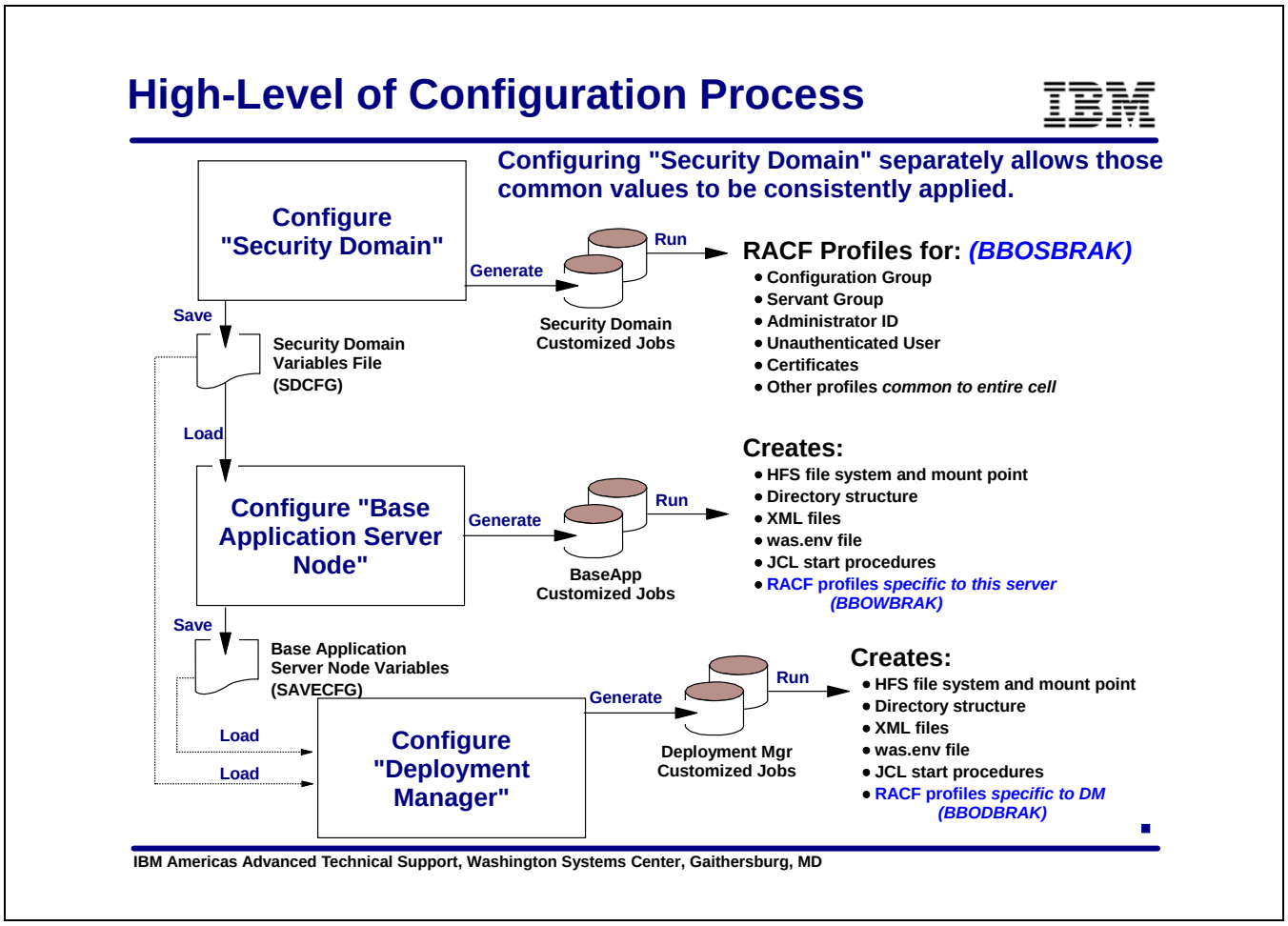
WebSphere Application Server
for z/OS and OS/390

WebSphere V5 for z/OS

Peeling Back the Layers of the RACF Jobs and Commands

IBM Americas Advanced Technical Support -- Washington Systems Center
Gaithersburg, MD, USA

(This page intentionally left blank)



Security Domain

As of WebSphere Application Server 5.02 there is an additional component of the ISPF build dialogs used to specify security information that will apply across the entire cell. This applies to both a Base Application Server configuration as well as a Network Deployment Configuration.

The Security Domain panels are used to provide RACF group IDs, an administrator ID, an ID to be used for unauthenticated users, a security domain name and certain certificate information.

The panels will generate two jobs in the .CNTL data set that you specify.

- BBOSBRAJ - a job to generate the RACF REXX job.
- BBOSBRAK - A job to run the REXX script created in the BBOSBRAJ job.

The variables set in the Security Domain panels must be saved in a .SDCFG data set. These variables will then be input to the ISPF dialogs used to create a Base AppServer and the ISPF dialogs used to create a Deployment Manager. This is to assure that the cell-wide security definitions are consistent.

When you run through the ISPF dialogs for the Base AppServer You must load the saved variables from the security domain, .SDCFG data set, as input. There will be two security related jobs created in your .CNTL data set. These jobs are generated using variables specified in the Base AppServer dialogs as well as variables saved in the Security Domain dialog.

- BBOWBRAJ - a job to create the REXX script containing all of the RACF definitons.
- BBOWBRAK - a job to execute the REXX script just created.

When you run through the ISPF dialogs for the Deployment Manager you provide as input the variables saved in the Base AppServer run and the variables saved in the Security Domain run. There will be two security related jobs created in your .CNTL data set.

- BBODBRAJ - a job to create the REXX script containing all of the RACF definitons.
- BBODBRAK - a job to execute the REXX script just created.

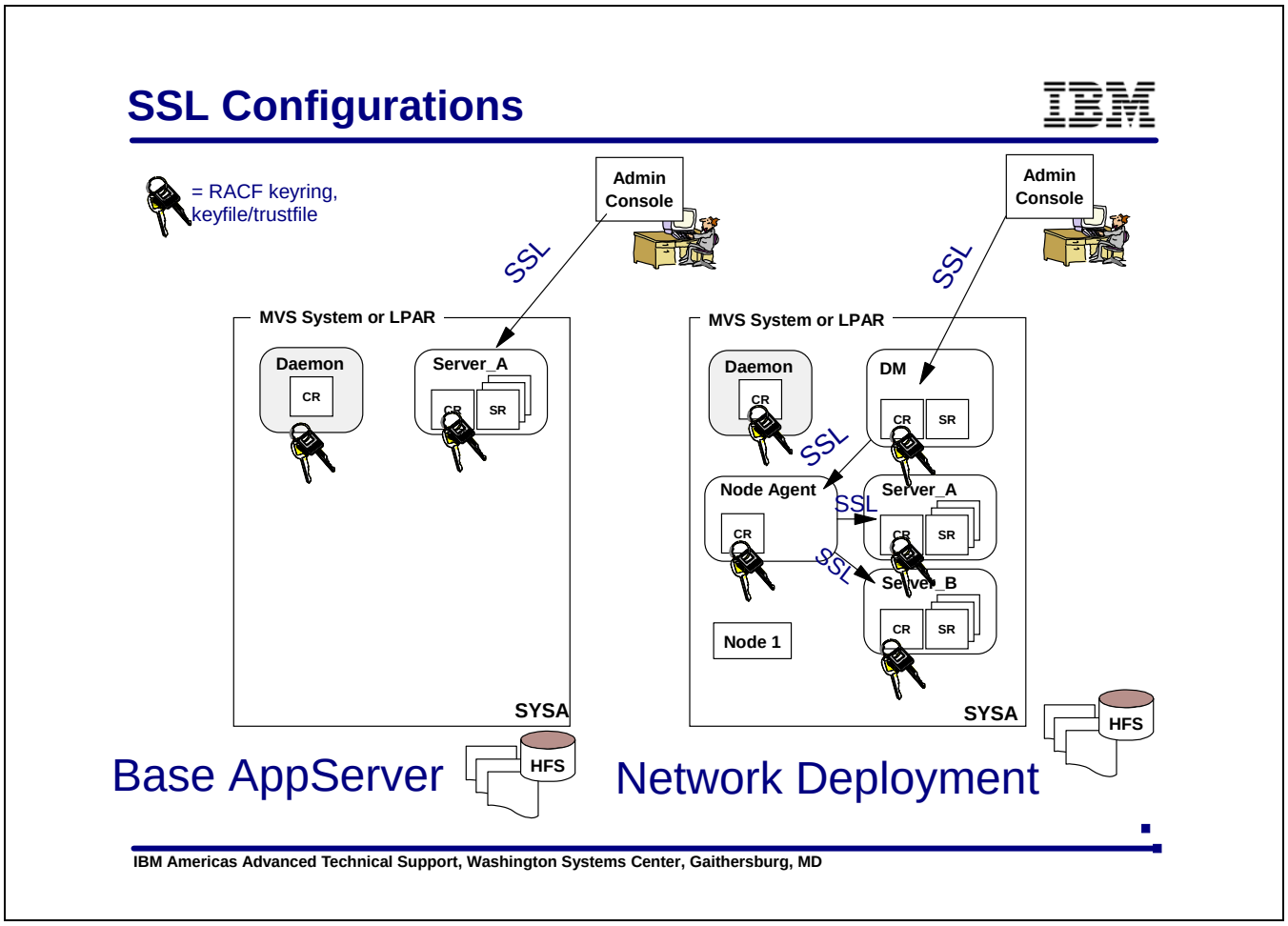
Output of Customization Scripts

- **The BBOxBRAK jobs contain RACF commands to create:**
  - Userids and groups used by the cell.
  - STARTED class profiles
  - CBIND class profiles
  - SERVER class profiles
  - EJBROLE class profiles
  - APPL class profile (optional)
  - PKTDATA class profile (optional)
  - Keyrings and Digital Certificates
  - FACILITY class profiles
- **We'll review these in the following charts.**

IBM Americas Advanced Technical Support, Washington Systems Center, Gaithersburg, MD

Each of the paths through the ISPF dialogs creates jobs to generate and run RACF commands. The RACF commands will create many different profiles and keyrings in RACF. These command are sufficient to set up an environment to run WebSphere whether Global Security is enabled or not.

In the charts to follow we will show where in the process these profiles are created and how they are used.



When global security is turned on in WebSphere Application Server, SSL will be used for much of the conversations.

The ISPF dialogs will create keyrings in RACF for each of the servers in your configuration. In the Base Application Server there is a keyring created for the application server and optionally one for the Daemon. In the Network Deployment Configuration there will be a keyring for the Deployment Manager and one for the Node Agent as well.

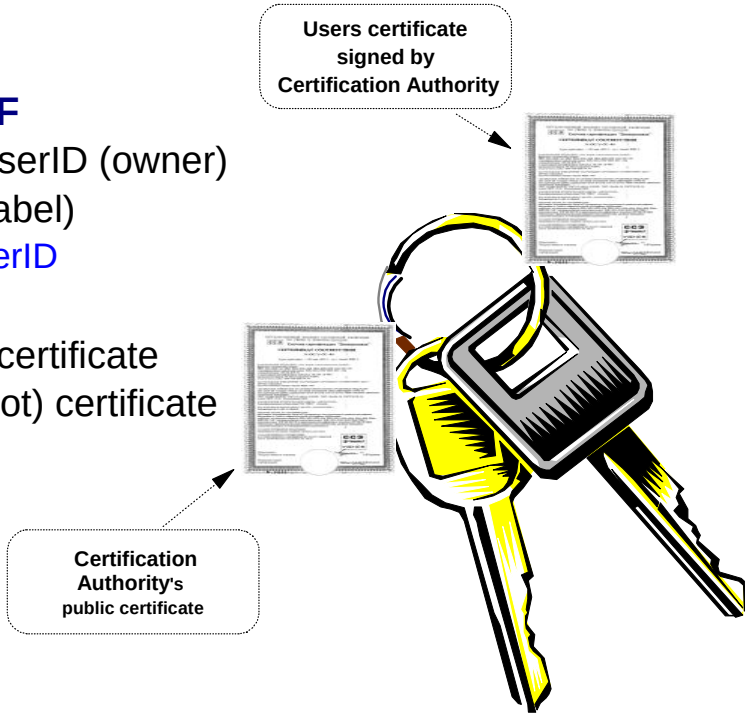
Keyring

Stored in RACF

- Identified by UserID (owner)
- Has a name (label)
  - unique by UserID

Contains

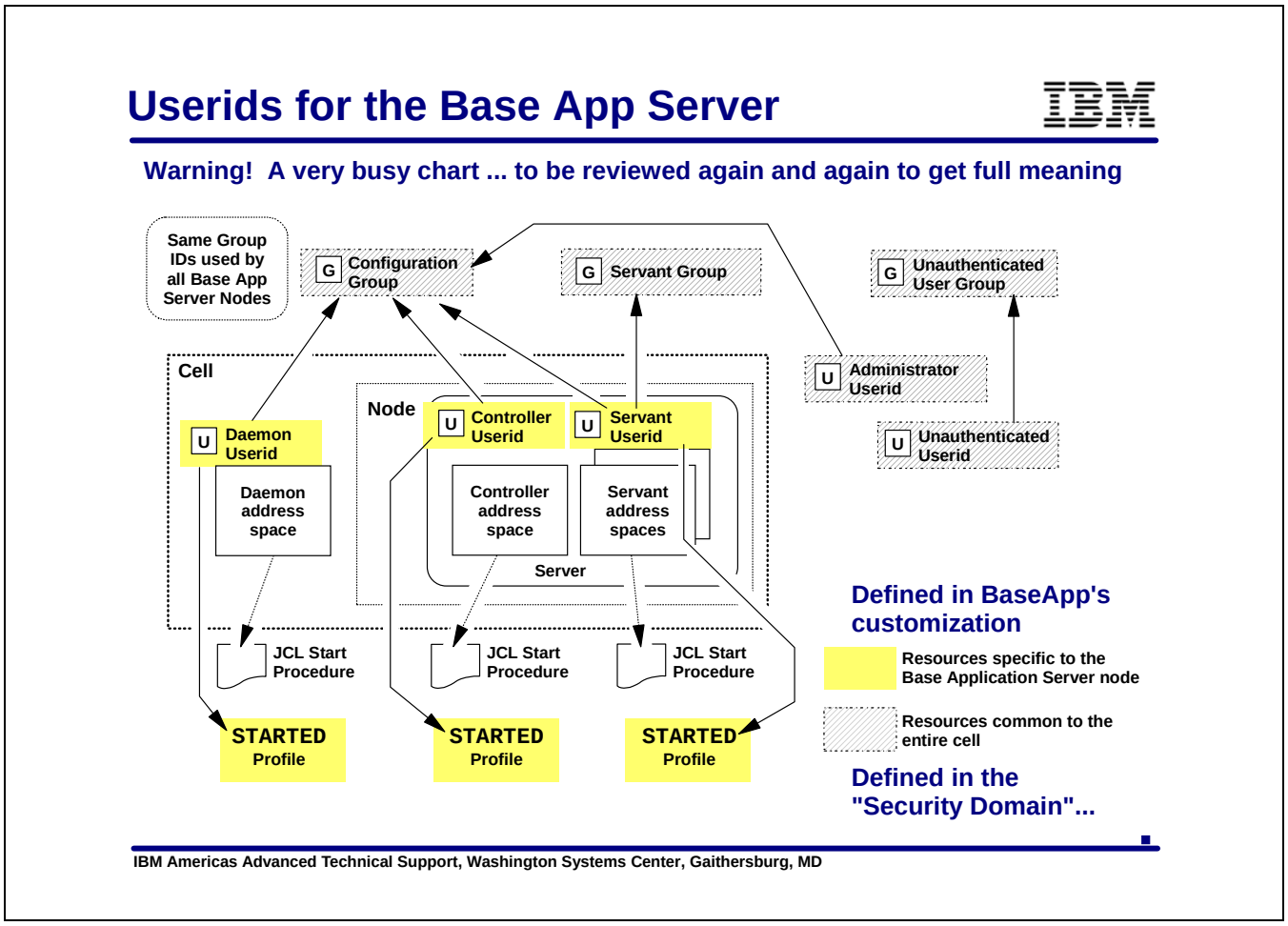
- User's private certificate
- CA's public (root) certificate



IBM Americas Advanced Technical Support, Washington Systems Center, Gaithersburg, MD

Keyrings are created and stored in RACF using the RACDCERT ADDRING command. They are stored by UserID and have a name associated with them. They will be used to hold the certificates associated with that UserID and the certificate of the Certification Authority used to sign the server's certificate. For example if I were to create a keyring for a server whose ID was WASW5C and I chose to call the keyring WASW5kring, I woould issue the following RACF command:

```
RACDCERT ADDRING (WASW5kring) ID( WASW5C )
```

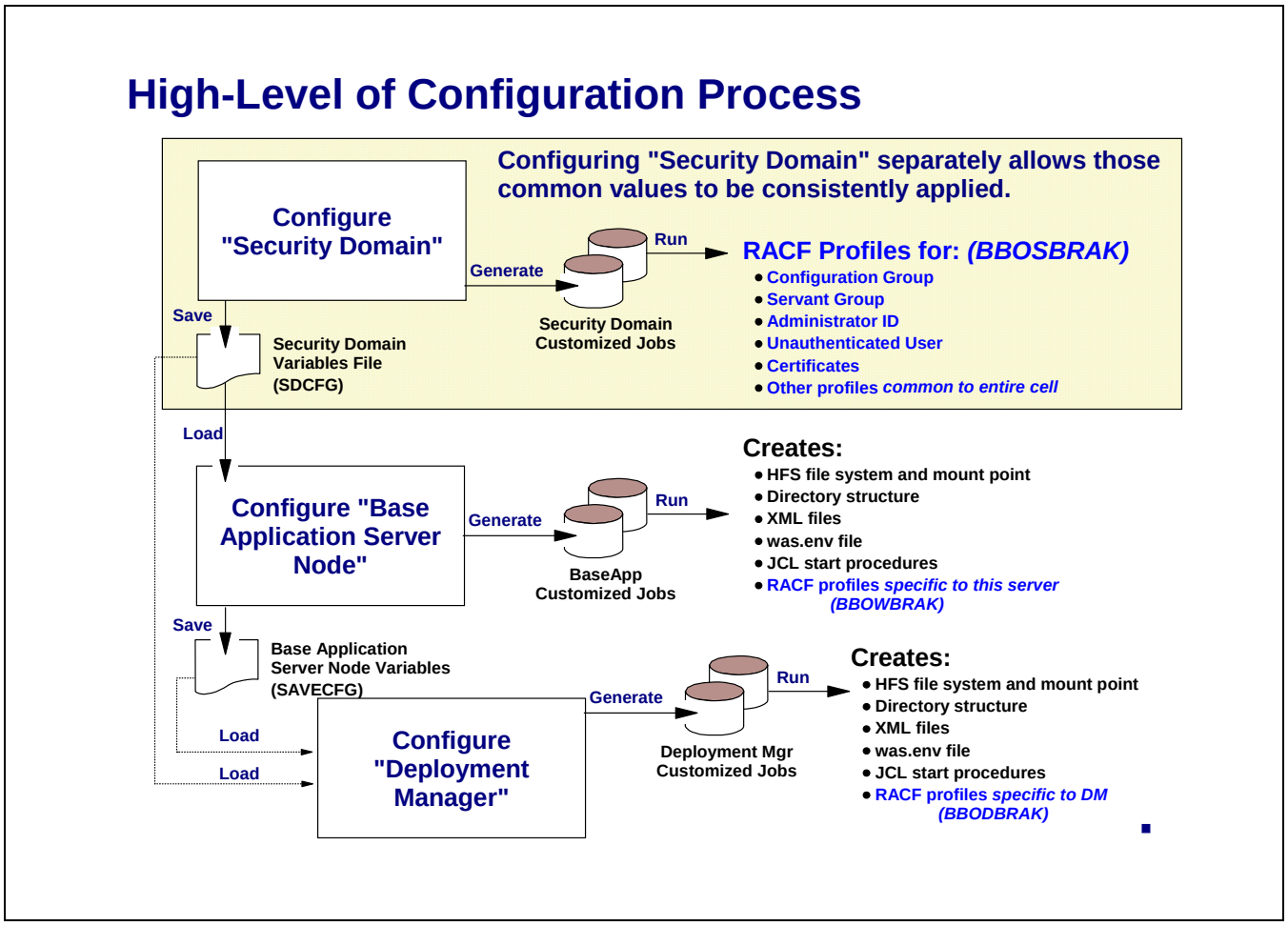


In a Base AppServer configuration there are three address spaces, the Daemon, the Controller and the Servant. Each address space has a PROC and will be associated with a UserID using a profile in the STARTED class. We will discuss STARTED class profiles, which are built during the Base AppServer build phase, later on.

In the BBOSDRAK job created for the Security Domain there will be definitions for RACF Groups used throughout the cell. The configuration group owns the configuration HFS and will have READ/WRITE access to that HFS. So that the Daemon, Controller and Servant can access the files they need in the HFS, they are all connected to the configurtion group.

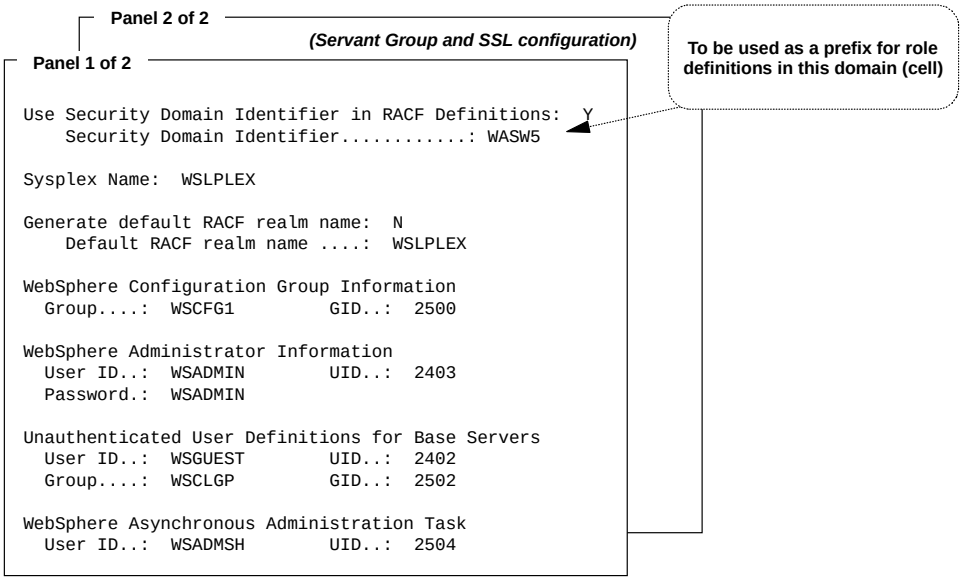
The Adminstrator ID is also defined and connected to the configuration group during the Security Domain portion of the build process.

The UserID and GroupID for unauthenticated users is also created at this point.



Each of the paths through the ISPF dialogs creates jobs to generate and run RACF commands. We will first take a look at the Security Domain panels. What do you need to specify? What will that generate for RACF?

Creating the Security Domain



Two panels that capture information about the security definitions common to the planned cell

"Planned cell" -- not just BaseApp, but into Network Deployment as well.

This is the first panel of the ISPF dialog to create the Security Domain.

This is where the configuration group, the administrator ID and initial password and the UserID for unathenticated users.

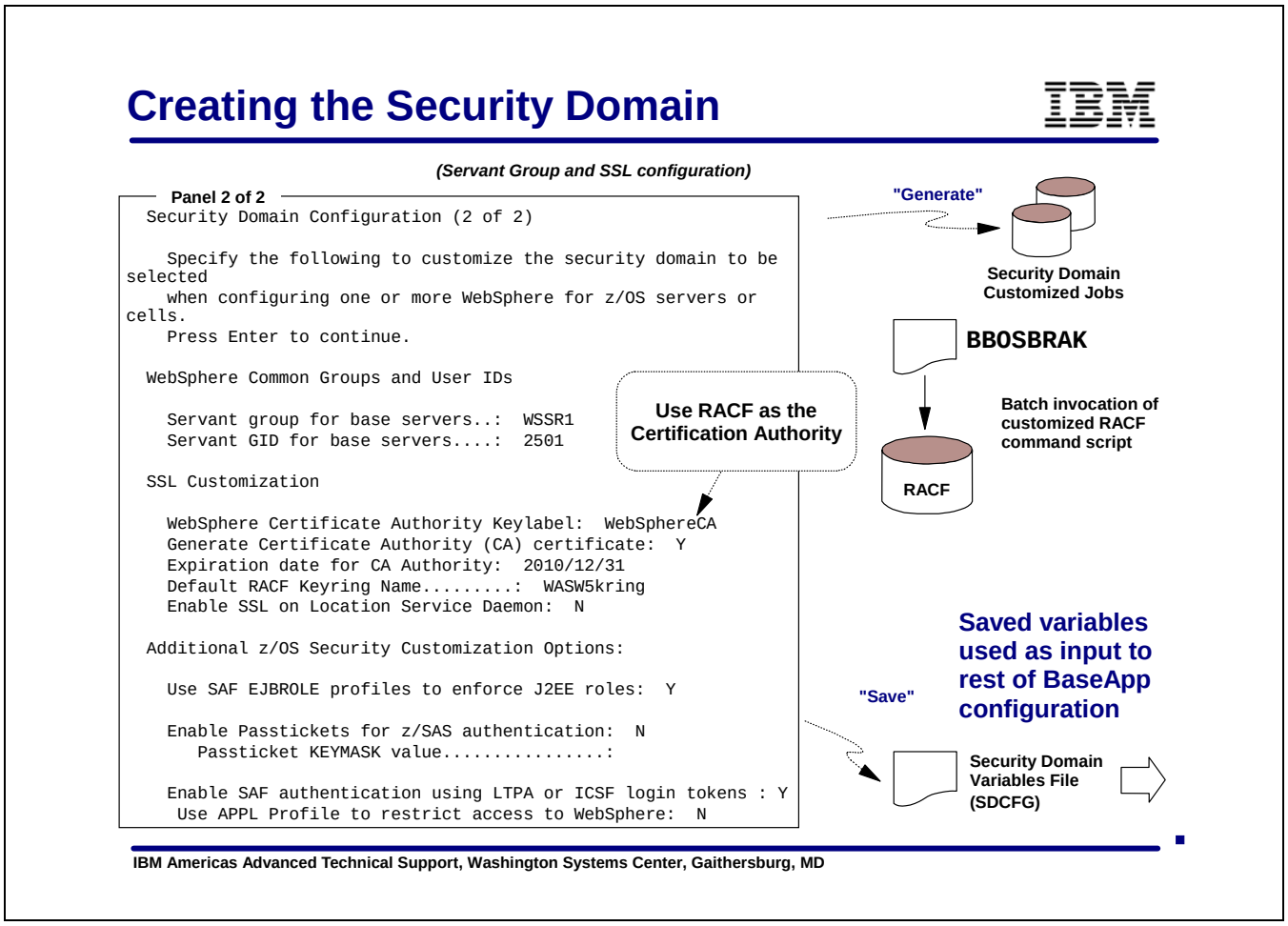
You have two other optional definitions on this pane the Security Domain Identifier and the RACF Realm name.

The Security Domain Identifier, if specified, is used as a prefix for EJBROLE profiles. This is so that you can differentiate between an administrator in one cell and an administrator in another cell. This feature is new with WebSphere 5.02. When WebSphere checks with RACF to see if a user is in a specified role, if the domain is specified then the check will be made for a profile of <domainID.rolename> instead of just using the rolename. No change need be made to the application or the deployment descriptors for the application.

More on EJBROLES later.

This also allows for an application that uses roles to be moved from one cell to another (test to production) without any application change.

The RACF Realm name is used in an environment where you are using kerberos to do your authentication.



Here is where you specify whether you want to use RACF as the Certification Authority and what the keyring name will be.

You can also specify whether to use SSL for the DAEMON. If you specify ‘Y” there will be a certificate generated for the DAEMON.

EJBROLES can either be defined and managed in RACF or defined and managed via the WebSphere Admin Console and maintained in xml files in the HFS. Specifying ‘Y’ will cause the dialogs to generate the appropriate RACF defintions for EJBROLES of

- <securitydomain.>administrator
- <securitydomain.>monitor
- <securitydomain.>operator
- <securitydomain.>configurator

If you did not specify a Security Domain value the EJBROLES generated and used for this cell will not contain a prefix.

If you specify ‘Y” for Enable Passtickets, you must specify a keymask value.

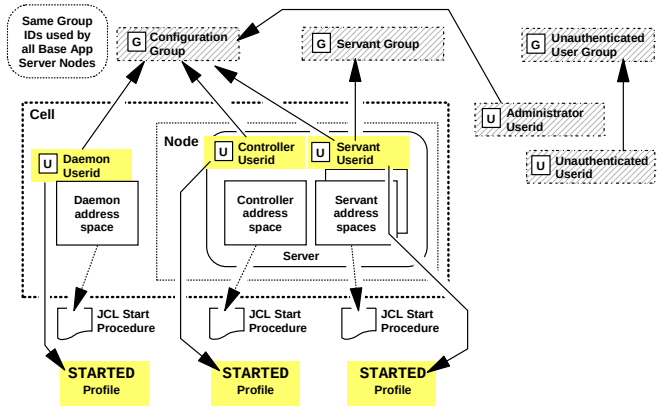
A PKTDATA class profile will be created and the PassTicket keymask value will be used by servers in the cell when handling PassTickets.

- If a Security Domain is specified, the PKTDATA profile name will be the Security Domain name.
- If no Security Domain is specified, the PKTDATA profile name will be CBS390.

Enable LTPA or ICSF for authentication - This will allow for WebSphere to create encrypted cookies for use in creating and passing login tokens. These tokens are used when implementing form based authentication.

You can also protect access to all applicatons in your WebSphere cell using the APPL profile in RACF. Specifying ‘Y’ to this option will result in an APPL profile to be created with the name of WebSphere Security Domain Identifier specified on the first panel. If you did not specify a Security Domain then CBS390 will be used for the APPL profile.

Define Common Group & User IDs




```
ADDGROUP WSSR1 OMVS(GID(2501))

ADDGROUP WSCFG1 OMVS(GID(2500))

ADDGROUP WSCLGP OMVS(GID(2502))
```

Administrator

Unauthenticated client

```
ADDUSER WSADMIN DFLTGRP(WSCFG1) OMVS(UID(2403) HOME(/tmp)
 PROGRAM(/bin/sh)) NAME('WAS ADMINISTRATOR')
 PW USER(WSADMIN) NOINTERVAL
ALU WSADMIN PASSWORD(WSADMIN) NOEXPIRED
ADDUSER WSADMSH DFLTGRP(WSCFG1) OMVS(UID(2504) HOME(/tmp)
 PROGRAM(/bin/sh)) NAME('WAS Asynch Admin Task')  NOPASSWORD
NOOIDCARD

ADDUSER WSGUEST RESTRICTED  DFLTGRP(WSCLGP) OMVS(UID(2402) HOME(/)
 PROGRAM(/bin/sh)) NAME('WAS DEFAULT USER')
```

IBM Americas Advanced Technical Support, Washington Systems Center, Gaithersburg, MD

These are the RACF commands generated in the BBOSDRAK member of the .DATA data set for group and user definitions. The names and UIDs are the defaults for these variables.

There are three group profiles created at this time.

- Configuration Group (WSCGF1) - all controller, servant and administrator UserIDs will be connected to this group. This groupID is also used as the group owner of the configuration HFS.
- Servant Group (WSSR1) - all servant UserIDs are also connected to this group.
- Unauthenticated Group (WSCLGP) - this is used for the unauthenticated UserID.

There are two userIDs created for administrator.

- WSADMIN for use when logging on to the admin console or executing administrator scripts.
- WSADMSH is used for asynchronous tasks initiated by funtions in the admin console.

The WSGUEST will be used by all application servers in the cell for unauthenticated client requests.

CA Certificate

```
/* Create CA Certificate for WebSphere Security Domain

RACDCERT CERTAUTH GENCERT
 SUBJECTSDN(CN('WAS CertAuth for Security Domain')  OU('WASW5.WebSphere for zOS'))
 WITHLABEL('WebSphereCA')
 TRUST NOTAFTER(DATE(2010/12/31))
```

Provide authority to read certificate and keyring

```
RDEFINE FACILITY IRR.DIGTCERT.LIST UACC(NONE)
RDEFINE FACILITY IRR.DIGTCERT.LISTRING UACC(NONE)
PERMIT IRR.DIGTCERT.LIST CLASS(FACILITY) ID(WSCFG1) ACC(READ)
PERMIT IRR.DIGTCERT.LISTRING CLASS(FACILITY) ID(WSCFG1) ACC(READ)
```

If you specified 'YES' to the question: "Generate Certificate Authority (CA) Certificate" the ISPF dialogs will generate a *RACDCERT* command to generate a certificate for the certification authority which is WebSphere itself. This certificate will be used to sign the certificates created later for the application server, the Daemon and the Deployment Manager. The label assigned to the certificate '**WebSphereCA**', the default, is used later to reference the CA certificate.

Note: All certificates used within a cell must be signed by the same CA certificate.

If you were using an outside Certification Authority you would have specified 'N' to the question: "Generate Certificate Authority (CA) Certificate" on the ISPF panel and this would not be generated.

It would then be your responsibility to

- Add the CA's certificate to RACF
- Add the Server and Daemon certificates to RACF
- Connect the CA's certificate to the application servers keyring.

Authority to Access Keyrings

The two *RDEFINE* commands shown here are to give the WebSphere address spaces, all members of the group WSCFG1 (configuration group), the authority to read the keyring and list the contents of the keyring.

EJBROLES

Prefix for role definitions specified in security domain

```
/*Defining roles used to access Administrator functions                */
RDEFINE EJBROLE WASW5.administrator UACC(NONE)

RDEFINE EJBROLE WASW5.monitor       UACC(NONE)

RDEFINE EJBROLE WASW5.configurator  UACC(NONE)

RDEFINE EJBROLE WASW5.operator      UACC(NONE)

/*Setting up  access to EJBROLES for administrator and CR          */
PERMIT WASW5.administrator  CLASS(EJBROLE)  ID(WSCFG1) ACCESS(READ)

PERMIT WASW5.monitor        CLASS(EJBROLE)  ID(WSCFG1) ACCESS(READ)

PERMIT WASW5.configurator   CLASS(EJBROLE)  ID(WSCFG1) ACCESS(READ)

PERMIT WASW5.operator       CLASS(EJBROLE)  ID(WSCFG1) ACCESS(READ)
```

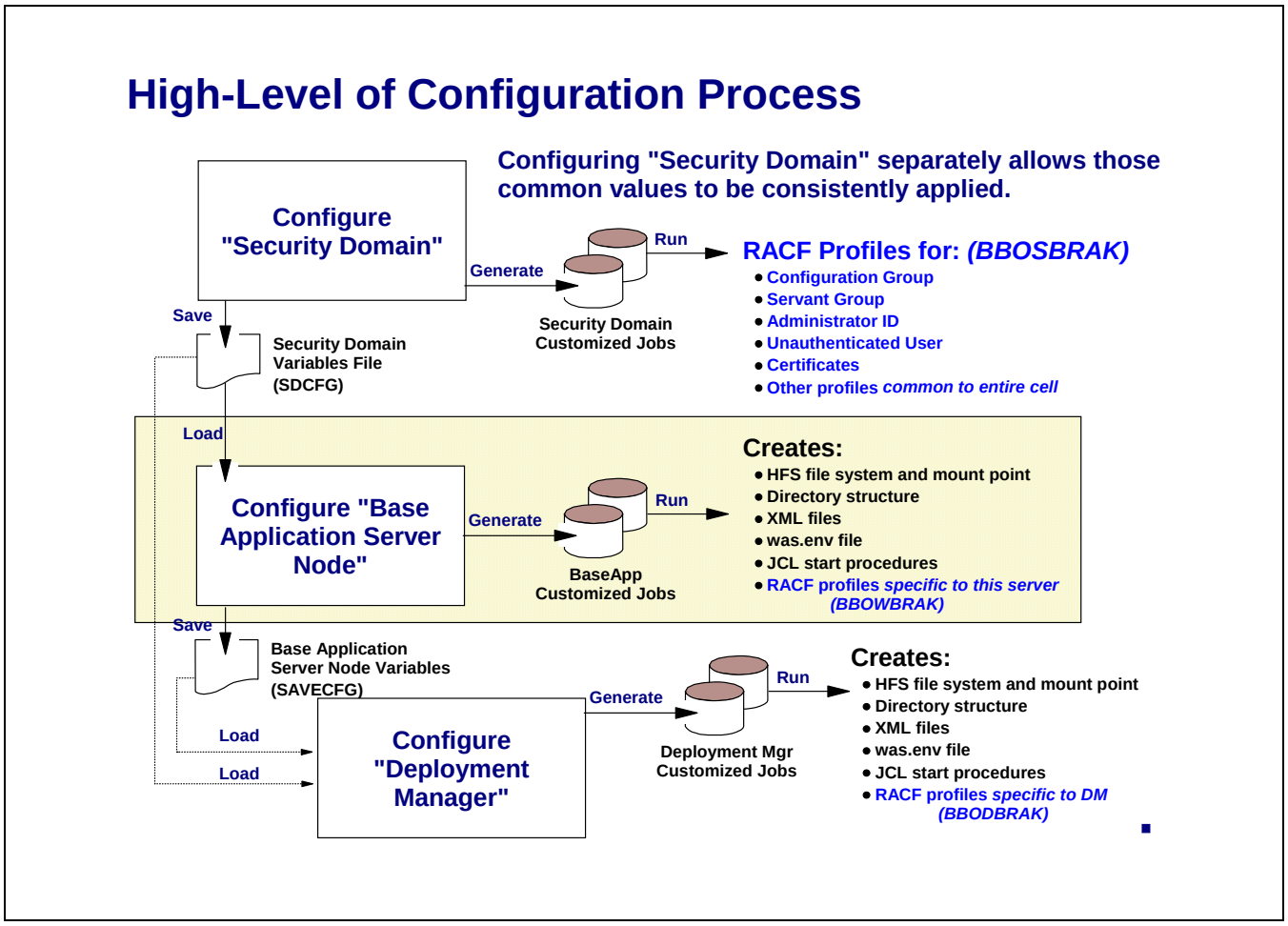
There are four EJBROLE profiles defined in the BBOSDRAK job. These are used to allow for different levels of access authority in administering WebSphere.

- **administrator** - full access to all administrative functions
- **monitor** - may only view configuration information
- **configurator** - may make changes to the configuration, deploy applications etc. However, a configurator may not start or stop servers.
- **operator** - may start and stop servers and view the configuration, but can not change the configuration.

If you specified a Security Domain, then these profiles will be prefixed with the name you specified as shown above.

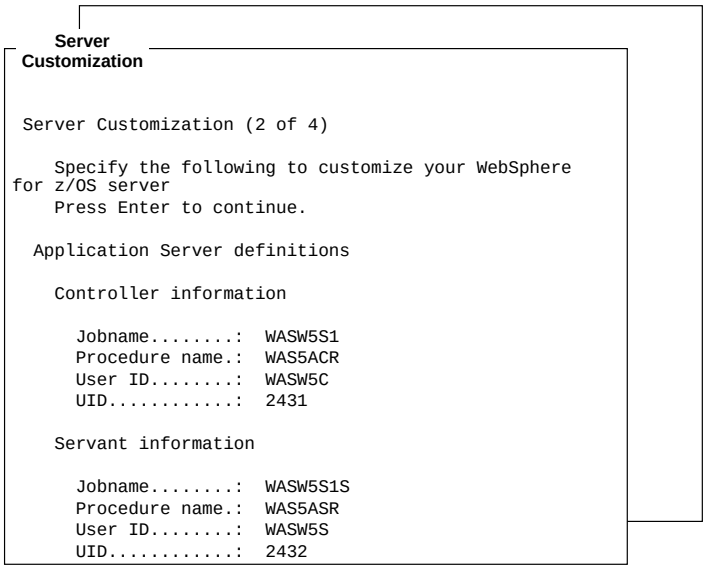
Note: EJBROLE profiles are case sensitive.

Once global security is turned on users who need access to the Admin Console must have authority to one of the above EJBROLES.



Now that we have generated and run the RACF commands that are used cell-wide we are now going to take a look at the application server specific definitions and what RACF definitions are generated.

Creating the Base AppServer



Two panels that capture information about the security definitions for the server and the daemon

IBM Americas Advanced Technical Support, Washington Systems Center, Gaithersburg, MD

When you configure the Base Application Server you always load the saved variables from the Security Domain definitions before you start entering the variables for the new server.

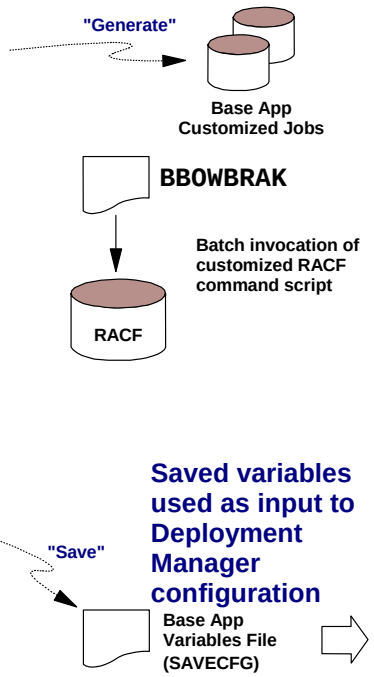
In the Server Customization panels there are some security variables that you must specify. Shown here is the panel on which you define the UserID and UID for the Controller and Servant, as well as the PROC name and JOB name for each. This information will be used to generate the RACF USER profiles and the RACF STARTED class profiles which we will take a look at shortly.

Creating the Base AppServer

Daemon Customization

```
  Server Customization (4 of 4)

    Specify the following to customize your WebSphere
for z/OS server.
    Press Enter to continue.

  Location Service Daemon definitions

    Daemon Home Directory:
      /WebSphereV5/W50200/WASW5/Daemon

    Daemon Job Name:  WASW5D

    Procedure Name.:  WASW5D
    User ID........:  WASW5DM1
    UID............:  2411

    IP Name........:  wg31.washington.ibm.com
    Port...........:  5655
    SSL Port.......:  5656

    Register Daemon with WLM DNS:  N
```

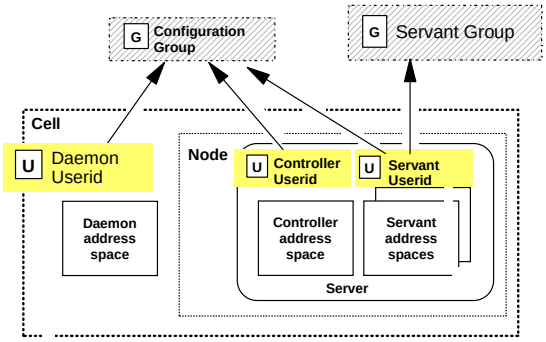
Two panels that capture information about the security definitions for the server and the daemon



IBM Americas Advanced Technical Support, Washington Systems Center, Gaithersburg, MD

On panel four of the Server Customization you provide UserID, PROC and JOB information to be used for the DAEMON. This information will be used to creat the RACF USER profile and the RACF STARTED class profile for the DAEMON.

Define User IDs




```
/* Adding Protected users for WAS Controller Regions.
ADDUSER WASW5DM1 DFLTGRP(WSCFG1) OMVS(UID(2411) HOME(/tmp)
PROGRAM(/bin/sh)) NAME('WAS DAEMON CR')  NOPASSWORD NOOIDCARD

ADDUSER WASW5C DFLTGRP(WSCFG1) OMVS(UID(2431) HOME(/tmp)
PROGRAM(bin/sh)) NAME('WAS APPSVR CR')  NOPASSWORD NOOIDCARD

/* Adding users for WAS Servant Regions.
ADDUSER WASW5S DFLTGRP(WSSR1) OMVS(UID(2432) HOME(/tmp)
PROGRAM(/bin/sh)) NAME('WAS APPSVR SR')  NOPASSWORD NOOIDCARD
CONNECT WASW5S GROUP(WSCFG1)
```

IBM Americas Advanced Technical Support, Washington Systems Center, Gaithersburg, MD

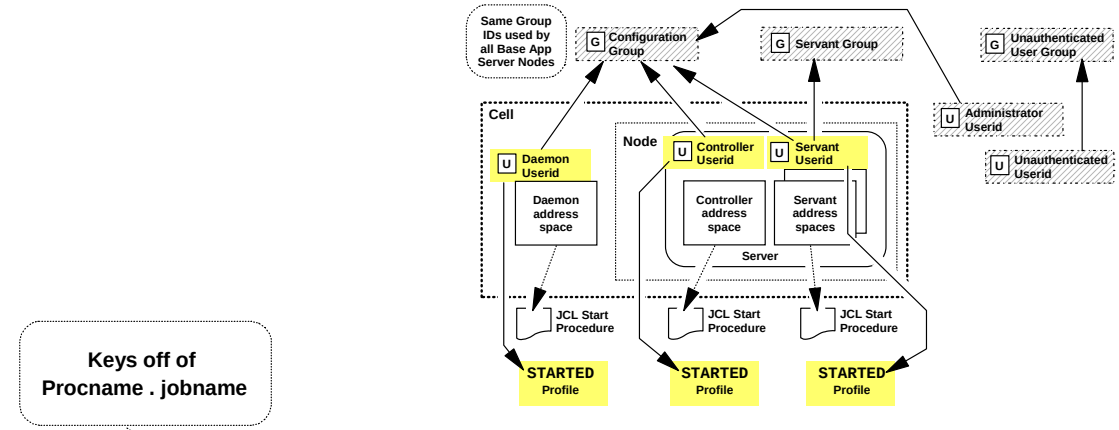
The BBOWBRAJ job will generate the RACF commands and place them in the BBOWBRAK member in your .DATA data set.

Using the values shown on the previous charts, these are the USER profile definitions that are generated.

The DAEMON and the Controller UserIDs are connected to the configuration group, WSCFG1, and the Servant is connected to the servant group, WSSR1. These Group profiles were created in the BBOSDRAK job that was run when you created the Security Domain.

The Servant is also connected to the configuration group.

Assigning User IDs to Started Tasks



Keys off of Procname . jobname

```
/* Assigning userids to started tasks.
RDEFINE STARTED WASW5D.* STDATA(USER(WASW5DM1) GROUP(WSCFG1)
TRACES))
RDEFINE STARTED WAS5ACR.* STDATA(USER(WASW5C) GROUP(WSCFG1)
TRACES))
RDEFINE STARTED WASW5S1S.* STDATA(USER(WASW5S) GROUP(WSCFG1)
TRACES))
```

Keys off of Jobname provided by WLM

Jobname consists of Server shortname followed by an 'S'

IBM Americas Advanced Technical Support, Washington Systems Center, Gaithersburg, MD

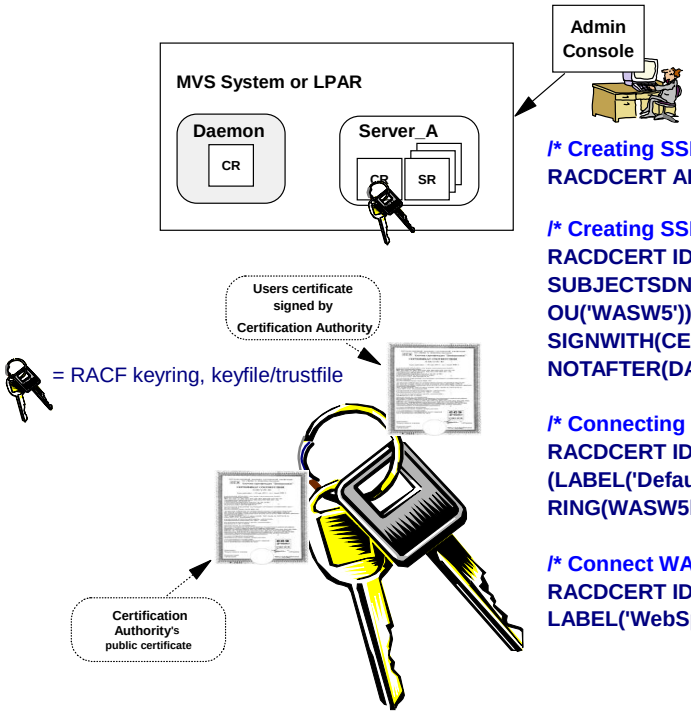
The RACF STARTED profile is used to assign a UserID to a started task. There are two different types of STARTED profiles used for WebSphere V5, one for processes started via start command from the console (Controller) and one for processes started internally (DAEMON) or by WLM (Servants).

The profile for tasks started via the console are specified as <procname.jobname>. The dialogs create a profile for the controller, **WAS5ACR.*,** which covers the proc and any jobname.

This will suffice if all of your application server controller address spaces are going to be run using the same UserID. If you want to differentiate the controller address spaces and assign unique UserIDs, then you will need to define a profile of <procname.jobname>, or in this case **WAS5ACR.WASW5S1**.

The profile used for internally started or WLM started processes is in the form <jobname.*>. For the DAEMON that would be the jobname you specified in the ISPF dialogs, **WASW5D**. For the Servant it will be the Server Short Name with an 'S' at the end, **WASW5S1S**.

SSL in Base Configuration



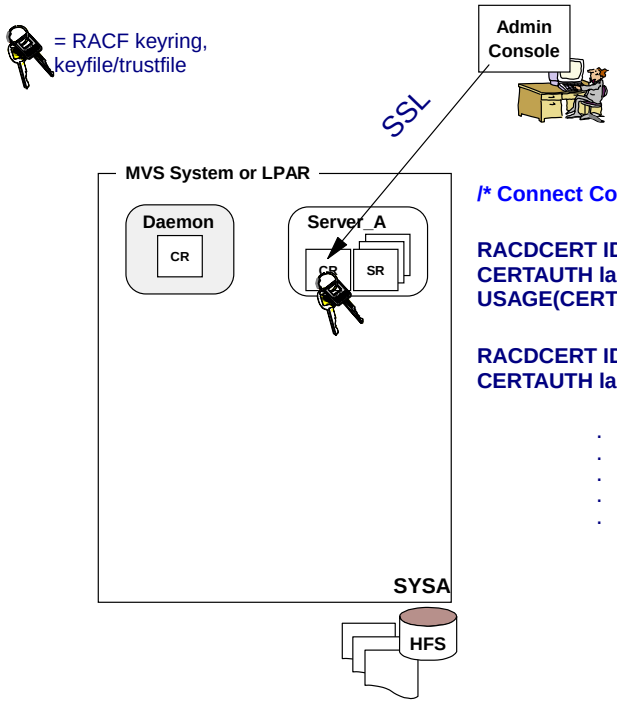

```
/* Creating SSL keyrings for WebSphere servers
RACDCERT ADDRING(WASW5kring) ID( WASW5C )

/* Creating SSL certificate for WebSphere server
RACDCERT ID (WASW5C) GENCERT
SUBJECTSDN(CN('WASW5C.WASW5S1') O('IBM')
OU('WASW5')) WITHLABEL('DefaultWASCert.WASW5S1')
SIGNWITH(CERTAUTH LABEL('WebSphereCA'))
NOTAFTER(DATE(2010/12/31))

/* Connecting Server Certificates to their keyrings
RACDCERT ID(WASW5C) CONNECT
(LABEL('DefaultWASCert.WASW5S1')
RING(WASW5kring ) DEFAULT)

/* Connect WAS CA Certificates to Servers keyring
RACDCERT ID(WASW5C) CONNECT (RING(WASW5kring)
LABEL('WebSphereCA) CERTAUTH)
```

IBM Americas Advanced Technical Support, Washington Systems Center, Gaithersburg, MD

This set of RACF commands creates a keyring for the application server. The name of the keyring, WASW5kring, was specified in the panels of the Security Domain definition. The RACDCERT..... GENCERT... command defines the certificate for the application server Controller. The ID specified is the UserID that the controller runs under. The certificate is signed by the Certification Authority. In this case it is signed by WebSphere using the CA certificate that was created in the Security Domain referenced by the label used when the CA certificate was created. In this case 'WebSphereCA'.

In addition, the CA's certificate must be connected to the application server's keyring. The CA certificate must always be connected to any keyring containing certificates signed by it.

SSL in a Base Configuration



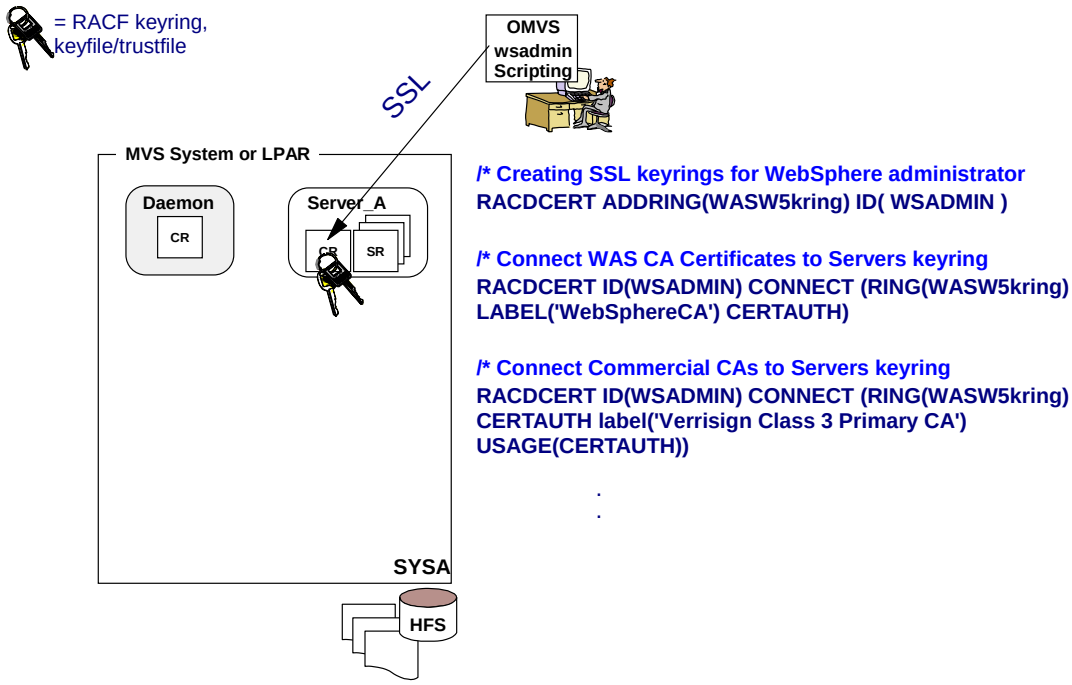

```
/* Connect Commercial CAs to Servers keyring

RACDCERT ID(WASW5C) CONNECT (RING(WASW5kring)
CERTAUTH label('Verrisign Class 3 Primary CA')
USAGE(CERTAUTH))

RACDCERT ID(WASW5C) CONNECT (RING(WASW5kring)
CERTAUTH label('RSA Secure Server CA') USAGE(CERTAUTH))
```

IBM Americas Advanced Technical Support, Washington Systems Center, Gaithersburg, MD

There are additional RACDCERT Connect commands generated to connect commercial CA certificates to the keyring.

This is in case your application server certificate is signed by a commercial authority rather that WebSphere. (It saves you the effort.)

SSL in a Base Configuration



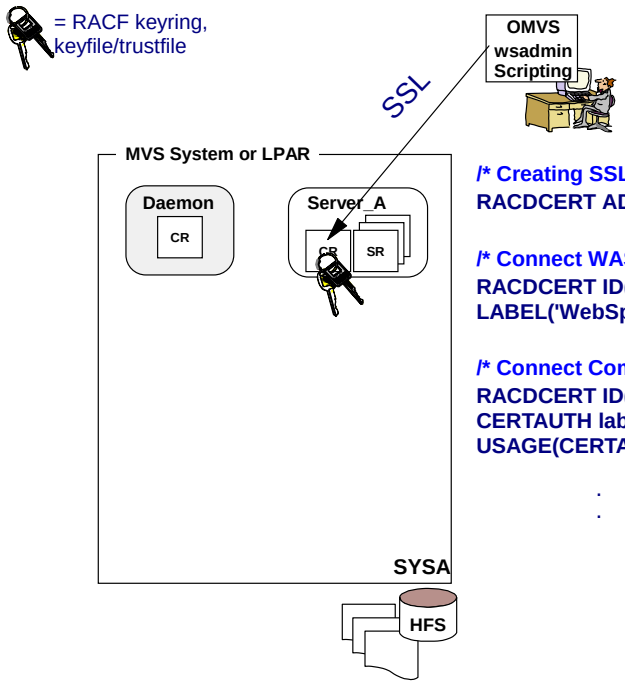

```
/* Creating SSL keyrings for WebSphere administrator
RACDCERT ADDRING(WASW5kring) ID( WSADMIN )

/* Connect WAS CA Certificates to Servers keyring
RACDCERT ID(WSADMIN) CONNECT (RING(WASW5kring)
LABEL('WebSphereCA') CERTAUTH)

/* Connect Commercial CAs to Servers keyring
RACDCERT ID(WSADMIN) CONNECT (RING(WASW5kring)
CERTAUTH label('Verrisign Class 3 Primary CA')
USAGE(CERTAUTH))
```

IBM Americas Advanced Technical Support, Washington Systems Center, Gaithersburg, MD

There are also commands generated to create keyrings and certificates for the two adminstrator IDs that were created during the Security Domain definitions, WSADMIN and WSADMSH (on the following chart).

These certificates are needed when you use wsadmin scripting to perform administrator functions and when the admin console kicks off asynchronous tasks to perform administrator functions.

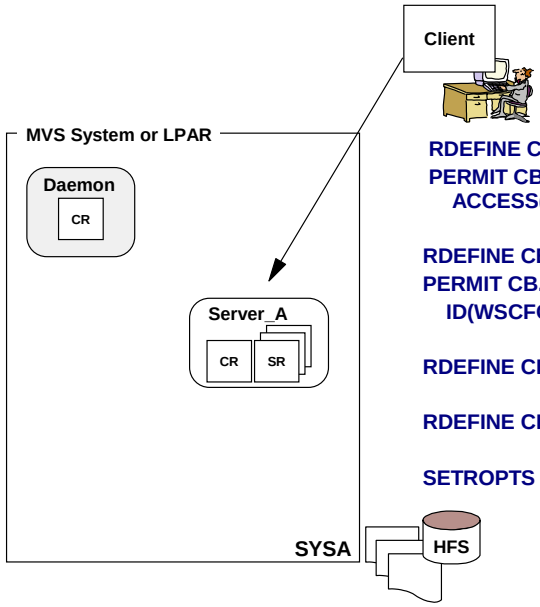
SSL in a Base Configuration




```
/* Creating SSL keyrings for WebSphere asynch administrator
RACDCERT ADDRING(WASW5kring) ID( WSADMSH )

/* Connect WAS CA Certificates to Servers keyring
RACDCERT ID(WSADMSH) CONNECT (RING(WASW5kring)
LABEL('WebSphereCA') CERTAUTH)

/* Connect Commercial CAs to Servers keyring
RACDCERT ID(WSADMSH) CONNECT (RING(WASW5kring)
CERTAUTH label('Verrisign Class 3 Primary CA')
USAGE(CERTAUTH))
.
.
```

IBM Americas Advanced Technical Support, Washington Systems Center, Gaithersburg, MD

Access to Server



RDEFINE CBIND CB.BIND.WASW5.BBO* UACC(READ)
PERMIT CB.BIND.WASW5.BBO* CLASS(CBIND) ID(WSCFG1)
ACCESS(CONTROL)

RDEFINE CBIND CB.BIND.WASW5.WASW5S1 UACC(READ)
PERMIT CB.BIND.WASW5.WASW5S1 CLASS(CBIND)
ID(WSCFG1) ACCESS(CONTROL)

RDEFINE CBIND CB.WASW5.WASW5S1 UACC(READ)

RDEFINE CBIND CB.WASW5.BBO* UACC(READ)

SETROPTS RACLIST(CBIND) GENERIC(CBIND) REFRESH

Profiles in the *CBIND* class are used to authorize access to servers (clusters) by clients.
A client might be another server in another domain or system.

IBM Americas Advanced Technical Support, Washington Systems Center, Gaithersburg, MD

The CBIND class is used to control access to application servers and to objects within application servers. The profiles are set up to allow access by anyone, UACC(READ). You could set this up so that you actually restrict access. This might be done if you have servers in other WebSphere domains accessing servers or objects in this domain. It that case you might set this upp with a UACC(NONE) and specifically permit the UserIDs of the servers that you would allow access from with access of CONTROL.

The profile names are different depending on whether you have specified a Security Domain or not.

The profiloe names are

CB.<websphere domain>.<cluster transition name>

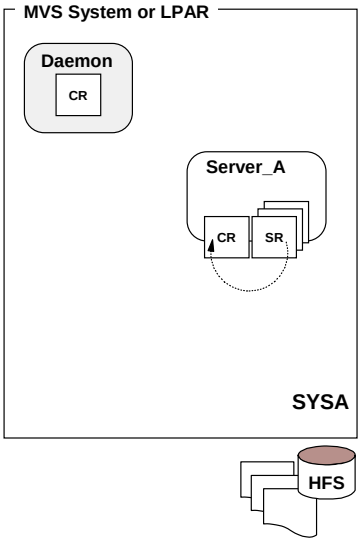
And

CB.BIND.<websphere domain>.<cluster transition name> .

If no Security Domain was specified the <websphere domain> prefix is not used. This is similar to what you saw in the EJBROLE profiles.

Access to Controller

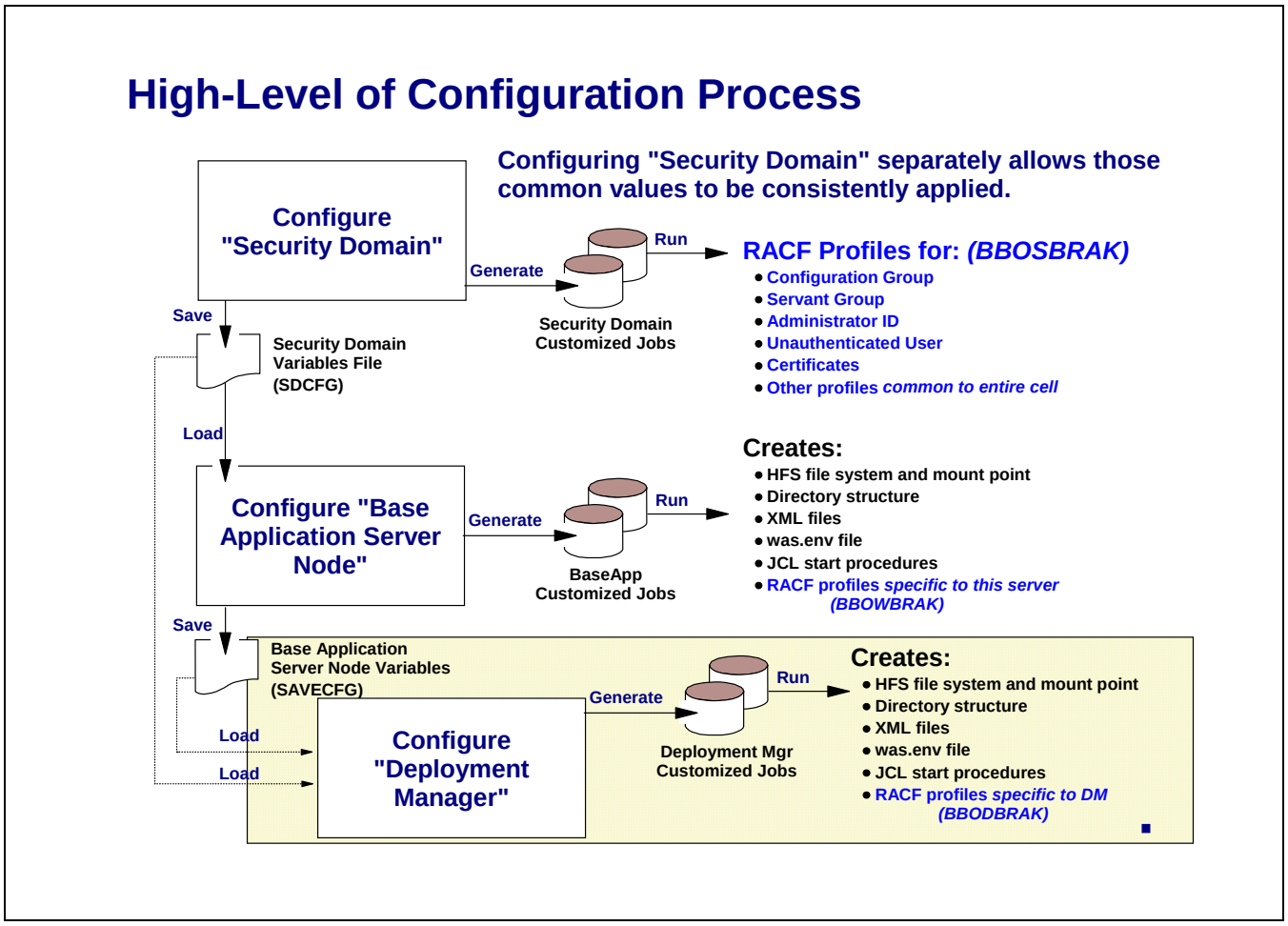
Profiles in the *SERVER* class control what servants may access the controller.




```
Defining SERVER CB.cluster.generic_server.
RDEFINE SERVER CB.*   UACC(NONE)

RDEFINE SERVER CB.*.BBO* UACC(NONE)

RDEFINE SERVER CB.*.BBO*.* UACC(NONE)

RDEFINE SERVER CB.*.WASW5S1 UACC(NONE)
PERMIT CB.*.WASW5S1 CLASS(SERVER)
   ID(WASW5S) ACC(READ)

RDEFINE SERVER CB.*.WASW5S1.* UACC(NONE)
PERMIT CB.*.WASW5S1.* CLASS(SERVER)
   ID(WASW5S) ACC(READ)

SETROPTS RACLIST(SERVER) GENERIC(SERVER) REFRESH
```

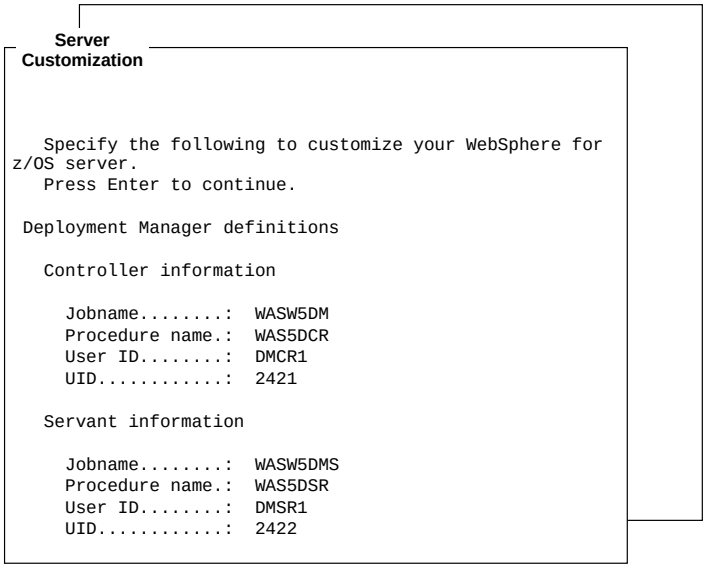
IBM Americas Advanced Technical Support, Washington Systems Center, Gaithersburg, MD

Controller address spaces run APF authorized code and provide services which need to restricted. The SERVER class provides that level of protection. In the SERVER class we define a profile of CB.*.<cluster transition name>.* . Where cluster transition name is the WLM Apllication Environment Name. Then we have a PERMIT command to allow access by the Servant's UserID.



Now that we have generated and run the RACF commands that are used cell-wide and the application server specific definitions, we are going to build the Deployment Manager. What RACF definitions are needed and generated for the Deployment Manager?

Creating the Deployment Manager



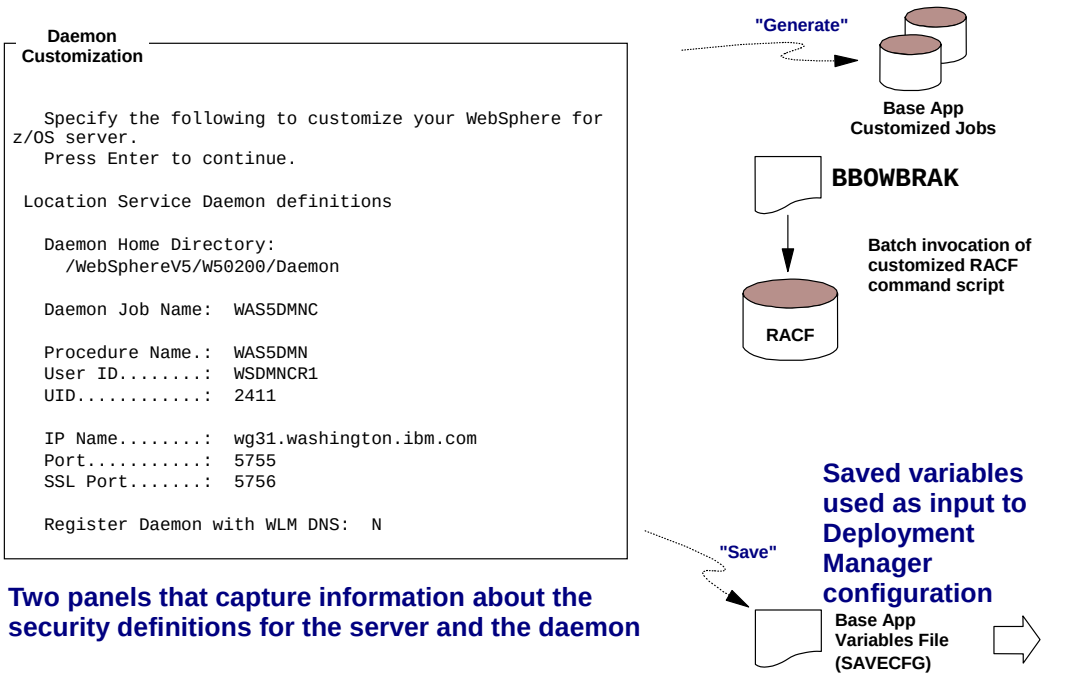
Two panels that capture information about the security definitions for the server and the daemon

IBM Americas Advanced Technical Support, Washington Systems Center, Gaithersburg, MD

When you create the Deployment Manager you must load both the saved variables from the Security Domain and the Base Application Server.

As with the Base server there are panels in the Server Customization where you get to specify UserID, Jobname and PROCname information for the Deployment Manager and the DAEMON.

Creating the Deployment Manager

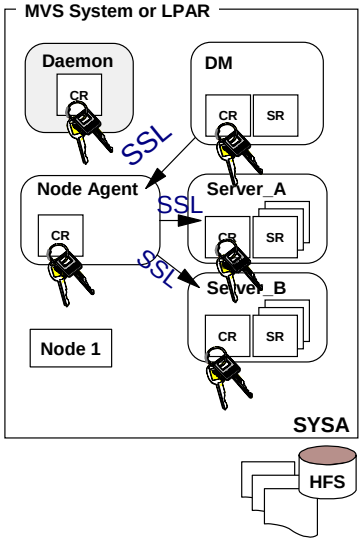


Two panels that capture information about the security definitions for the server and the daemon

IBM Americas Advanced Technical Support, Washington Systems Center, Gaithersburg, MD

I do not show the generated RACF commands here since these UserIDs, Jobnames and PROCnames are used to generate USER profiles, and STARTED class profiles very similar to what was created when the Base Application Server was created.

Certificate for the Deployment Manager




```
/* Generating certificates for WebSphere Deployment Manager
RACDCERT ID (DMCR1) GENCERT
SUBJECTSDN(CN('DMCR1.WASW5DM') O('IBM')
      OU('WASW5')) WITHLABEL('DefaultWASDmgrCert')
      SIGNWITH(CERTAUTH LABEL)
      ('WebSphereCA')) NOTAFTER(DATE(2010/12/31))

/* Connecting Certificates to the Deployment Manager keyring
RACDCERT ID(DMCR1) CONNECT
(LABEL('DefaultWASDmgrCert') RING(WASW5kring )
DEFAULT)

/* Connect WAS CA Certificate to Deployment Manager keyring
RACDCERT ID(DMCR1) CONNECT (RING(WASW5kring)
      LABEL('WebSphereCA')   CERTAUTH)
```

IBM Americas Advanced Technical Support, Washington Systems Center, Gaithersburg, MD

When you create a Deployment Manager cell and then federate the Base Application Server into that cell you end up with a configuration that looks something like this.

The Deployment Manager will get it's own keyring in addition to the keyrings we saw for the DAEMON and Application Server in the Base configuration. The commands shown here create the certificates and connect them to the keyring.

The piece that's missing here is; where does the Node Agent's keyring come from. Typically the Node Agent runs under the same UserID as the Application Server created in the Base configuration. Therefore it actually shares the keyring with the Application Server.

If you create additonal Application Servers in your Network Deployment configuration, as shown in the diagram, you have a choice of using the same UserIDs for the additional server or creating a new set of IDs.

If you use the same UserIDs, then you can share the keyring and certificate created for the first server.

If however you choose to define your additional server with unique UserIDs then you must create a keyring and certificate for that server as well. You could just take the set of commands that was created for the Base Application Server and simply change the ID and create a new certificate for this application server.

Deployment Manager Profiles

Profiles are defined for:
- UserIDs
- Started Class
- CBIND Class
- SERVER Class

The same type of definitions are created as were in the Base AppServer.

There are RACF User profiles created for the Daemon, Controller and Servant.

For each of the the started address spaces a STARTED class profile is created to assign the appropriate UserIDs.

There are CBIND and SERVER class profiles created just like those you already saw in the Base AppServer but with the WLM Application Environment name used for the Deployment Manager.

Summary

- The ISPF dialogs for create jobs to generate and run RACF commands to set up your security environment.
  - Security Domain
  - Base Application Server
  - Deployment Manager
- Profiles are defined for:
  - User and Group
  - Started Class
  - CBIND Class
  - SERVER Class
- RACF keyrings and certificates are created.

IBM Americas Advanced Technical Support, Washington Systems Center, Gaithersburg, MD

Each of the paths through the ISPF dialogs creates jobs to generate and run RACF commands. These command are sufficient to set up an environment to run WebSphere with security turned on.

End of Document