



# Identity Propagation

## Distributed platforms to z/OS

### New York RACF Users Group

Simon Dodge

zSeries Security Architecture/Engineering

November 2011

Together we'll go far



# Identity Propagation.. Agenda

- Identity Propagation within z/OS (refresher)
- Identity Propagation from Distributed environments to z/OS
  - Mapping of Distributed Identity to RACF userid
  - Tracking of both RACF Userid + Distributed Identity in SMF
- RACMAP - New RACF command to build mappings
- Mapping algorithm
- Exploiters / Software requirements
- Samples of SMF audit trail
- Summary
- References

# Identity Propagation.. Agenda

- Identity Propagation within z/OS (refresher)
- Identity Propagation from Distributed environments to z/OS
  - Mapping of Distributed Identity to RACF userid
  - Tracking of both RACF Userid + Distributed Identity in SMF
- RACMAP - New RACF command to build mappings
- Mapping algorithm
- Exploiters / Software requirements
- Samples of SMF audit trail
- Summary
- References

# Identity Propagation is not Authentication

- In our context, almost by definition, Propagation refers to copying an Identity
  - without authentication
  - accepting an identity from a trusted source
- CICS to CICS connections on same plex, shared RACF db (Duh?)
- CICS to CICS connections on different plexes, different RACF db's
  - Your RACF db's ? Perhaps kept in sync via RRSF ?
  - Or someone else's RACF db such as a business partner
- Similarly for JES NJE
  - RACFVARS & RACLNDE for local trusted nodes
  - NODES: you may trust, you may translate

# Distributed Identity

- Distributed Identity characteristics
  - A user identity in the distributed world, in contrast to z/OS UserId
  - Security Registry that was used to authenticate that identity, eg LDAP

**Uid=Joe,Ou=Dept,O=company**

**Registry.Domain**

# Without Identity Propagation



User's Identity  
• DN & Realm

User's Identity  
• RACF userid



RunAs

userid

z/OS

SubSystem

RACF

Audit

Audit Record

✓ RACF userid

Distributed applications often use a common RACF user-ID when invoking CICS, DB2, IMS to process the request. This distributed User ID is not passed to CICS, etc. and on to RACF, making end-user accountability difficult to determine.

Audit records do not show who the user is; accountability difficult to determine who the originating user is.

# 3 problems / challenges

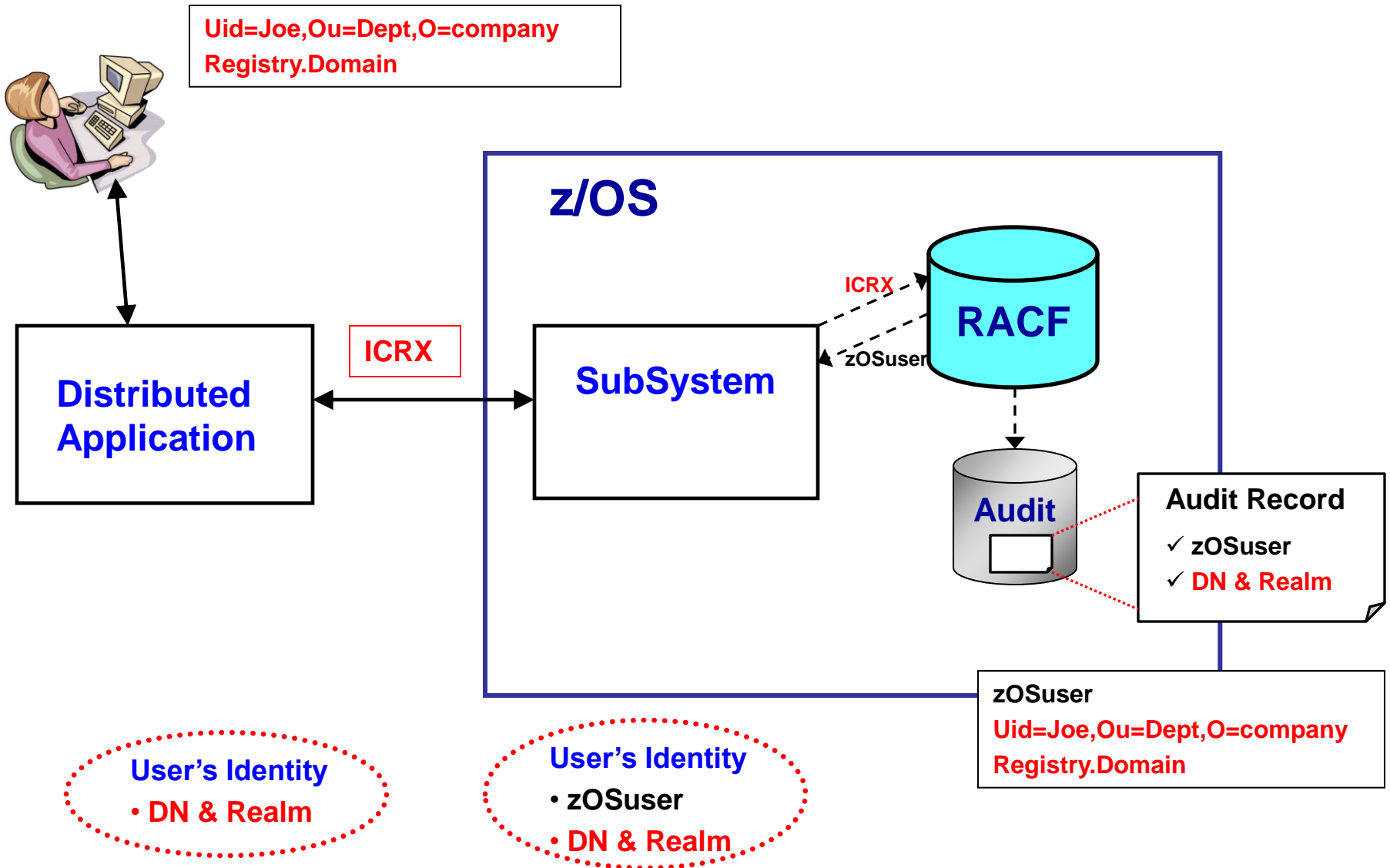
1. Determination of the z/OS identity is performed outside of z/OS
  - Often within an application
  - Are you really comfortable with that ?
2. Accountability in z/OS audit trail does not reflect end user identity
  - A server ID gives no End to End accountability
  - Identity is not propagated across platform boundary
3. RACF has a limit of 8 characters for Userid
  - Often used as a weakness against RACF

# Identity Propagation.. Agenda

- Identity Propagation within z/OS (refresher)
- Identity Propagation from Distributed environments to z/OS
  - Mapping of Distributed Identity to RACF userid
  - Tracking of both RACF Userid + Distributed Identity in SMF
- RACMAP - New RACF command to build mappings
- Mapping algorithm
- Exploiters / Software requirements
- Samples of SMF audit trail
- Summary
- References



# With Identity Propagation



# How does RACF do it ?

- New form of RACROUTE VERIFY allows for
  - Distributed Identity + Registry/Realm  
instead of
  - Userid + Password
- RACF searches mappings to find a RACF userid
  - No mapping → ICH408I “No mapping found”
  - Match found → Build ACEE for RACF userid
    - Also saves Distributed Id + Registry
- SMF record from RACF now includes Distributed Id + Registry (new relocate sections).

# Accessing Distributed Identity

- RACF has new relocate sections in SMF records
  - In UTF-8 format
  
- CICS application can use: EC INQ ASSOCIATION
  - also in UTF-8 format

# Identity Propagation.. Agenda

- Identity Propagation within z/OS (refresher)
- Identity Propagation from Distributed environments to z/OS
  - Mapping of Distributed Identity to RACF userid
  - Tracking of both RACF Userid + Distributed Identity in SMF
- **RACMAP - New RACF command to build mappings**
- Mapping algorithm
- Exploiters / Software requirements
- Samples of SMF audit trail
- Summary
- References

# Command syntax - RACF

```
RACMAP ID(userid) MAP USERDIDFILTER(name('.....'))  
      REGISTRY(name('.....'))  
      LABEL(xyz)
```

```
RACMAP ID(userid) LIST
```

```
RACMAP ID(userid) DELMAP LABEL(xyz)
```

```
RACMAP QUERY USERDIDFILTER(name('.....'))  
      REGISTRY(name('.....'))
```

# Security Administration

- RACF Resource access is unaffected. Still controlled via permissions based on Userid / Group(s)
- Mapping of Distributed Identity to RACF Identifier can be
  - One to One Full match on DN
  - Many “One to One”s A shared userid
  - Many/Partial to One A generic z/OS identity
- DN(\*) REALM(\*) allows for a catchall
  - “UNKNOWN” / “UNMAPPED” / “Guest”
- No mapping → “Logon violation: Unknown Distributed Identity”
- Mapping filters includes Registry, you decide which authenticators you trust.

# Identity Propagation.. Agenda

- Identity Propagation within z/OS (refresher)
- Identity Propagation from Distributed environments to z/OS
  - Mapping of Distributed Identity to RACF userid
  - Tracking of both RACF Userid + Distributed Identity in SMF
- RACMAP - New RACF command to build mappings
- **Mapping algorithm**
- Exploiters / Software requirements
- Samples of SMF audit trail
- Summary
- References

# Mapping

- New research class IDIDMAP
- New command RACMAP to define mappings
  - Mapping can be One to One
    - DN + Registry → Userid
  - Mapping can be Many to One
    - Partial DN + Registry → Userid
    - Algorithm for parsing DN, not a generic mask
    - Allows multiple DN's to map to single userid
- Can have a “fall through” mapping via “\*”
- Registry can be full name or “\*”
  - No partial matchings, Either full or “\*”



# Mapping algorithm

Iteratively:

- Search for match
- If match found then “Mapping found”
- Remove leftmost RDN
- If end of DN then “No mapping found”
  - RACINIT event qualifier 39
- Try again

\*

<b>Uid=Joe,Ou=Dept,O=company</b>	<b>Registry.Domain</b>	<b>➔</b>	<b>UserId</b>
<b>Ou=Dept,O=company</b>	<b>Registry.Domain</b>	<b>➔</b>	<b>DeptId</b>
<b>O=company</b>	<b>Registry.Domain</b>	<b>➔</b>	<b>Compld</b>

# Possible set of mappings

<b>USERDIDFILTER</b>	<b>Userid</b>
<b>Uid=Hayim,Ou=NYRUG,C=RUG</b>	<b>NYRUG</b>
<b>Uid=Stu,Ou=NYRUG,C=RUG</b>	<b>NYRUG</b>
<b>Ou=NYRUG,C=RUG</b>	<b>NYUSER</b>
<b>C=RUG</b>	<b>RUGUSER</b>

# Case Sensitivity

*RDNname=value,*

- RDN *name* is not sensitive to case  
→ RACF upper cases RDN name in db
- RDN *value* is sensitive to case

uid=sdodge  
uiD=sdodge  
UId=sdodge  
UID=sdodge

**Same results.**  
Case of RDN name  
does **not** matter

uid=sdodge  
uiD=sDodge  
UId=SDodge  
UID=SDODGE

**Different results.**  
Case of RDN value  
**does** matter

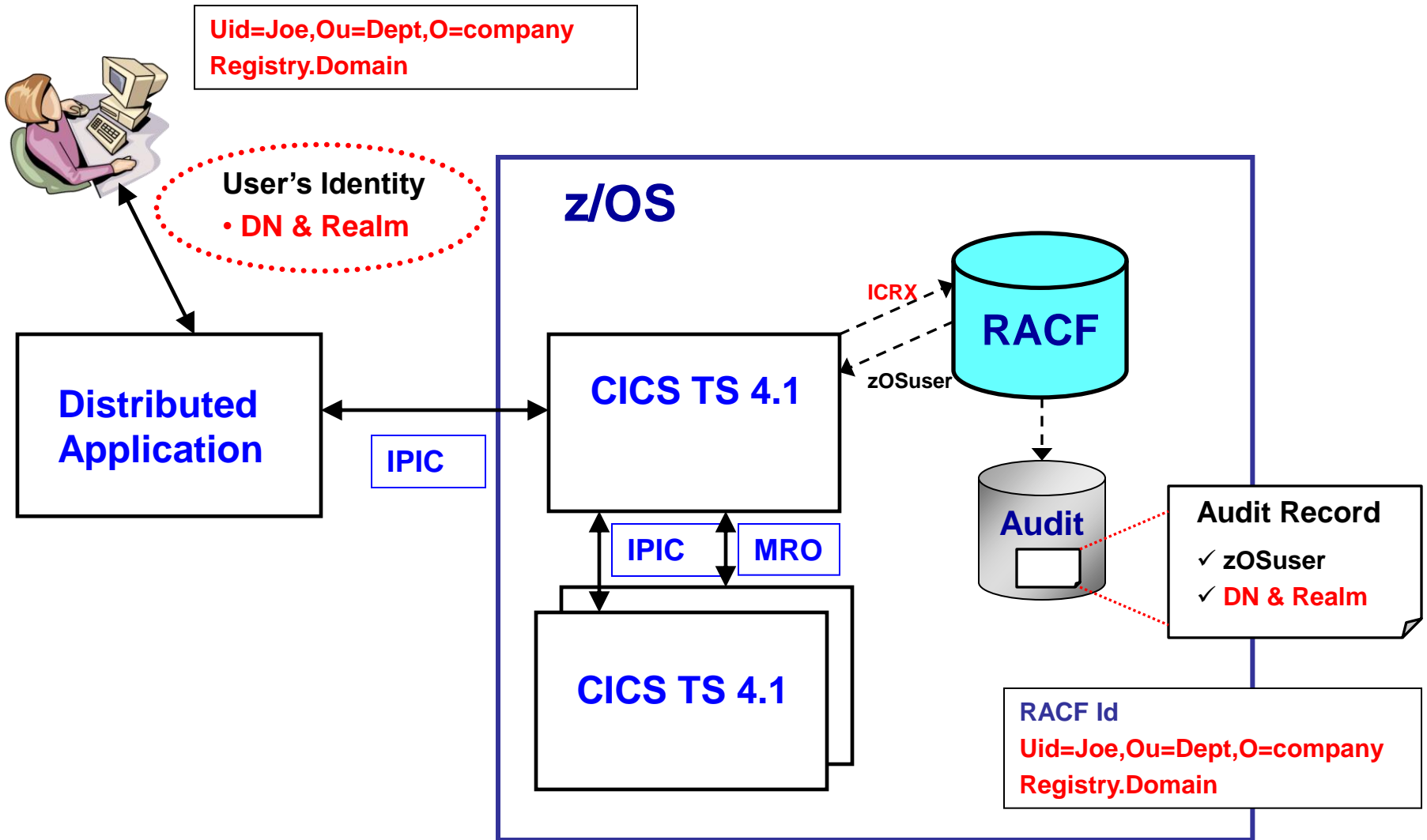
# Identity Propagation.. Agenda

- Identity Propagation within z/OS (refresher)
- Identity Propagation from Distributed environments to z/OS
  - Mapping of Distributed Identity to RACF userid
  - Tracking of both RACF Userid + Distributed Identity in SMF
- RACMAP - New RACF command to build mappings
- Mapping algorithm
- **Exploiters / Software requirements**
- Samples of SMF audit trail
- Summary
- References

# Software support

- z/OS release 1.11 base support
  - PTFs recommended for enhanced support
  - APARs: OA34258, OA34259
- CICS TS 4.1
  - PTFs needed to support Identity Propagation
  - APARs: PK83741, PK95579, PM01622, PK98426
  - Needs IPIC connections
- CICS Transaction Gateway V8
  - Uses IPIC server to CICS
- DB2 V10
  - Needs Trusted Context
- WebSphere Application Server V??

# CICS configuration





# CTG configuration

User's Identity  
• DN & Realm

Distributed Application

IPIC

z/OS

CTG 8.0

IPIC

CICS TS 4.1

IPIC MRO

CICS TS 4.1

Uid=Joe,Ou=Dept,O=company  
Registry.Domain

ICRX  
zOSuser

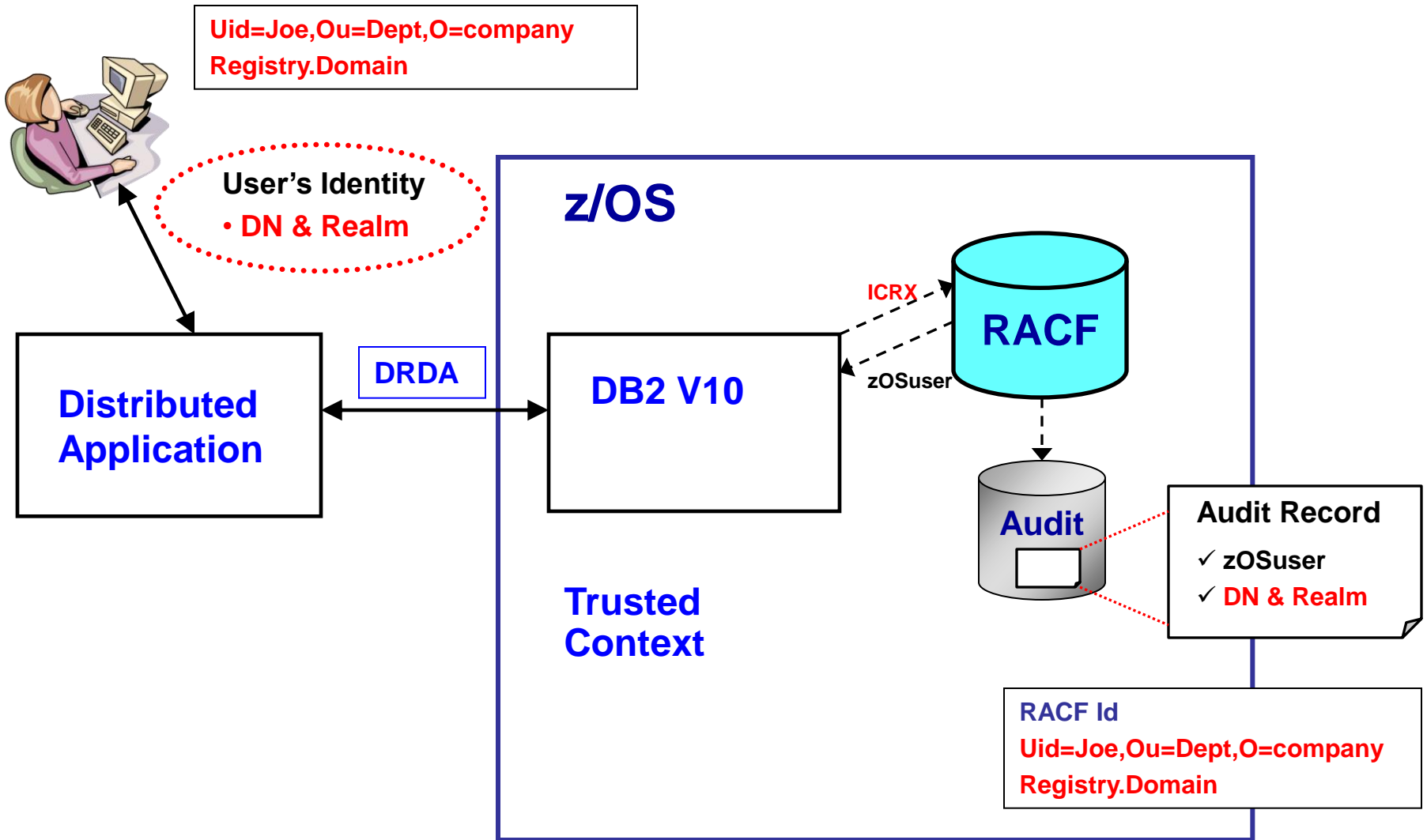
RACF

Audit

Audit Record  
✓ zOSuser  
✓ DN & Realm

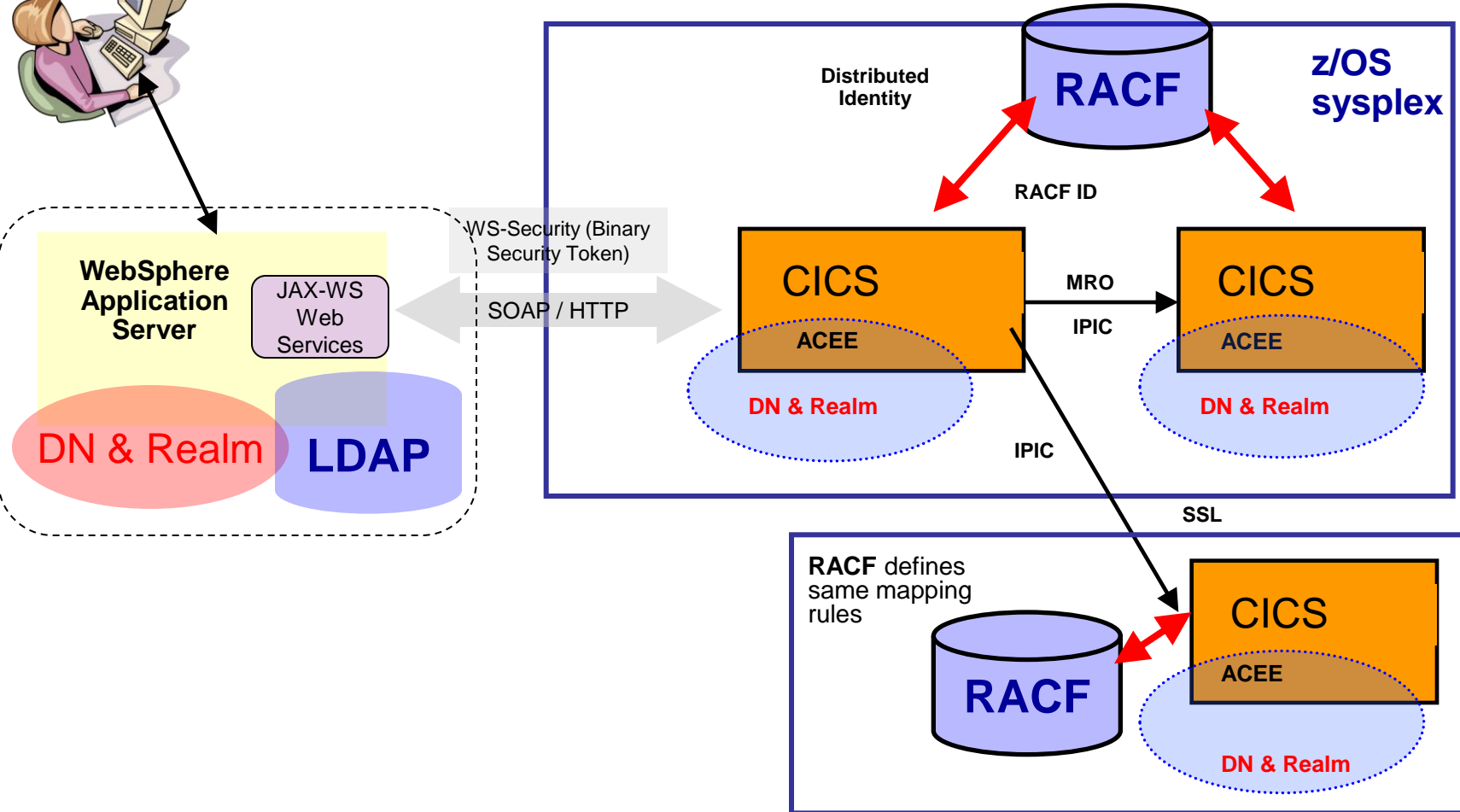
RACF Id  
Uid=Joe,Ou=Dept,O=company  
Registry.Domain

# DB2 configuration





# Identity Context Propagation – WebSphere (Web Services)



# Identity Propagation.. Agenda

- Identity Propagation within z/OS (refresher)
- Identity Propagation from Distributed environments to z/OS
  - Mapping of Distributed Identity to RACF userid
  - Tracking of both RACF Userid + Distributed Identity in SMF
- RACMAP - New RACF command to build mappings
- Mapping algorithm
- Exploiters / Software requirements
- **Samples of SMF audit trail**
- Summary
- References

# Logon failure

```
16.42.52 ICH408I USER(CICS ) GROUP(STC ) NAME(STARTED TASK
008      DISTRIBUTED IDENTITY IS NOT DEFINED:
008      uid=martina,ou=swg,o=ibm wtsc58.itso.ibm.com:389
16.42.52 IRR012I VERIFICATION FAILED. USER PROFILE NOT FOUND.
```

S M F R E C O R D L I S T I N G 19Jan11 11:43 to 19Jan11 18:02

Date	Time	Typ	Event	Eq	Userid / DistName + Registry
19Jan2011	16:42:52	80	RACINIT	39	CICS uid=martina,ou=swg,o=ibm wtsc58.itso.ibm.com:389

Event Qualifer  
39

Distinguished  
Name

Registry

# Resource access

Date	Time	Class	Intent	Tranid	Userid / DN + Registry
19Jan2011	17:32:20	TCICSTRN	READ	CSMI	SWGRES UID=MARTINA,OU=SWG,O=IBM wtsc58.itso.ibm.com:389

Mapped Userid

Distinguished Name

Registry

# Identity Propagation.. Agenda

- Identity Propagation within z/OS (refresher)
- Identity Propagation from Distributed environments to z/OS
  - Mapping of Distributed Identity to RACF userid
  - Tracking of both RACF Userid + Distributed Identity in SMF
- RACMAP - New RACF command to build mappings
- Mapping algorithm
- Exploiters / Software requirements
- Samples of SMF audit trail
- **Summary**
- References

# Summary: Does this address our issues ?

- “z/OS ESM’s have a limit of 8 char max for userid”
  - Still a limit of 8, but So What ? Now that we have both identifiers, the limit of 8 on z/OS identity seems irrelevant
- “Audit trail on z/OS just reflects RACF identity, not Distributed Identity; No End to End accountability”
  - SMF now has both the DN/Realm as well as z/OS identifier
- Distributed applications decide what identity to “Assert”  
“RunAs”
  - z/OS Security Administrator controls the mappings to z/OS Identity, not the application.

# Identity Propagation.. Agenda

- Identity Propagation within z/OS (refresher)
- Identity Propagation from Distributed environments to z/OS
  - Mapping of Distributed Identity to RACF userid
  - Tracking of both RACF Userid + Distributed Identity in SMF
- RACMAP - New RACF command to build mappings
- Mapping algorithm
- Exploiters / Software requirements
- Samples of SMF audit trail
- Summary
- **References**



- z/OS Identity Propagation
  - SG247850
- <http://www.redbooks.ibm.com/abstracts/sg247850.html?Open>
- Examples showing Id Prop for
  - CICS and CTG
  - DB2
  - CICS Web services



# Other references

- “CICS and Identity Propagation:  
Solving the End-to-End Security Challenge”
  - Phil Wakelin, Nigel Williams, Martin Brown
  - z/Journal December 2010
  - Mainframezone.com
- CICS SupportPac CH51 for CTG
  - VERY helpful when troubleshooting CICS / CTG connection

# Possible twist

- Middleware to adjust DN by inserting an **Appl=xxxx** identifier as an RDN to identify the distributed application
- RACF mappings based on application identifier
- Will enable smooth transition from current environment
  - Moves Id establishment outside of application
  - Allows continued use of application specific identities
  - Will now have full End to End accountability
- Concern about distributed assertion of Identity moves from Application to middleware
  - IE Will it insert the correct /appropriate Appl=xxxx value

# Application RDN driving mapping to a Appl/Server Id

Uid=Joe,Ou=Dept,O=company  
Registry.Domain



Distributed  
Application

ICRX

z/OS

SubSystem

Mapping based on application identifier:  
ApId=xxxx,O=company → SAPLxxxx

ICRX

SAPLxxxx



Uid=Joe,Ou=Dept,ApId=xxxx,O=company  
Registry.Domain

SAPLxxxx

Uid=Joe,Ou=Dept,ApId=xxxx,O=company  
Registry.Domain