Information On Demand

DB2/LUW Use of Trusted Context for a VM/VSE-Environment

GSE Leipzig 27.10.2009

weberg@de.ibm.com
IM Techsales
Günter Weber

ON DEMAND BUSINESS™

Disclaimer/Trademarks

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements, or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility, or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information may contain examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious, and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

Trademarks

The following terms are trademarks or registered trademarks of other companies and have been used in at least one of the pages of the presentation:

The following terms are trademarks of International Business Machines Corporation in the United States, other countries, or both: AIX, AS/400, DataJoiner, DataPropagator, DB2, DB2 Connect, DB2 Extenders, DB2 OLAP Server, DB2 Universal Database, Distributed Relational Database Architecture, DRDA, eServer, IBM, IMS, iSeries, MVS, Net.Data, OS/390, OS/400, PowerPC, pSeries, RS/6000, SQL/400, SQL/DS, Tivoli, VisualAge, VM/ESA, VSE/ESA, WebSphere, z/OS, zSeries

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.
Intel and Pentium are trademarks of Intel Corporation in the United States, other countries, or both.
UNIX is a registered trademark of The Open Group in the United States and other countries.
Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.
Other company, product, or service names may be trademarks or service marks of others.

Agenda

- ***Trusted Context***
  - Why TC
  - Scenarios for TC
  - Roles
  - TC
  - SECADM

Why do we need more Security enhancements ?

We don‘t want to allow any application server / client to access our DB2 production system

We cannot identify the individual user, if they come through our Web Application Server

Current problems with 3-tier architectures

Common application server userid used for all communication with DB2

End-user not propagated to DB2

Not possible to audit actions performed by different end-users

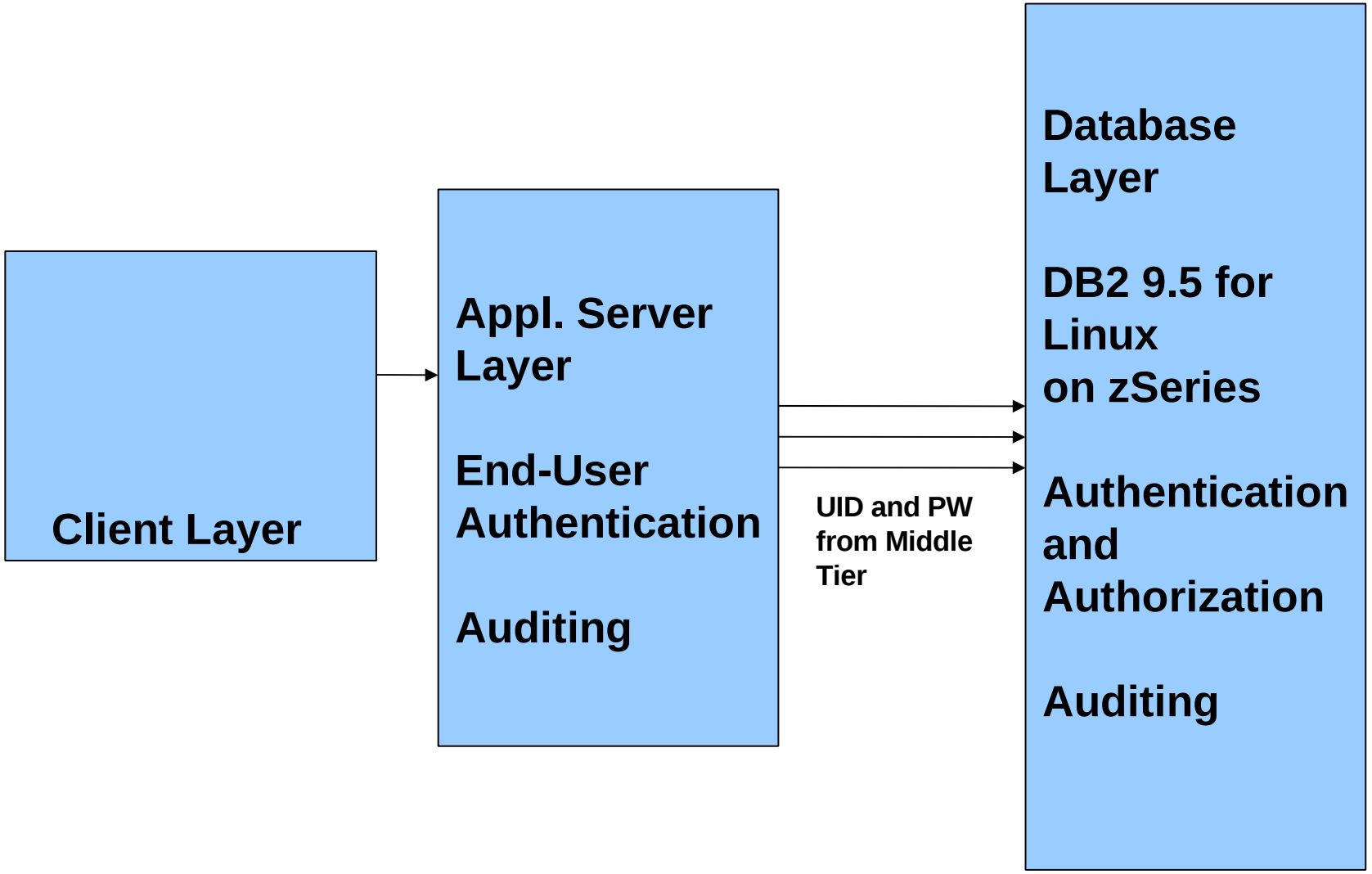
Application server userid must hold all authorizations needed to perform all parts of the application

Application server must make sure that end-users are only allowed to use what they are authorized for

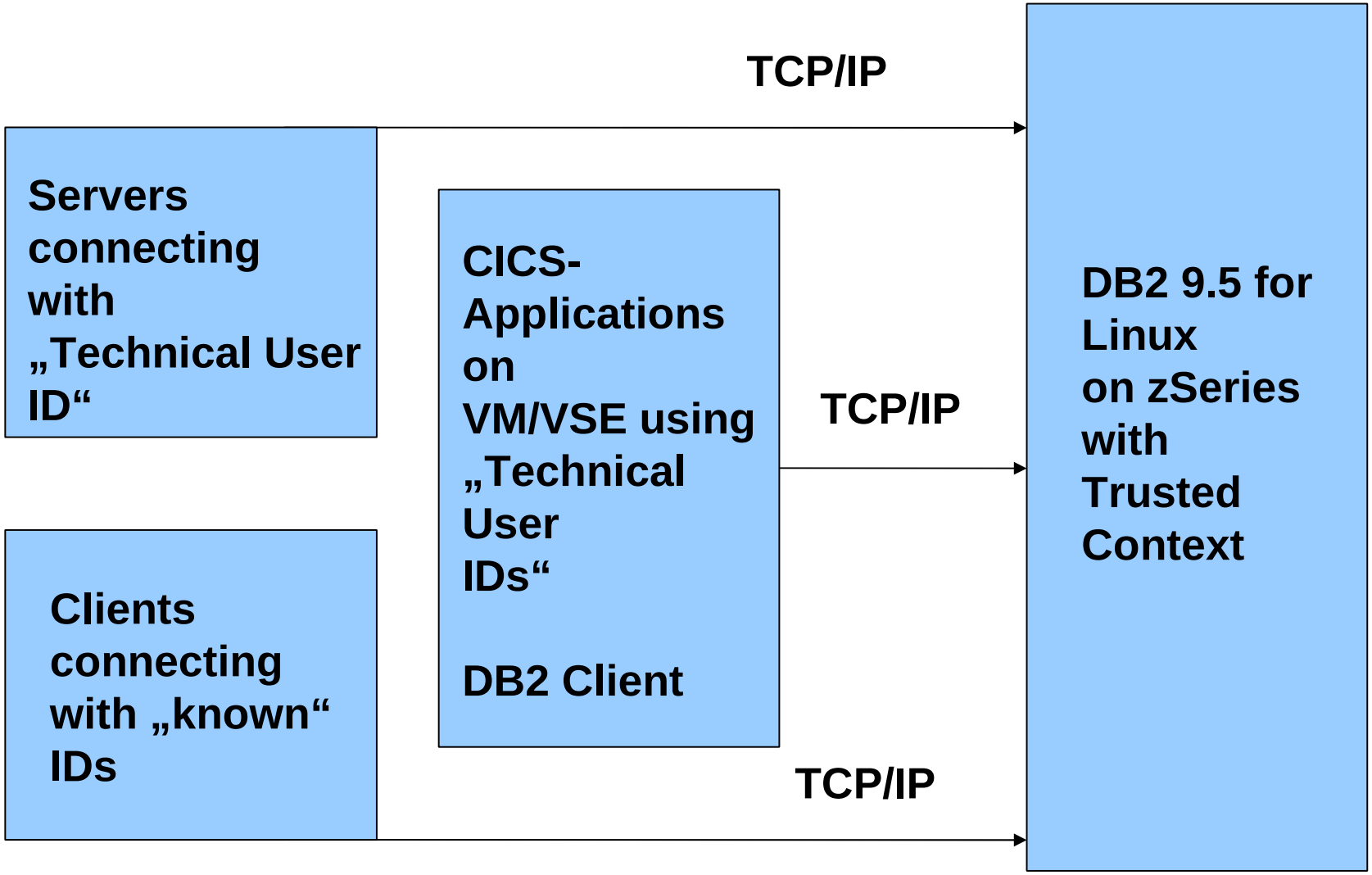
Privileges held by application server userid are valid from any “location”

If application server userid is compromised, the impact is large

Scenarios for TC (1)



Scenarios for TC (2)



Solution: Trusted context and roles

A trusted context is a database object that identifies a certain “location”

Connections created using a trusted context are trusted connections

Within a trusted connection you can
- Switch to another authid (with or without authentication)
  allows the reuse of connections !
- Exercise the use of roles

A role provides context dependent privileges
- The role is available only within the trusted context
- The privileges granted to the role can be exercised only
  through the trusted connection

Role

New database entity
- Not a userid
- Holds privileges like a userid (or group)

Can be associated with a userid within a trusted context via the “Default Role” clause

What is the intention?

Limiting privileges to a particular context

Creating a Role

Creating a Role:

```
CREATE ROLE My_ROLE;
```

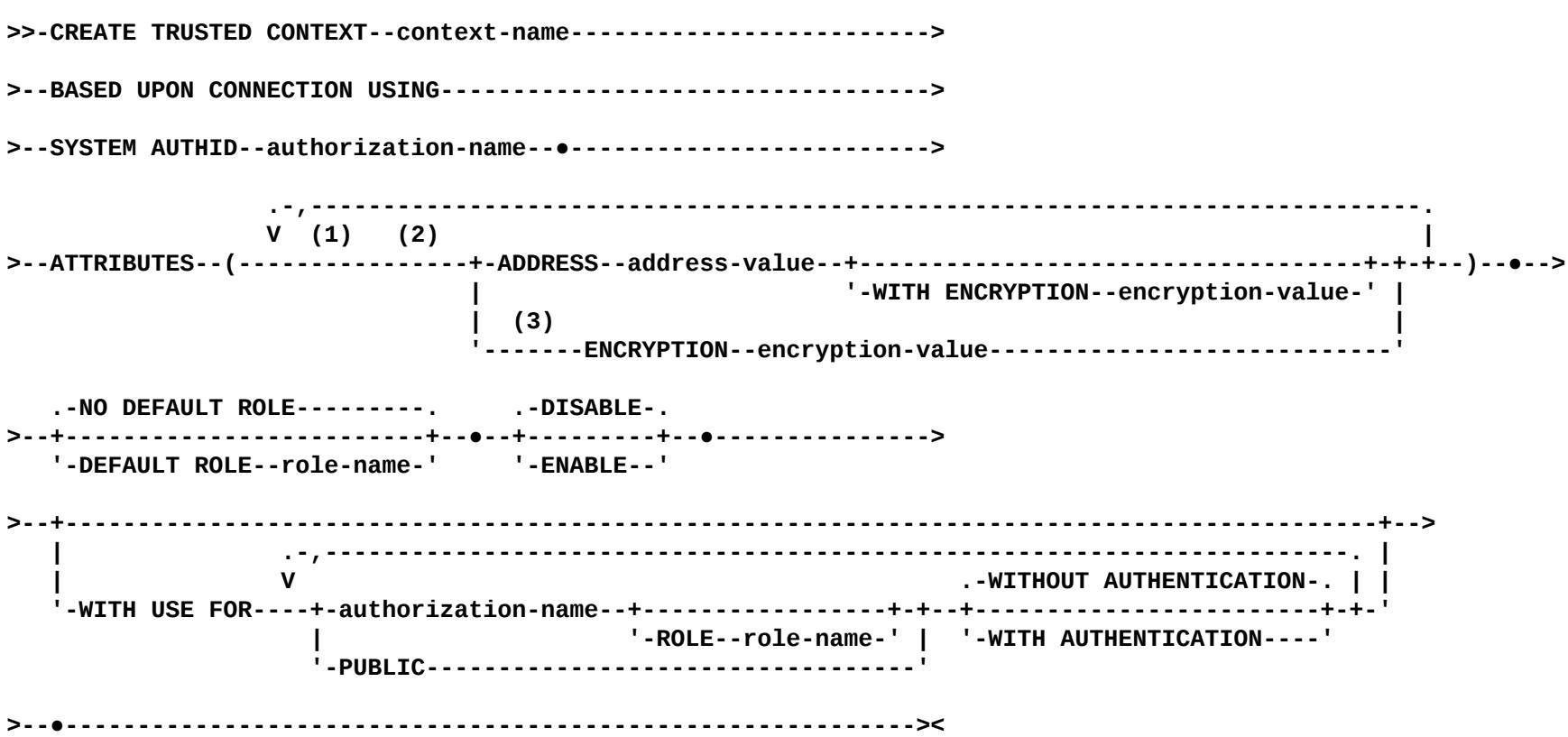
Grant privleges to a ROLE:

```
GRANT SYSADM TO My_ROLE;
```

Use in a trusted context, as default or by user:

```
CREATE TRUSTED CONTEXT My_TC
BASED UPON CONNECTION USING SYSTEM AUTHID aUserID
ATTRIBUTES (ADDRESS ‘1.2.3.4‘)
DEFAULT ROLE My_Role
ENABLE;
```

Creating a TC



A sample Trusted Context DDL

```
CREATE TRUSTED CONTEXT CTX1
BASED UPON CONNECTION USING SYSTEM AUTHID WASADM1
ATTRIBUTES (ADDRESS ‘1.2.3.4',
ADDRESS ‘1.2.3.5')
ENCRYPTION LOW
ENABLE;
```

Trusted Context and Role Definition are reflected in the DB2 Catalog

sysibm.syscontexts, sysibm.syscontextattributes
and corresponding catalog-views
syscat.contexts, syscat.contextattributes

A sample Trusted Context DDL

```
CREATE TRUSTED CONTEXT CTX1
BASED UPON CONNECTION USING SYSTEM AUTHID WASADM1
ATTRIBUTES (ADDRESS ‘1.2.3.4',
ADDRESS ‘1.2.3.5')
ENCRYPTION LOW
ENABLE;
```

The possible Connection trust attributes are:

PROTOCOL: The communication protocol trust attribute. This is used to control which network communication protocols can use the trusted context.

ADDRESS: The network address trust attribute. This is used in conjunction with the PROTOCOL attribute to control which addresses the trusted context can be used with. This is the actual client's IP address or domain name, used by the connection to communicate with the database manager.

ENCRYPTION: The network encryption trust attribute. This specifies the minimum level of encryption of the data stream ("network encryption") for the connection.

AUTHENTICATION: The authentication trust attribute. The attribute specifies the level of authentication required to be performed on the system authorization ID during the establishment of the connection.

Connection Types

Explicit Trusted Connection

You request that the connection be trusted AND the connection meets the server's criteria for being trusted

Implicit Trusted Connection

You do not request that the connection be trusted AND the connection meets the server's criteria for being trusted

can be any kind of connection

-> a regular connection with add. Role privleges

Regular Connection

The connection does not meet the server's criteria for being trusted – if an explicit trusted connection was requested, it results in a regular connection and warning SQL20360W (SQLSTATE 01679) is returned

User Switching with explicit trusted connections

An explicit trusted connection is created via:

- CLI/ODBC SQLConnect, SQLSetConnectAttr
- XA CLI/ODBC XA_open
- JAVA getDB2TrustedPooledConnection, getDB2TrustedXAConnection

Switching to a different user is done via:

- CLI/ODBC SQLSetConnectAttr
- XA CLI/ODBC SQLSetConnectAttr
- JAVA getDB2Connection, reuseDB2Connection

TC and CLI

```
/* set attribute to enable a trusted connection */
  SQLSetConnectAttr(hdbc1,
                    SQL_ATTR_USE_TRUSTED_CONTEXT, ... );
/* Establish a trusted connect to a testdb with SQLConnect() */
/* as user newton                                              */
  SQLConnect( hdbc1, "testdb", SQL_NTS, "newton", SQL_NTS,
  "xxxxx", SQL_NTS );
// Perform some work like creating objects, inserting data etc.
// All the work is performed as user newton
/* Switch the user from newton to zurbie on a trusted connection */
  SQLSetConnectAttr( hdbc1,
                SQL_ATTR_TRUSTED_CONTEXT_USERID,"einstein",
                SQL_IS_POINTER );
  SQLSetConnectAttr( hdbc1,
                SQL_ATTR_TRUSTED_CONTEXT_PASSWORD,"yyyyy",
                SQL_NTS );
```

TC and XA

```
#-- Allocate the environment handle
sqlallocenv 1
#-- Set the Trusted Context bit, System Authid and Password
xaopen 10 "DB=stlec1,sreg=t,SPM=domino,TCTX=TRUE,
uid=newton,PWD=xxxxx" TMNOFLAGS
#-- Allocate the connection handle
sqlallocconnect 1 1
sqlconnect 1 stlec1 -3 newton -3 xxxxx -3
#-- switch the userid to „einstein“ & set the password
sqlsetconnectattr 1 SQL_ATTR_TRUSTED_CONTEXT_USERID einstein
sqlsetconnectattr 1 SQL_ATTR_TRUSTED_CONTEXT_PASSWORD yyyyy
#-- Start a transaction
#-- This will switch the user to einstein
xastart 10 99 gtrid bqual TMNOFLAGS
```

SECADM

SECADM authority

Very powerful authority level introduced in DB2 9.1

Trusted Contexts and roles can only be managed by a user with SECADM authority.

SECADM can only be granted by SYSADM, and only to a user (not a group !) -> GRANT secadm ON DATABASE TO db2sec;

SECADM also manages security labels, policies, and Label-Based Access Control (LBAC)

Securing a VSE/CICS Appl. via TC

Uses an implicit Trusted Context – no change for appl. needed

On DB2 zLinux create `TRUSTED CONTEXT vseappl`

```
BASED UPON CONNECTION USING SYSTEM AUTHID cicsid
ATTRIBUTES (ADDRESS ‘VSE_IP',....
DEFAULT ROLE cicsrole
```

UserID „cicsid“ is only granted „connect“ priv. in DB2 zLinux

Role „cicsrole“ is created an has all needed privs.

When connecting from VSE_IP all the priv. from Role „cicsrole“ determine what userid „cicsid“ can do

More Information

Redbooks:
http://www.redbooks.ibm.com/

DB2 Security and Compliance Solutions for Linux, UNIX, and Windows – SG247555

Developerworks:
www.ibm.com/developerworks/db2/library/techarticle/

Implement DB2 for Linux, UNIX, and Windows trusted contexts and roles in a Web application

Use trusted context in DB2 client applications

End-to-end federated trusted contexts in WebSphere Federation Server V9.5

Questions ?