

ibm.com



# Overview of OS/390 Security for eBusiness



**Redbooks**

International Technical Support Organization

© Copyright IBM Corp. 2001

IBM

# Agenda



- Network Level Protection
- Platform Security
- Transaction Security
- LDAP Directory

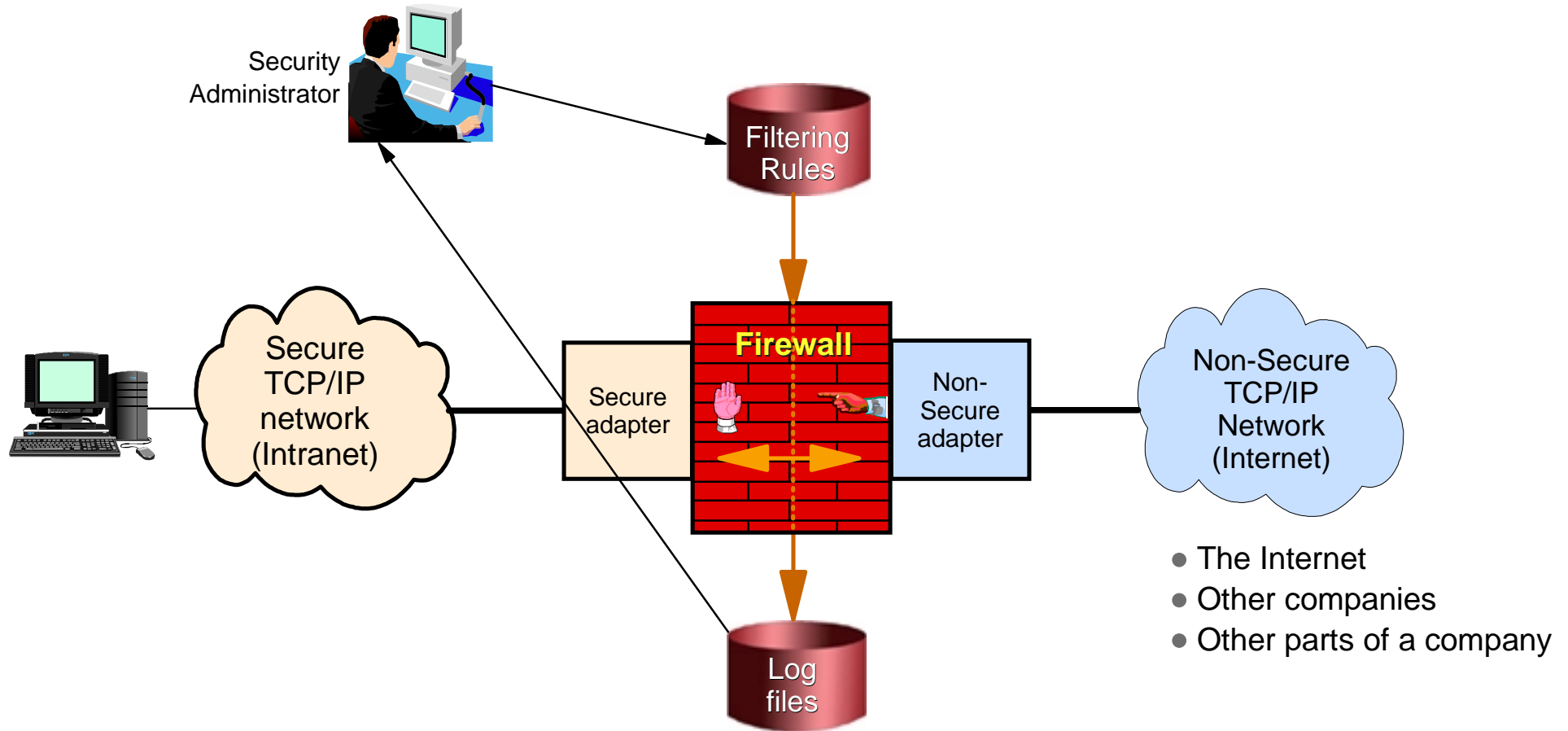
# Network Security



**Redbooks**

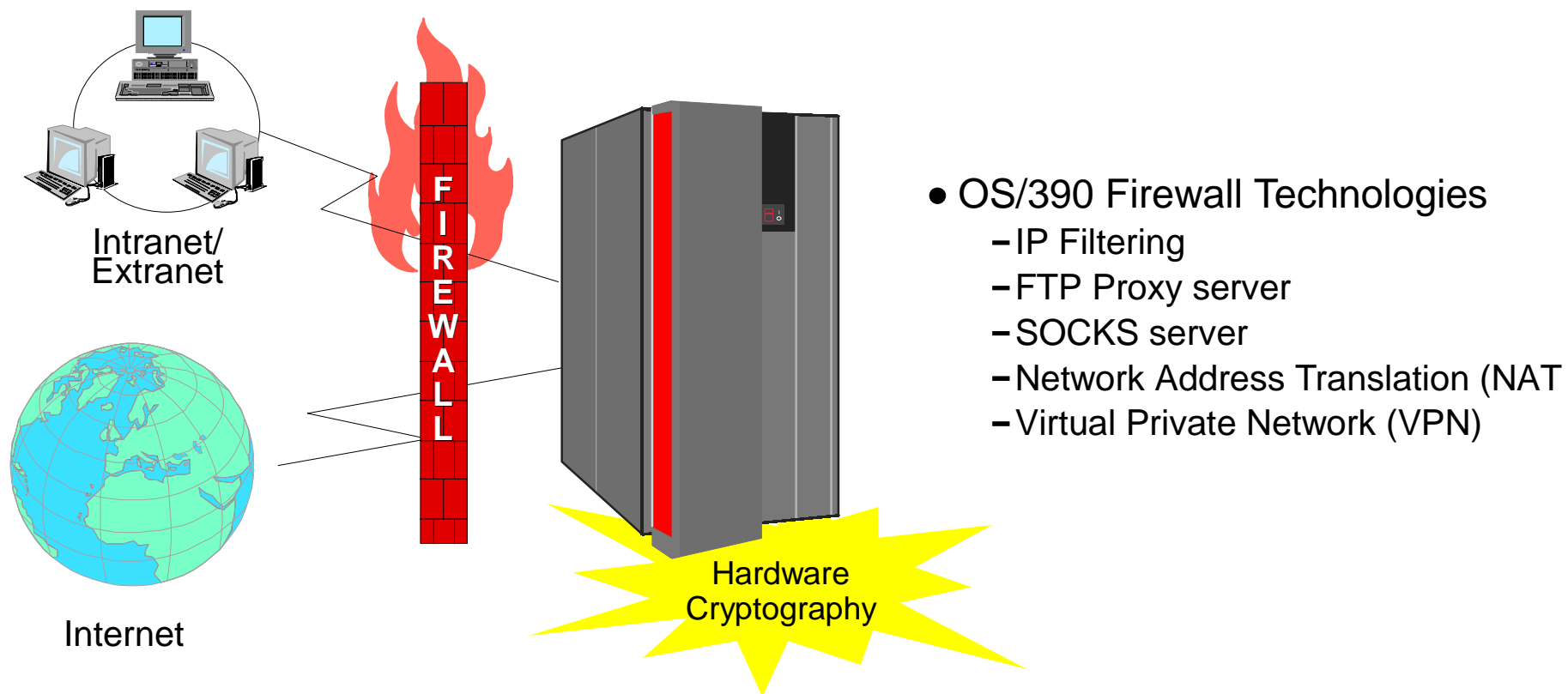
International Technical Support Organization

# What is a Firewall doing?



- Controls TCP/IP traffic in and out of secure network
- Isolates secure network from non-secure network
- Maintains log files on suspicious events

# Network Level Protection with OS/390



Firewall Technologies are delivered in OS/390 as part of Communication Server and Security Server, since OS/390 V2R5

# OS/390 Firewall Technology Deliverables...

In OS/390 since OS/390 V2R5

- In OS/390 Security Server (licensed products)
  - FTP proxy server
  - Socks V4 server
  - Command Line Configuration (no Security Server license required)
  - Configuration Server (GUI) (OS/390 R7)
  - IPSec VPN Key server (dynamic tunnels) (OS/390 R8)
- In OS/390 Communications Server
  - IP filters
  - Real Audio Support
  - IPSec VPNs (manual tunnels)
  - Network Address Translation (N.A.T.)
  - Enhanced Syslog Daemon

# OS/390 V2R10 Additional TCP/IP security

- TCP/IP service policies support Traffic Regulation Management
  - can limit amount of connections requested by the network (flooding attack)
- Control of network access by userid
  - destination network associated with a RACF resource
  - verify userid's permission to send data at TCPIP connect or UDP/TCP/RAW send
- Control of port access by userid
  - ports defined as RACF protected resources
  - verify userid's permission to access port at bind()
  - port can be locked out from any user
- Control of stack access by userid
  - new RACF resource for stack access
  - verify userid's permission to access stack during socket() call

# Ethical Hacking

- Started in OS/390 R4 with Firewall GA.
- Partnership with GSAL (Hawthorn Research).
- Incorporated into OS/390 process as of OS/390 R6.
- 2 people in z-series focused full time on Security/Integrity, CERTs.
- 30 people in IBM Research that focus on Ethical Hacking and Penetration Testing (Zurich and Watson).
- Communication Server and Firewall teams run research test cases.
- Research comes and Ethically Hacks z/OS
- 200 people in IGS trained to do Ethical Hacking for customers.



# Platform Security



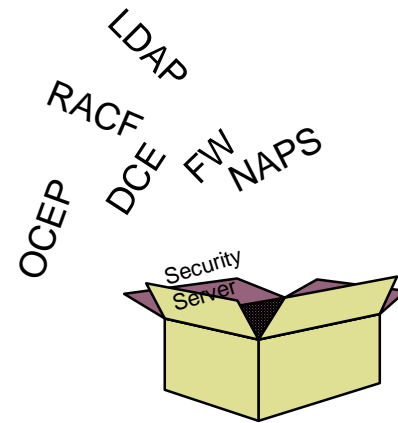
**Redbooks**

International Technical Support Organization

# SecureWay Security Server for OS/390

## OS/390's Integrated Set of Security Functions

- Optional feature of OS/390 V1 and V2
- Formerly named OS/390 Security Server
- Integrated package
  - RACF
  - Firewall Technology
  - LDAP Directory Server
  - DCE Security Server
  - OCEP (CDSA extensions)
  - Network Authentication Services (Kerberos, OS/390 2.10, 2.8)



# The OS/390 Security Architecture

- Identification and authentication of users and other accessors

- UserID and Password or Passticket
- Digital Certificate
- Kerberos ticket (OS/390 V2R10)



- Protect resources from unauthorized usage

- Exploitation of hardware security architecture
- Access checking and Authorization points imbedded within OS/390
- All accesses to all resources checked for user's authority

- Resources

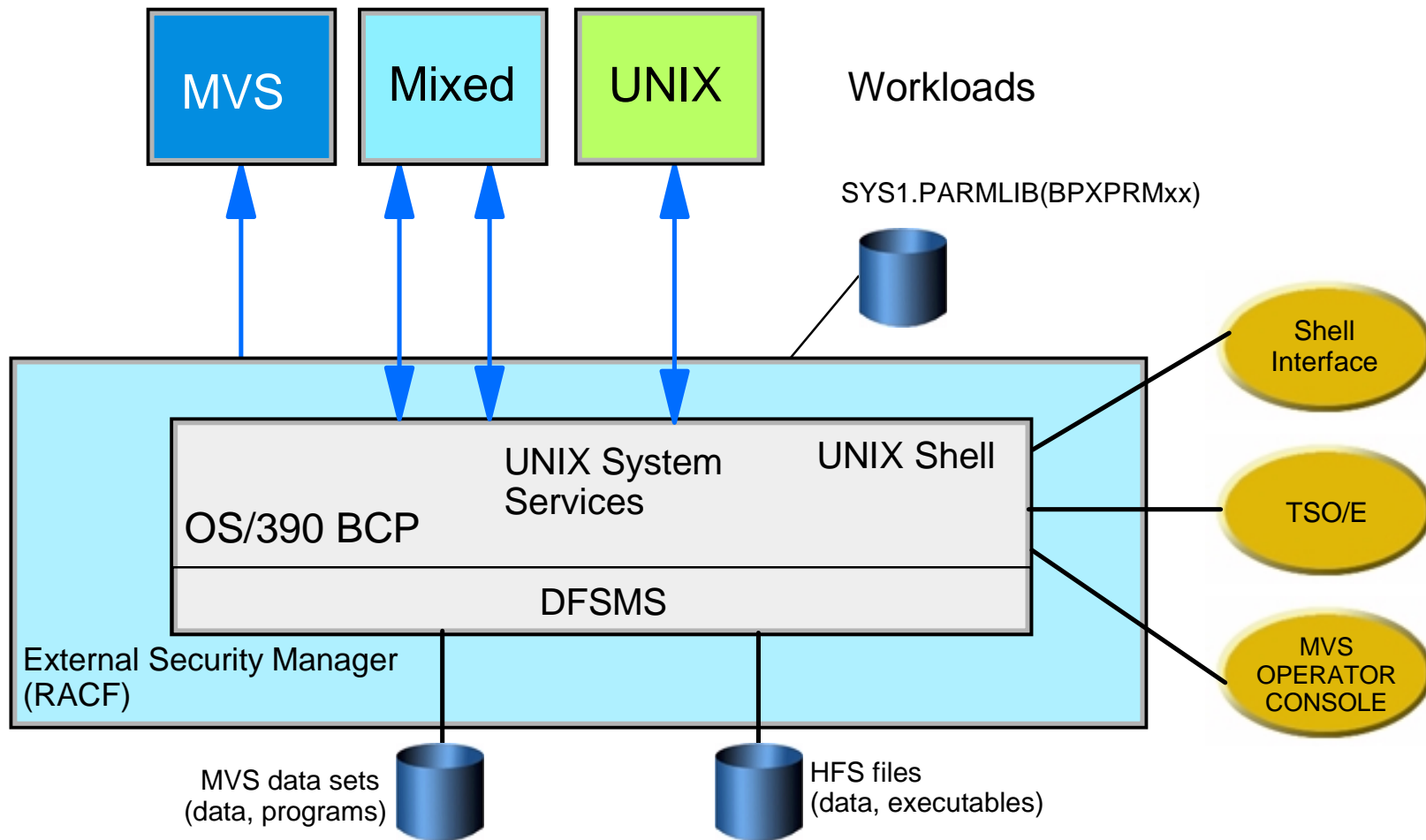
- business data, databases, transaction systems, programs, batch jobs, operator functions, user commands, networks, print facilities, UNIX ...



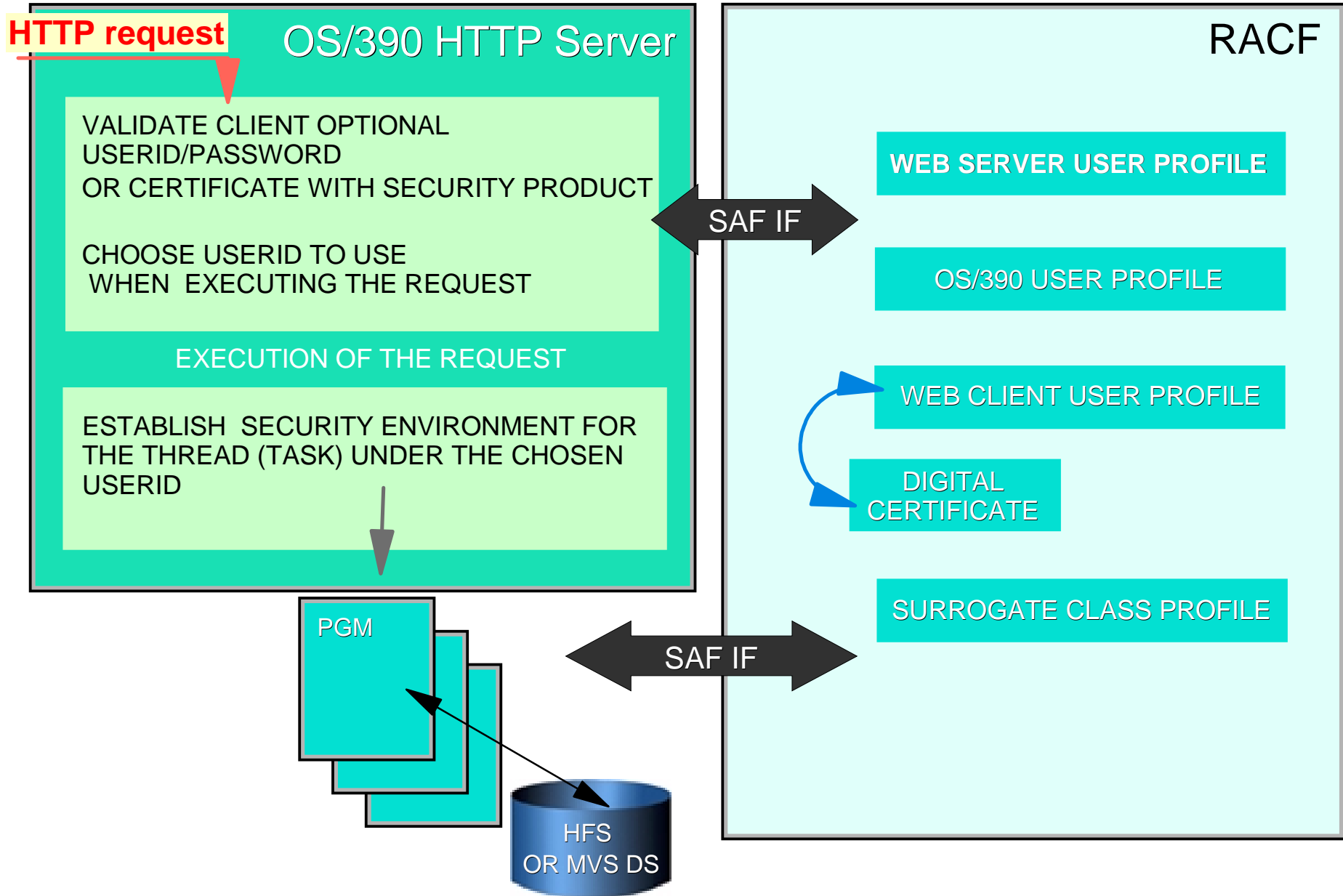
- Audit trail of security activity

Formal commitment to System Integrity since 1973

# OS/390 MVS and UNIX System Services



# OS/390 HTTP Server Protection Directives and SAF



# RACF New user attributes

'Protected User' (OS/390 R8)

## ADDUSER/ALTUSER *userid* NOPASSWORD

- For started procedures (and daemons)
- No logon, no SU, no revoked userid from Password guessing

## 'RESTRICTED' User

(OS/390 R8 with  
APAR OW40129  
and OW40130)

## ADDUSER/ALTUSER *userid* RESTRICTED

- The RESTRICTED attribute prevents a user from gaining access to a protected resource unless the user ID is specified on the access list.  
The following facilities do not apply for giving resource access to a RESTRICTED user
  - Global access checking
  - the ID(\*) entry on the access list
  - the UACC

A user can be both protected and restricted

# Platform Security with S/390 JAVA

<http://www1.s390.ibm.com/java>

**Java classes provided in Java for OS/390 JDK 1.1.6:**

**PlatformAccessControl  
PlatformThread  
PlatformSecurityServer  
PlatformAccessLevel  
PlatformReturned**

**These classes allow a Java application to interact with SAF to:**

**Check to see if the Security Server or a specific security  
server class is active  
Extract the userid in effect for the current running thread  
Check the userid in effect for access rights to a resource**

**New classes complying with the JAAS (Java Authentication and  
Authorization Services) Framework delivered with JDK 1.3**

# Transaction Security



**Redbooks**

International Technical Support Organization

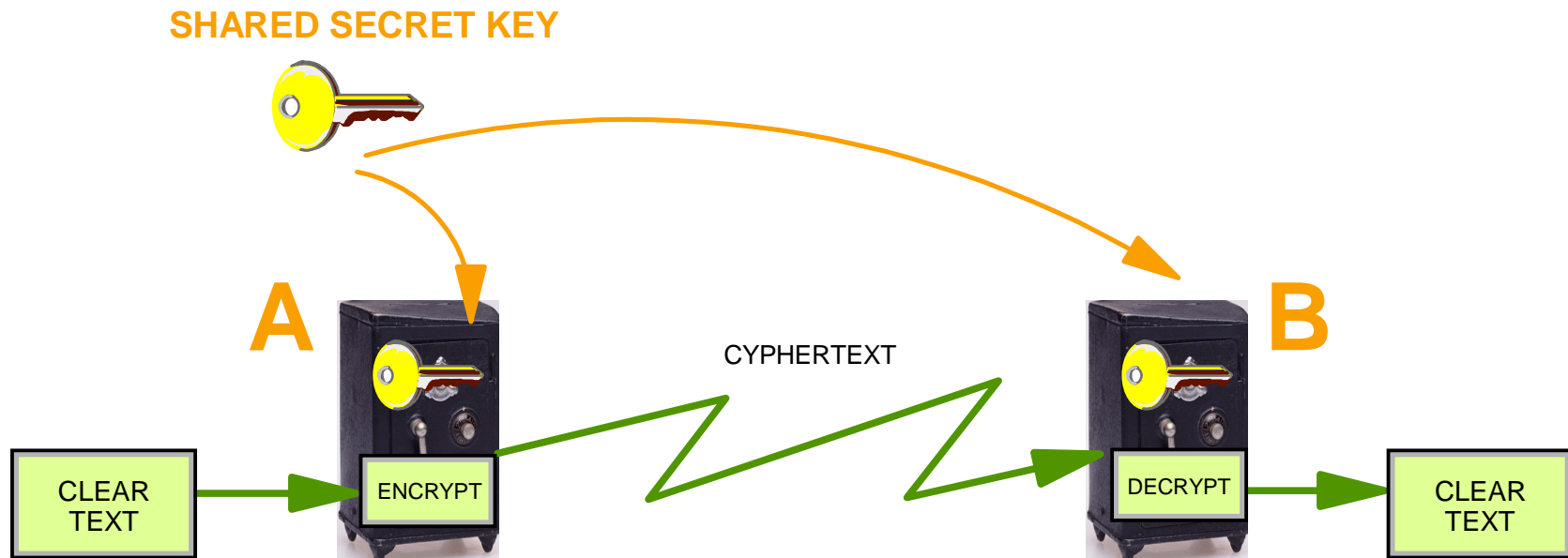


# Security at the Transaction Level

- Integrity
- Confidentiality
- Authentication
- Non-Repudiation

The Internet new security model is implemented via Public Key Cryptography and digital certificates

# Shared Secret Key (Symmetric Algorithms)



**DES :** 56-bit key

**Triple-DES :** 168-bit key

**CDMF :** 40-bit key

**RC2 :** 40-bit, 128-bit key

**RC4 :** 40-bit, 128-bit key

**AES (Rijndael):** up to 256-bit key

...

**Hardware  
Cryptography**

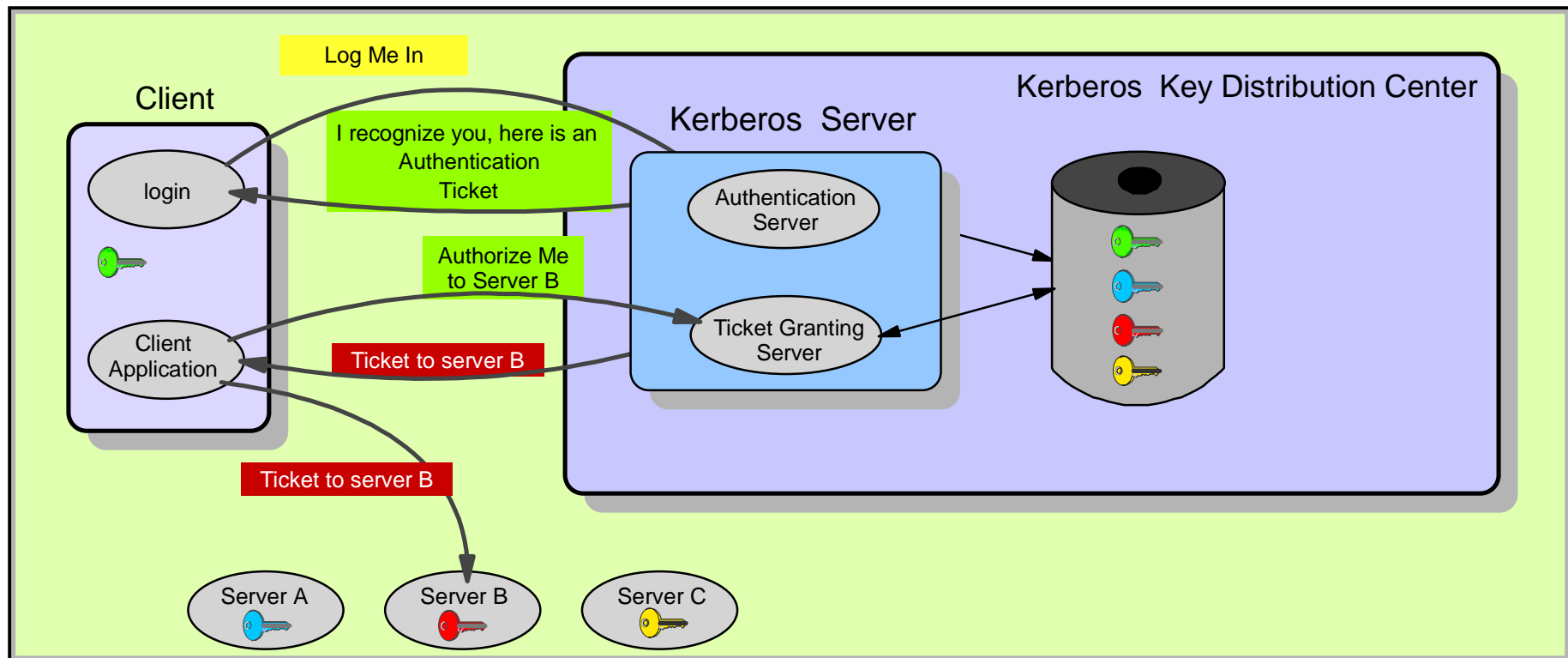
## Applicability to Internet :

- **Security issue : secrecy of shared key**
- **Key management issue : volume of secret keys to manage**

# What is Kerberos ?

- A distributed authentication service developed by MIT
- Currently at Version 5
- Allows user authentication over a physically untrusted network without transmitting password
- Tickets are issued by a Kerberos authentication server: both users and servers are required to have keys registered with the authentication server
- Flows to and from the authentication server establish a session key, used in a direct exchange between a user and service
- Provides optionally data privacy

# Kerberos Authentication Overview



Kerberos KDC Security Realm

- Uses symmetric algorithm (DES), for authentication and data privacy
- **no password in clear on the network**
- A KDC keeps a copy of DES keys for all entities in the KDC 'Realm'
- Transitive trust can be established between realms
- Used by several OS (e.g. AIX, OS/400, WIN2K, ...) for network users authentication, including OS/390 2.10, 2.8

# Kerberos Authentication Overview

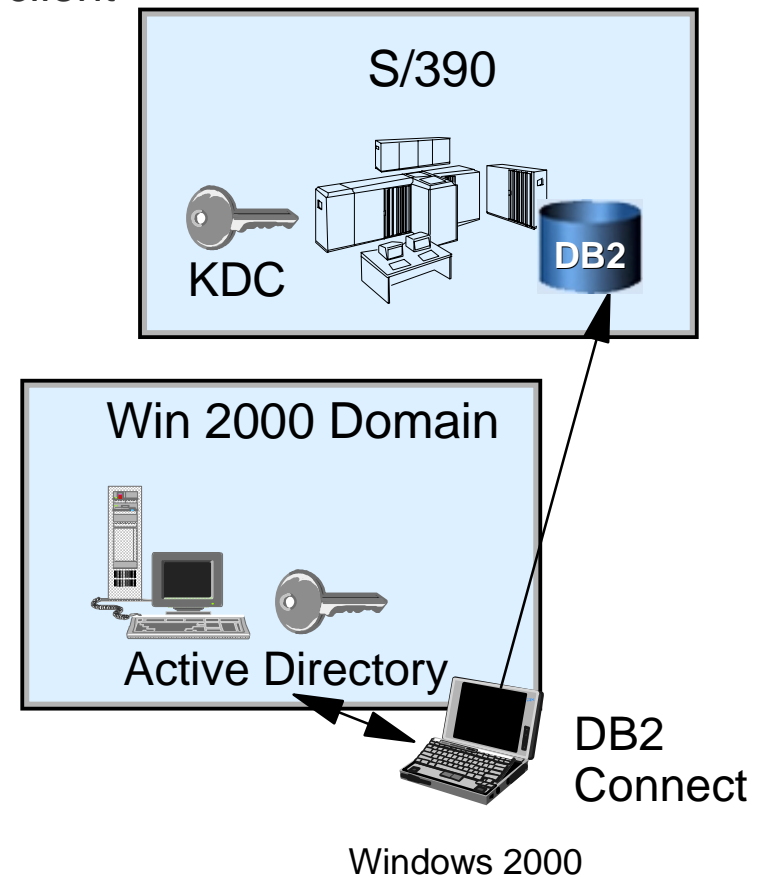
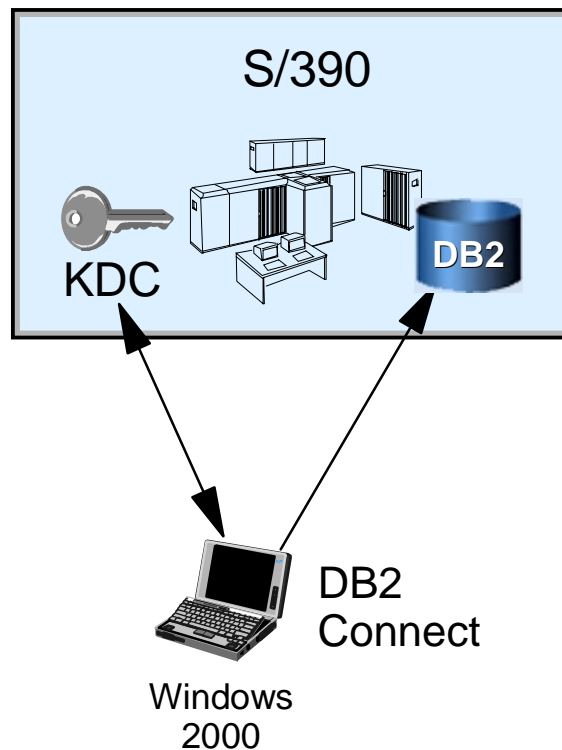
- Application (client and server) APIs
  - GSS (Generic Security Services) API
    - Enable an application to determine other application's user identification
    - Enable an application to delegate access rights to another application
    - Apply security services, such as confidentiality, integrity, on a per-message basis.
    - GSS\_
  - Kerberos API
    - krb5\_  
kadm5\_
  - Microsoft Windows 2000 SSPI
    - ISC\_

# Kerberos Authentication Overview

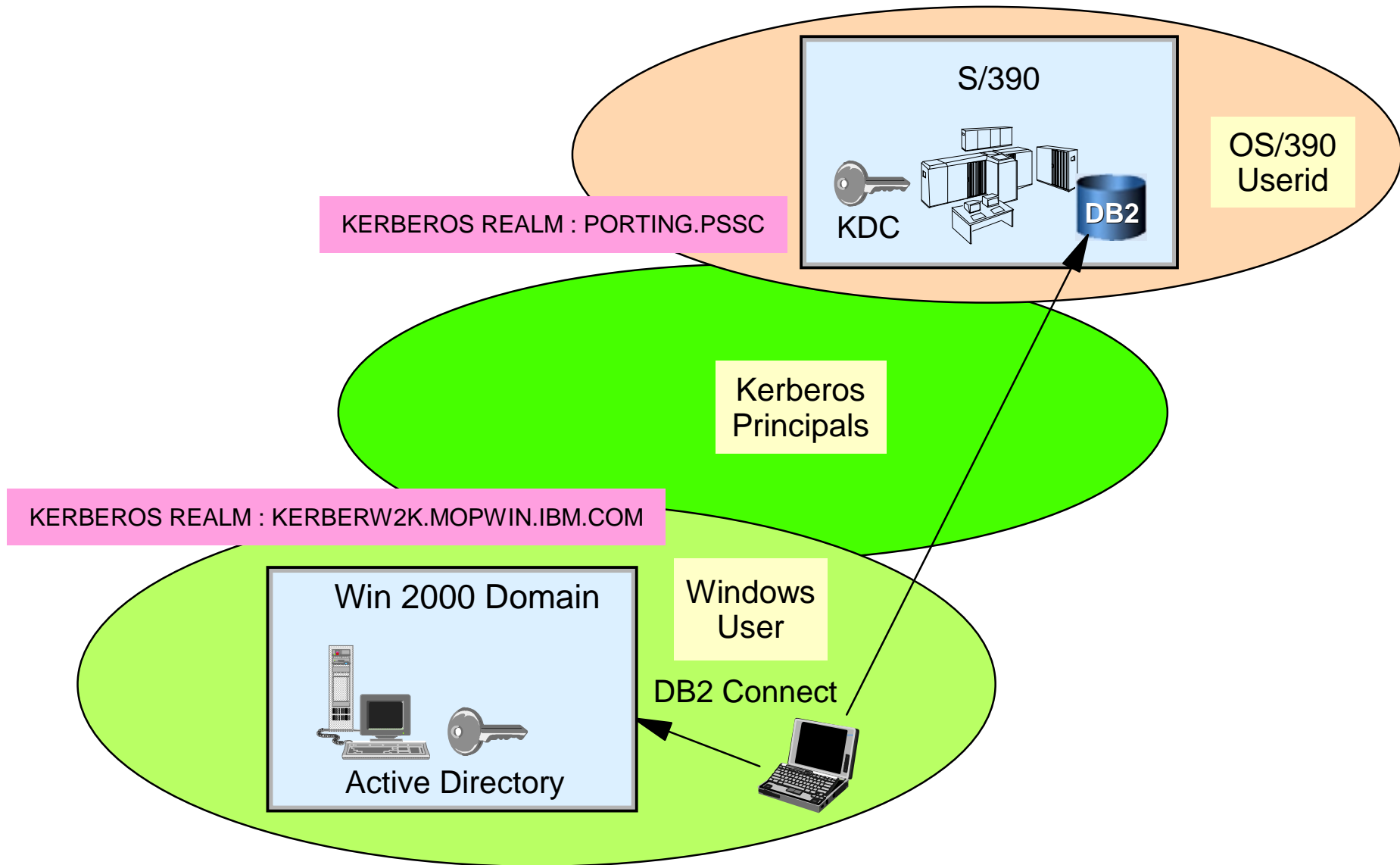
- **kinit**  
asks for a user password and obtains a Ticket Granting Ticket
- **klist**  
displays the list of tickets obtained so far and kept in a credentials cache
- **kdestroy** destroys all obtained tickets kept in a credentials cache
- **keytab** allows to manage a local key table
- **ksetup**  
manages service entries in the LDAP directory for a Kerberos realm
- **kadmin**  
miscellaneous administration of the Kerberos principals  
(via the Kerberos Administration Server)
- **kpasswd**  
to change principal's password (via the Kerberos Password Server)
- **kvno** query key version number

# Network Authentication Services on OS/390 (Kerberos)

- Kerberos uses a symmetric algorithm (DES) to identify and authenticate network entities
  - supported by many platforms, including Windows 2000
  - scalability constrained to corporate networks
- The OS/390 Network Authentication Services (OS/390 2.10, 2.8) provide Kerberos authentication - E.g. a Windows 2000 client can authenticate to OS/390 DB2 V7 server

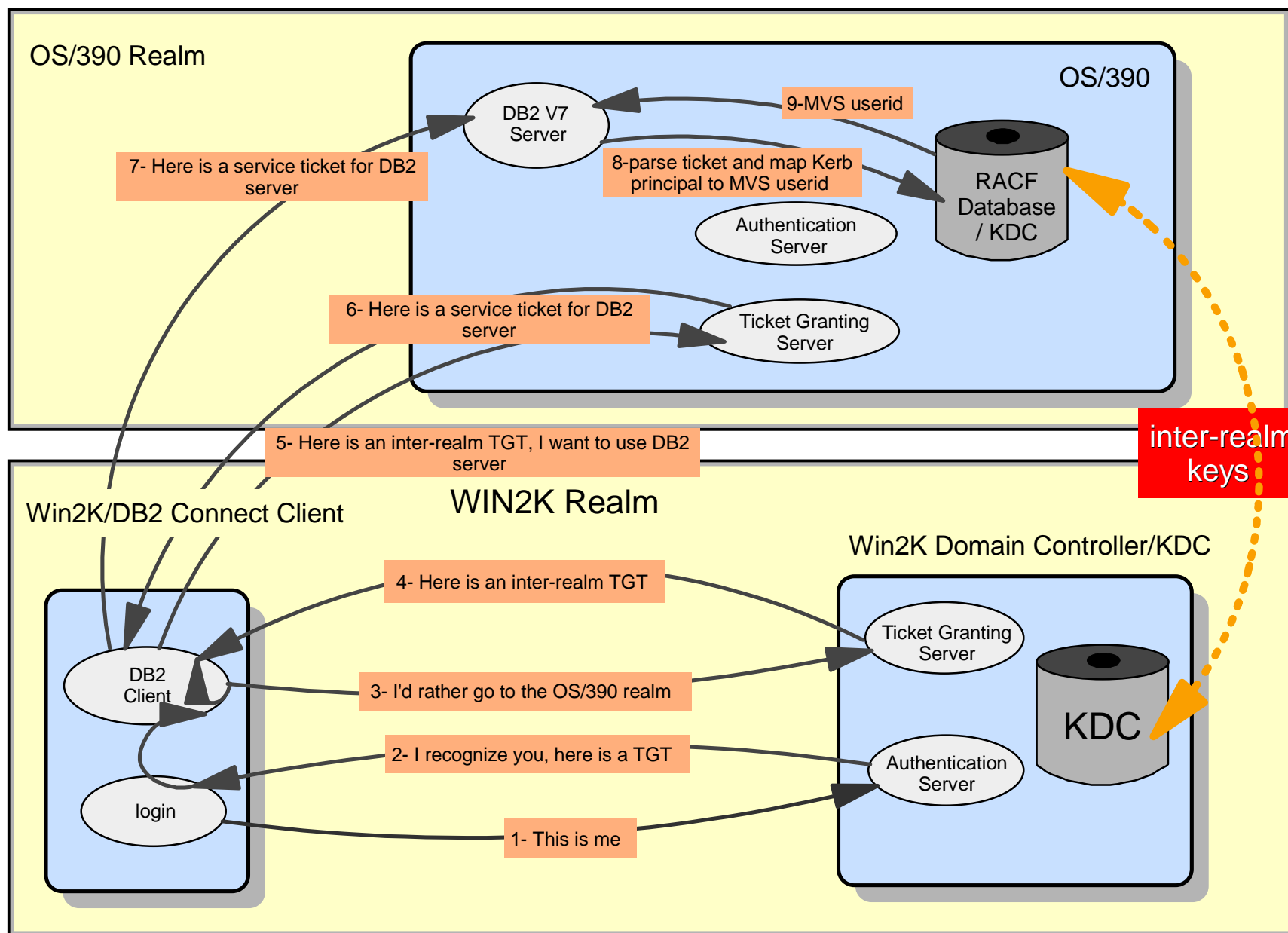


# OS/390 Network Authentication Service - Example

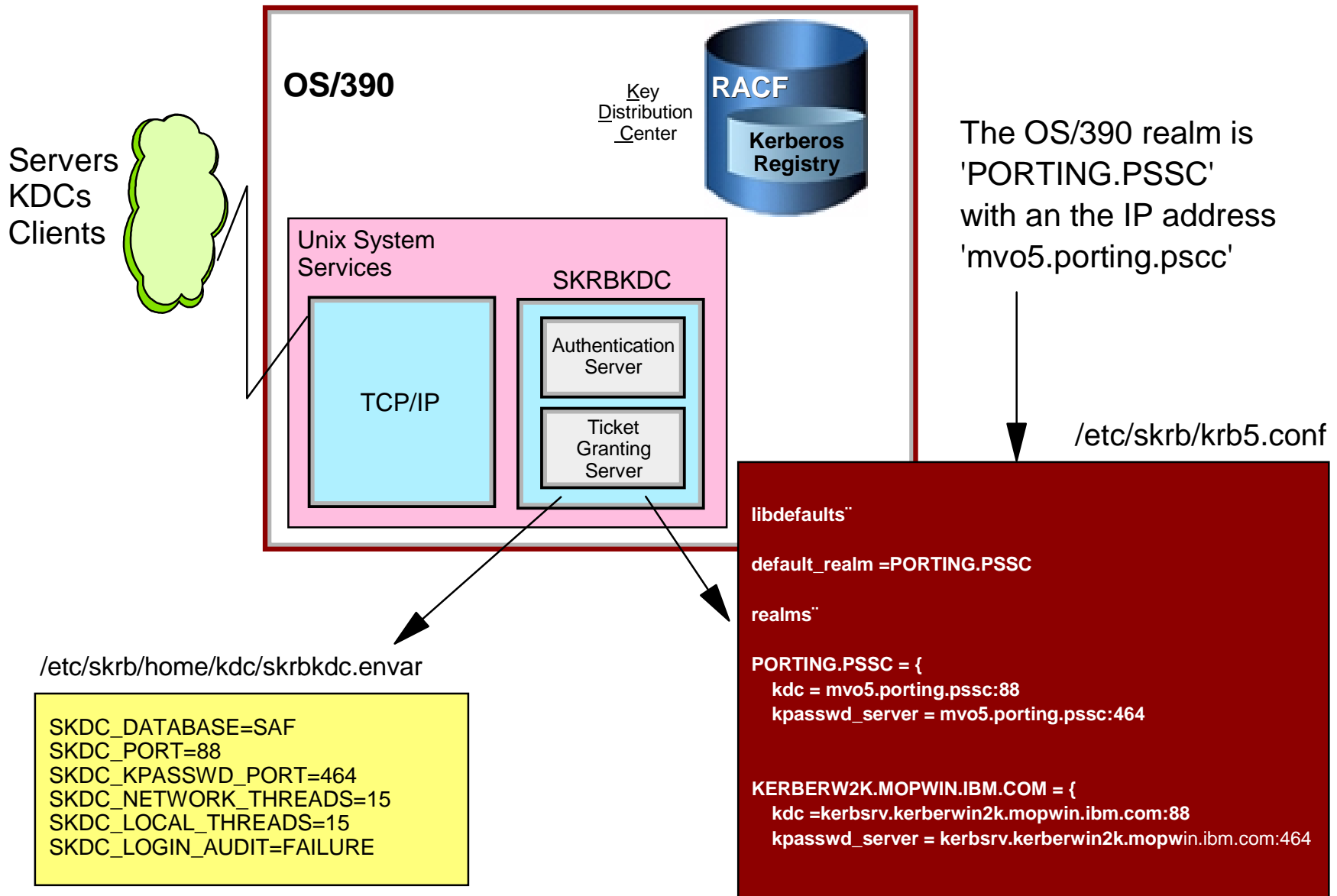




# OS/390 Network Authentication Services - Inter-realms



# OS/390 Kerberos Server Setup



# OS/390 Network Authentication Services - RACF Setup

- RACF must be setup as a local RRSF node
- Definition of RACF profiles
  - definition of the local Kerberos realm & foreign realms
    - **REALM class**
  - local Kerberos principals (users)
    - **KERB segment in user profiles**
    - **KERBLINK class profiles**
  - definition of foreign Kerberos principals with a local identity
    - **KERBLINK class profiles**

# OS/390 Network Authentication Services - Administration

## RACF Setup - Steps for getting started

- Define local realm

- RDEFINE REALM KERBDFLT KERB(KERBNAME(realm\_name)  
PASSWORD(realm\_password))

- Define inter-realm relationship

- RDEFINE REALM /.../local\_realm/krbtgt/foreign\_realm  
KERB(PASSWORD(inter\_realm\_password1))

- RDEFINE REALM /.../foreign\_realm/krbtgt/local\_realm  
KERB(PASSWORD(inter\_realm\_password2))

- Define local principals

- ALTUSER local\_useruser KERB(KERBNAME(principal\_name)) PASSWORD(password)  
NOEXPIRED

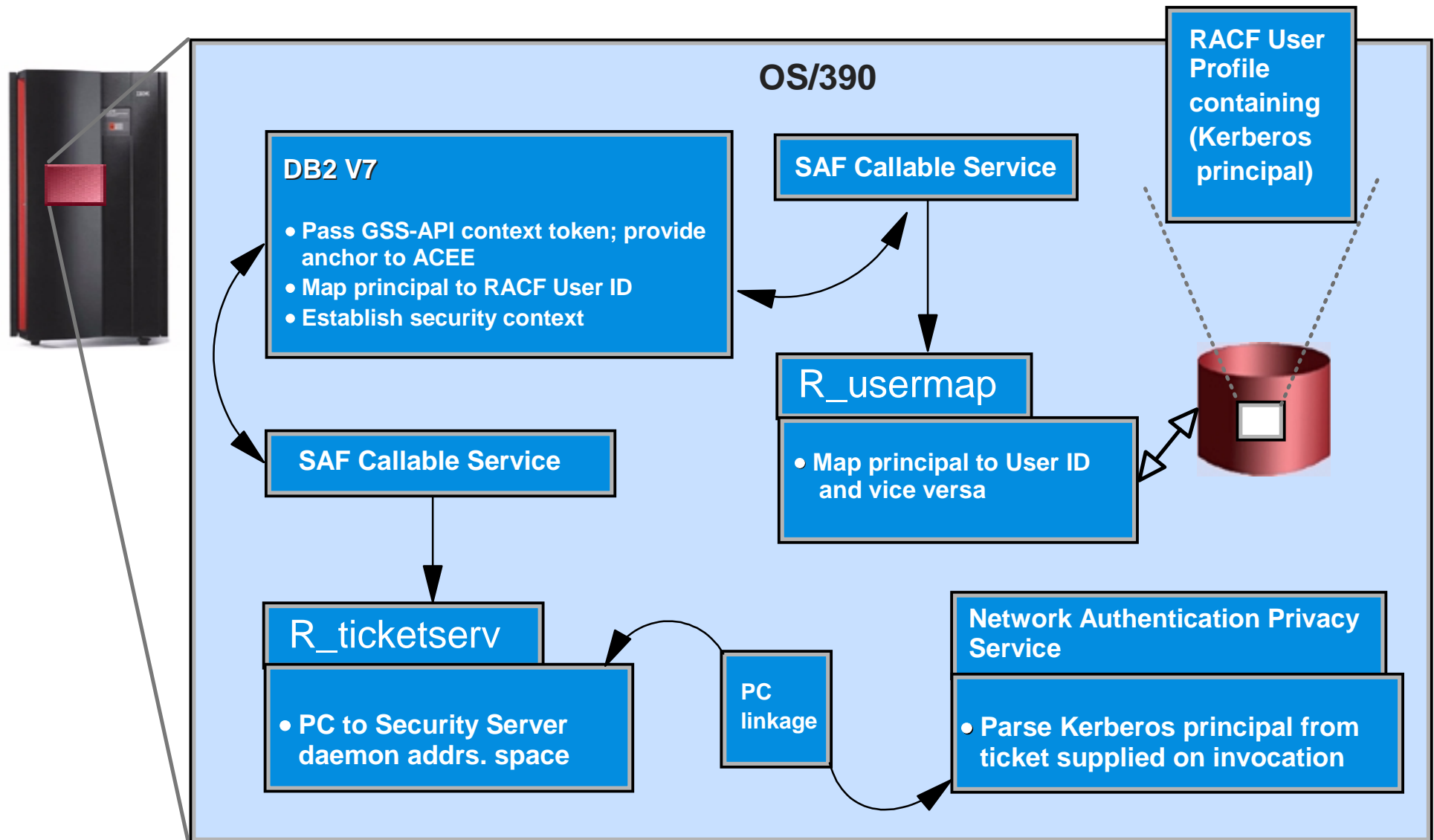
- Define foreign principals

- RDEFINE KERBLINK /.../foreign\_realm/foreign\_principal APPLDATA(local\_userid)
    - maps single principal to a RACF user

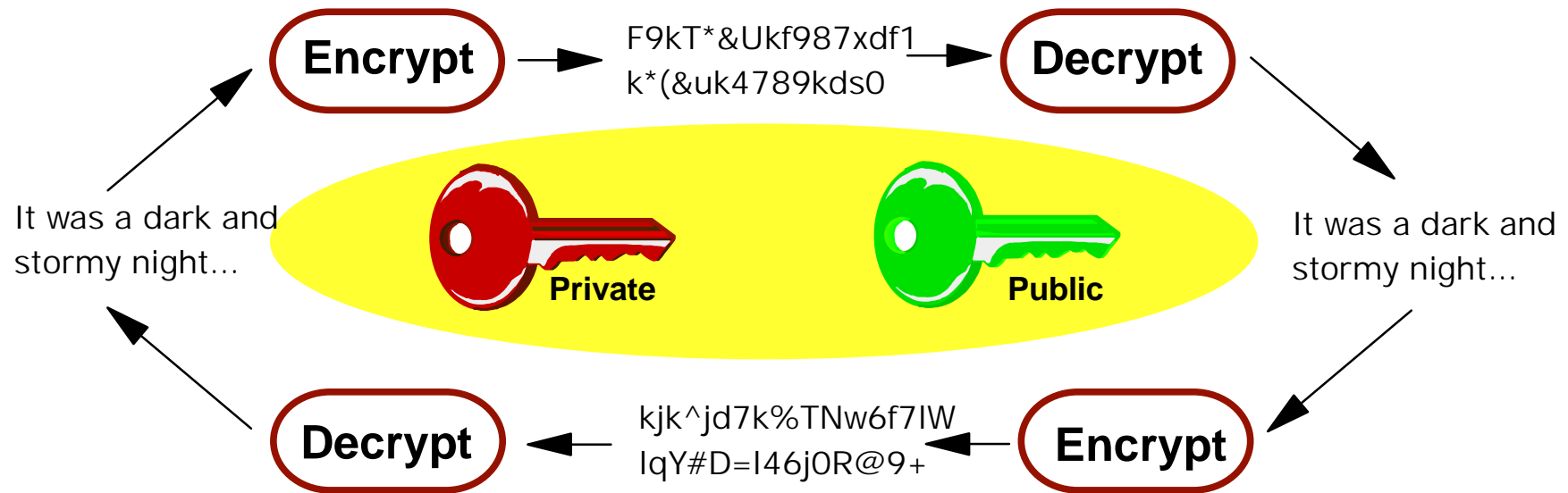
- RDEFINE KERBLINK /.../foreign\_realm/ APPLDATAlocal\_userid)

- Maps all principals for a single realm to a RACF userid

# OS/390 Network Authentication Services - Authentication



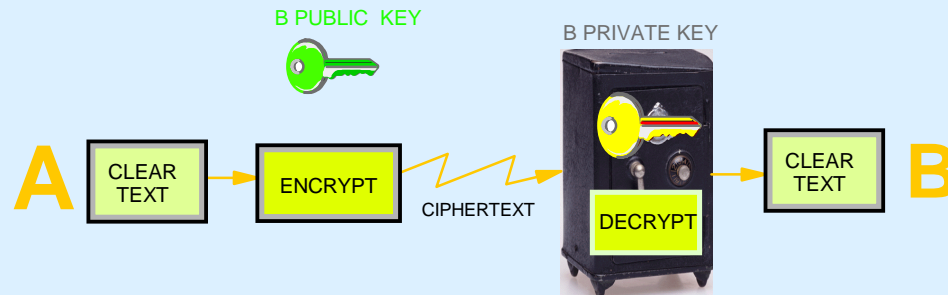
# Public Key Cryptography (Asymmetric Algorithms) 1/2



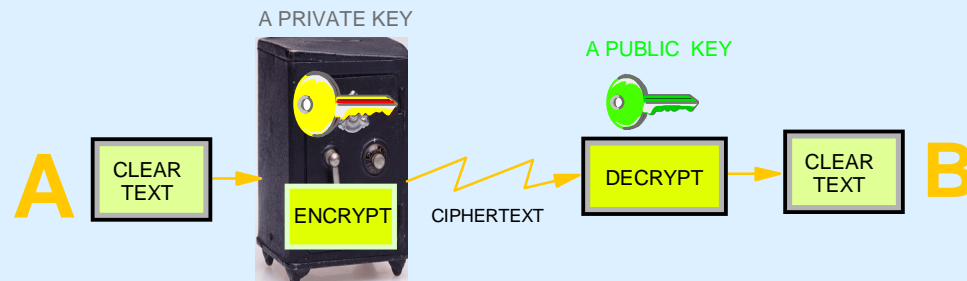
- A pair of keys is needed. The two keys are generated together
- One key of the pair becomes public knowledge ("public key")  
As many copies as needed
- One key of the pair is kept secret ("private key")  
Only one single instance of the private key  
This is the only secret to manage.

# Public Key Cryptography (Asymmetric Algorithms) 2/2

## Data Confidentiality



## Authentication (also used for digital signature)



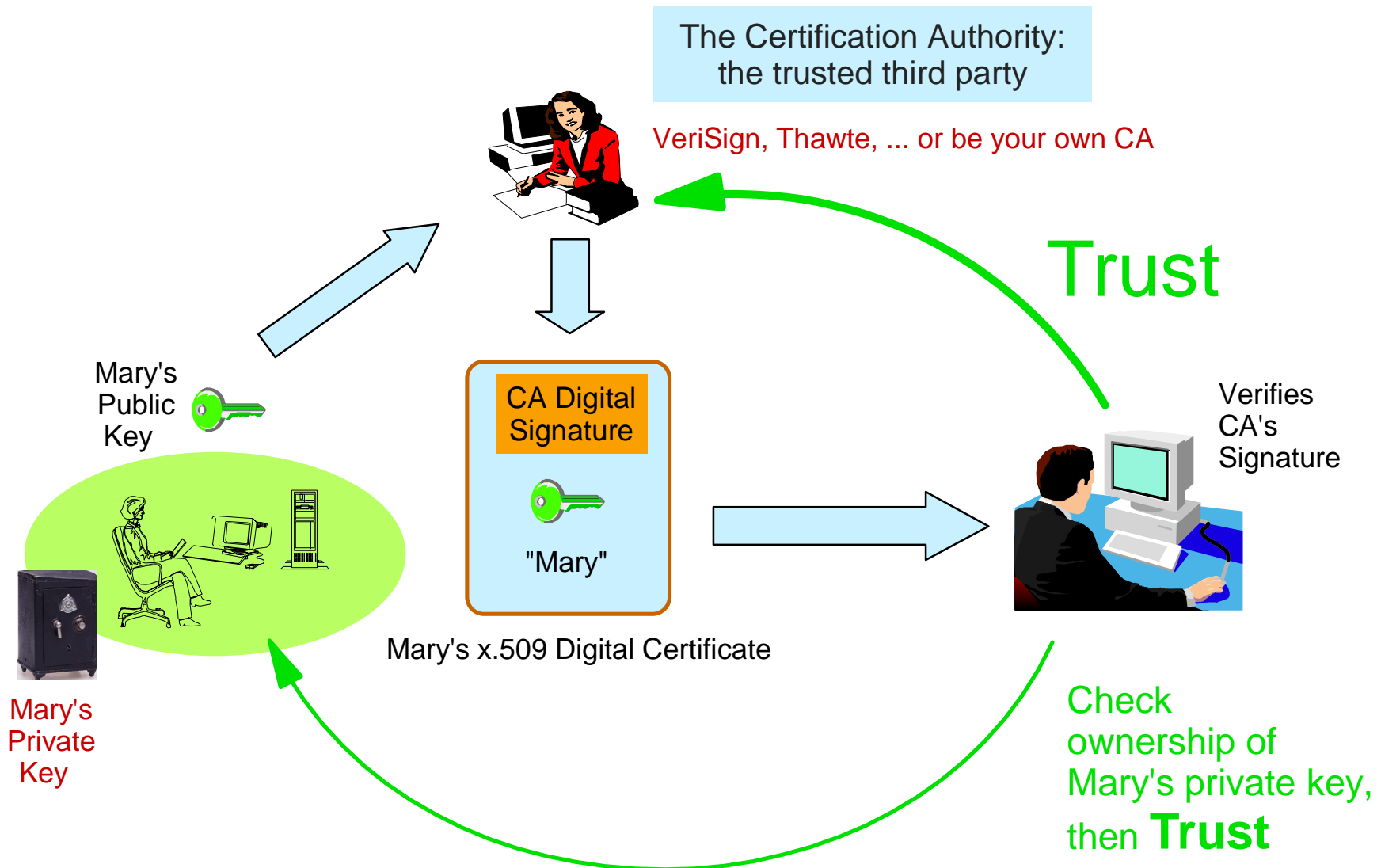
RSA : 512-bit to 2048-bit key



### Applicability to Internet :

- Solves the key management problem
- Performance issue : very computing intensive
- Security issue : certification of the public key

# The Internet Trust Model



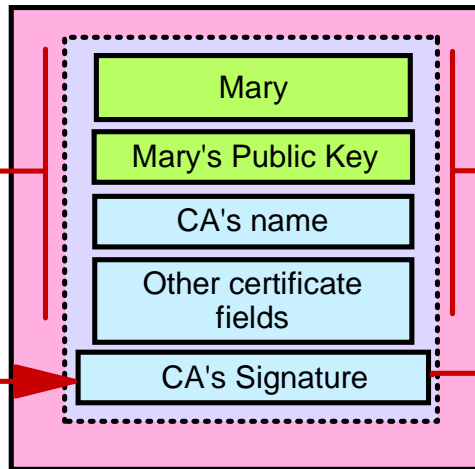


# Signature and signature verification of a Digital Certificate

Certifying Authority (CA)  
signs certificate



X.509 Digital Certificate



Most used  
'hash'  
algorithms:  
MD5: 128  
bits  
SHA-1: 160  
bits

hash value

Encrypt



CA's Private Key

Most used  
encryption  
algorithm: RSA  
at 1024-bit key

Decrypt



CA's Public Key

Certificate Exploiter  
verifies certificate



hash value

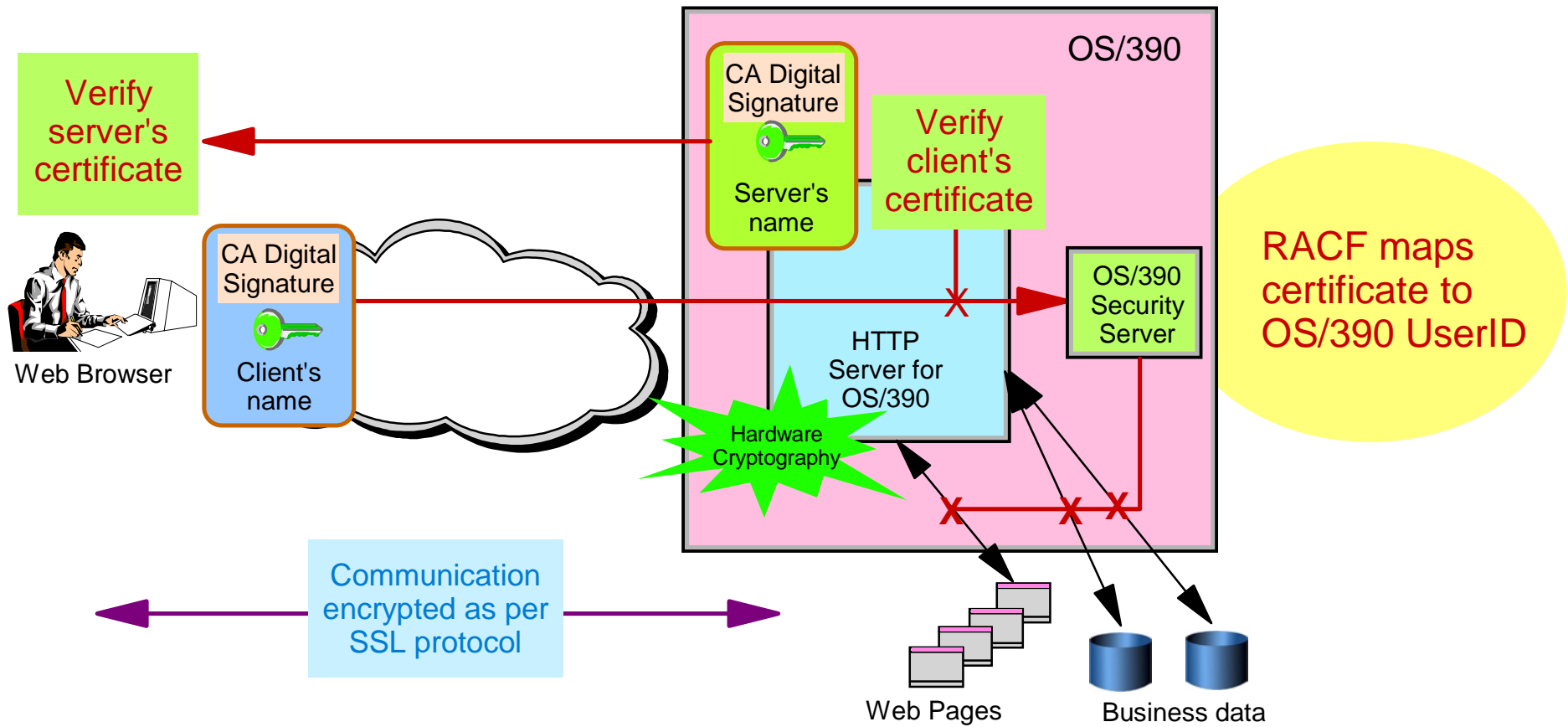
Equal ?

Hardware  
Cryptography

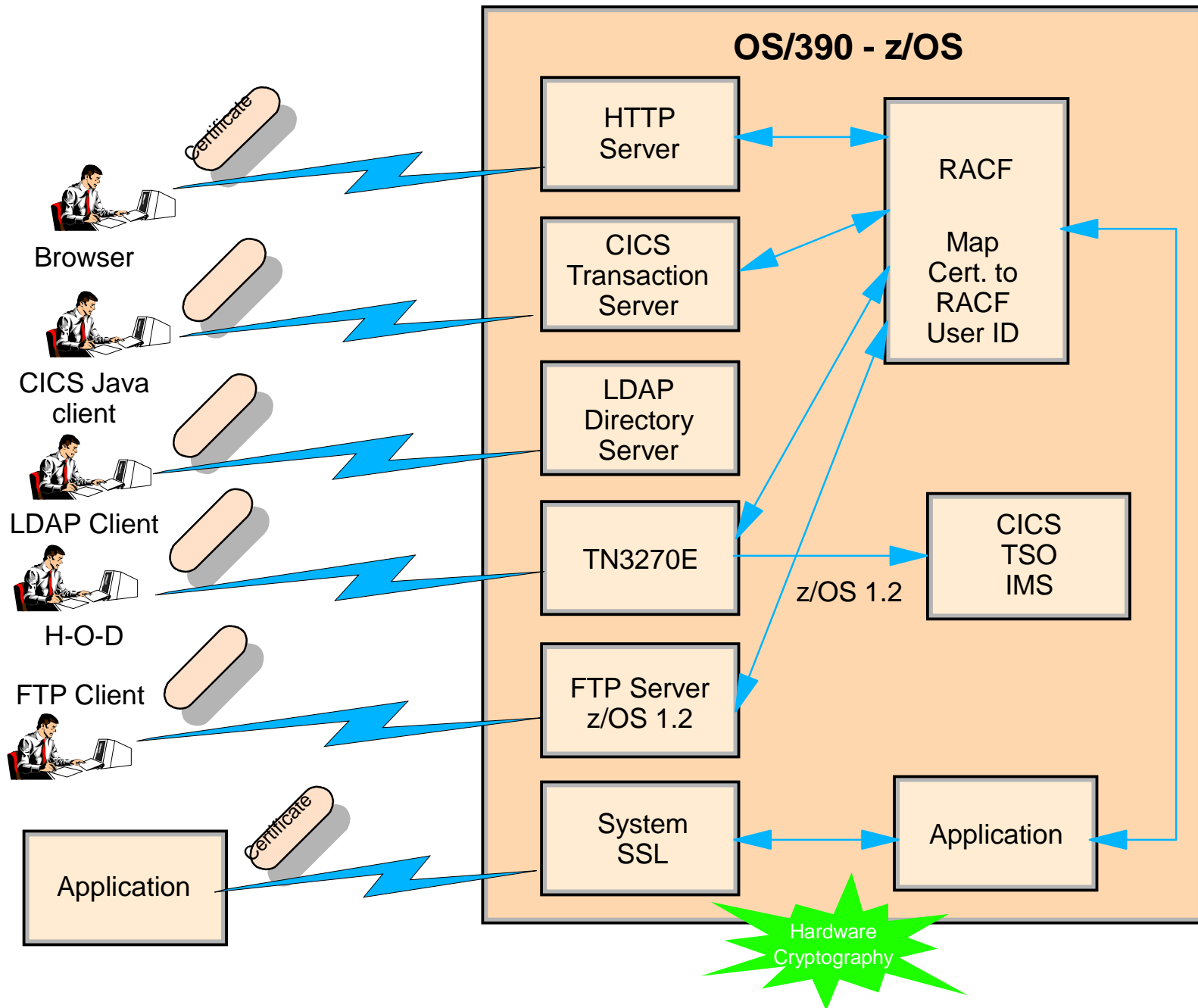
# What is a digital certificate?

- An electronic form of 'strong' identification
  - The CA's signature binds the subject's name to the subject's public key
  - further communication implies ownership of the private key
- Most common use is probably the SSL (Secure Socket Layer) protocol with X.509 certificates
- Also in use by other protocols, such as SET (Secure Electronic Transaction)

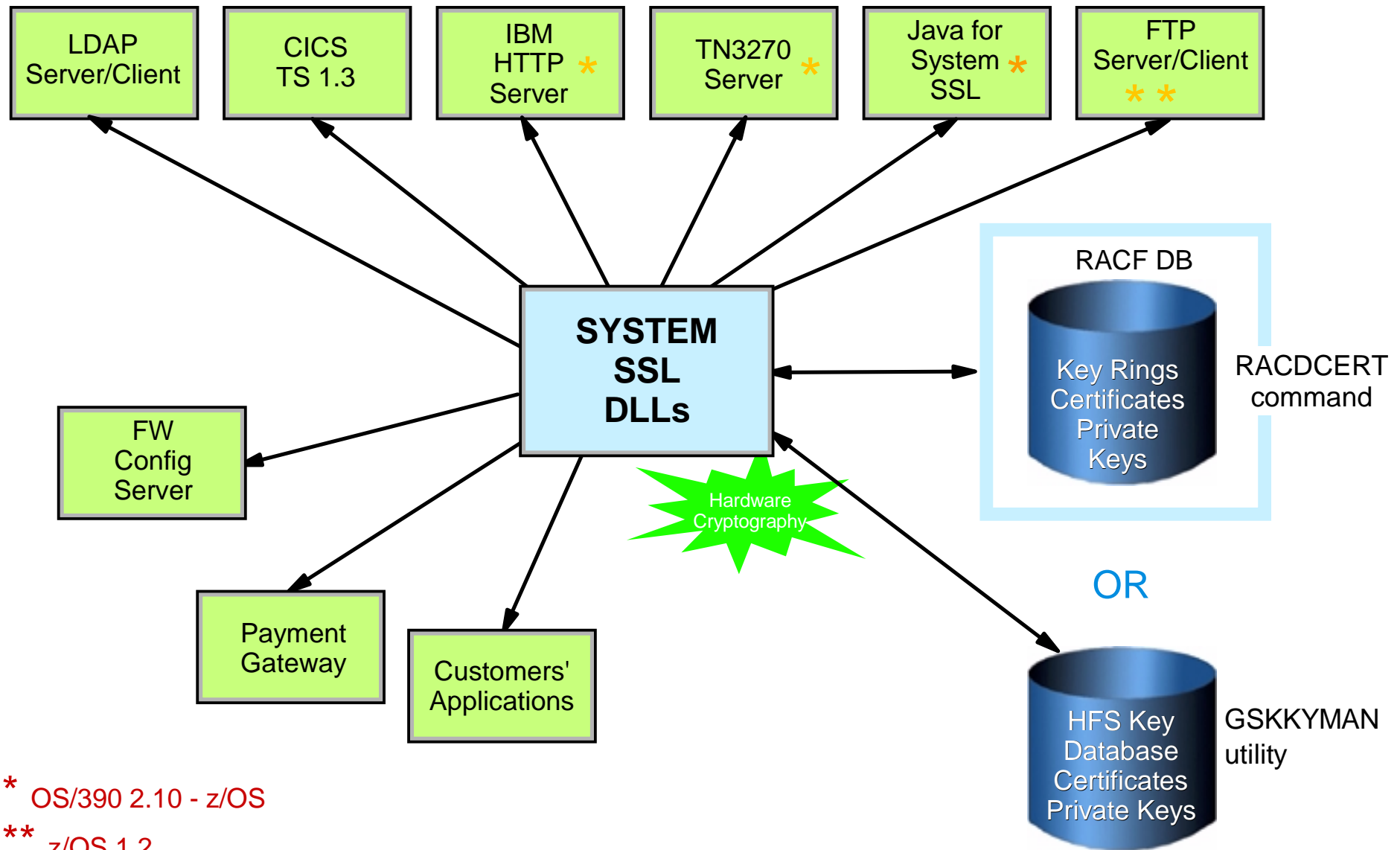
# SSL in OS/390



# SSL and Certificates on OS/390...



# OS/390 'System SSL' DLLs



\* OS/390 2.10 - z/OS

\*\* z/OS 1.2

# OS/390 RACF keys and certificates support

- RACF generation and management of RSA key pairs and certificates (OS/390 2.8)
  - New RACDCERT command functions that allow :
    - The generation of RSA key pair, X.509 certificates and certificate requests
    - The aggregation of certificates into key rings in the RACF DB
    - The importation of PKCS-12 certificates into the RACF DB
    - optionally: storing of the private key in the cryptographic key data set

● **RACDCERT Example #1- Create a public/private key pair and a certificate for the user ID (SRVR01) which is the user ID associated with the inventory server:**

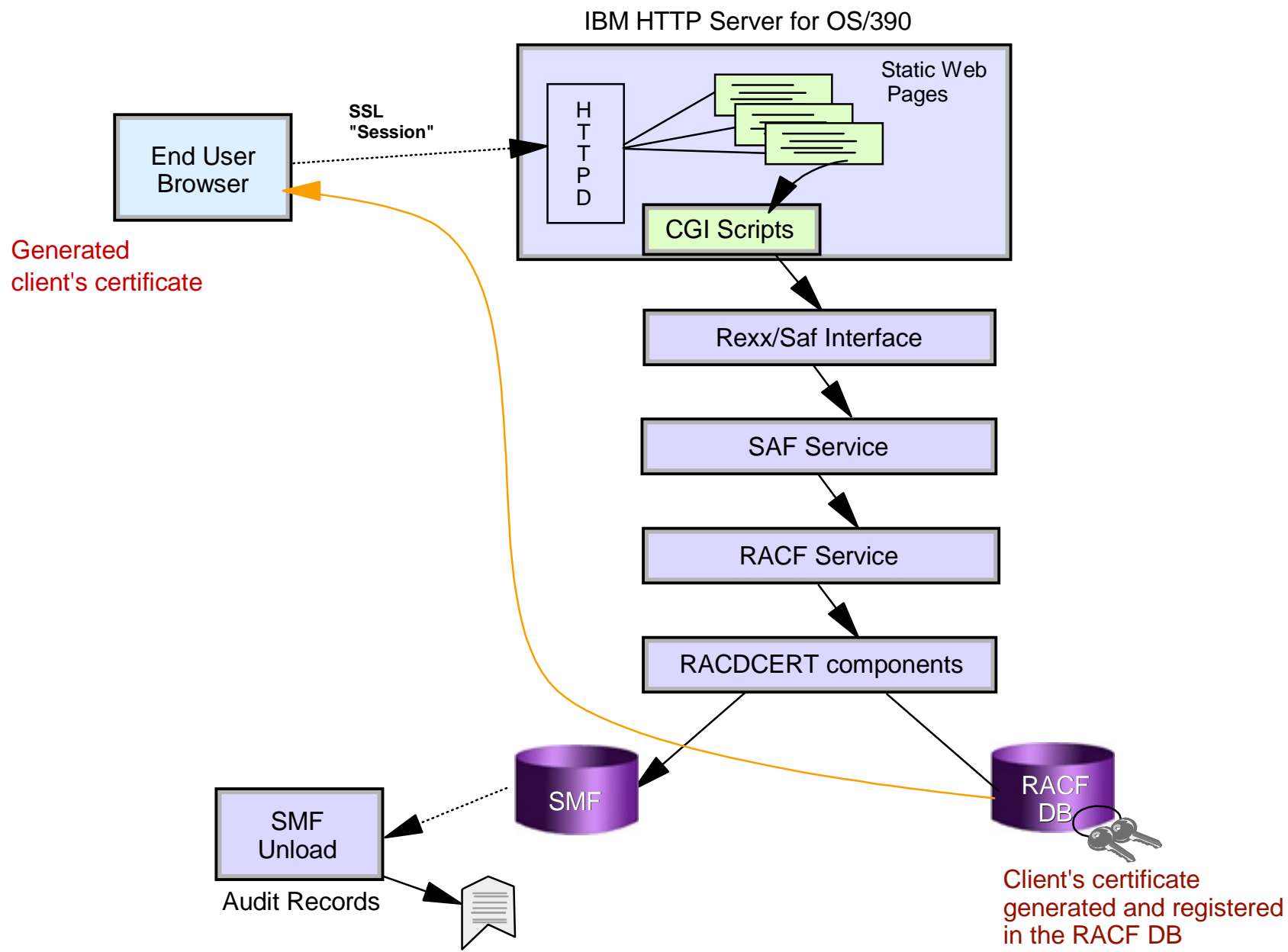
```
RACDCERT ID(SRVR01)
  GENCERT
  SUBJECTSDN(CN('co-name.com')
    OU('Inventory')
    C('US'))
  WITHLABEL('Inventory Server')
```

- The certificates are kept in the DIGTCERT class profiles

# OS/390 RACF keys and certificates support

- OS/390 Server keys and certificates
  - can be imported into the RACF database
  - or can be
    - generated by RACF RACDCERT
    - signed by outside CA or in RACF with a local CA private key
- OS/390 Clients' certificates
  - can be imported into the RACF database to be mapped to an MVS userid
  - or can be
    - generated by RACF RACDCERT
    - signed by outside CA or in RACF with a local CA private key
  - RACF provides a limited CA capability with the PKIServ function via web access protocol (OS/390 2.10)
  - based on RADCERT certificate generation capability

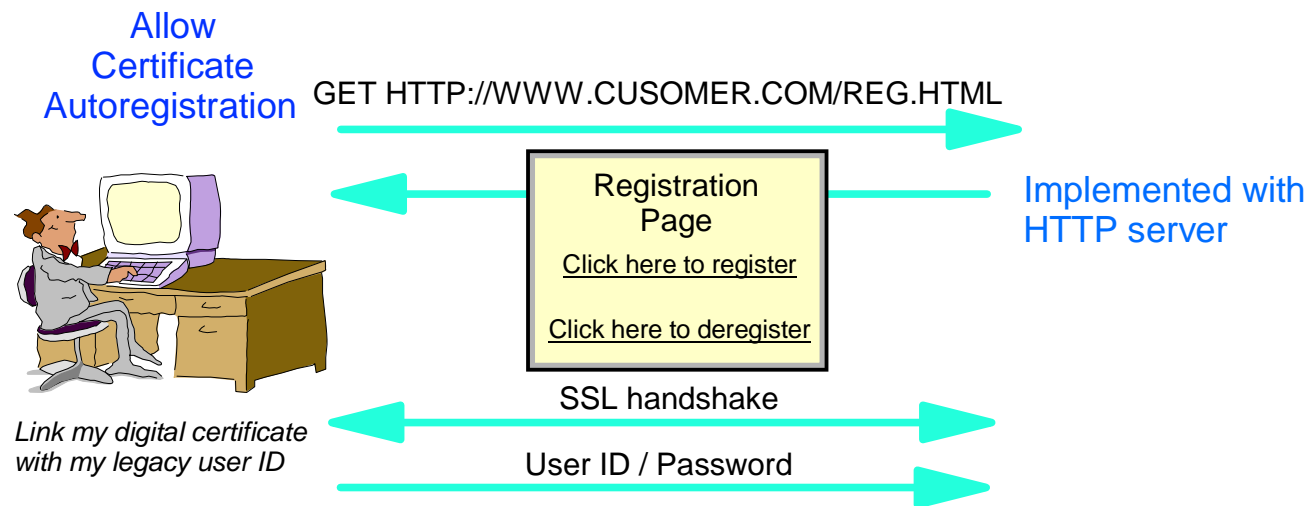
# PKIServ at OS/390 2.10





# Mapping of Clients' Digital Certificates to RACF userids

- Storing of **individual clients' certificates images** in the RACF DB, by the RACF administrator.  
Compare to certificate passed by the server (OS/390 2.4)
- **Auto registration** of individual clients' certificates at OS/390 2.5, via the http server



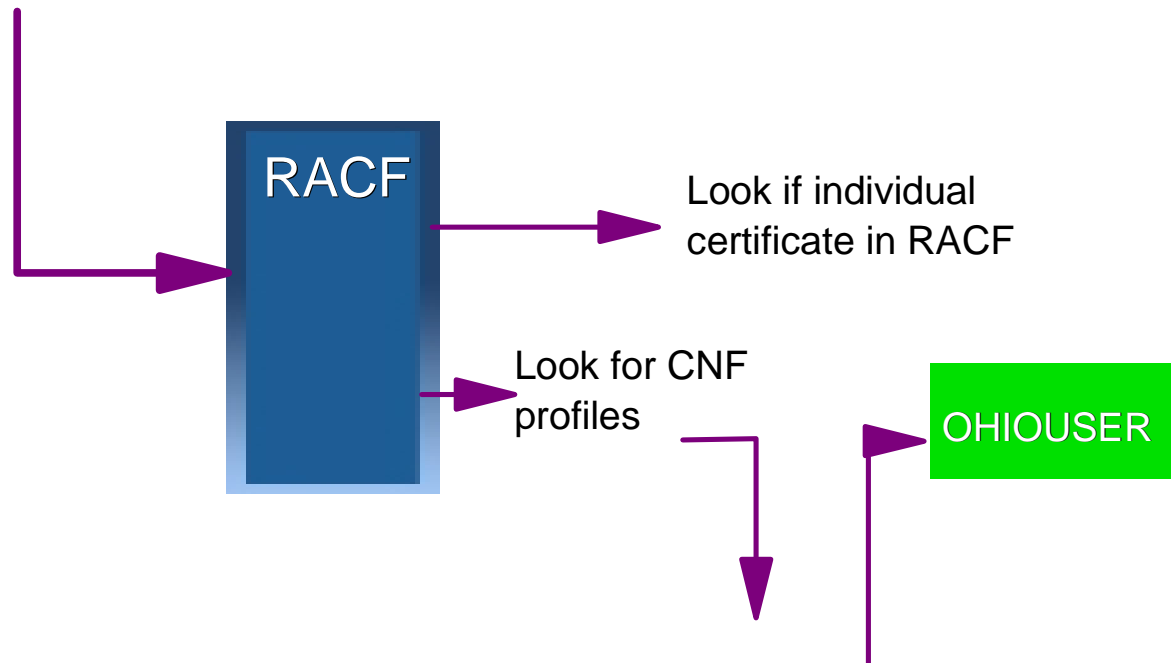
- **Certificate Name Filtering** at OS/390 2.9 (2.8 + APAR 40129) to select a RACF user ID based on clients' certificate's fields contents (no clients' certificates in RACF DB)

# RACF Certificate Name Filtering

- Without CNF
  - Every client's certificate must be installed into the RACF DB
- Certificate Name Filtering (OS/390 2.8 + APAR OW40129)
  - Filtering rules for mapping of certificates to userIDs in the DIGTNMAP and DIGTCRIT class profiles
  - Clients' certificates are not stored in the RACF DB
    - can map many certificates to one generic userID
    - minimizes RACF administration burden
  - Accountability is maintained
  - Access by generic userIDs can be restricted ('Restricted' user attribute)
- RACF still looks for certificates in the RACF DB, then for filtering rules

# Examples of client's certificates and filters

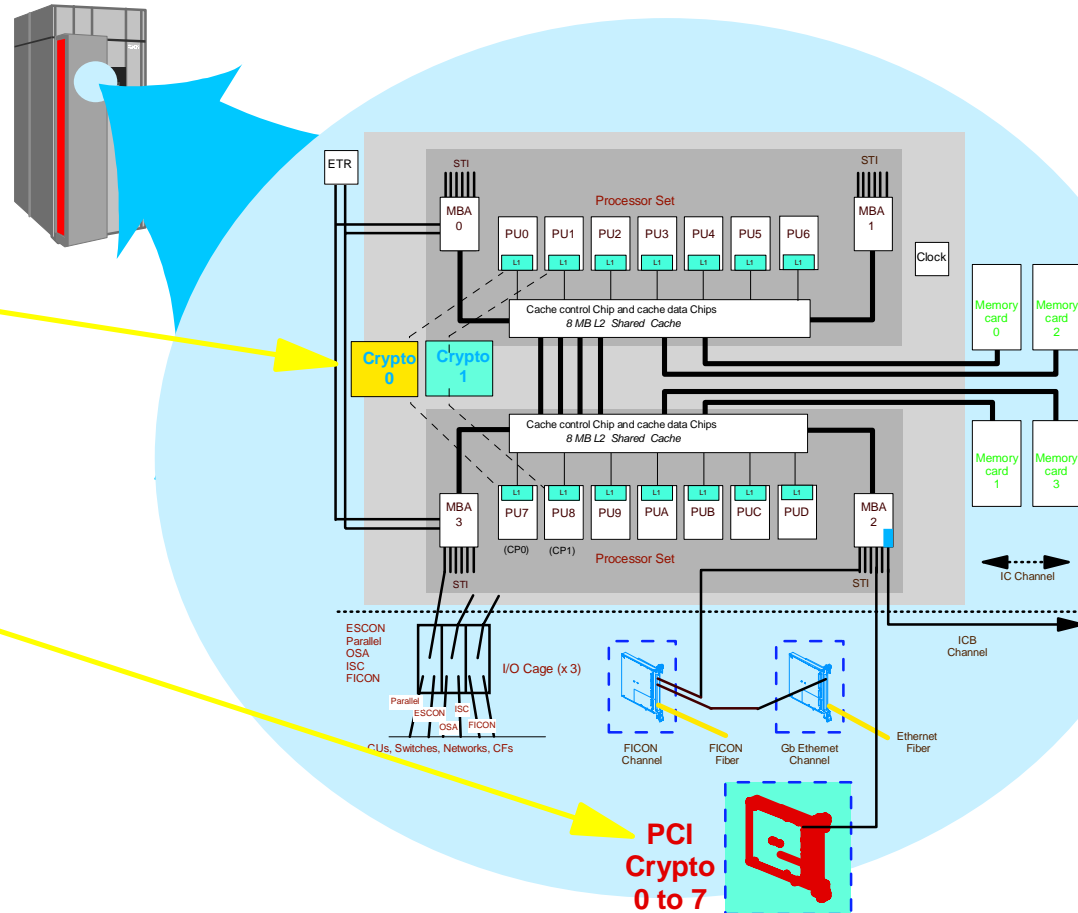
- A customer's certificate
  - Subject DN: CN=Sid Shopper.OU=Customer.O=Ohio.C=US
  - Issuer DN : OU=BobsMart  
Subscriber.O=Verisign,Inc.L=Internet



- Filtering rule = gives userid OHIOUSER when finding in certificate :
  - OU=Customer.O=Ohio.C=US||OU=BobsMart  
Subscriber.O=Verisign,Inc.L=Internet
- That is: customers in Ohio, with a certificate issued by Bobsmart Subscriber

# The S/390 Integrated Cryptographic Coprocessors

- 1994 : S/390 CMOS Cryptographic Coprocessor Facility (CCF)
  - priced feature on 9672 G3
  - standard feature on 9672 G4, G5, G6, z900 MP2000, MP3000
- 2000 : S/390 PCI Cryptographic Card (PCICC)
  - priced feature on 9672 G5, G6, z900
  - 0 to 8 cards in a system



IBM CCA compliant ...

FIPS 140-1 Level 4

# Who uses Crypto on S/390?

- IBM HTTP Server for OS/390
- System SSL
- OS/390 LDAP Directory Server
- CICS Transaction Gateway
- IBM Payment Suite e-commerce solutions on OS/390
- OS/390 TN3270 Server
- OS/390 Firewall Technology IPSec (VPN) and IKE (Internet Key Exchange)
- DCE Security Server
- VTAM
- BSAFE Toolkit - for applications and subsystems
- Financial Institution Applications
- CBT (Crypto Based Transactions) banking solution
- Open Cryptographic Services Facility (CDSA APIs)
- RACF

# LDAP Directory



**Redbooks**

International Technical Support Organization

# What is a directory Service ?

- Specialized database of information: people, systems parms, security, ...
- Simplified access methods, optimized for reading static data
- Highly scalable
- Security based on authentication and access control lists (ACLs)
- An integration point for many distributed computing environments

# Directory services on OS/390

- LDAP (Lightweight Directory Access Protocol) server
  - Part of OS/390 Security Server at OS/390 V2R5, with DB2 backend
  - RACF 'backend' since OS/390 V2R7
  - HCD 'backend' since OS/390 V2R9
  - Basic in OS/390 since OS/390 V2R8
  - New optimized (TDBM) DB2 backend at OS/390 V2R10
- LDAP client for C/C++ since OS/390 V2R4
  - LDAP client for Java since OS/390 V2R7
  - Socksified LDAP client at OS/390 V2R10
- NDS (Novell Directory Services) for OS/390
  - Non priced additional product



# What is LDAP ?

- Lightweight Directory Access Protocol is a subset of the Directory Access Protocol and the X.500 OSI directory service :
  - client-server communication over TCP/IP only
  - Streamlined functionality
  - Simplified data representation and encoding
- LDAP is the directory service of choice for the Internet
  - rich variety of features to support any kind of information, any kind of applications
  - open standards
  - -well documented, well known easy to use APIs
  - Today at V3 (RFCs 2251-2256)

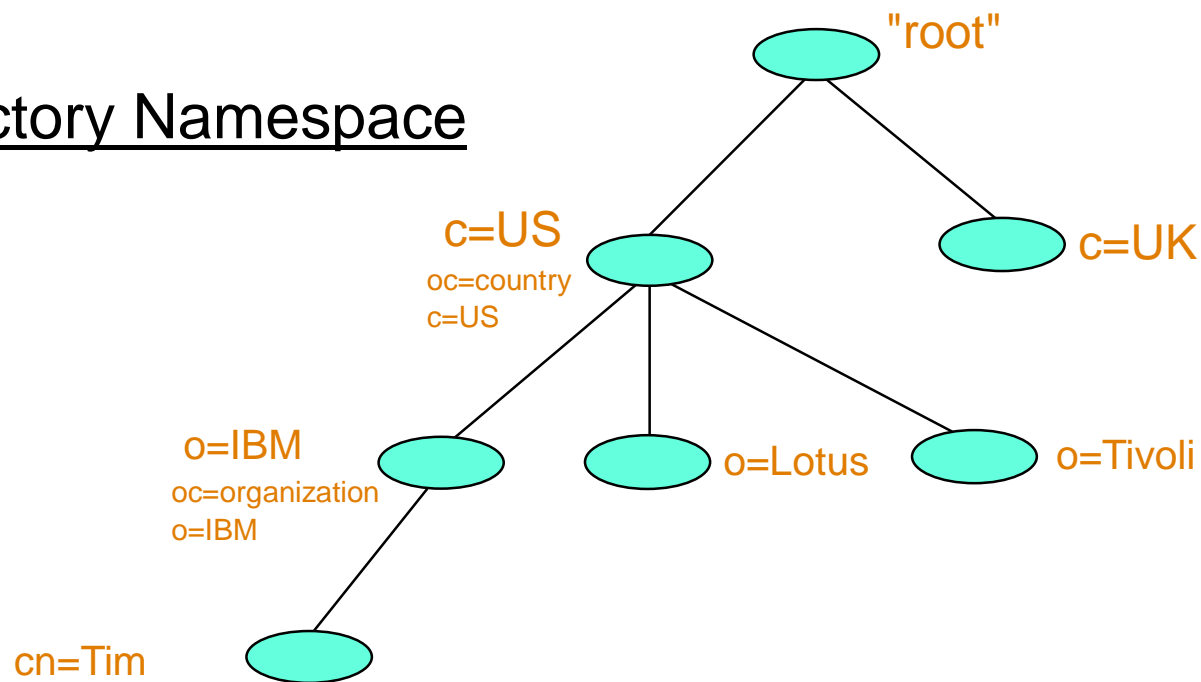
# The X.500 Directory Model

Directory is a hierarchy of entries

- Entries contain attributes
- Attributes have one or more values
- An entry's attributes (not their values) are defined by the entry's object class
- Each entry has a name relative to its parent. This is a relative distinguished name (RDN).
- All RDNs from root to entry put together form a distinguished name (DN)

# The X.500 Directory Model

## Directory Namespace



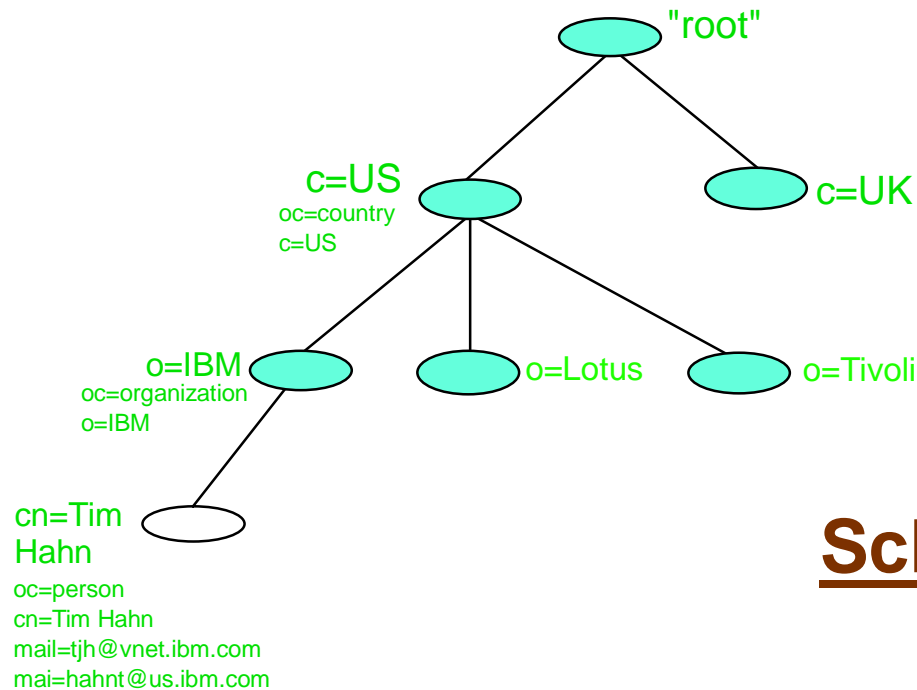
cn=Tim  
Hahn  
oc=person  
cn=Tim Hahn  
mail=tjh@vnet.ibm.com  
mai=hahnt@us.ibm.com

**RDN:** cn=Tim Hahn

**DN:** cn=Tim Hahn, o=IBM,  
c=US

- All entries have attributes (and values)
- Object class (oc) is an attribute in all entries
- Attributes grouped into mandatory and optional

# The LDAP Directory Model



## Schema

RDN: cn=Tim Hahn  
DN: cn=Tim Hahn, o=IBM,  
c=US

## Object Classes definitions

objectclass **person**

requires : cn  
          objectClass

allows: mail

objectclass **organization**

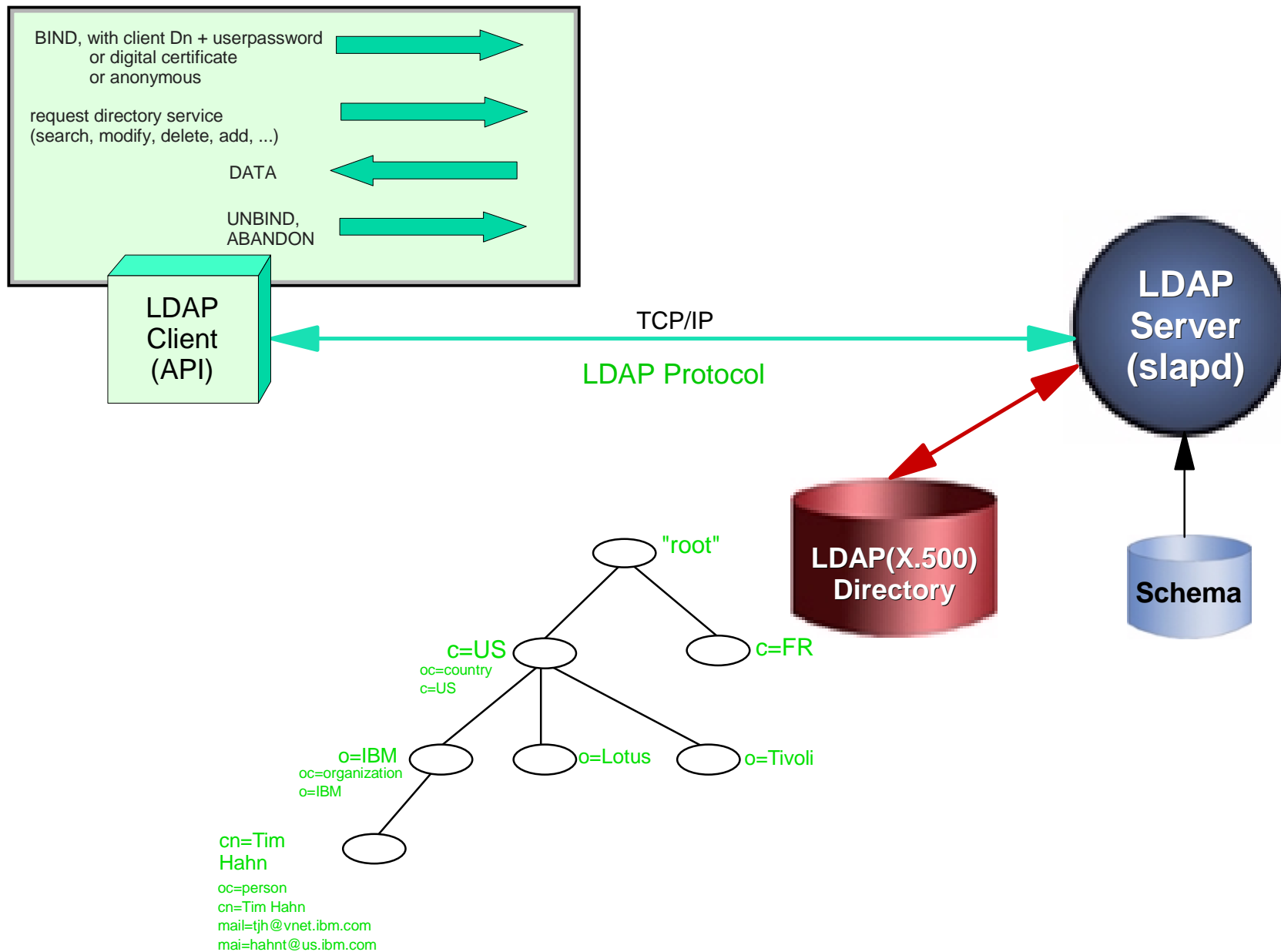
requires: o  
          objectClass

allows: description  
         businessCategory  
      .....

## Attribute types definitions

objectClass	oc	cis	128	normal
commonName	cn	cis	128	normal
organizationName	o	cis	128	normal
....				

# LDAP Operations



# The LDAP Functional Model

**BIND, with client Dn + userpassword  
or digital certificate  
or anonymous**



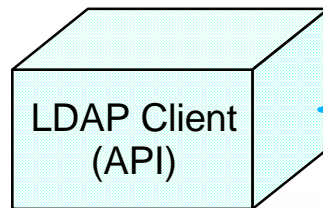
**request directory service  
(search, modify, delete, add, ...)**



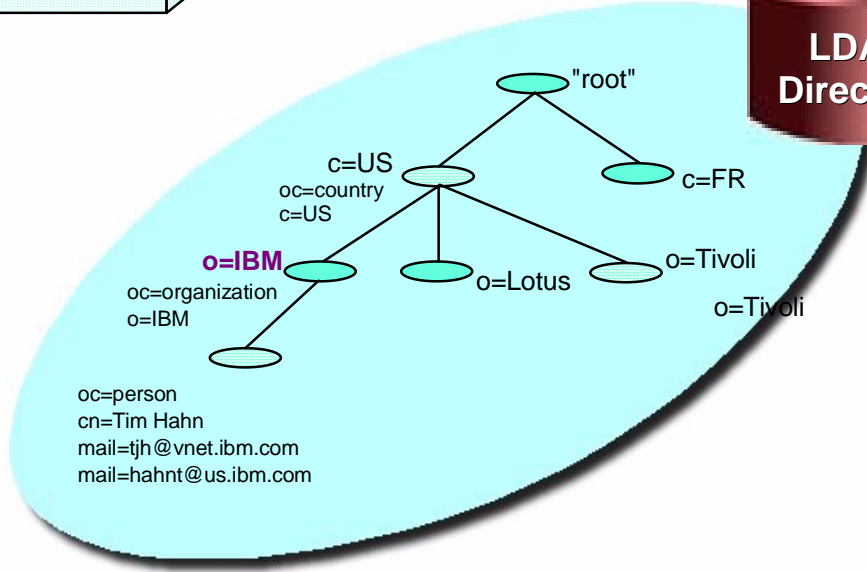
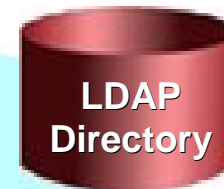
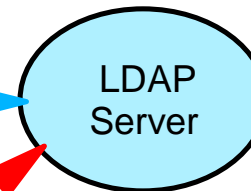
**DATA**



**UNBIND/ ABANDON**



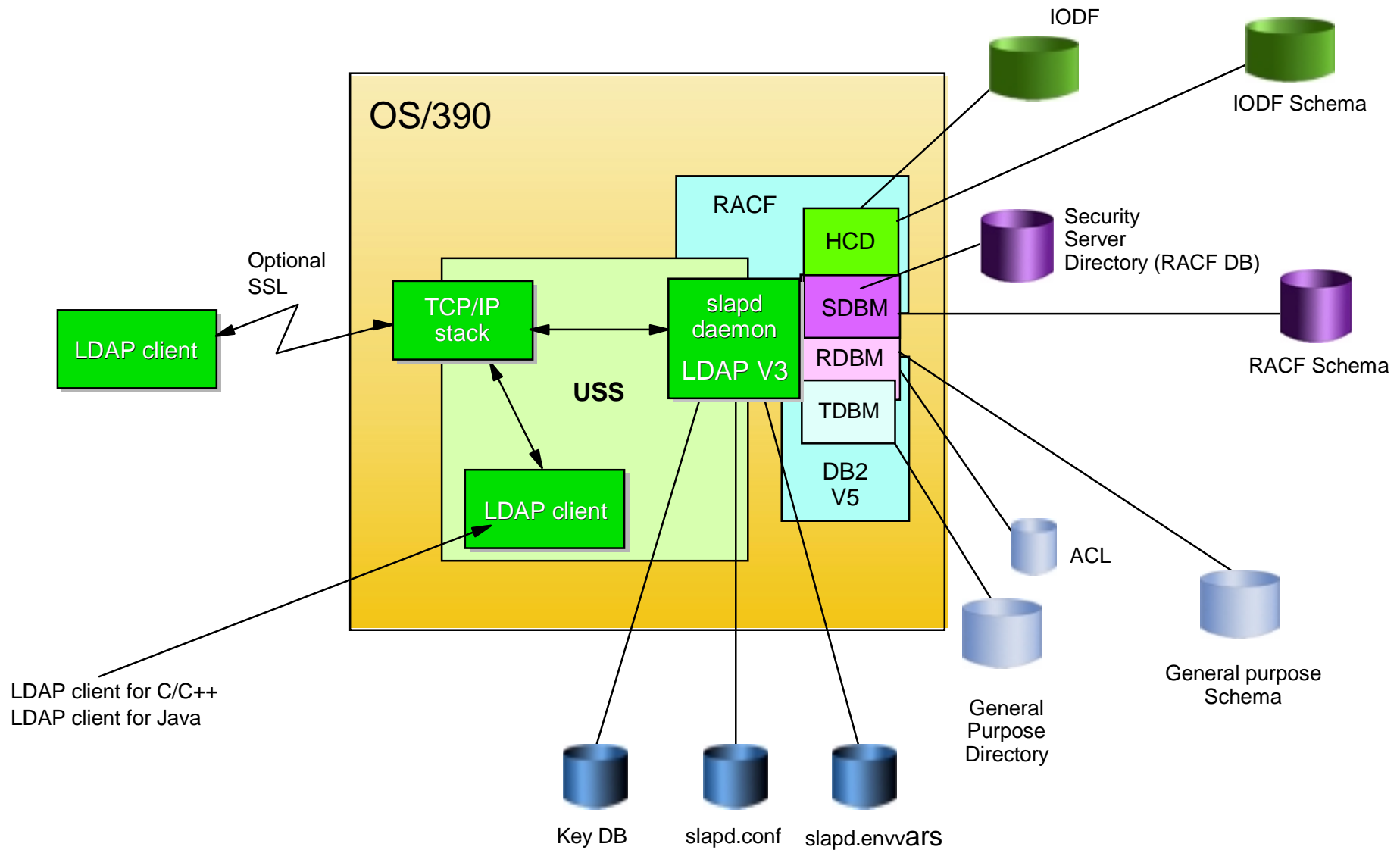
**LDAP Protocol on TCP/IP**



**In OS/390 or z/OS, can be:**

- DB2 tables
- RACF data base
- IODF data

# The OS/390 LDAP Server (as of z/OS 1.1)



# Brief History of LDAP on OS/390

## **OS/390 R5**, GA'd 3/1998

- LDAP V2 protocol
- Server includes DB2 backing store (RDBM back end), access control and replication support

## **OS/390 R6**, GA'd 6/1998

- Added remote ACL Admin

## **OS/390 R7**, GA'd 3/1999

- Sysplex Support
- Security Server SDBM back end support

## **OS/390 R8**, GA'd 9/1999

- Partial LDAP V3 protocol support

## **OS/390 R9**, GA'd 3/2000

- HCD backend
- Password encryption. Retrofitted R8 (APAR OW41326)

## **OS/390 R10**, GA 9/2000

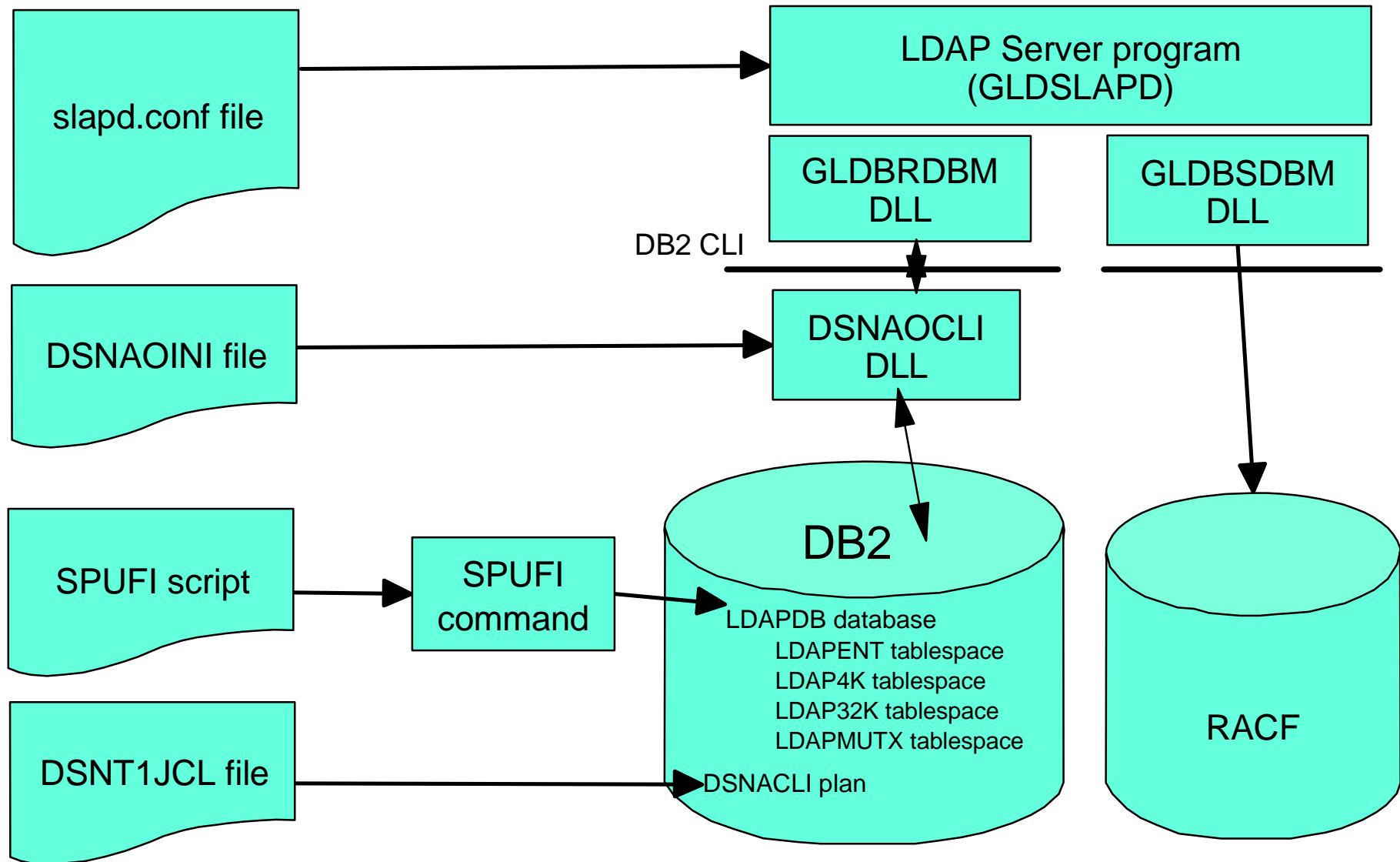
- new TDBM DB2 back end support



# Brief History of LDAP on OS/390 (cont.)

- LDAP client for C/C++
  - OS/390 R4, GA'd 9/1997, V2 protocol
  - OS/390 R5, GA'd 3/1998
  - OS/390 R6, GA'd 9/1998, V3 protocol
  - OS/390 R7, GA'd 3/1999
  - OS/390 R8, GA'd 9/1999
- LDAP client for Java
  - OS/390 R7, GA'd 3/1999, V2 & V3 protocols
  - OS/390 R8 PTF, added SSL support
- Socksified LDAP client
  - OS/390 R10 - Socks V4

# Configuring the LDAP Server



# LDAP and RACF

