#SHAREorg

# Enterprise Security: Building On All Your Assets

Neil Beesley
Covariant Inc
neil.beesley@covariant-systems.com

Friday August 10th 2012
Session: 11369

#SHAREorg

# Part 1: Introduction

# Security requirements

- Authentication of legitimate users, i.e. knowing who someone is… despite:
  - Different ID's and multiple passwords across platforms
  - z/OS, UNIX, LINUX, Windows [Active Directory]
- Prevent unauthorized access to applications and data
- Integration with platform access control mechanisms
- Accountability and providing reliable audit trail
- Satisfy regulatory requirements

# Security Challenges

- Additional cost to deploy applications
- Inconvenience and lost time for users
- Cost of security administration
- Effort required for successful audit
- Multiple layers of security – perimeter, IP security, servers and OS, and the 'applications'
- Different approaches to authentication: Password, X509 certificate, LDAP, custom password managers and **Kerberos**

- Which facilities are available with vendor products ?

# Heterogeneous challenge

- No longer simple security requirement with single environment, TP monitor, language and DBMS
- Extensible platforms: Windows, UNIX, z/OS, .NET, Web services and SOA
- Integration connects separate application components executing on different platforms
  - Typical user may have access to applications on:
    - Windows - Active Directory Forest/Domain(s)
    - Unix Server(s) – Often Kerberos managed
    - Mainframe server(s) e.g. z/OS – RACF/ACF2/TSS
- Different user ID password on each platform

#SHAREorg

# Part 2: Kerberos

# What is Kerberos

- A network authentication service
- MIT, Heimdal, Microsoft implementations
- Authentication between disparate security authorities in z/OS, UNIX, Windows
- Kerberos issues tickets for authentication against a service and optionally a session key for data encryption
- Kerberos KDC has the following components
  - Authentication Server (AS)
  - Ticket Granting Server (TGS)
  - Database (KDB)

# Kerberos Terminology

- Ticket: opaque envelope enabling user identity, dynamic session key, time stamp, ticket lifetime and the service name to be derived
- Realm: Kerberos domain which includes all entities known to the KDC
- Principal: an identity which is defined to a realm, such as a user
- Service Principal: principal under which a service executes
- Trust: Two or more realms can be configured to trust the authentication of each other's Principals, e.g. a Windows AD user can be accepted in a Unix or z/OS KDC realm

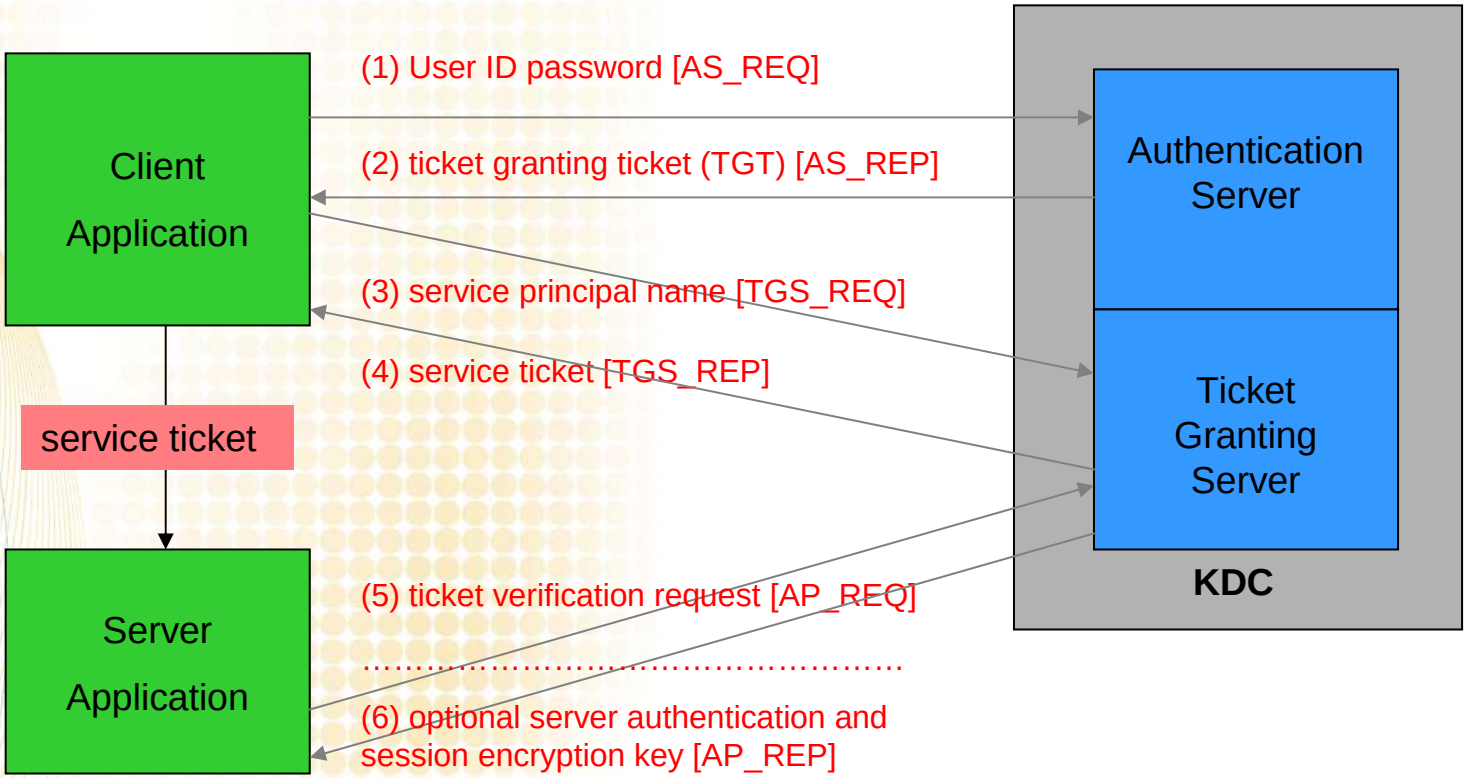
# Why is Kerberos interesting..

- Method to establish authentication without sending user ID password (user already authenticated to local security authority)
- Leverages one time ticket, valid only for a finite period
- Ticket gives access to just one particular *service*
- Authentication between separate Kerberos realms
- Incremental deployment plan across databases or other services
- Kerberos suitable for heterogeneous environments

- Therefore a user with different ID's on each platform can be mapped and the need for storing or supplying password across platforms is avoided

#SHAREorg

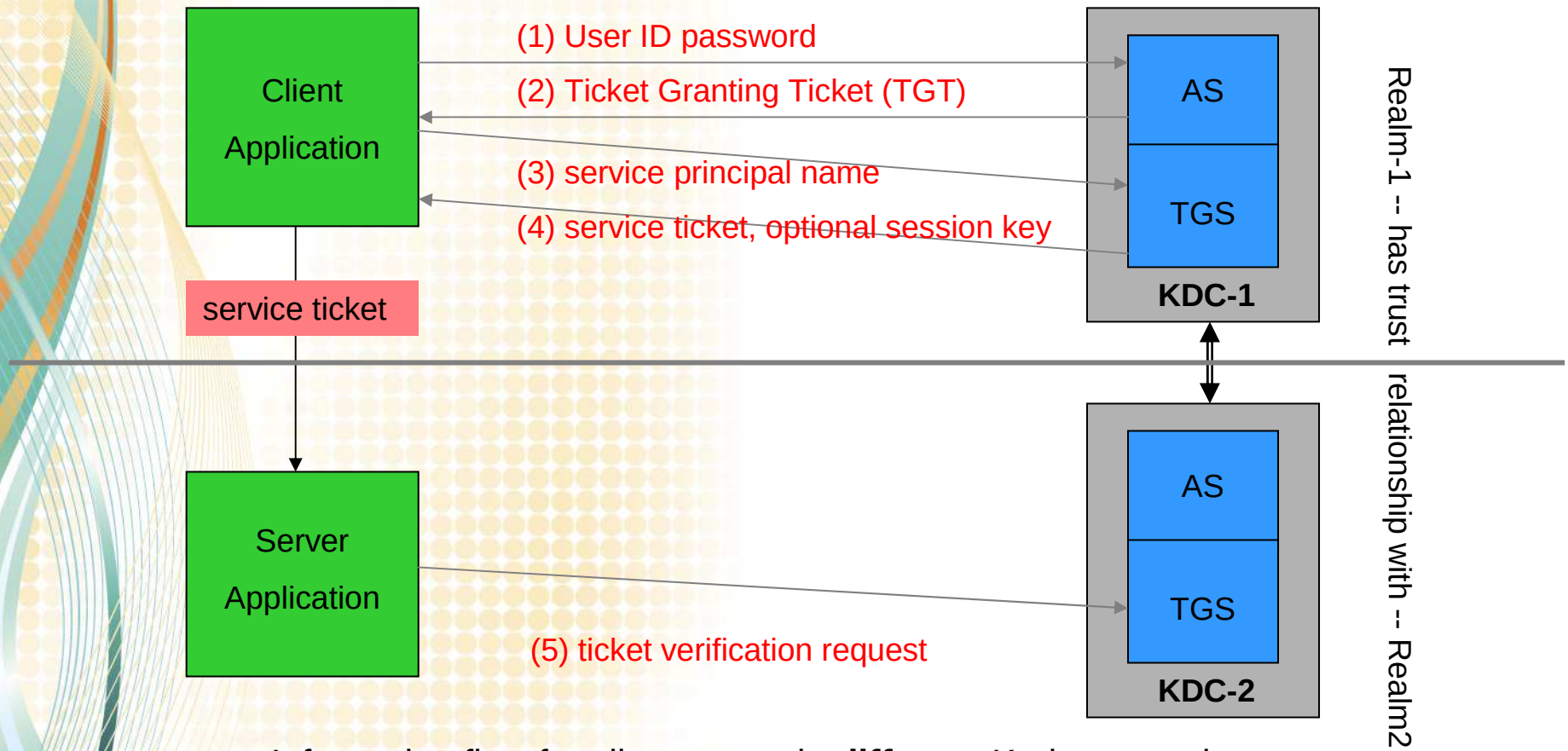
# Overview of Kerberos




Information flow for client server in same Kerberos realm

#SHAREorg

# Overview of Kerberos




Information flow for client server in **different** Kerberos realms



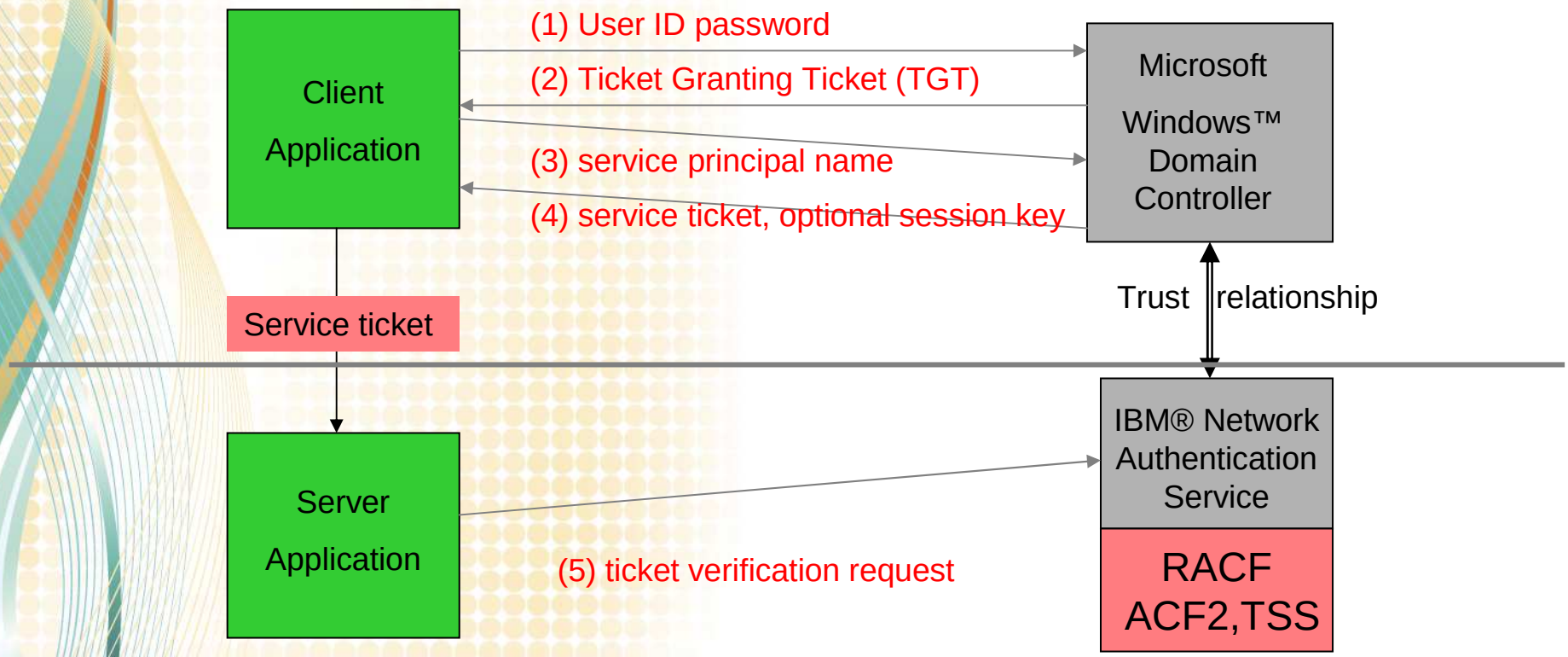
SHARE in Anaheim 2012

# Kerberos on z/OS

- IBM Network Authentication Service
- Integration with RACF/ACF2/Top Secret enables authorization and audit functionality
- RACF/ACF2/Top Secret contain user ID mappings between different platforms
- RACF/ACF2/Top Secret contain the trust relationships between different Kerberos realms
- DB2, WAP, FTP, Telnet, LDAP, Rshd… etc

#SHAREorg

# Kerberos on z/OS




Information flow for client server from Windows to z/OS



SHARE in Anaheim 2012

# Kerberos benefits

- Authentication across Windows, UNIX, z/OS
- Authorization and audit can be applied, for example using RACF/ACF2/Top Secret
- Ability to generate session keys (encryption)
- Applicable to any heterogeneous application

- So, why isn't Kerberos supported by all applications?

# Kerberos requirements

- Peer [components] must exchange ticket data
  - Kerberos does not provide means for transporting ticket to remote peer application components
- Configuration not straight-forward
  - Trust configuration is tricky
  - Mapping to local user profile is platform specific
  - Errors can be cryptic
- Not all vendor products do this yet, but this has not prevented customers implenting Kerberos with them...

# Part 3: User Scenarios

- DB2 implementation with DB2 Connect from LINUX, UNIX and Windows clients
- SAP Sybase access from mainframe client
- Software AG webMethods EntireX implementation

# DB2 z/OS from DB2 Connect

- Standard IBM product functionality for LUW client authentication against DB2 z/OS for 3GL/Perl/Java
- Finesse is GSS-API DB2 Connect (Kerberos) client side plug-in module for UNIX/LINUX -- 3GL/non JAVA
- Implementation requires revised security definitions on z/OS and distributed systems plus software configuration changes – SPN definitions client side
- JAVA applications require additional properties for JDBC type 4 call
- Non JAVA applications unchanged – DB2 Connect catalog definitions needed
- Standard SAF security access rules used
- PTF to cater for multi-LPAR DB2 servers (different service principal names with different LPARs here)

# DB2 z/OS from DB2 Connect: JAVA

**JAVA code fragment for JDBC type 4 Kerberos properties (common code -- can be encapsulated)**

```
java.util.Properties properties = new java.util.Properties();
// Set security mechanism to Kerberos
properties.put("securityMechanism",
new String("" + com.ibm.db2.jcc.DB2BaseDataSource.KERBEROS_SECURITY + ""));
// Set Kerberos Service Principal Name
properties.put("kerberosServerPrincipal",
 new String("" + “db2mstr/lpar1.krbzos.covnt.com@KRBZOS.COVNT.COM" +""));
url = "jdbc:db2://" + server + ":" + portNumber + "/" + dbName +
":traceDirectory=/var/tmp" +
";traceFile=db2_javat4_trace" +
";traceFileAppend=false" +
";traceLevel=" +
com.ibm.db2.jcc.DB2BaseDataSource.TRACE_ALL +";";
Class.forName("com.ibm.db2.jcc.DB2Driver").newInstance();
con = DriverManager.getConnection( url, properties );
```

# DB2 z/OS from DB2 Connect: Non JAVA

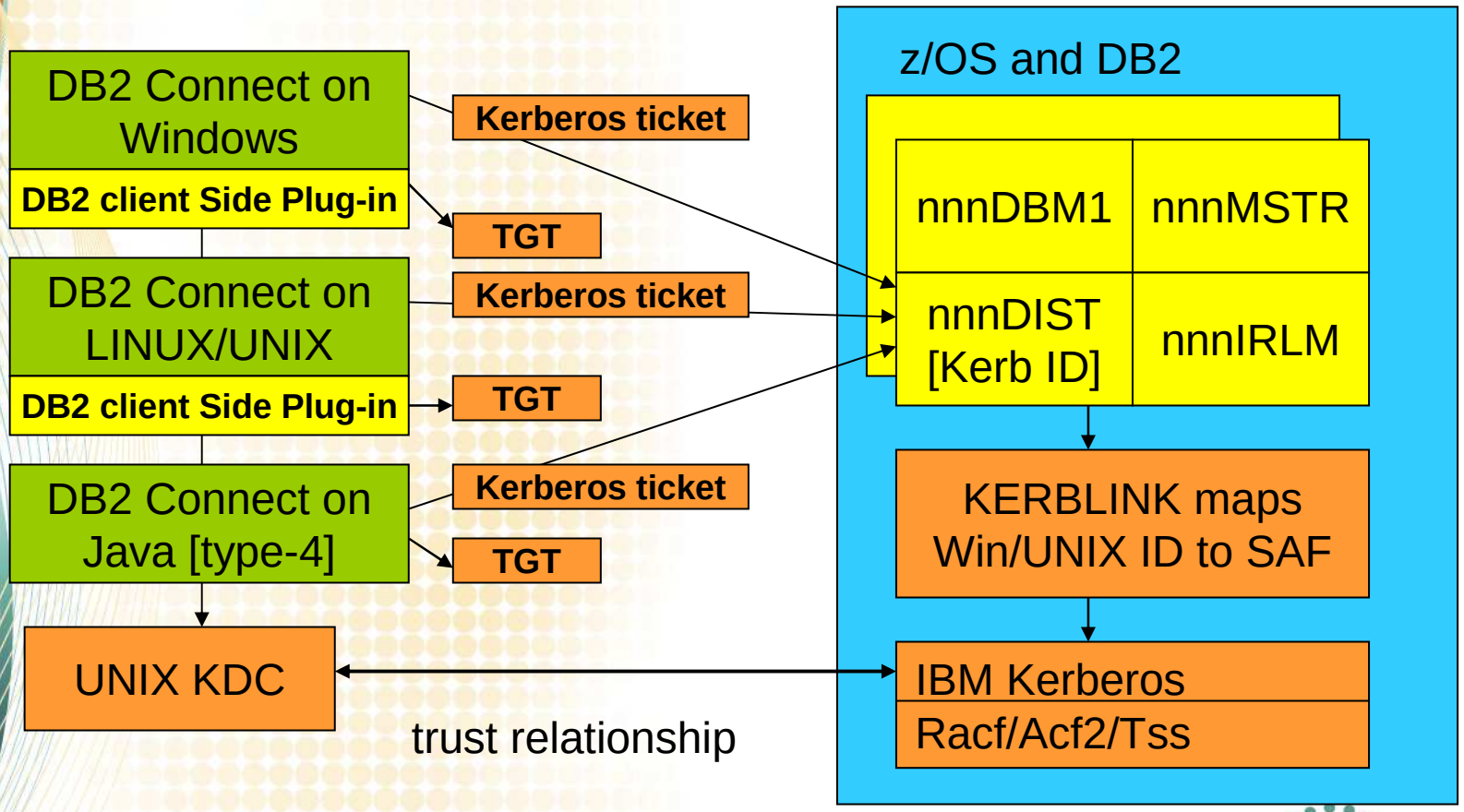
- DB2 Connect catalog definitions -- database alias and client side plug-in:

db2 CATALOG DATABASE TESTDB2 AS TESTDB2K AT NODE TESTNOD1 AUTHENTICATION KERBEROS TARGET PRINCIPAL db2mstr/lpar1.krbzos.covnt.com@KRBZOS.COVNT.COM

db2 UPDATE DBM CFG USING CLNT_KRB_PLUGIN covnt_krb5_plugin

#SHAREorg

# DB2 z/OS from DB2 Connect





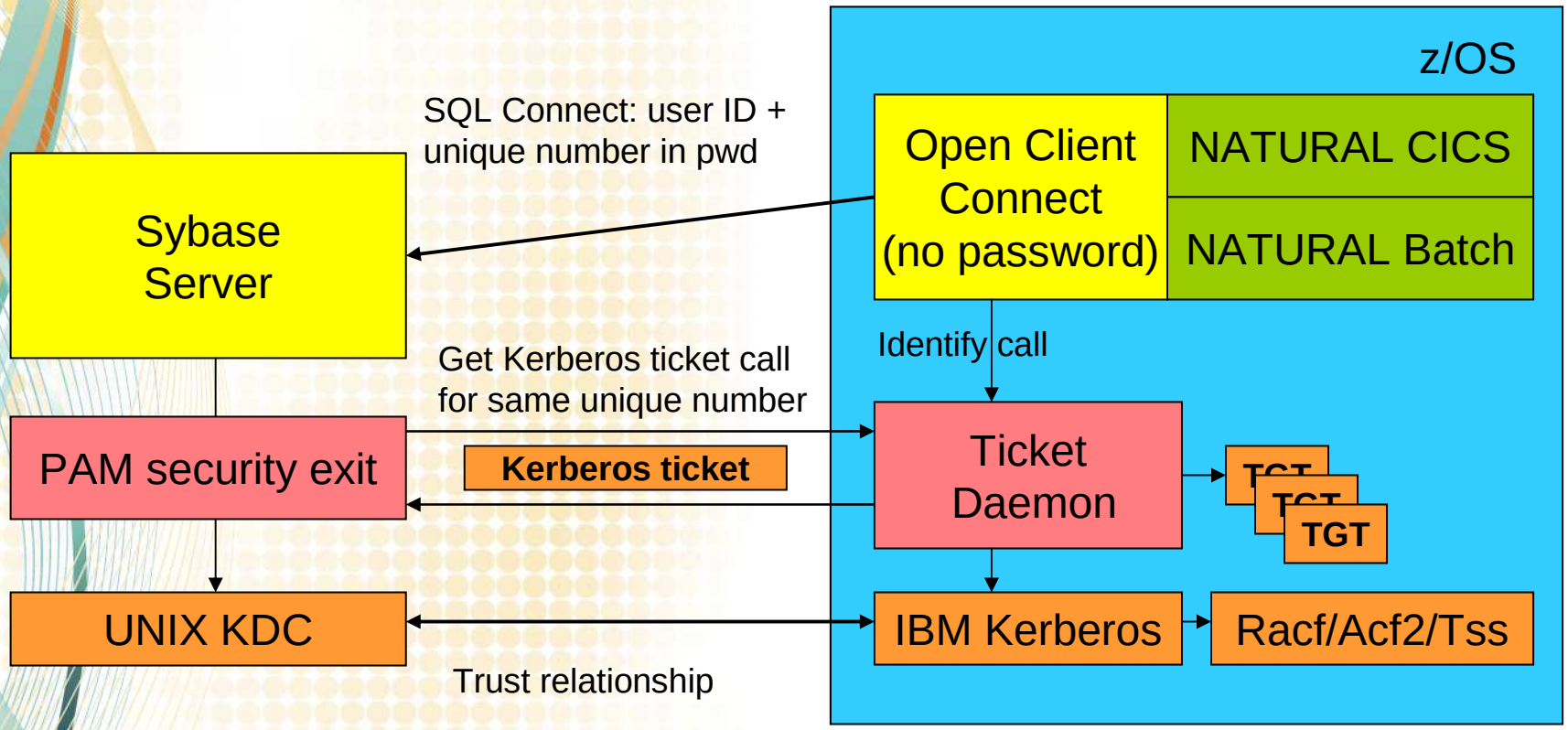

SHARE in Anaheim 2012

# Sybase access from z/OS Client

- Client application z/OS batch/CICS – Sybase servers in UNIX
- No vendor functionality available: implemented with user exits
- Obtain client side Kerberos ticket in application user exit and transport **out-of-band** to UNIX server where client side ticket is validated via GSS-API in PAM module (pluggable authentication)
- Implementation requires revised security definitions on z/OS and distributed systems plus additional software components and configuration changes
- TSS fixes required for obtaining client side Kerberos service tickets (R_Gensec call)

#SHAREorg

# Sybase access from z/OS Client





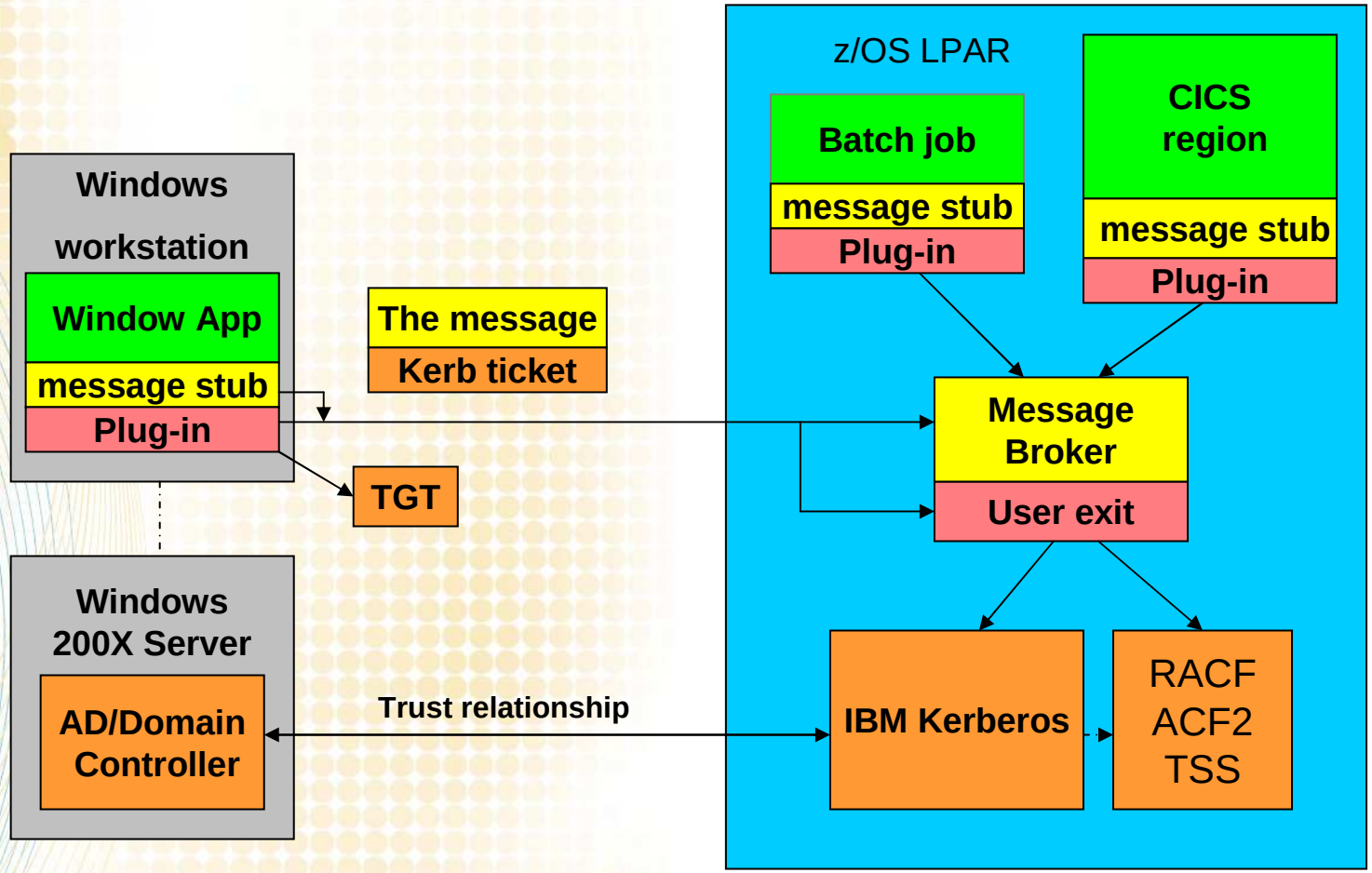

SHARE in Anaheim 2012

# Implementation for Software AG EntireX

- LUW clients using z/OS message service on z/OS – but can be other way round. Similar to IBM MQ
- Implemented using 3-rd party solution
- **Out-of-band** Kerberos ticket acquisition/verification across platforms using standard IBM and GSS-API interfaces. IBM C/LE written code.
- Implementation requires revised security definitions on z/OS and distributed systems, plus additional software components and configuration changes
- Standard SAF security access rules used
- Newer Kerberos encryption protocols for WS-2008

#SHAREorg

# Implementation for Software AG EntireX






SHARE in Anaheim 2012

# Part 4: Configuration tips

- KDC configuration
  - More than one KDC for resilience
  - Client members
  - Application Service members
- RACF user definitions
- RACF Kerberos Realm Definitions
- Kerberos service principal names
- Kerberos name space considerations
- TCP/IP and network requirements
- Gotchas… and diagnostics

#SHAREorg

# KDC Configuration on z/OS

```
[realms]
KRBZOS.COVNT.COM = {
  kdc            = nasdc1.krbzos.covnt.com:88
  kpasswd_server = nasdc1.krbzos.covnt.com:464
  admin_server   = nasdc1.krbzos.covnt.com:749
  auth_to_local  = { RULE:[2:$1](joe)s/^.*$/guest/
                     RULE:[2:$1;$2](^.*;admin$)s/;admin$//
                     DEFAULT } ← UNIX only: z/OS uses KERBLINK
}
KRBWIN.COVNT.COM = {
  kdc            = addc1.krbwin.covnt.com:88
  kpasswd_server = addc1.krbwin.covnt.com:464
}
[domain_realm]
.krbzos.covnt.com = KRBZOS.COVNT.COM
.krbwin.covnt.com = KRBWIN.COVNT.COM
```

/etc/skrb/krb5.conf

# RACF Definitions: Users

- Local RACF User
  - ADDUSER SAFUSR1 KERB(KERBNAME(kerbusr1))
  - Then logon and change password (if not NDBM)
- Foreign users such as defined on Windows
  - RDEFINE KERBLINK /../KRBWIN.COVNT.COM/winusr2 APPLDATA('SAFUSR2') maps between winusr2 and SAFUSR2
  - RDEFINE KERBLINK /../KRBWIN.COVNT.COM/ APPLDATA('SAFUSR3') maps all AD Domain users to SAFUSR3

# RACF Definitions: Realms

- Local Realm (RACF)
  - RDEFINE REALM KERBDFLT KERB(KERBNAME(KRBZOS.COVNT.COM) PASSWORD(pwd) MINTKTLFE(15) DEFTKTLFE(36000) MAXTKTLFE(86400))
- Foreign Realm (Windows domain controller)
  - RDEFINE REALM /.../KRBZOS.COVNT.COM/krbtgt/KRBWIN.COVNT.COM KERB(PASSWORD(trustpwd))
  - RDEFINE REALM/.../KRBWIN.COVNT.COM/krbtgt/KRBZOS.COVNT.COM KERB(PASSWORD(trustpwd))

# Service Principal Name (SPN)

- Server application must run under user ID valid for context acceptance
- KERB segment specifies Kerberos principal name for service, e.g. db2mstr
- If service exists on multiple LPAR use APAR OA32352 – e.g. relevant to DVIPA and DB2 database groups – IBMKLINK statements
- Client application supplies SPN: db2mstr/lpar1.krbzos.covnt.com@KRBZOS.COVNT.COM

# TCP/IP Configuration

- Incorrect z/OS and OMVS hosts definitions
- Inconsistencies with Kerberos principals for KDC/password server/admin server
- Required ports to allow AD and KDC (IBM) to communicate blocked by firewall
  - kdc port: 88
  - kpasswd: 464
  - kadmin: 749 (normally not required)

# Network Design

- Bad choices for Kerberos Realm / AD domain name
  - Avoid using default names from sample configurations
  - Design your name space
    - Avoid name clashes with registered Internet domain names (e.g. krb390.**ibm.com**) or different user ID's in separate domains, e.g. msmith@ibm.krbzos.com != msmith@ibm.krbwin.com
    - Windows AD Forest / Domain structure
    - Kerberos realm name fits AD Forest name hierarchy

# Windows AD configuration

- Registry setting to make AD use TCP vs UDP for Kerberos traffic (Windows 2003)
  - RFC 4120 now obsoletes RFC 1510. RFC 4120 specifies that a KDC must accept TCP requests and this is what Windows 2008 Server and Windows 7 and onwards do
- Registry settings for debugging Kerberos on Windows
  - http://support.microsoft.com/kb/262177
  - HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Lsa\Kerberos\Parameters
    - LogLevel=1

# RACF Configuration tips

- Missing or invalid RACF definitions
  - Kerberos Realm
  - Kadmin
  - Kerberos segments of participating user/server
- If KERB segment of RACF user profile is changed, password must be changed
  - Password change triggers Key generation
- RRSF (Remote RACF) must be configured and running

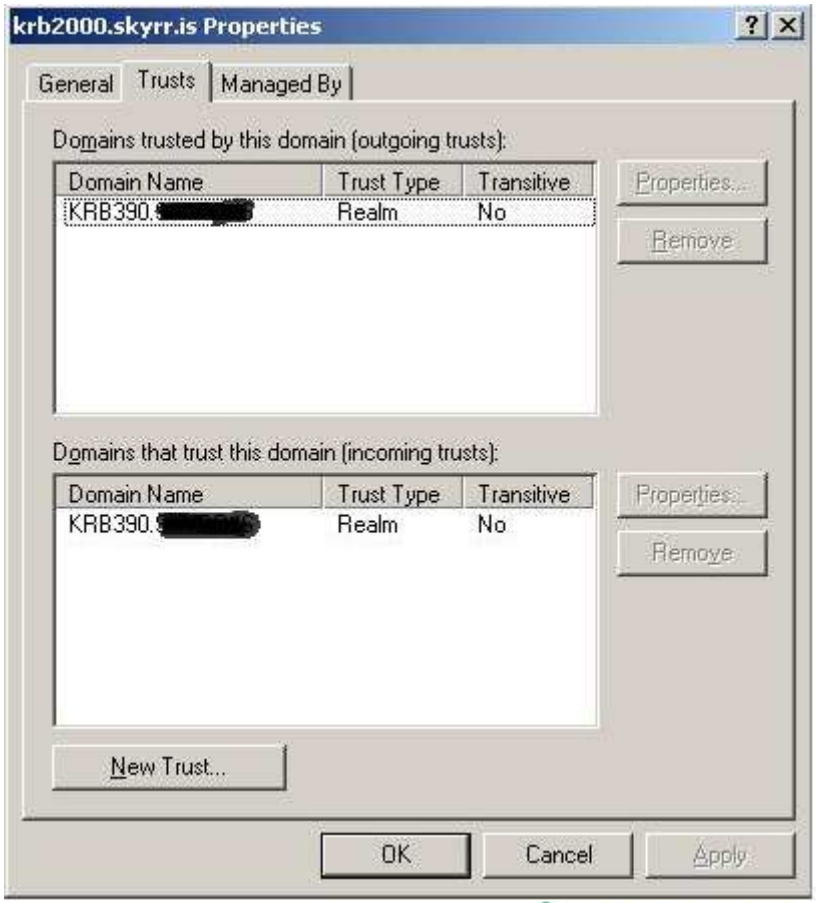
# Clock Skew tips

- Kerberos tolerates up to 5 minutes clock skew
  - If more, cryptic errors occur and authentication will fail
  - NTP time synchronization of all AD DC and all KDC is highly recommended (actually all servers should be time synchronized)
- z9 and z10 and up support STP for time synchronization
  - Licensed feature #1021 required
  - z10+ supports STP syncronization using external NTP time source

#SHAREorg

# Case Sensitivity tips

- Some places are case senstitive
  - Using mixed case Kerberos trust password when RACF is configured to support only upper case
  - Realm/Domain names MUST be upper case in Trust configuration
  - Compare SAF to NDBM Kerberos repository






SHARE in Anaheim 2012

# Summary

- Kerberos provides cross platform enterprise security mechanism. Large investment already on LUW side
- Do a POC especially with existing product already having Kerberos functionality
- Involve cross platform groups and security staff from beginning
- Ensure vendor fixes are researched: talk to vendors up front
- When vendor product does not support Kerberos on z/OS then customer written or 3rd party add-ons can…

# References

Session: 11369

- http://publib.boulder.ibm.com/infocenter/db2luw/v9/topic/com.ibm.db2.udb.apdv.java.doc/doc/cjvjcskb.htm
- http://pic.dhe.ibm.com/infocenter/db2luw/v9r7/topic/com.ibm.db2.luw.admin.sec.doc/doc/c0012013.html
- http://dfusion.com.au/wiki/tiki-index.php?page=Kerberos+Tutorial#The_Authentication_and_Ticket_acquisition_process
- http://en.wikipedia.org/wiki/Kerberos_%28protocol%29
- http://web.mit.edu/kerberos/krb5-current/doc/krb_admins/troubleshoot.html
- http://support.microsoft.com/kb/907272

- http://covariant-systems.com