IBM Policy Director Authorization Services for z/OS and OS/390

Including Tivoli Policy Director Overview and Security Server (RACF) Support

Arthur Fitzpatrick
Security Development and Test
arthurf@us.ibm.com

Agenda

- **Introduction to Tivoli Policy Director**
- **Policy Director Authorization Services for z/OS and OS/390**
- **Security Server (RACF) support and SAF enhancements for Policy Director Authorization Services**

Trademarks

- IBM, the IBM logo, OS/390, RACF, S/390, SecureWay, System/390, Tivoli, the Tivoli logo, and z/OS are trademarks or registered trademarks of International Business Machines Corporation or Tivoli Systems, Inc. in the United States, other countries, or both
- Microsoft and Windows are trademarks of Microsoft Corporation in the United States, other countries, or both
- Other company, product, or service names may be trademarks or service marks of others

Introduction to Tivoli Policy Director

Tivoli software

What are the requirements?

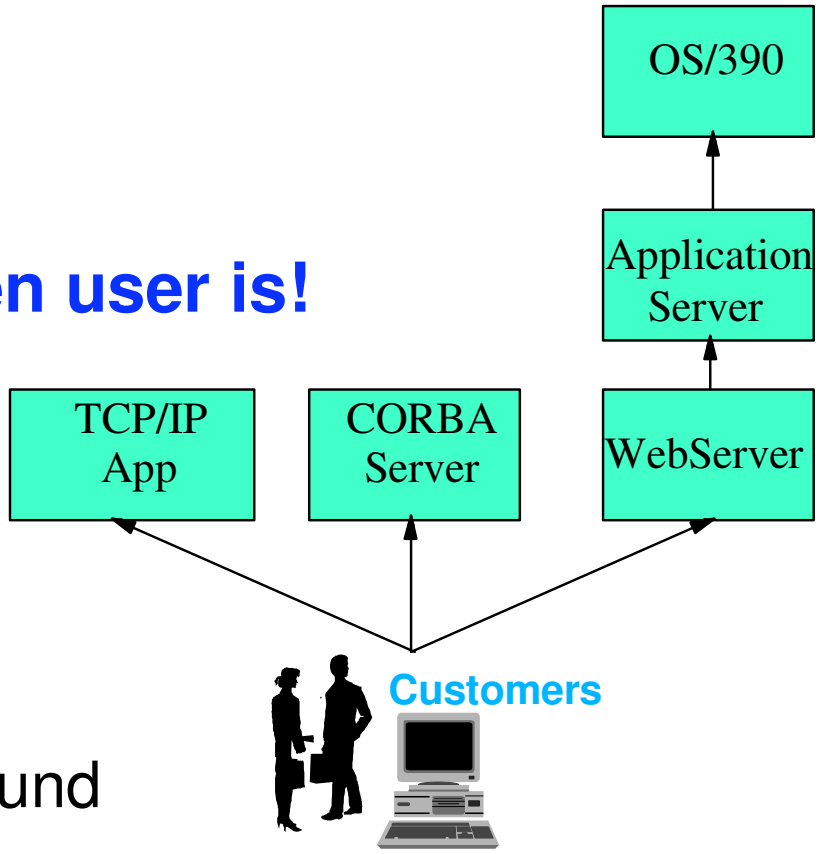
- **Customers Require**
  - Ease of use
  - Security
- **Security Requirements**
  - Authentication
  - Authorization
  - Auditing
  - Quality of Protection (QoP)
    - Integrity
    - Confidentiality
  - Enforced Security Policy

What are the problems?

Distributed applications have their own:

- Security administration tools
- Authentication models
- Authorization policies

Do not agree on who a given user is!



Tivoli Policy Director

- Unites core technologies around common security policies
- Addresses inability to implement security policies across multiple distributed platforms

What does Tivoli Policy Director do?

Maintains a central user registry

- Users and groups (LDAP)
- Authentication information

Maintains a model of the Protected Objectspace

- Hierarchically organized

Defines permitted actions on objects

- Uses Access Control List templates
- ACL represents relationship between user or group of users and some resource
- ACLs are attached to entries in objectspace

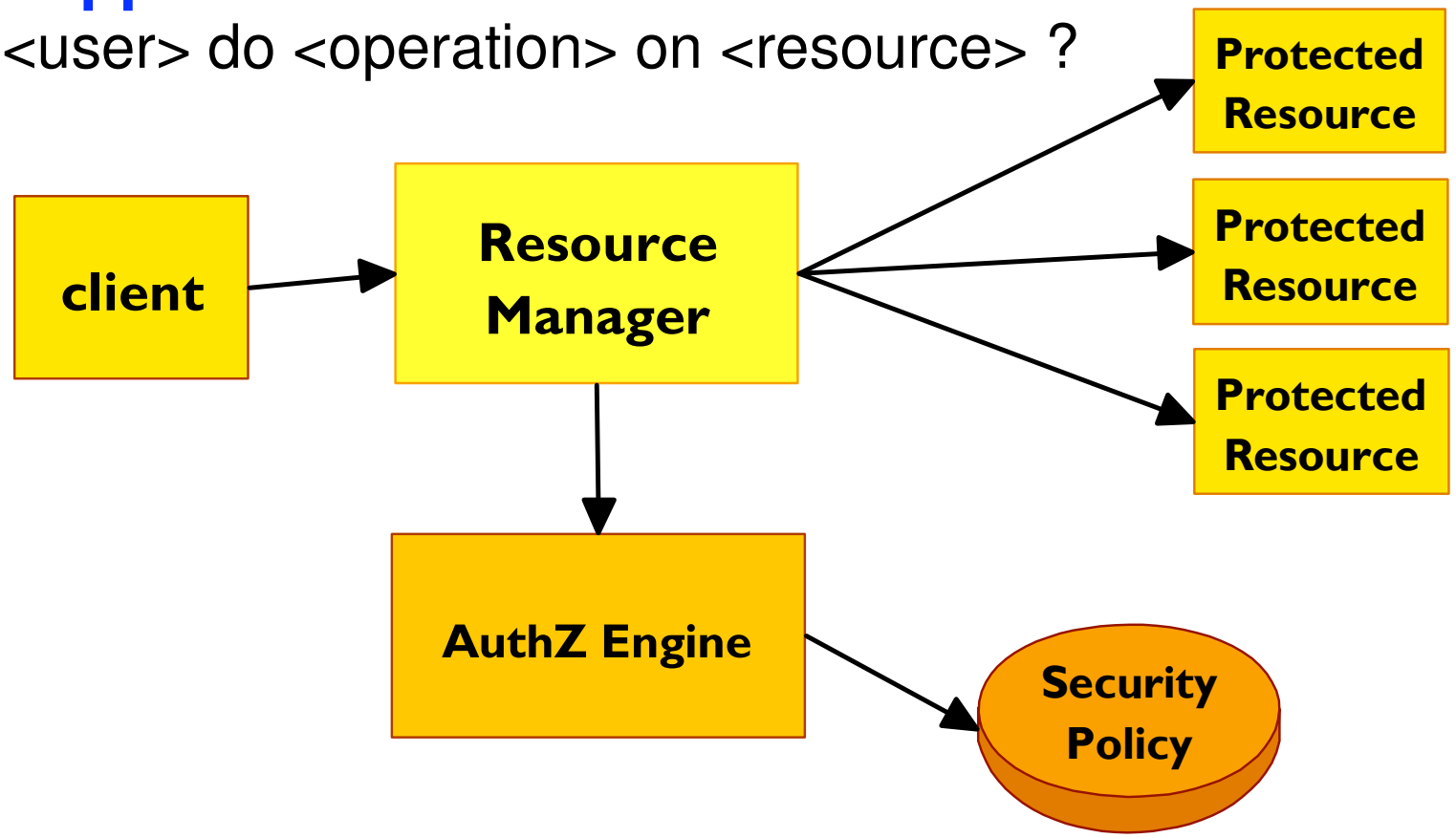
Provides an API for making authorization queries

- And provides exploiting applications (NetSEAL, WebSEAL, PD for MQ Series)

The Authorization API

Allows applications to ask

- ★ can <user> do <operation> on <resource> ?



An Open Group ratified standard programming model

Security Policy

Defines...

The participants

- Employees, customers, partners, anonymous

The objects requiring protection

- Web, TCP port, CORBA methods and interfaces, customer/application defined
- MQ Series queues, etc.

What actions can be performed on the objects:

- Read, modify, administer, execute, queue
- Can also be customer/application defined

Access Control List (ACL)

Authorization is the process of determining a subject's rights to perform an operation on any given object

- Rights are stored in the Tivoli Policy Director authorization policy database in the form of Access Control Lists (ACLs)
- If permissions on a resource are compatible with a user's credentials (from the User Registry), Tivoli Policy Director will respond that the user request should be allowed
- ACL templates can also be extended to include customer defined objects and actions

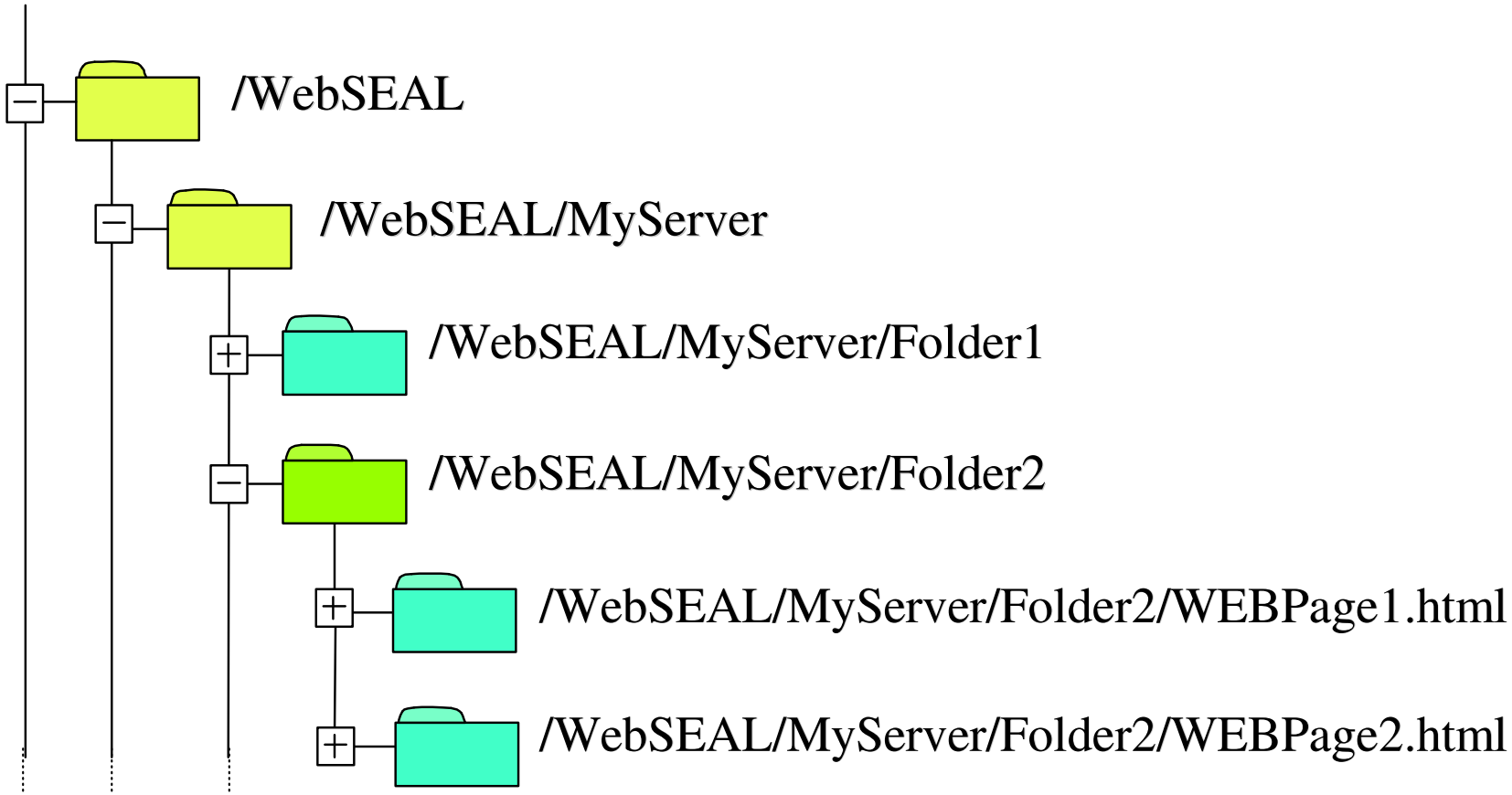
User Credentials

- **Built from information in the user registry (LDAP)**
- **Independent of the authentication mechanism**
  - Username/Password
  - Token
  - X.509 Certificate
  - etc

- **Contains information about the user**
  - Universally Unique Identifier (UUID)
  - Group membership (group UUIDs)
- **Used by the Policy Director Authorization Engine to render an access decision**

The Protected Objectspace

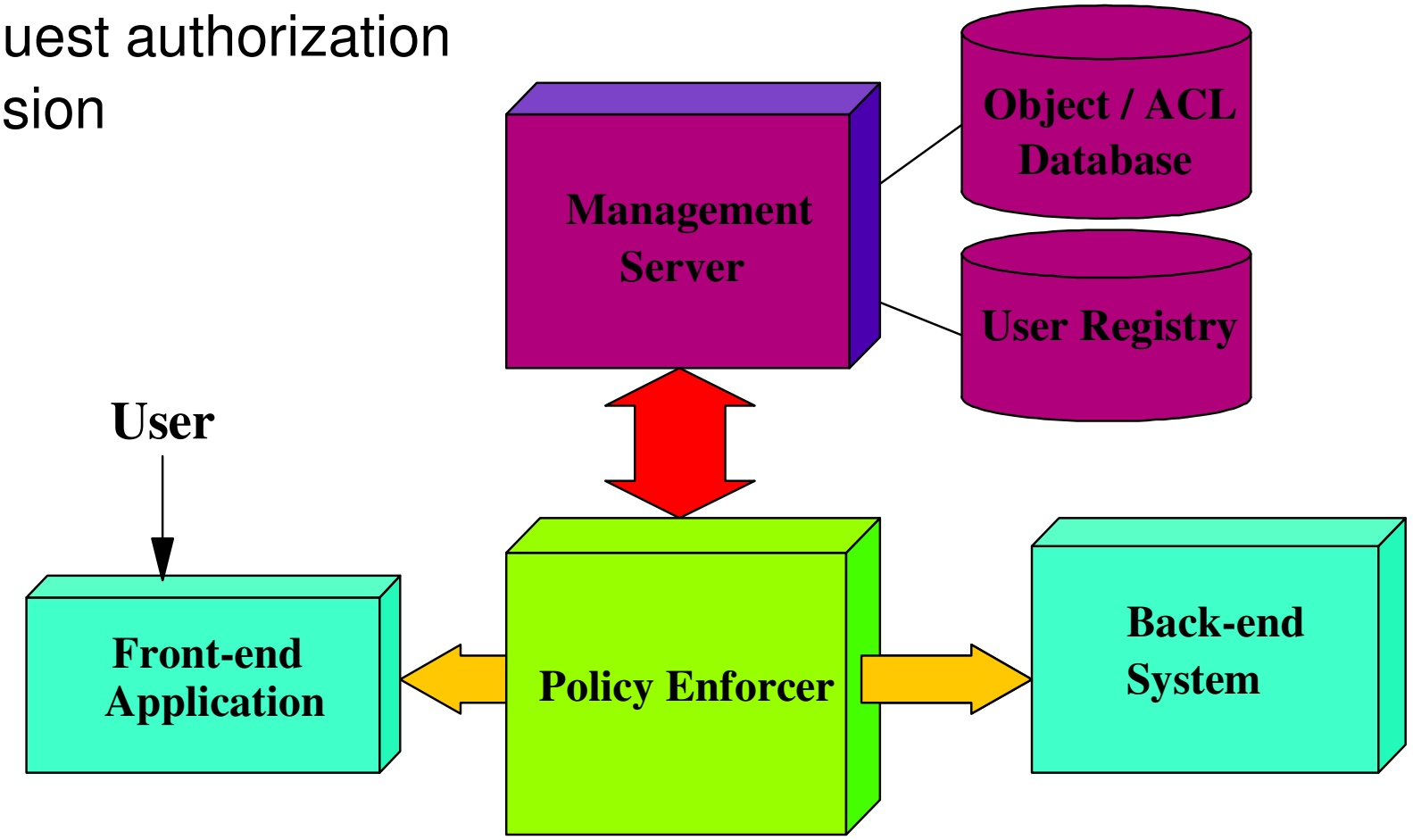
- **Hierarchy of protected objects, maintained in the master authorization policy database**
- **Rules can be defined explicitly or at higher levels and inherited by lower levels**



Where does Policy Director fit?

Policy Enforcer invokes API to

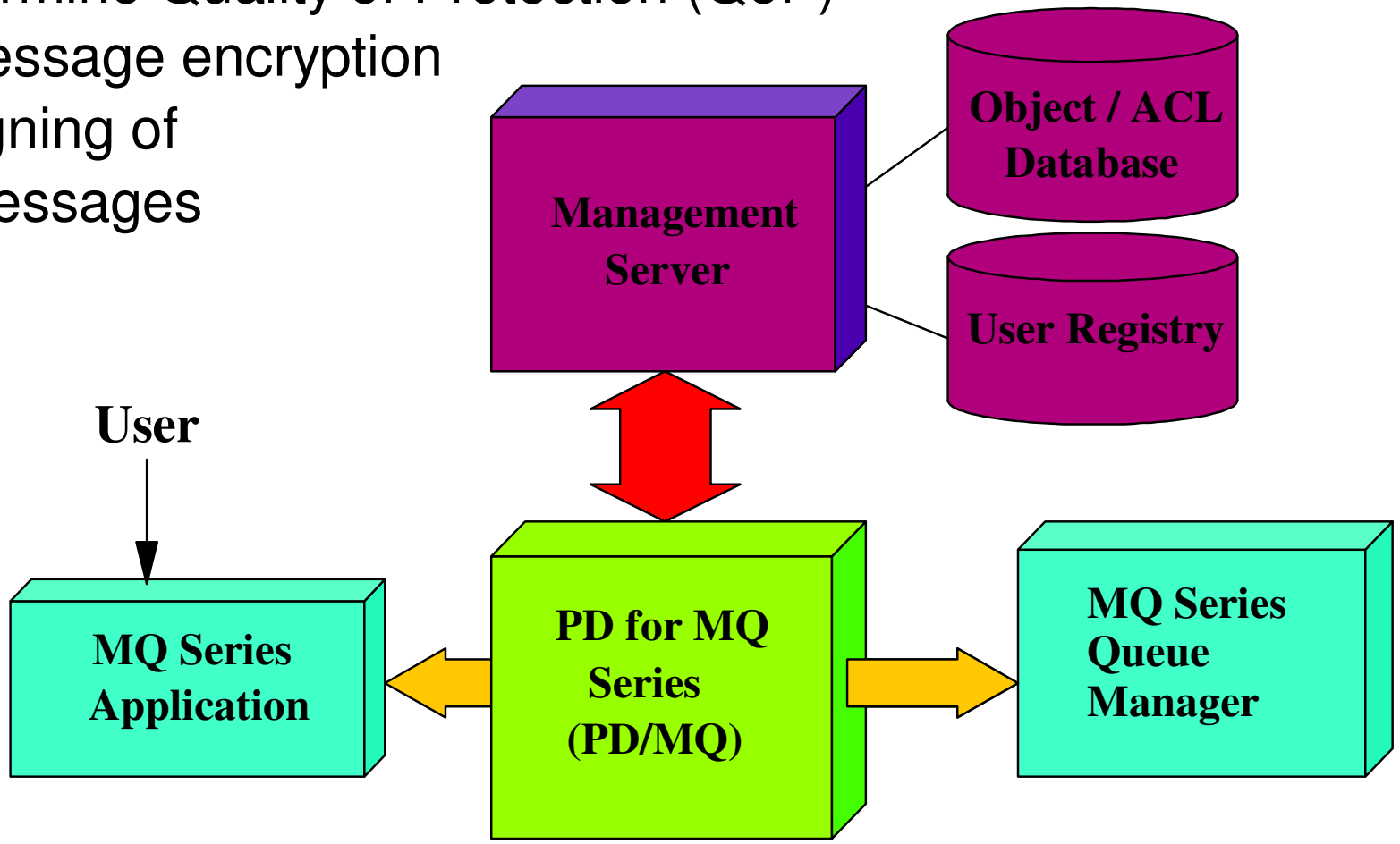
- Obtain user credentials
- Request authorization decision



For example...

Security policy can be used to

- Control access to queues
- Determine Quality of Protection (QoP)
  - message encryption
  - signing of messages



Summary

Tivoli Policy Director

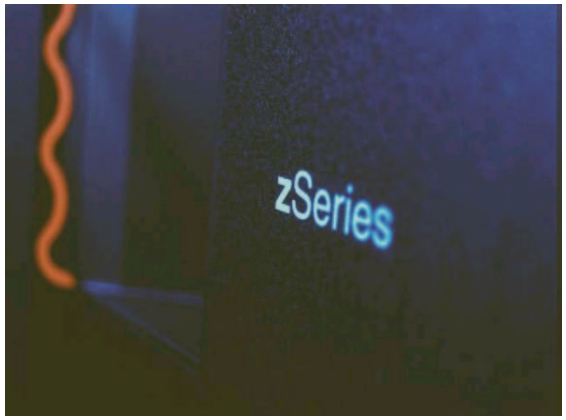
- Is a distributed e-business security solution
- Provides a mechanism for defining common security policy
- Provides a standardized interface for applications to determine user authorization rights to resources
- Supports many different operating system platforms
- Has an extendable model
- Also provides out-of-the-box solutions (WebSEAL, NetSEAL, PD for MQ Series ...)

Tivoli software

IBM Policy Director Authorization Services for z/OS and OS/390

IBM Policy Director Authorization Services for z/OS and OS/390

- **Provides support for PD-enabled applications on z/OS and OS/390**
  - Runs on OS/390 V2R10 and z/OS V1R1 or later
  - Independent product, free to current licensees of z/OS and OS/390
- **Allows z/OS and OS/390 to plug-and-play in an existing Tivoli Policy Director secure domain**
- **pdacld is ported to USS and handles replication of policy data**
- **PD applications invoke new SAF services for authorization decisions**

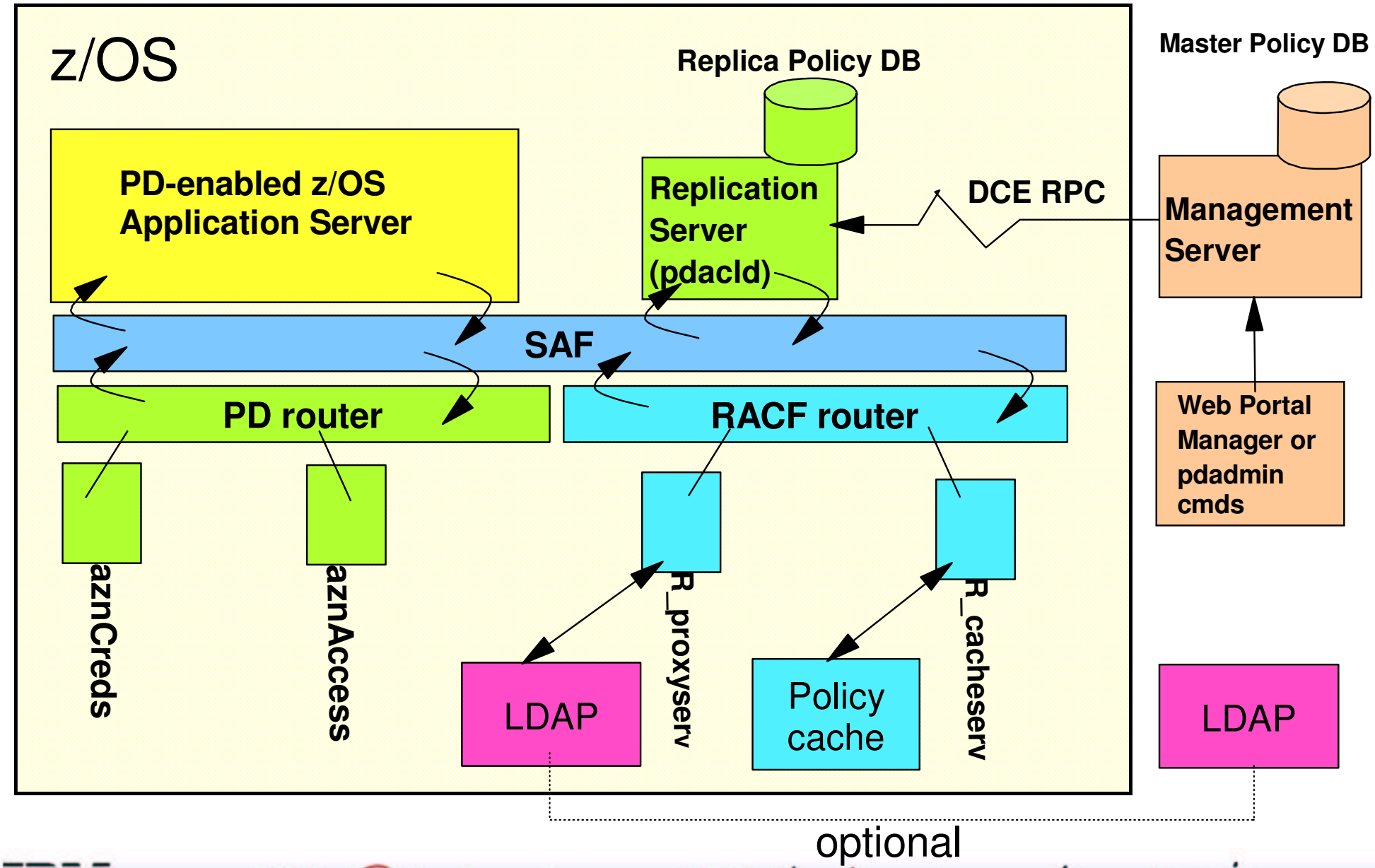


Benefits

- **Extends the value of Tivoli Policy Director to the z/OS and OS/390 platforms**
  - Allows customers to leverage existing Tivoli Policy Director security policies and LDAP user registry information
  - Provides infrastructure required by PD/MQ on z/OS and OS/390

- **New SAF authorization interfaces allow a user to be identified via host identity (ACEE)**
  - Establishes a relationship between native identity and Policy Director identity (typically an LDAP DN)

Components


z/OS
Replica Policy DB
Master Policy DB
PD-enabled z/OS Application Server
Replication Server (pdacld)
DCE RPC
Management Server
SAF
PD router
RACF router
Web Portal Manager or pdadmin cmds
aznCreds
aznAccess
R_proxyserv
R_cacheserv
LDAP
Policy cache
LDAP
optional

How does pdacld fit in a Policy Director domain?

- **Tivoli Policy Director function is included in the box with Policy Director Authorization Services**
  - Base Services for AIX (Version 3.7.1)
  - Base Services for Windows (Version 3.7.1)
  - Base Services for Solaris (Version 3.7.1)
  - Base Services for HP-UX (Version 3.7.1)
  - Web Portal Manager (Version 3.7.1)

- **Management Server must be installed on workstation of choice**

Policy administration is off platform

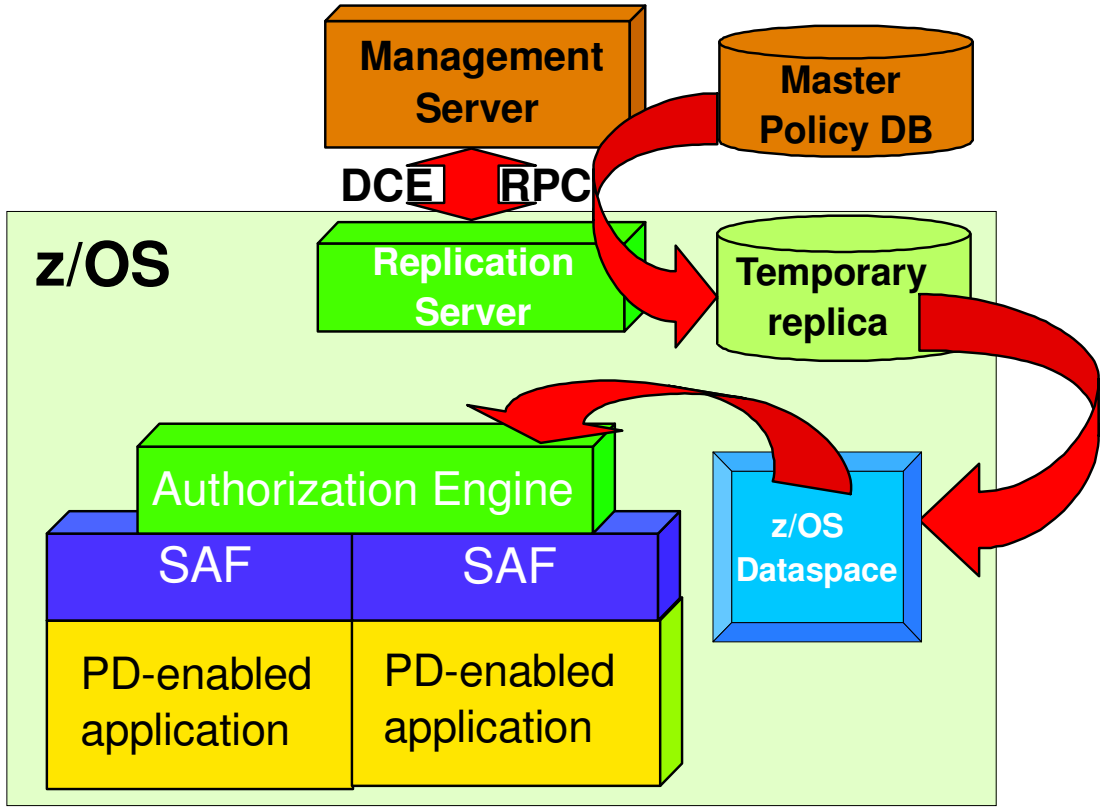
- **Web Portal Manager is a Web-based GUI used to manage security policy**
  - Provides management and administration of users, groups, roles, permissions, and policies
  - Use of Web Portal Manager is optional

- **The pdadmin command line utility also provides full support for all administrative tasks**

Replication Server (pdacld) on z/OS

Instance of pdacld process hosted on z/OS

- Runs as UNIX System Services process
- Communicates via DCE secure RPC to PD Management Server on a distributed platform
- Interacts with PD 3.7.1
- Maintains replica ACL database
- Also enhanced to invoke SAF service for mirroring policy to in-memory cache (for aznAccess)



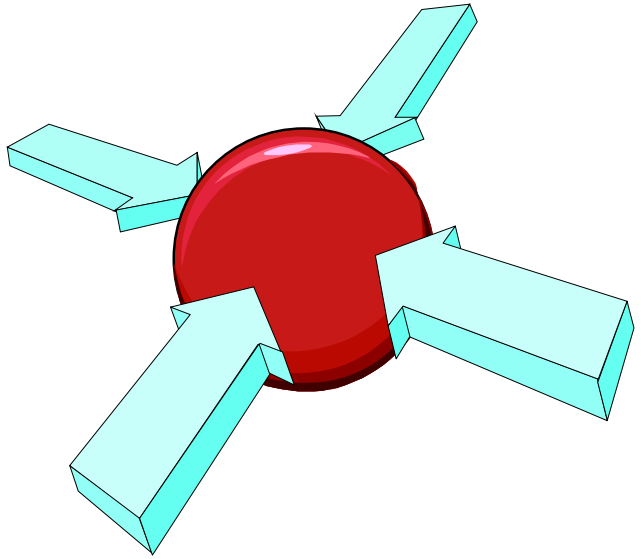
Getting Started

- The z/OS LDAP server must be started and configured
- The z/OS host system must be configured into a DCE cell
- Tivoli Policy Director 3.7.1 must be installed and configured
- A Tivoli Policy Director secure domain must be installed and configured
- z/OS must join that domain
- The z/OS pdacld daemon must be running and configured into the same DCE cell as the off-platform Management Server

How does LDAP fit in?

Support for OS/390 V2R10 and z/OS V1R2 provided by APAR OW50971 includes

- **Program Call (PC) interface**
  - Supports LDAP requests without requiring LE environment
  - Not a full implementation - limited set of Policy Director Authorization Services functions supported



How Does LDAP fit in? (cont)

- **Extended Operation (ExOp) function**
  - Provides functionality of multiple LDAP search calls with a single request
  - 2 ExOps are provided for Policy Director Authorization Services
    - Get DN for user ID
    - Get user privileges
  - ExOp must be defined in slapd.conf
  - Can support Local or Remote LDAP
    - LDAP hosted on z/OS
    - LDAP running on a distributed server

Aside: Mainframe LDAP now supported as PD User Registry for PD 3.7.1 and PD 3.8

How does RACF fit in?

Support for OS/390 V2R10 and z/OS V1R2 provided by APAR OW49959/OW49960 includes

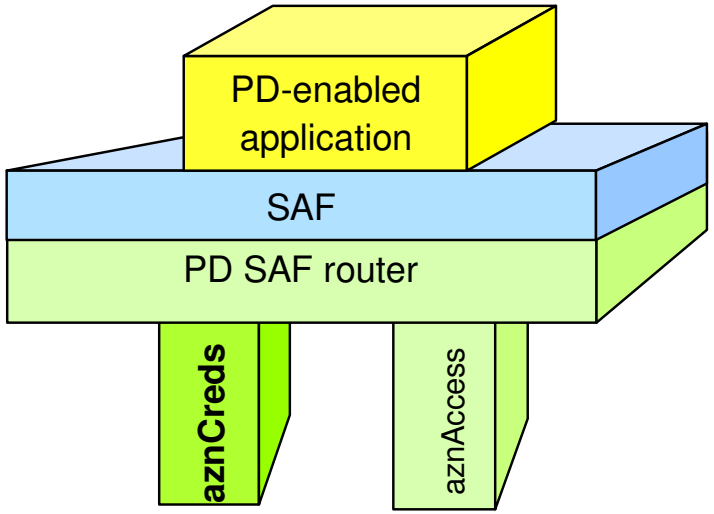
- **Management of the local policy cache** (R_cacheserv)
- **Retrieval of user credential information from LDAP** (R_proxyserv)
  - Exploits new LDAP Program Call and ExOp functionality

aznCreds and aznAccess callers not in system key or supervisor state must be authorized to the IRR.AZNSERVICE profile in the FACILITY class

aznCreds (IRRSZC00) SAF Callable Service

Allows z/OS and OS/390 application servers to perform credential related services

- Equivalent to azn_id_get_creds and azn_creds_delete C language function
- Invokers may supply either Policy Director credential or host user identity (ACEE)

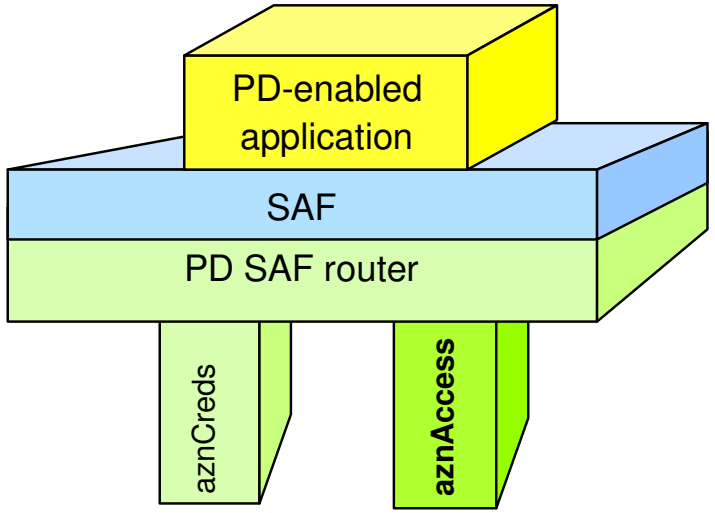


- Credential can also be requested for default (unauthenticated) identity

The Virtual Lookaside Facility (VLF) can be configured to cache credentials for enhanced performance

aznAccess (IRRSZA00) SAF Callable Service

Allows z/OS and OS/390 application servers to make access decisions using Tivoli Policy Director distributed ACLS



- Equivalent to azn_decision_access_allowed C language function
- Invokers may supply either Policy Director credential or host user identity (ACEE)
- Will write SMF Type 80 records if audit requested in protected object policy (POP)
- Protected resource attribute and POP values can also be optionally returned

Summary

IBM Policy Director Authorization Services for z/OS and OS/390

- Can leverage existing Tivoli Policy Director policy and LDAP user registry information (plug-and-play)
- The Replication Server is ported and communicates with an off-platform Management Server
- A Security product which supports both the R_cacheserv and R_proxyserv SAF callable services is required
- Applications can obtain credentials using the aznCreds SAF callable service and request an access decision using the aznAccess SAF callable service
- This infrastructure is exploited by PD/MQ on z/OS and OS/390

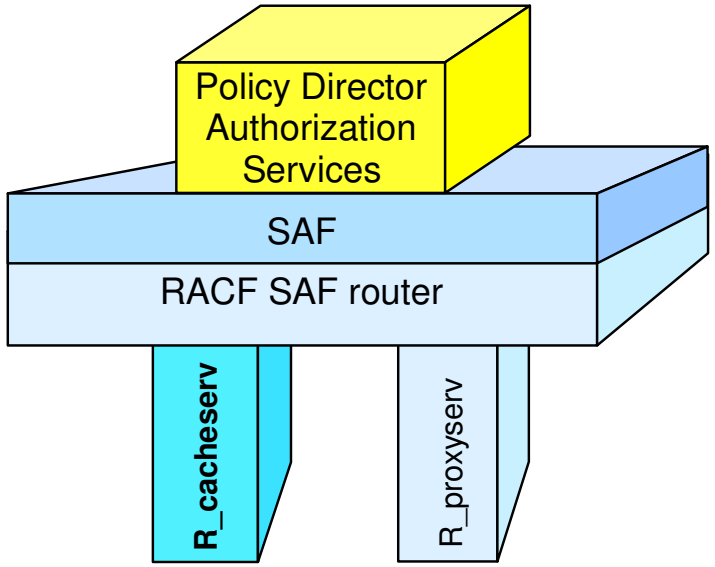
Security Server (RACF) Support and SAF Enhancements for Policy Director Authorization Services for z/OS and OS/390

Security Server (RACF) Support

- **The Security Server (RACF) provides infrastructure required by Policy Director Authorization Services**
  - R_cacheserv for management of the local policy cache
  - R_proxyserv for extract of User registry information from LDAP
- **The RACF database can be used for harden and restore of the Policy Director Authorization Services local policy cache**
- **The RACF SMF Unload utility will unload Policy Director Authorization Services SMF audit records**
- **SAF trace support includes not only R_cacheserv and R_proxyserv, but also aznCreds and aznAccess**

R_cacheserv (IRRSCH00) SAF Callable Service

- **Store and retrieve security relevant information from a cache**
- **RACF implements the cache using a dataspace**
- **Optionally harden (backup) to the security product (RACF) database**
- **Functions:**
  - Start a new cache
  - Add a record to the new cache
  - End cache creation
  - Fetch a record from the cache
  - Delete the cache



New Class

- CACHECLS profiles control save and restore of the R_cacheserv cache from the RACF database
- Save (hardening) to the RACF database requires that CACHECLS be active and a base profile defined with the same name as the cache
- Similar to processing done for the RACGLIST class

New Class (cont)

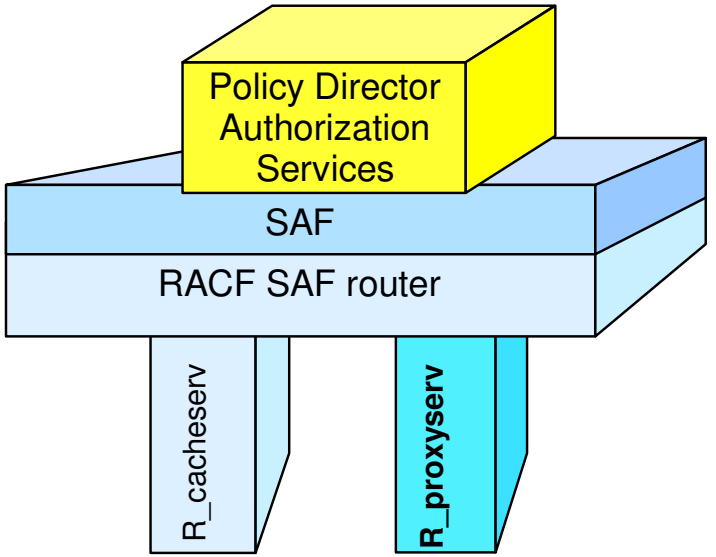
Policy Director Authorization Services recommends that customers enable hardening of the local policy cache

```
RDEFINE CACHECLS RZPDIR
SETROPTS CLASSACT(CACHECLS)
```

This allows the local policy cache to be backed up to the RACF database and restored for an aznAccess decision prior to start of pdacld

R_proxyserv (IRRSPY00) SAF Callable Service

- **Obtains data from an LDAP directory**
- **Invokers are not required to be LE-enabled**
- **Not a generic front-end to LDAP, at this point**
- **Relies on LDAP Program Call and ExOp function**
- **Requires LDAP bind info (URL, DN, PW)**
- **Functions:**
  - Return the distinguished name (DN) of the user identified by the host user identity
  - Return the Tivoli Policy Director attributes for the specified DN



User and General Resource Commands

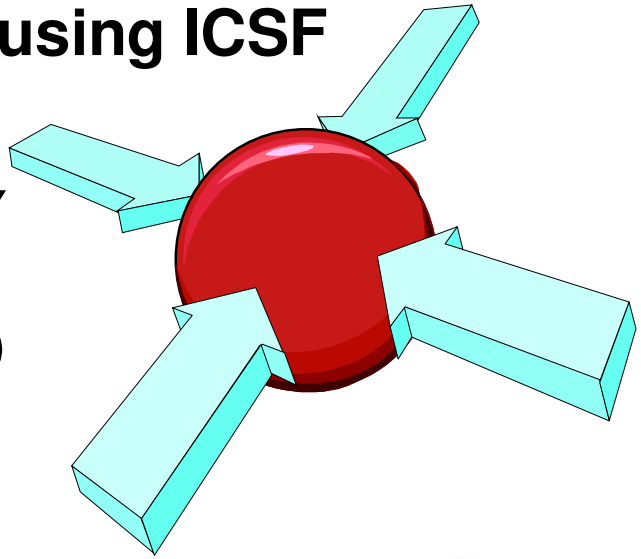
- Can be used to associate PROXY information with a server's (started task's) RACF user identity
- Alternatively, the IRR.PROXY.DEFAULTS profile in the FACILITY class can be used for defining PROXY information for all servers
- PROXY segment values can be specified on ADDUSER, ALTUSER, RDEFINE, RALTER, and displayed via LISTUSER and RLIST
- LISTUSER and RLIST will display BINDPW=YES or NO, indicating whether or not the password has been specified
- New fields are added to the RACF templates (IRRTEMP1), so IRRMIN00 PARM=UPDATE must be run

Defining PROXY segments

Policy Director Authorization Services requires that customers define PROXY information to identify the LDAP directory

- **Information required to locate (URL) and authenticate/BIND to the directory (DN and password)**
- **Define a KEYSMSTR profile to specify how the BINDPW will be stored (masked or encrypted using ICSF cryptographic services)**

```
RDEFINE KEYSMSTR LDAP.BINDPW.KEY
    SIGNON(KEYMASKED(key-value )
                | KEYENCRYPTED(key-value))
SETROPTS CLASSACT(KEYSMSTR)
```



Defining PROXY segments (cont)

- **Add a PROXY segment to the User profile of the server (started task) that will be invoking aznCreds**

```
ADDUSER mrserver
  PROXY(LDAPHOST(LDAP://SOME.LDAP.HOST:389)
  BINDDN('cn=Joe User,ou=Poughkeepsie,o=IBM,c=US')
  BINDPW(MYPASS1))
```

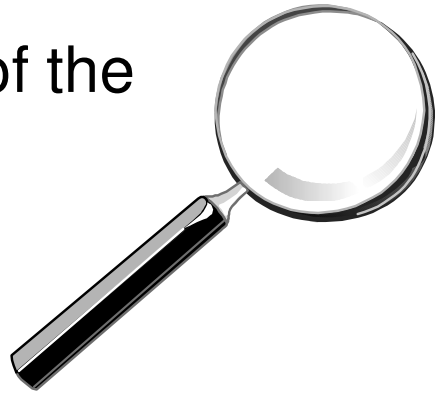
- **Or add a PROXY segment to a FACILITY class profile that will be used as a default for all servers**

```
RDEFINE FACILITY IRR.PROXY.DEFAULTS
  PROXY(LDAPHOST(LDAP://SOME.LDAP.HOST:389)
  BINDDN('cn=Joe User,ou=Poughkeepsie,o=IBM,c=US')
  BINDPW(MYPASS1)
```

SMF Data Unload Utility

The Policy Director Authorization Services aznAccess SAF callable service writes SMF Type 80 audit records when audit is requested

- IRRADU00 will unload this information along with RACF Type 80 information
- Similar to support provided for Network Authentication Services component of the Security Server
- The minimum logical record length (LRECL) of the RACF SMF data unload utility (IRRADU00) output dataset (OUTDD) is increased from 5096 to 8192



The SET Command

Can be used to enable or disable SAF Trace for the new SAF services: R_cacheserv, R_proxyserv, aznCreds, and aznAccess

- SAF trace copies the parmlists of the services traced into a GTF record before and after the service runs
- SAF Trace and the SET TRACE command are described in the SecureWay Security Server RACF Diagnosis Guide and Command Language Reference publications
- Trace records are intended only for diagnosis use when requested by the IBM support center

Activating a Trace

[subsystem-prefix] SET [TRACE(
[RACROUTE(ALL | NONE | TYPE(t1, t2,...)) |
NORACROUTE] |

[DATABASE ([ALL | NONE] | [ALTER |
NOALTER] | [ALTERI | NOALTERI] | [READ | NOREAD])
|
NODATABASE] |

[CALLABLE(ALL | NONE | TYPE(t1, t2, ...)) |
NOCALLABLE]

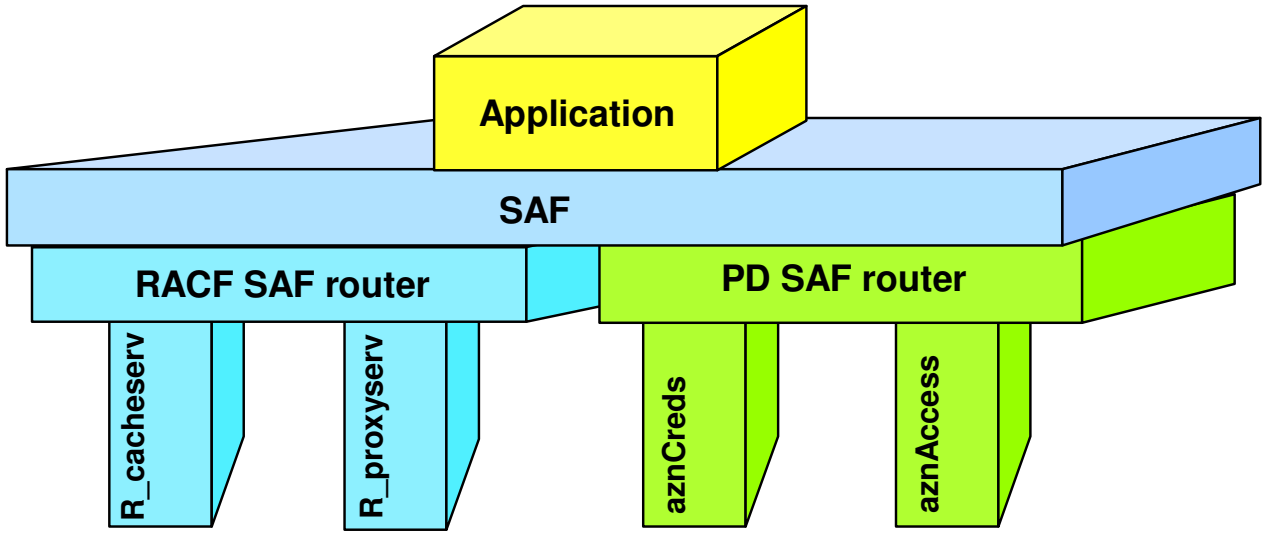
NEW

[PDCALLABLE(ALL | NONE | TYPE(t1,t2,...)) |
NOPDCALLABLE}

[ASID(asid1, asid2, ... | *) | NOASID | ALLASIDS]
[JOBNAME(jobname1, jobname2, ... | *) |
NOJOBNAME | ALLJOBNAMES]

SAF Support

- The SAF interface has been extended for the new aznCreds and aznAccess SAF callable services



- Control structure is modeled after existing SAF/RACF callable service support for services such as R_admin, initACEE, etc.

Summary

The Security Server (RACF)

- Provides required infrastructure for Policy Director Authorization Services
- Provides a secure data store for harden and restore of the local policy cache
- Integrates Policy Director Authorization Services with other security related SMF audit information unloaded by the SMF Unload utility
- Supports enablement of SAF Trace for new Policy Director Authorization aznCreds and aznAccess callable services

SAF has been extended to provide an interface for Policy Directory Authorization Services SAF Callable Services which is consistent with prior Security Server support

Questions???