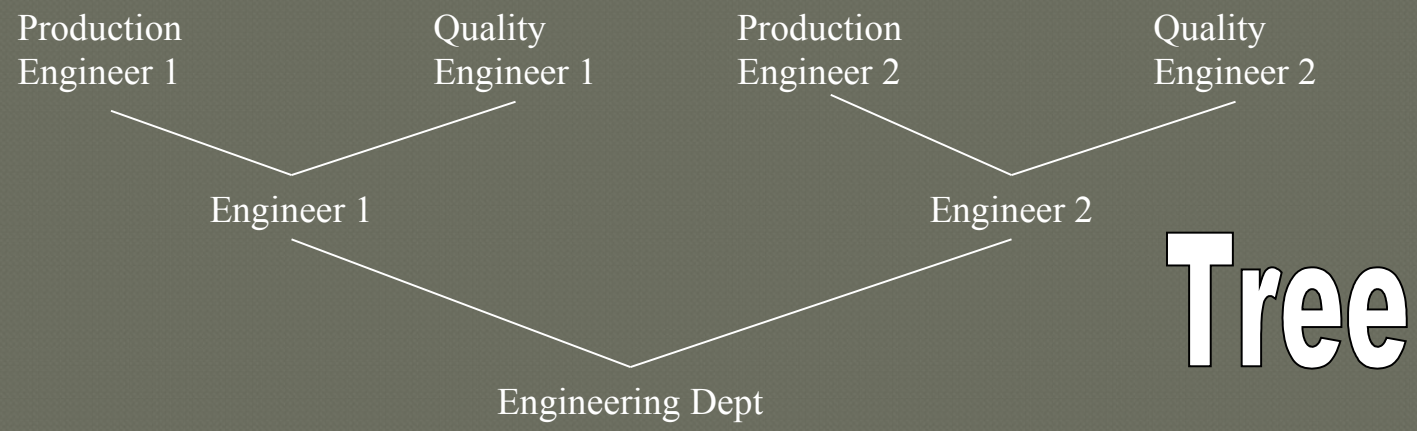
Hierarchal RBAC



Role Hierarchies (*rh*)

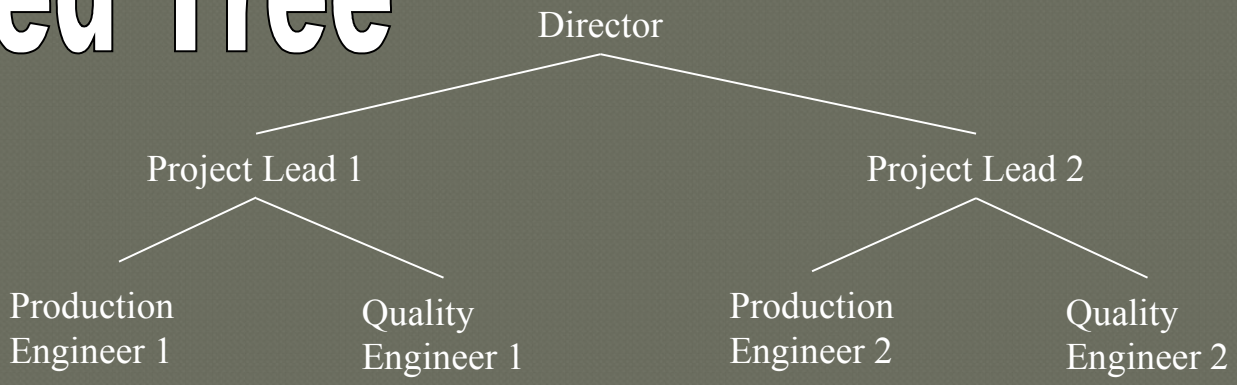
- General
- Limited

Tree Hierarchies



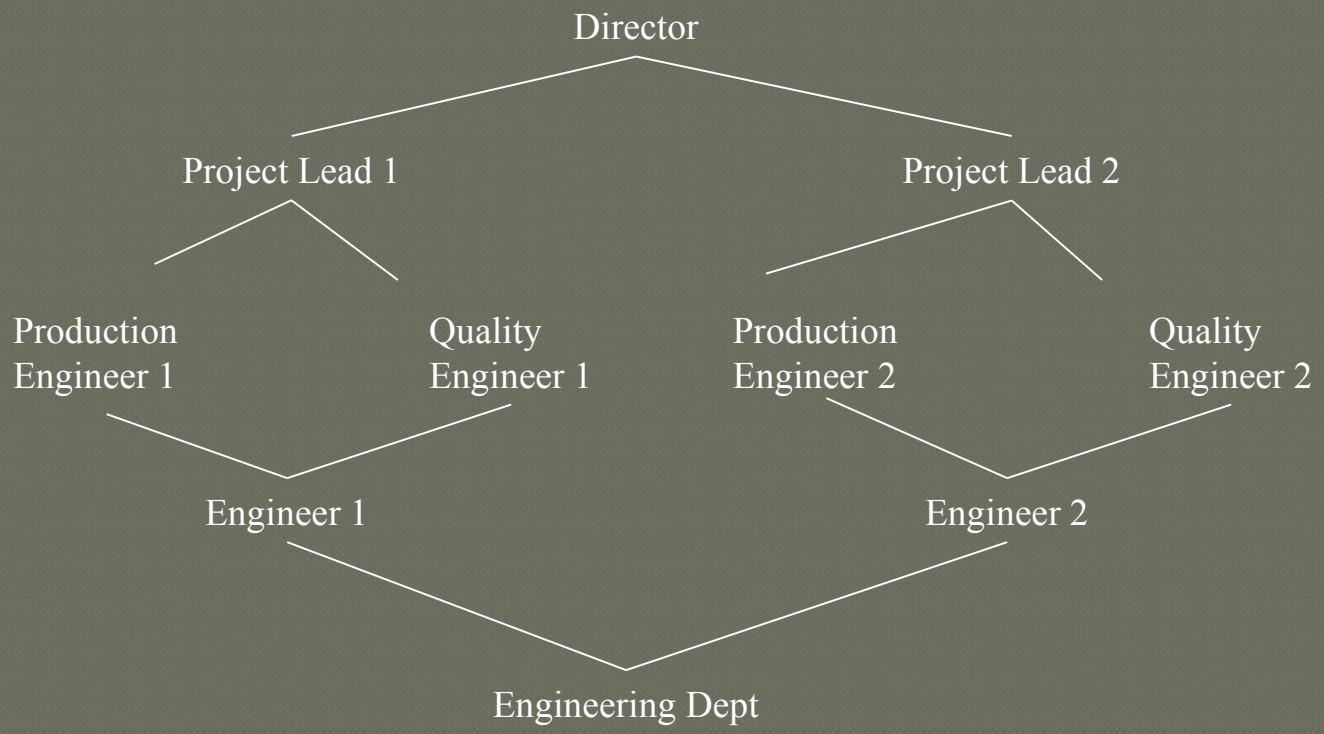
Tree

Inverted Tree

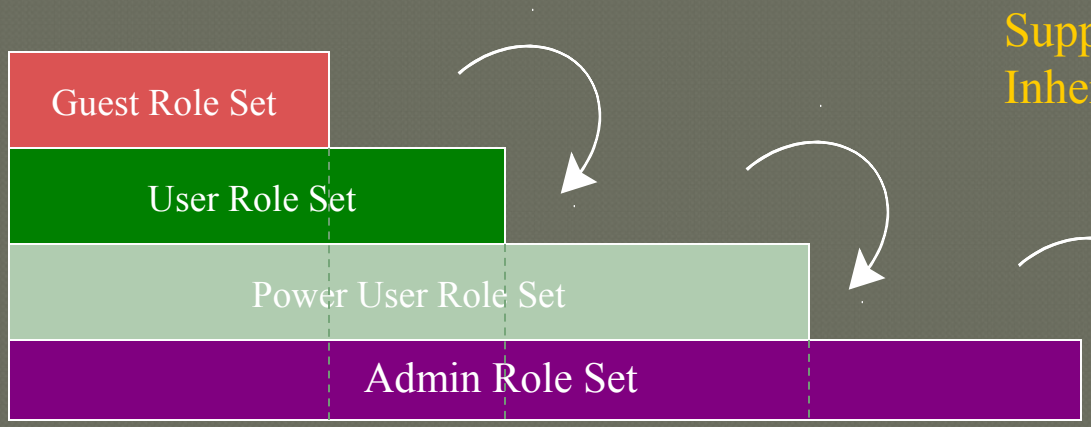


Lattice Hierarchy

Upper roles have all the access rights of the lower roles as well other access rights not available to a lower role



General RH



Support Multiple Inheritance

Only if all permissions of r_1 are also permissions of r_2

i.e. r1 *inherits* r2

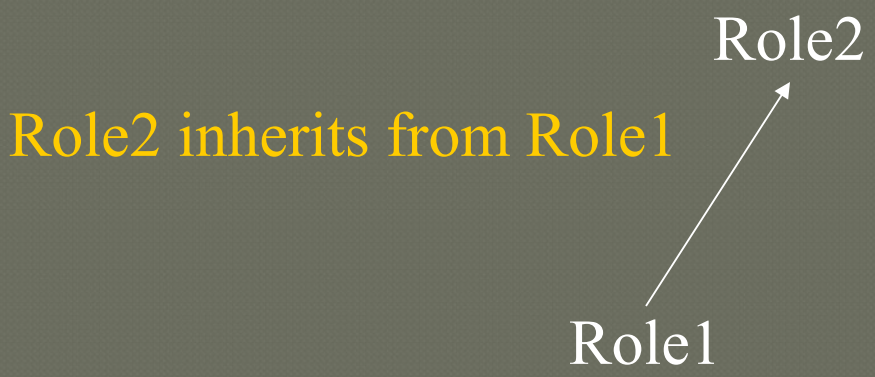
Only if all users of r_1 are also users of r_2

User r-w-h

Guest -r-

Limited RH

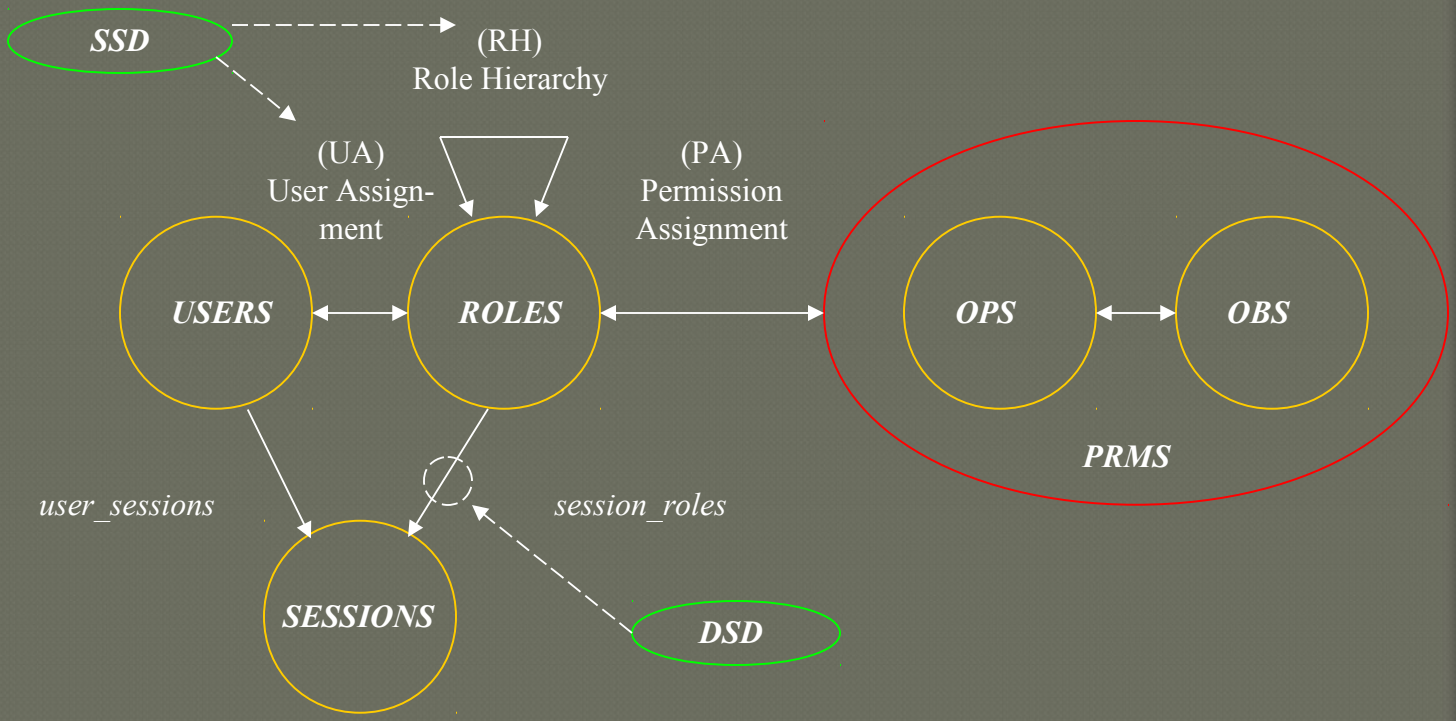
A restriction on the immediate descendants of the general role hierarchy



- Role3

Role3 does not inherit from Role1 or Role2

Constrained RBAC



Constrained RBAC

- Static
- Dynamic

Separation of Duties

- Enforces conflict of interest policies employed to prevent users from exceeding a reasonable level of authority for their position.
- Ensures that failures of omission or commission within an organization can be caused only as a result of collusion among individuals.
- Two Types:
  - Static Separation of Duties (SSD)
  - Dynamic Separation of Duties (DSD)

SSD

- SSD places restrictions on the set of roles and in particular on their ability to form *UA* relations.
- No user is assigned to *n* or more roles from the same role set, where *n* or more roles conflict with each other.
- A user may be in one role, but not in another—mutually exclusive.
- Prevents a person from submitting and approving their own request.

SSD in Presence of RH

- A constraint on the authorized users of the roles that have an SSD relation.
- Based on the authorized users rather than assigned users.
- Ensures that inheritance does not undermine SSD policies.
- Reduce the number of potential permissions that can be made available to a user by placing constraints on the users that can be assigned to a set of roles.

DSD

- Places constraints on the users that can be assigned to a set of roles, thereby reducing the number of potential permission that can be made available to a user.
- Constraints are across or within a user's session.
- No user may activate *n* or more roles from the roles set in each user session.
- *Timely Revocation of Trust* ensures that permission do not persist beyond the time that they are required for performance of duty.

RBAC Defense in Depth

RBAC for GIAC Enterprises

- The small scale of GIAC Enterprises is both a plus and minus for implementing RBAC
- Smaller companies will most likely mean users will be assuming multiple roles within the organization thus making it difficult to create static roles for each users or process.
- At first glance the implementation of RBAC in a company with under 10 employees may seem simple. If roles are not properly identified and categorized, scalability becomes a problem. The sooner you can implement principles of least privilege and segregation of duties, the more reliable your process will become.
- At a high level GIAC Enterprises can be broken into four divisions
  - Business (CEO, CFO, Sales Manager, Product Manager)
  - Development (Developer)
  - Administration (System Administrator)
  - Audit (External Resource)

RBAC in the DMZ

- The DMZ houses the Email gateway, IPS, Web Server, and MetaFrame Presentation Server
- Windows systems (Email, MetaFrame) use Active Directory (AD) for maintaining role-based access controls
- Linux systems (Web, App, IPS) use Vintela Authentication Services (VAS) which sits on the AD framework for administering role-based access controls
- Within AD, the following roles are defined specific to the DMZ:
  - User - read-only access to web pages
  - Administrator - read/write access to deploy changes made by developer
  - Auditor – read-only access to specified systems

RBAC for Internal Systems

- Access to the majority of GIAC Enterprise's internal systems (Email, File, HR, Antivirus, DC, DNS) is governed by Windows Active Directory (AD)
- Access to the Linux/Apache web server and the Solaris/Weblogic App Server is controlled via Vintela Authentication Services (VAS) managed through AD
- Internally the following roles are defined:
  - User - read-only access to web pages
  - Administrator - read/write access to deploy changes to production after they've been made by a developer
  - Developer – read/write access to development partitions of web/app/db servers
  - Auditor – read-only access to specified systems
- Employees access the sales and HR database utilizing a web-to-app interface thereby abiding by a 3-tier architecture
- Systems are partitioned and segmented into development and production environments to facilitate configuration management practices

RBAC for Network Devices

- Cisco's Network Admission Control (NAC) is used to control workstations and laptop access to the internal network
- IBNS and 802.1x is integrated into NAC (next slide)
- 802.1x provides controls for both wired and wireless devices
- NAC Profiler is used to automatically identify and assess non-PC devices such as Voice over IP phones and printers
- Appropriate device roles are created. For example, business user, guest user, etc...
- NAC is used to isolate vender connections (i.e. visiting laptops), while still allowing Internet access
- Ensure that authorized endpoint devices have been patched (operating systems, critical applications, anti-virus, anti-spyware, etc..) via the policy server.

RBAC for Infrastructure

- Use Cisco's AAA & TACACS+ via Cisco Secure Access Control Server & Active Directory for centralized router and firewall Authentication, Authorization, and Accounting.
- Use Cisco's Identity-Based Networking Services (IBNS) identity management solution
- IBNS is based on 802.1x and offers authentication, access control, and user policies to secure the network
- 802.1X allows enforcement of port based network access control when devices attempt to access the network
- IBNS leverages Cisco's switches, Wireless APs, Cisco Secure ACS and Cisco Secure Services Client
- Cisco's Role-Based CLI Access is used to define auditor and helpdesk views
- These views are configured to restrict access to Cisco IOS commands and configuration while allowing timely problem resolution and audit access to the IOS

RBAC for Auditing

- RBAC will ease auditing of network and systems
- Enforces unique usernames; only one username per user
- Define ‘read’ or ‘view’ only access to auditing roles
- Auditors can then be granted access to audit roles
- Appropriate event logs from servers, Active Directory, IPS, routers, Vintela Authentication Services, NAC, key card system and other network infrastructure devices are stored in a centralized log server
- Access to the centralized log server data is restricted, IT can not access, modify or delete logs without audit’s permission
- An event correlation and reporting server is used by both IT and audit to correlate and review the data

References

1. NIST documents at http://csrc.nist.gov/rbac/
2. D. Ferraiolo, R. Sandhu, S. Gavrila, D.R. Kuhn, R. Chandramouli, "A Proposed Standard for Role Based Access Control (PDF)," *ACM Transactions on Information and System Security* , vol. 4, no. 3 (August, 2001) - draft of a consensus standard for RBAC.
3. The ARBAC97 model for role-based administration of roles (1999)
4. Symbiosis
    1. Neha Kabra
    2. Jayesh Singhal
    3. Rohit Gedam
    4. Sunil Saroj

THANK YOU