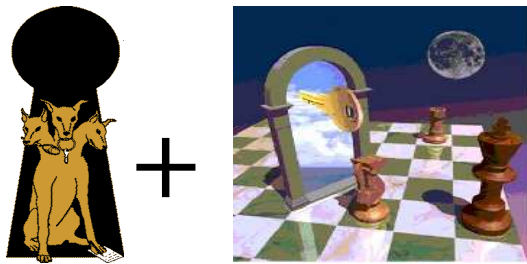


# 5595 – Implementing Kerberos on z/OS

with  
**Network Authentication Service**  
and  
**Resource Access Control Facility**



Eric Rosenfeld, CISSP  
z/OS Security Development  
rosenfel@us.ibm.com

February 2008

SHARE IBM Systems and Technology Group IBM

## Agenda

- ☒ General Kerberos Overview
- ☒ Base Kerberos Registry Support Overview
- ☒ Getting Started
  - ▶ Server Information
  - ▶ Registry set-up
- ☒ SAF Callable Services
- ☒ Dependencies and Considerations
- ☒ Session Summary

© 2008 IBM Corporation 2

SHARE IBM Systems and Technology Group IBM

## Trademarks

- 1 The following are trademarks or registered trademarks of the International Business Machines Corporation:
  - ▶ IBM, DB2, OS/390, RACF, SecureWay, z/OS, AS/400, AIX
- 1 UNIX is a registered trademark of The Open Group in the United States and other countries.
- 1 Microsoft, Windows, and Windows NT are trademarks of Microsoft Corporation in the United States, other countries, or both.
- 1 SOLARIS is a trademark or registered trademark of Sun Microsystems, Inc. in the United States and other countries
- 1 Kerberos is a trademark of MIT
- 1 Other company, product, and service names may be trademarks or service marks of others.


© 2008 IBM Corporation 3

SHARE IBM Systems and Technology Group IBM

## Greek Mythology

Kerberos (Cerberus) was the mythological three-headed dog that guarded the entrance to the underworld.

Unless you could get past Kerberos, you could not enter (or leave!) the underworld



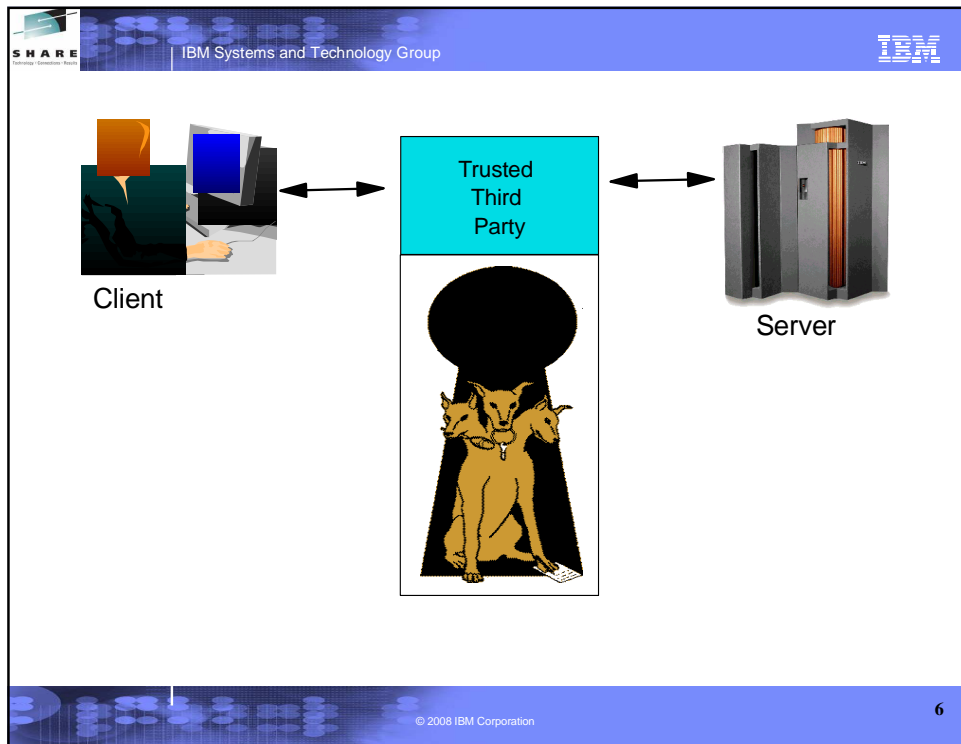
© 2008 IBM Corporation 4

SHARE IBM Systems and Technology Group IBM

## What is Kerberos?

- §A distributed authentication service developed by MIT
- §Allows user authentication over a physically untrusted network
- §Tickets are issued by a Kerberos authentication server
  - §Users and servers are required to have keys registered with server
- §Flows to and from server covered by a session key
  - §used in a direct exchange between a user and a service
- §V5 implemented in OS/390, z/OS, AIX, AS/400, Windows, Solaris and others
  - §**Network Authentication Service** component of Integrated Security Services on z/OS

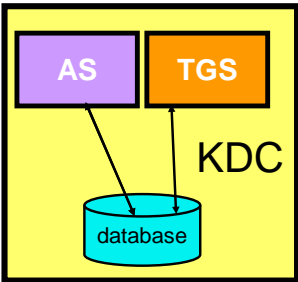
© 2008 IBM Corporation 5



SHARE | IBM Systems and Technology Group | IBM

## Key Distribution Center (KDC)

- § Trusted "third party"
  - § Both client and server trust the information in/decisions of the KDC
- § Responsible for issuing user credentials and tickets
- § Consists of
  - ▶ an authentication server (KAS)
    - ▶ Authenticates users
    - ▶ Grants Ticket Granting Tickets
  - ▶ a ticket granting server (TGS)
    - ▶ Generates session key
    - ▶ Grants service tickets
  - ▶ a Kerberos Data Base (KDB)
    - Contains keys for each user and server



© 2008 IBM Corporation 7

SHARE | IBM Systems and Technology Group | IBM

## Terms

- § Ticket
  - ▶ An encrypted electronic authentication token including:
    - client's identity
    - a dynamically created session key
    - a time stamp
    - lifetime for the ticket
    - a service name
- § Realm
  - ▶ The Kerberos domain: the set of entities which authenticate using the domain of authority served by one KDC.
- § Principal
  - ▶ Anything that is defined to a realm
  - ▶ *name@realm*
    - Can be a user, service or relationship

© 2008 IBM Corporation 8

SHARE | IBM Systems and Technology Group | IBM

## Ticket Use

- At logon (kinit) Ticket Granting Ticket returned
- To use a service, TGT presented w/request
- Server returns service ticket
  - Contains session key
  - Client presents service ticket to server as part of authentication protocol
    - GSS-API gss\_init\_sec\_context method
  - Can be used until expiration
  - Avoids repeated authentication

9

© 2008 IBM Corporation

SHARE | IBM Systems and Technology Group | IBM

## Kerberos on z/OS

(Its own component, integrated with RACF via SAF)

Key Distribution Center

zSeries with z/OS

Standards

RFC 1510 => Kerberos V5  
RFC 1964 => GSS-API

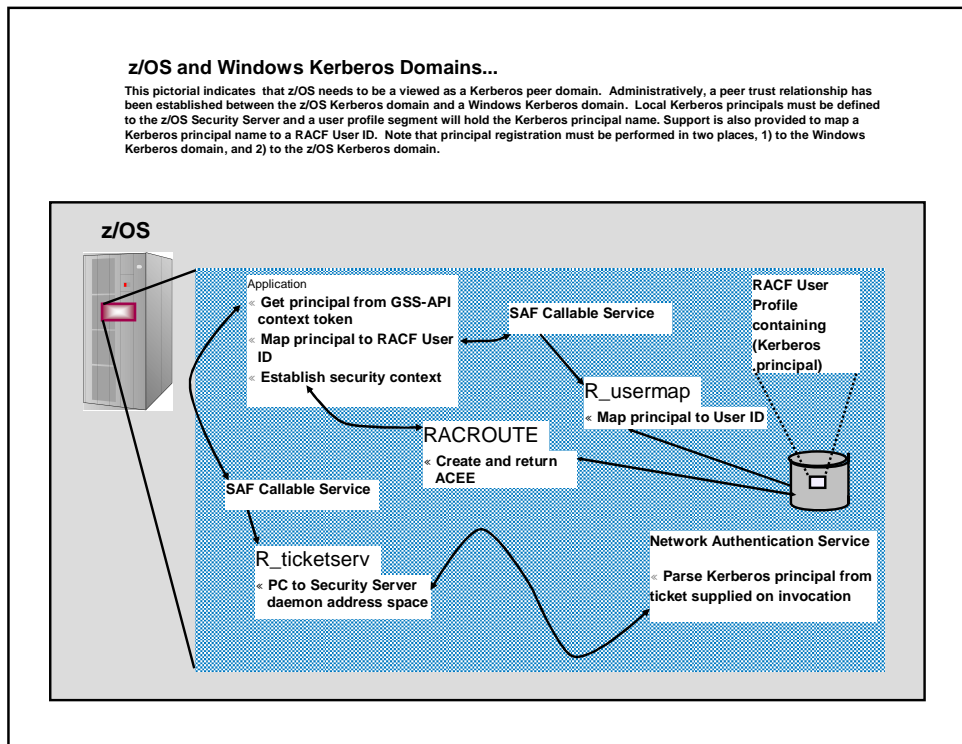
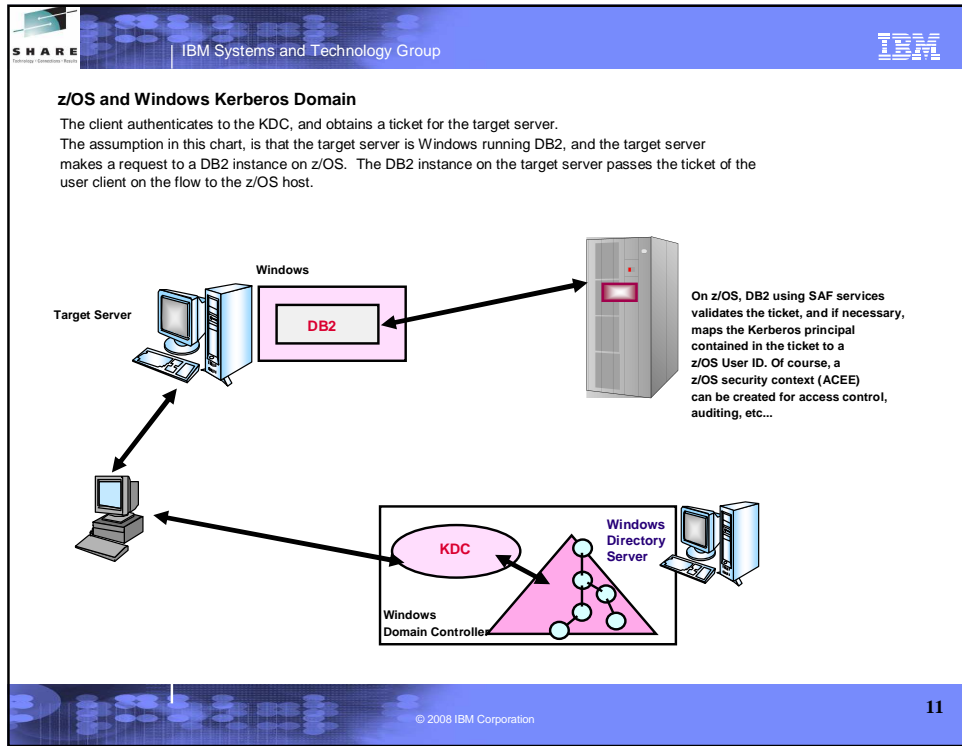
**(AS)**  
Authenticates Users  
Grants TGTs

**(TGS)**  
Generates Session Keys  
Grants service tickets based on TGT

1. Kerberos registry integrated into RACF registry
2. Kerberos KDC executes within z/OS address space
3. z/OS KDC behaves like any other Kerberos "Realm"
4. Kerberos Realm to Realm function supported

10

© 2008 IBM Corporation



SHARE | IBM Systems and Technology Group | IBM

## Network Authentication Service – Commands

- Ø **kinit** - obtains or renews the Kerberos ticket-granting ticket.
- Ø **klist** - displays the contents of a Kerberos credentials cache or key table.
- Ø **kdestroy** - destroys a Kerberos credentials cache.
- Ø **keytab** - manages a key table (z/OS likely will use RACF).
- Ø **ksetup** - manages Kerberos service entries in the LDAP directory for a Kerberos realm.
- Ø **kpasswd** - allows principal to change password
- Ø **kvno** - returns key version number.
- Ø **kadmin** - administer non z/OS KDC with Kerberos commands
  - Ø help, list\_principals, add\_principal, delete\_principal, change\_password, rename\_principal, list\_policies, add\_policy, delete\_policy, add\_key, etc.

© 2008 IBM Corporation 13

SHARE | IBM Systems and Technology Group | IBM

## RACF is the Kerberos Registry

- § The Network Authentication Server requires a registry of principal information, global information, etc.
- § This security information is stored in RACF User and General Resource profiles
- § Kerberos administration is done via RACF commands/panels
- § The Network Authentication Server obtains it's registry information via SAF callable service
- § Kerberos application servers can use SAF callable services to parse Kerberos tickets to obtain principal names, and to map from principal to RACF user and vice versa

© 2008 IBM Corporation 14

SHARE IBM Systems and Technology Group IBM

## Classes

- §KERBLINK
  - ▶ Maps Kerberos principal to RACF userid
    - ADDUSER/ALTUSER defines local profiles
    - RDEF/RALT used to define foreign profiles
- §REALM
  - ▶ Defines default information for local realm (KERBDFLT)
  - ▶ Defines inter-realm trust
    - ▶ A TGT issued in one realm can be used in another

© 2008 IBM Corporation 15

SHARE IBM Systems and Technology Group IBM

## Kerberos Registry

- ▶ Local Kerberos principals are defined as RACF users with a KERB segment
- ▶ REALM class profiles are used to define information about the local Kerberos realm and foreign realms
  - Local realm information includes name, key, and ticket lifetime (MIN, MAX, and DEFAULT in seconds)
  - Foreign realm trust relationships are defined in pairs (A to B and B to A) which also include a key
- ▶ Foreign Kerberos principals are mapped to a RACF identity using KERBLINK class profiles

© 2008 IBM Corporation 16



SHARE IBM Systems and Technology Group IBM

## Kerberos Registry

§ The RACF user password and the Kerberos local principal's password are integrated

- ▶ Kerberos key will be generated when the user's password changes and is **not** expired
  - TSO/application logon
  - ALU NOEXPIRED
  - PASSWORD command
- ▶ The Kerberos password is subject to RACF SETROPTS rules and installation defined rules via password exit

© 2008 IBM Corporation 17

SHARE IBM Systems and Technology Group IBM

## SAF Services

§ [R\\_kerbinfo](#) is called by the server to

- ▶ Retrieve principal information
- ▶ Retrieve realm information
- ▶ Update the count of invalid key attempts
  - similar to an invalid logon attempt
- ▶ Reset the count of invalid key attempts
  - like when you remember your password, on your 2nd or 3rd try

§ [R\\_ticketserv](#) is called by applications to determine the principal name associated with a credential

§ [R\\_usermap](#) is called by applications to map from principal to RACF identifier

© 2008 IBM Corporation 18

SHARE IBM Systems and Technology Group IBM

## SAF Services (cont)

GSS-API support

- Allows Kerberos GSS-API function via non-LE interface
- **R\_GenSec** service provides following GSS-API functions:
  1. GSEC\_INIT\_SEC\_CONTEXT
  2. GSEC\_CONT\_SEC\_CONTEXT
  3. GSEC\_ACC\_SEC\_CONTEXT
  4. GSEC\_DEL\_SEC\_CONTEXT
  5. GSEC\_REL\_CRED
  6. GSEC\_GET\_MIC
  7. GSEC\_VER\_MIC
  8. GSEC\_WRAP\_MSG
  9. GSEC\_UNWRAP\_MSG
  10. GSEC\_EXPORT\_SEC\_CONTEXT
  11. GSEC\_EXPORT\_CRED
  12. GSEC\_IMPORT\_SEC\_CONTEXT
  13. GSEC\_IMPORT\_CRED
  14. GSEC\_ACQUIRE\_CRED

© 2008 IBM Corporation 19

SHARE IBM Systems and Technology Group IBM

## Steps for Getting Started

- ☒ Install/Customize Network Authentication Server
- ☒ Set up registry
  - ▶ Define local realm
  - ▶ Define inter-realm relationships
  - ▶ Define local principals
  - ▶ Define foreign principals

© 2008 IBM Corporation 20

SHARE IBM Systems and Technology Group IBM

## Network Authentication Service - Installation

§ Installs into

- ▶ UNIX file system
  - executables in directory /usr/lpp/skrb
  - /etc/skrb files need access 755
  - /var/skrb/creds needs access 1777
- ▶ System datasets
  - EUVF.SEUVFLPA
  - SYS1.SIEALNKE (EUVF.SEUVFLNK Pre V1R6)
  - EUVF.SEUVFEXC for SYSEXEC DD concatenation for TSO

© 2008 IBM Corporation 21

SHARE IBM Systems and Technology Group IBM

## Network Authentication Service - Installation

§ Configuration in krb5.conf file

- ▶ KRB5\_CONFIG environment variable
- ▶ default is /etc/skrb/krb5.conf
- ▶ sample in /usr/lpp/skrb/examples/krb5.conf
- ▶ permissions should be read for everyone, only administrator may modify
- ▶ modified only in code page 1047

© 2008 IBM Corporation 22

SHARE IBM Systems and Technology Group IBM

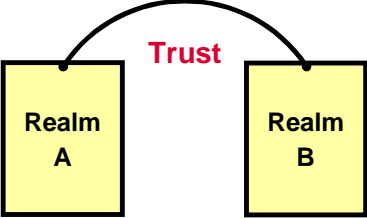
## Network Authentication Service - Installation ...

- ⌘ Set-up RRSF (RACF Remote Sharing) in local mode
- ⌘ Define SKRBKDC application and USERID as started task
- ⌘ Copy SKRBKDC environment variables definitions to /etc/skrb/home/kdc/envar
- ⌘ Set TZ and RESOLVER\_CONFIG for your installation

© 2008 IBM Corporation 23

SHARE IBM Systems and Technology Group IBM


## Registry Definitions




Commands must be entered to define:

- A local realm
- Inter-realm trust relationships (between KDCs)
- Local and foreign principals

© 2008 IBM Corporation 24



IBM Systems and Technology Group




## Realm Commands


- Ø Realm definition with RDEFINE/RALTER
  - ▶ Realm class profile
  - ▶ Ticket life values
    - DEFTKTLFE - default ticket life
    - MAXTKLFE - maximum ticket life
    - MINTKTLFE - minimum ticket life
    - Only valid for local realm
    - If one is specified all three values must be for RDEFINE
    - All three values must be on command or in DB for RALTER
    - Range from 1 to 2,147,483,647 seconds

© 2008 IBM Corporation

25



IBM Systems and Technology Group




## Realm Commands ...


- Ø **KERBNAME** - unqualified name of the local Kerberos realm
  - Max length of 117 characters
  - Can not contain '/'
  - EBCDIC variant characters should not be used
- Ø **PASSWORD** - realm password
  - Max length of 8 characters
  - EBCDIC variant characters should not be used
- Ø **ENCRYPT** – Supported encryption types
  - Choice of DES, Triple DES and DES with Derivation
- Ø **NODEFTKTLFE, NOMAXTKLFE, NOKERBNAME, NOMINTKTLFE, NOPASSWORD, NOENCRYPT** and **NOKERB** only for RALTER

© 2008 IBM Corporation

26



IBM Systems and Technology Group




## Realm Commands ...


§ Profile naming

- ▶ Defining a local realm
  - Profile name must be KERBDFLT
  - KERBNAME field has unqualified local realm name
  - Realm name is rolled to upper case
- ▶ Defining an inter-realm trust relationship
  - Can consist of two REALM class profiles
  - Profile name: /.../LOCAL\_REALM/krbtgt/REALM\_2  
w krbtgt/REALM\_2@LOCAL\_REALM
  - Profile name: /.../REALM\_2/krbtgt/LOCAL\_REALM  
w krbtgt/LOCAL\_REALM@REALM2

© 2008 IBM Corporation 27



IBM Systems and Technology Group



## Realm Command *Examples*

§ Local Realm example:

- ▶ RDEFINE REALM KERBDFLT KERB(KERBNAME(KRB390.IBM.COM)  
PASSWORD(xxxx) MINTKTLFE(15) DEFTKTLFE(36000)  
MAXTKTLFE(86400))

§ Inter-realm trust example:

- ▶ RDEFINE REALM /.../KRB390.IBM.COM/krbtgt/KRB2000.IBM.COM  
KERB(PASSWORD(passwr1 ))
- ▶ RDEFINE REALM /.../KRB2000.IBM.COM/krbtgt/KRB390.IBM.COM  
KERB(PASSWORD(passwr2))

© 2008 IBM Corporation 28

SHARE IBM Systems and Technology Group IBM

## User Commands

§ Local principal definition with ADDUSER/ALTUSER

- Ø Local realm must exist before issuing command
- Ø **MAXTKLFE** specifies the local principal maximum ticket life
- Ø **KERBNAME** is the unique name of a local principal.
  - Can not contain '@'
  - Variant characters should not be used
  - Can not exceed 240 characters when fully qualified with the local realm name  
 ../local\_realm/kerbname\_1
  - Must be entered unqualified
- Ø **ENCRYPT** specifies supported encryption types
  - Choice of DES, Triple DES and DES with Derivation
- Ø **NOMAXTKLFE, NOKERBNAME, NOENCRYPT, NOKERB** only valid on ALTUSER
- Ø Kerberos keys generated at non-expired password setting
- Ø KERBLINK mapping profile created/updated

© 2008 IBM Corporation 29

SHARE IBM Systems and Technology Group IBM

## LISTUSER - Key information

When the initial KERB segment is added via  
**ADDUSER USER1 KERB(KERBNAME(User1))**  
 the password is not yet synchronized with the Kerberos local principal's password:

```
LISTUSER USER1 KERB NORACF


USER=USER1
KERB INFORMATION
-----
KERBNAME= User1
```

After a password change, the key is generated !


```
USER=USER1
KERB INFORMATION
-----
KERBNAME= User1
KEY VERSION= 001
```

← key

© 2008 IBM Corporation 30



IBM Systems and Technology Group




## Mapping Foreign Users


- § Foreign Kerberos principals are mapped to a RACF identity using KERBLINK class profiles
  - § RDEFINE KERBLINK /.../foreign\_realm/foreign\_principal APPLDATA('racf\_user')
    - ▶ Maps single foreign principal to a RACF userid
  - § RDEFINE KERBLINK /.../foreign\_realm/ APPLDATA('racf\_user')
    - ▶ Maps all principals for a single realm to a RACF userid
- § Realm names are rolled to upper case

© 2008 IBM Corporation

31



IBM Systems and Technology Group



## Steps for Getting Started

- § Install/Customize Server
- § Define local realm
  - ▶ RDEFINE REALM KERBDFLT KERB(KERBNAME(realm) PASSWORD(realmpass))
- § Define inter-realm relationship
  - ▶ RDEFINE REALM /.../realm1/krbtgt/realm2 KERB(PASSWORD(TrustP1))
  - ▶ RDEFINE REALM /.../realm2/krbtgt/realm1 KERB(PASSWORD(TrustP2))
- § Define local principals
  - ▶ ALTUSER user1 KERB(KERBNAME(KerbUSER1)) PASSWORD(usrp) NOEXPIRED
- § Define foreign principals
  - ▶ RDEFINE KERBLINK /.../foreign\_realm/foreign\_principal APPLDATA('racf\_user')
    - maps single principal to a RACF user
  - ▶ RDEFINE KERBLINK /.../foreign\_realm/ APPLDATA('racf\_user')
    - Maps all principals for a single realm to a RACF userid

© 2008 IBM Corporation

32



SHARE | IBM Systems and Technology Group | IBM

## z/OS Tid Bits

- TCP/IP V6 supported
- NDBM (New DataBase Manager) support
  - UNIX backed SAF database alternative
  - Not shared by SYSPLEX
  - SAF still required to map principals to RACF IDs
  - kadmin used for administration

© 2008 IBM Corporation 33

SHARE | IBM Systems and Technology Group | IBM

## Dependencies and Gotchas

- § Network Authentication Service implements V5 standard
- § Any application can use R\_ticketerv and R\_usermap to map Kerberos information to RACF
- § Kerberos sever required to be installed prior to any key generation
- § RRSF local node must be defined to allow for keys to be generated for user password application updates
- § Password must be changed after user definition to generate initial keys

© 2008 IBM Corporation 34

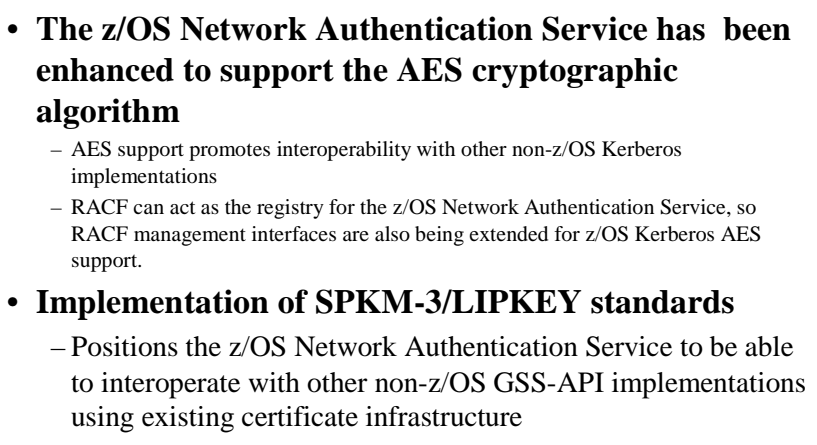


SHARE  
IBM Systems and Technology Group  
IBM

**z/OS V1R9**

© 2008 IBM Corporation 35

The slide features a blue header with the SHARE logo, 'IBM Systems and Technology Group', and the IBM logo. The main content is a large, stylized graphic of a z/OS V1R9 logo, consisting of a grey rounded rectangle with the text 'z/OS V1R9' in bold black font, set against a white background with diagonal lines. The footer contains the copyright notice '© 2008 IBM Corporation' and the page number '35'.



SHARE  
IBM Systems and Technology Group  
IBM

- **The z/OS Network Authentication Service has been enhanced to support the AES cryptographic algorithm**
  - AES support promotes interoperability with other non-z/OS Kerberos implementations
  - RACF can act as the registry for the z/OS Network Authentication Service, so RACF management interfaces are also being extended for z/OS Kerberos AES support.
- **Implementation of SPKM-3/LIPKEY standards**
  - Positions the z/OS Network Authentication Service to be able to interoperate with other non-z/OS GSS-API implementations using existing certificate infrastructure

© 2008 IBM Corporation 36

The slide features a blue header with the SHARE logo, 'IBM Systems and Technology Group', and the IBM logo. The main content is a list of two bullet points, each with a sub-bullet. The first bullet point is bolded and describes enhancements to the z/OS Network Authentication Service to support the AES cryptographic algorithm. The second bullet point is bolded and describes the implementation of SPKM-3/LIPKEY standards. The footer contains the copyright notice '© 2008 IBM Corporation' and the page number '36'.

SHARE IBM Systems and Technology Group IBM

## RACF interface changes for AES

- **Commands, panels, utilities, and SAF callable services which support Kerberos encryption types are enhanced to also support 128-bit and 256-bit AES.**

**Allowed on both USER and REALM class profiles**

```
ADDUSER RONTOMS KERB(KERBNAME(raeburn) ENCRYPT(NOAES256))

LISTUSER RONTOMS NORACF KERB
USER=RONTOMS

KERB INFORMATION
-----
KERBNAME= raeburn
KEY ENCRYPTION TYPE= DES DES3 DESD AES128 NOAES256
```

- **Note that using a command or panel to enable use of AES keys, does not generate new keys...a password change is also required!**

© 2008 IBM Corporation 37

SHARE IBM Systems and Technology Group IBM

## The GSS-API

- **Generic Security Service Application Programming Interface (GSS-API) support is provided by the z/OS Network Authentication Service**
  - The GSS-API is a set of programming interfaces which abstract identity authentication, message origin authentication and integrity, and message confidentiality
  - In concept, a secure application developed using the GSS-API should be able to work over different security mechanisms without changes to the application
- **Previously, the z/OS Network Authentication Service GSS-API offering only supports the Kerberos security mechanism**
- **LIPKEY and SPKM-3 support has been added as extensions to the GSS-API**

© 2008 IBM Corporation 38

SHARE IBM Systems and Technology Group IBM

## SPKM-3

- **The Simple Public-Key GSS-API Mechanism (SPKM) is based on a public key infrastructure, not the Kerberos symmetric-key infrastructure**
  - SPKM-3 does not use secure timestamps, enabling secure authentication in environments without access to secure time
  - Designed to be flexible, for example providing Algorithm Identifiers for specifying various algorithms to be used by communicating peers
  - Provides support for asymmetric algorithm-based digital signatures
  - Data formats and procedures are designed to be as similar to the Kerberos mechanism as possible for ease of implementation by applications which are already Kerberos enabled
- **SPKM-3 uses the same certificate infrastructure as SSL**

© 2008 IBM Corporation 39

SHARE IBM Systems and Technology Group IBM

## LIPKEY

- **LIPKEY (a Low Infrastructure Public Key Mechanism using SPKM) is a GSS-API security mechanism which can be used when the initiator (client) does not have a certificate and instead uses user ID and password for authentication**
- **It consists of a client with no public key certificate, accessing a server with a public key certificate (in contrast, in SPKM-3, both client and server require access to certificates)**
- **The server must have access to a user ID/password repository (we use the \_\_passwd system routine, with setup/restrictions documented in the z/OS Network Authentication Service Programming Guide)**


© 2008 IBM Corporation 40

SHARE IBM Systems and Technology Group IBM

## How LIPKEY works

A client using the LIPKEY mechanism

- Obtains the server's certificate
- Verifies that it was signed by a trusted CA
- Generates a random session symmetric key
- Encrypts the session key with the server's public key
- Sends the encrypted session key to the server
- At this point, the client and server have a secure channel, so the client can provide a user name and password for authentication



© 2008 IBM Corporation 41

SHARE IBM Systems and Technology Group IBM

## What externals were changed?

- New z/OS Network Authentication Service environment variables are added, such as `GSS_KEYRING_NAME` (specifies the name of the key database HFS file or the SAF key ring)
- New messages are added
- GSS-API descriptions, parameter descriptions, and parameter format descriptions are modified to indicate/provide support for the two new security mechanisms, SPKM-3 and LIPKEY

For example, the `desired_mech` parameter of the `gss_acquire_cred` function is modified to indicate that `gss_mech_spk3` and `gss_mech_lipkey` are now supported in addition to `gss_mech_krb5`

© 2008 IBM Corporation 42

SHARE IBM Systems and Technology Group IBM

## Migration & Coexistence Considerations

- Problems can occur when RACF is the Kerberos registry and the database is shared between z/OS V1R9 and lower-level systems
  - As always, administration should be done on the higher level system
  - The fix for RACF APAR OA20304 (V1R7 PTF UA33765 / V1R8 PTF UA33766) must be applied in order for Kerberos to use **triple DES** and **DES with derivation** correctly on the lower-level systems

© 2008 IBM Corporation 43


SHARE IBM Systems and Technology Group IBM

## Session Summary


What we have covered:

- ▶ What Kerberos is and does
- ▶ How SAF/RACF interacts with the Network Authentication Service
- ▶ How an application would interact with SAF to map Kerberos constructs to RACF constructs
- ▶ How to install and configure Kerberos support
- ▶ An overview of newer support

© 2008 IBM Corporation 44



IBM Systems and Technology Group



## References

**Ø IBM Books**

- § SA22-7691 z/OS Security Server RACF Callable Services
- § SA22-7687 z/OS Security Server RACF Command Language Reference
- § GA22-7680 z/OS Security Server RACF Data Areas
- § SA22-7682 z/OS Security Server RACF Macros and Interfaces
- § SA22-7686 z/OS Security Server RACF Messages and Codes
- § SA22-7683 z/OS Security Server RACF Security Administrator's Guide
- § SC24-5926 z/OS Integrated Security Services Network Authentication and Privacy Service Administration
- § SC24-5927 z/OS Integrated Security Services Network Authentication and Privacy Service Programming


**Ø RFCs**

- § RFC 1510 - The Kerberos Network Authentication Service (V5)
- § RFC 1964 - The Kerberos Version 5 GSS-API Mechanism
- § RFC 2078 - Generic Security Service Application Program Interface (V2)
- § RFC 2744 - Generic Security Service Application Program Interface (V2): C Bindings


**Ø Internet**

- § <http://web.mit.edu/kerberos/www/>

© 2008 IBM Corporation 45



IBM Systems and Technology Group



## References for V1R9 Enhancements

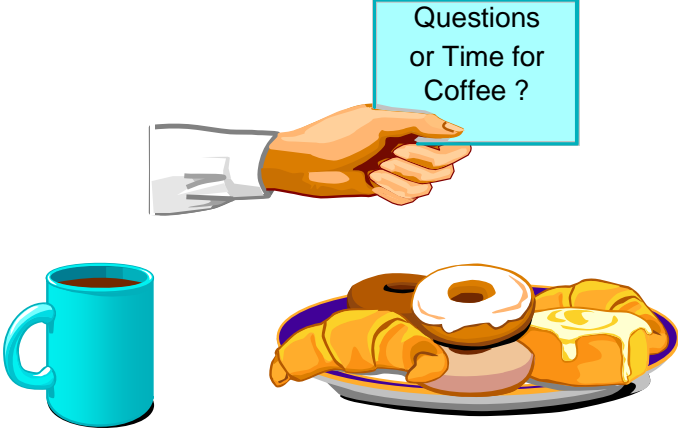
- RFC archives
  - RFC 2025 - The Simple Public-Key GSS-API Mechanism (SPKM)
  - RFC 2847 - LIPKEY - A low infrastructure mechanism Using SPKM
  - RFC 3962 - Advanced Encryption Standard (AES) Encryption for Kerberos
  - RFC 4121 - The Kerberos V5 GSSAPI Mechanism: Version 2
  - RFC2253 UTF-8 String Representation of Distinguished names
  - RFC2459 X.509 Public Key Infrastructure
- SC24-5926 z/OS Network Authentication Service Administration
- SC24-5927 z/OS Network Authentication Service Programming
- SC24-5901 Cryptographic Services System Secure Sockets Layer Programming
- GA22-7800 z/OS Unix System Services Planning
- SA22-7803 z/OS Unix System Services Programming: Assembler Callable Services Reference

© 2008 IBM Corporation 46

SHARE IBM Systems and Technology Group IBM

# Questions ?

Questions or Time for Coffee ?



© 2008 IBM Corporation 47

This slide features a blue header with the 'SHARE' logo, 'IBM Systems and Technology Group', and the 'IBM' logo. The main content area is white with the title 'Questions ?' in large black font. Below the title, a hand in a white sleeve holds a light blue sign that says 'Questions or Time for Coffee ?'. To the left of the sign is a blue coffee cup, and to the right is a plate of pastries including a croissant, a donut, and a slice of pie. The footer is blue with a pattern of dots and contains the text '© 2008 IBM Corporation' and the number '47'.

SHARE IBM Systems and Technology Group IBM

# Reference

© 2008 IBM Corporation 48

This slide features a blue header with the 'SHARE' logo, 'IBM Systems and Technology Group', and the 'IBM' logo. The main content area is white with the word 'Reference' in large black font, centered within a grey rounded rectangle that has a drop shadow. The footer is blue with a pattern of dots and contains the text '© 2008 IBM Corporation' and the number '48'.



SHARE IBM Systems and Technology Group IBM

## R\_ticketserv (IRRSPK00)

§ Parse or extract Kerberos principal

- ▶ Function code
  - TKTS\_RETURN\_NAME (1) - Parse specified ticket and return Kerberos principal name
    - GSS-API context token is input
    - Principal name is output

© 2008 IBM Corporation 49


SHARE IBM Systems and Technology Group IBM

## R\_usermap (IRRSIM00)


§ Map application user

- ▶ Function codes:
  - UMAP\_R\_TO\_K (5) -- return the Kerberos application user identity for the supplied RACF user ID
  - UMAP\_K\_TO\_R (6) -- return the RACF user ID associated with the supplied Kerberos application user identity

© 2008 IBM Corporation 50



SHARE  
IBM Systems and Technology Group



## R\_admin (IRRSEQ00)

§ Functions supported

- ADMN\_ADD\_USER, ADMN\_ALT\_USER, ADMN\_LST\_USER  
ADMN\_ADD\_GENRES, ADMN\_ALT\_GENRES,  
ADMN\_LST\_GENRES to support KERB segment fields

§ Fields

- KERBNAME - realm or principal name
- MAXTKTLF - realm or principal maximum ticket life
- MINTKTLF - realm wide minimum ticket life
- DEFTKTLF - realm wide default ticket life
- PASSWORD - realm password

© 2008 IBM Corporation

51