

Information Assurance



Oracle Corporation

David Knox

Chief Engineer

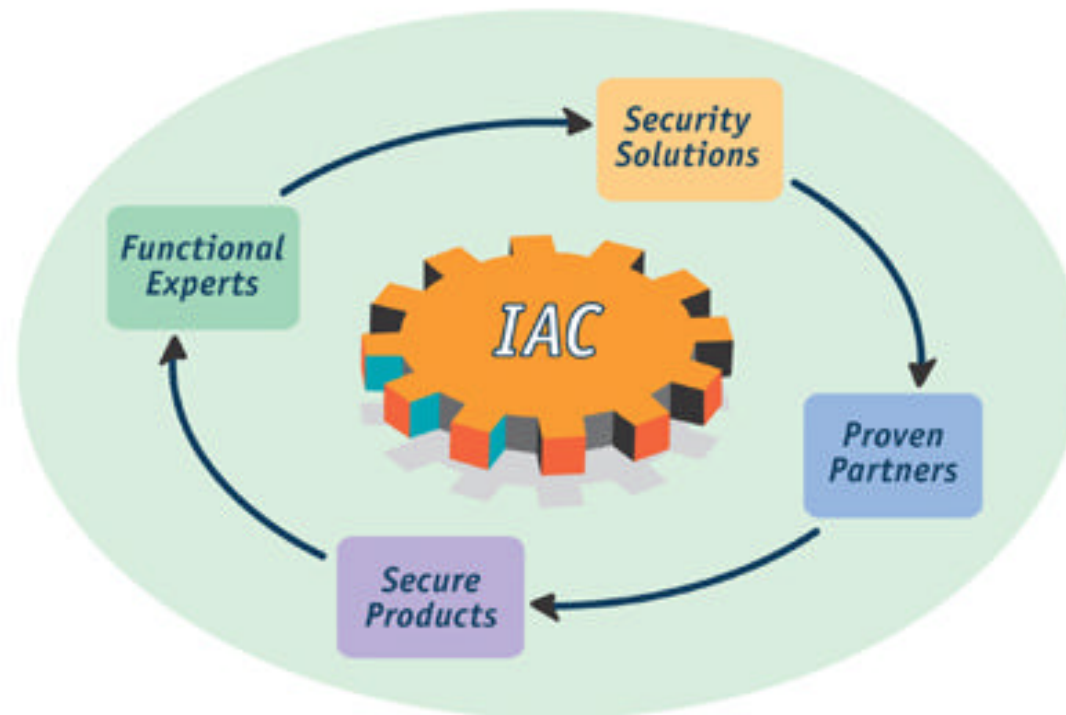
Information Assurance Center

ORACLE

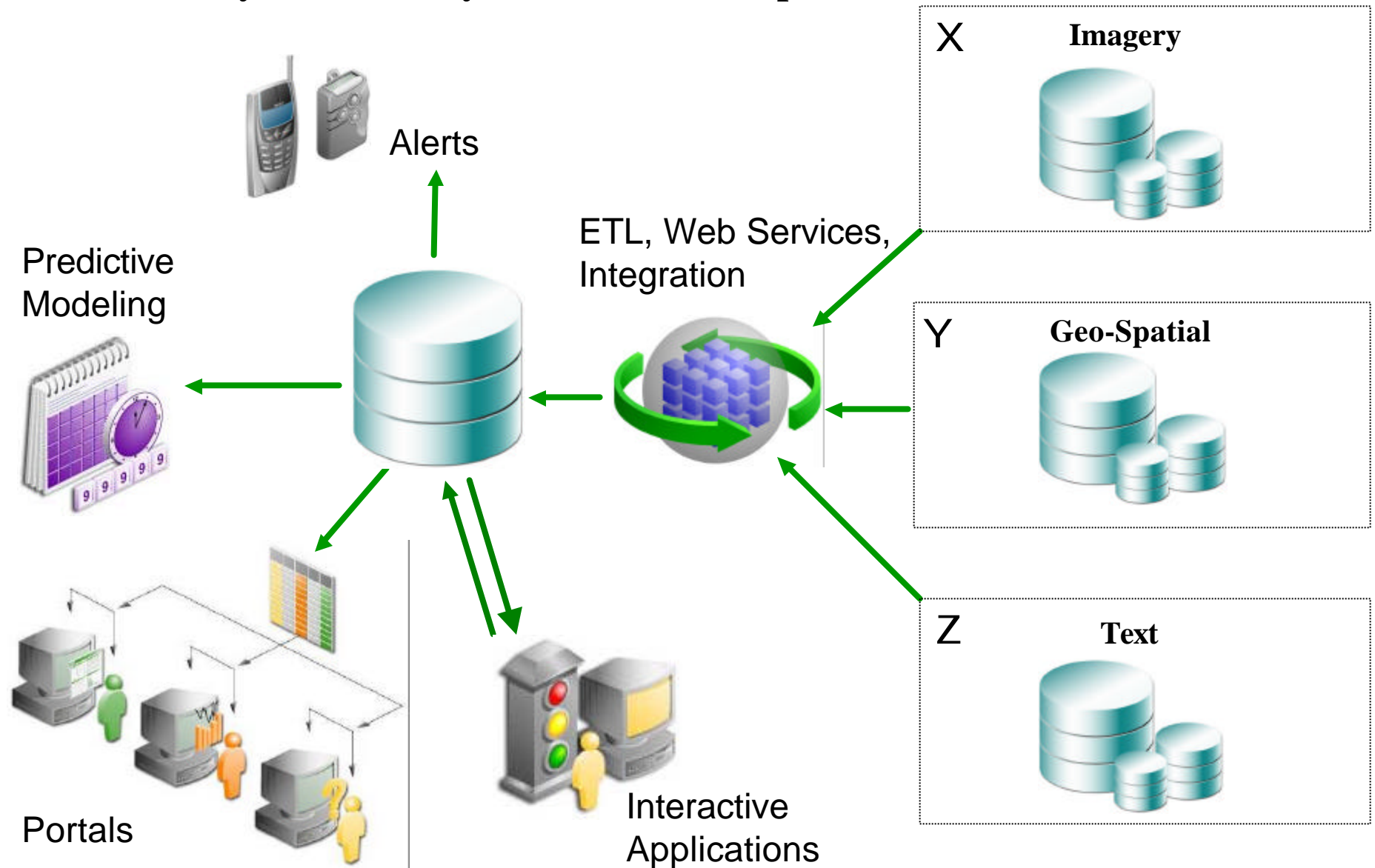
© 2003 David Knox, Oracle Corporation

Information Assurance Center

- ✍ Led by Dave Carey, former CIA Executive Director
- ✍ Focus on customer education
- ✍ Physical and virtual center
- ✍ Test bed for leading-edge security solutions



1. As data is consolidated it is more usable and less costly to manage
2. **Availability and Security are now more important**



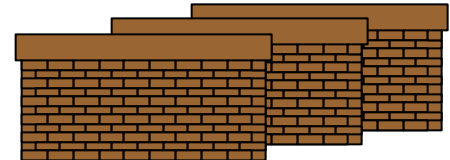
Information Assurance Tenets


 Security has to be built in to the system, not bolted on afterwards



 Defense in depth

- Security in layers for higher assurance



 Not feasible to build systems 100% secure

- Risk mitigation, not risk avoidance
- Security implementations compete with usability, performance, cost, and administration



 The database is your vault, choose wisely



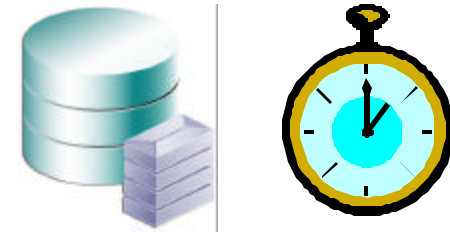
Information Assurance in a Nutshell

- 1. Ensuring information availability**
- 2. Securing your information assets**

Business Continuity - Availability

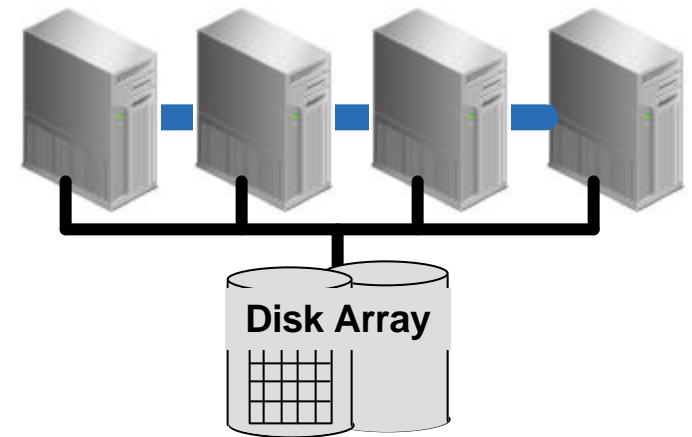
Continuous Operations – 24 x 7 x 365

- Online rebuilds
- Hot backup/recovery
- Fault isolation



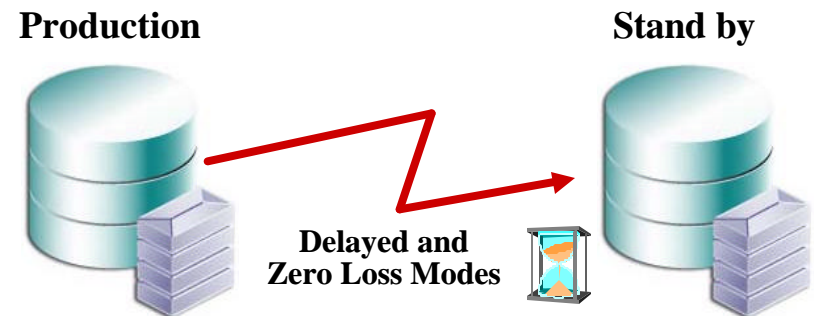
High Availability – No SPOF

- “Real” Application Clusters (RAC)
- Linear scalability
- Localized high availability



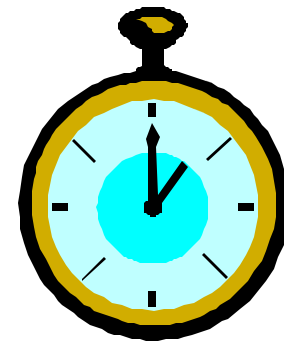
Recoverability/Survivability

- Fail over operating environment
- Geographically separated
- Graceful recovery



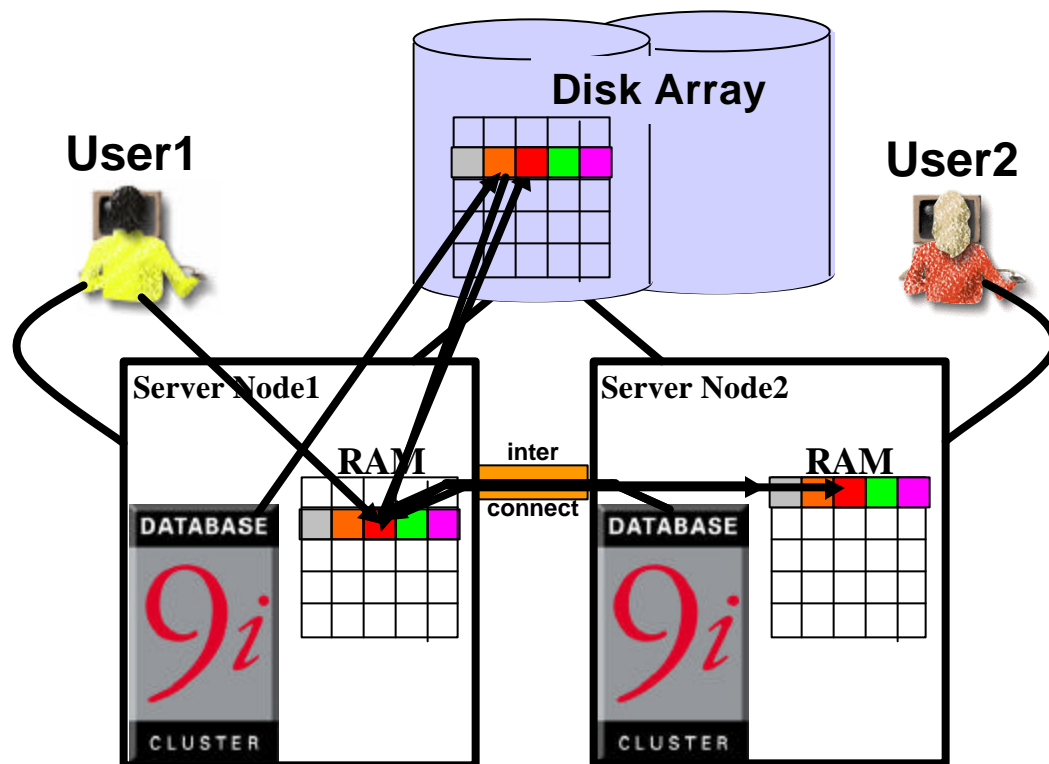
Continuous Operations – 24 x 7 x 365

- ✍ Automated space management
- ✍ Online rebuilds
- ✍ Hot backup/recovery
- ✍ Block-level recovery
- ✍ Fault isolation
 - Data partitioning
 - Self-service error correction



Real Applications Clusters - Cache Fusion

Fast

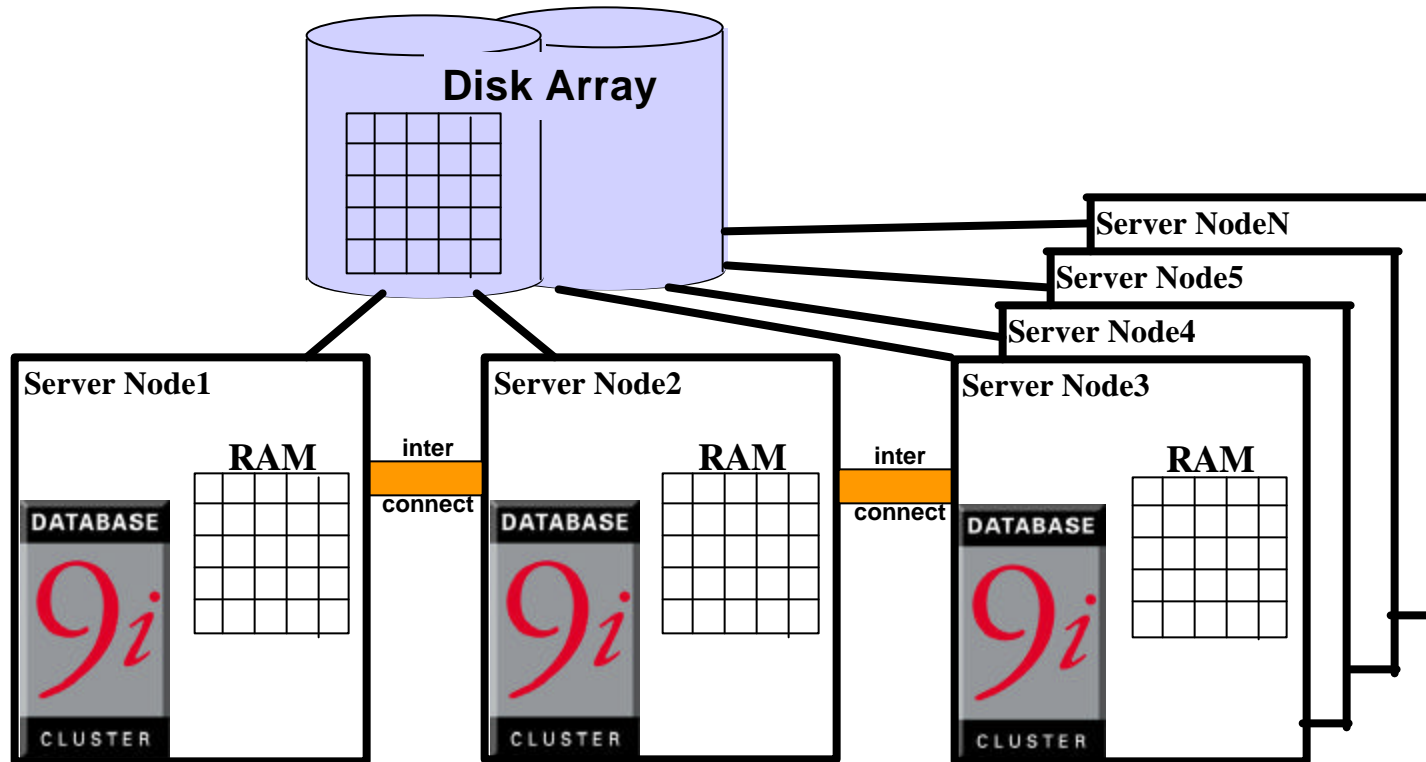


1. User1 queries data
2. User2 queries same data - via interconnect with no disc I/O
3. User1 updates a row of data and commits
4. User2 wants to update same block of data – 9i keeps data concurrency via interconnect

ORACLE

Real Applications Clusters

Scalability

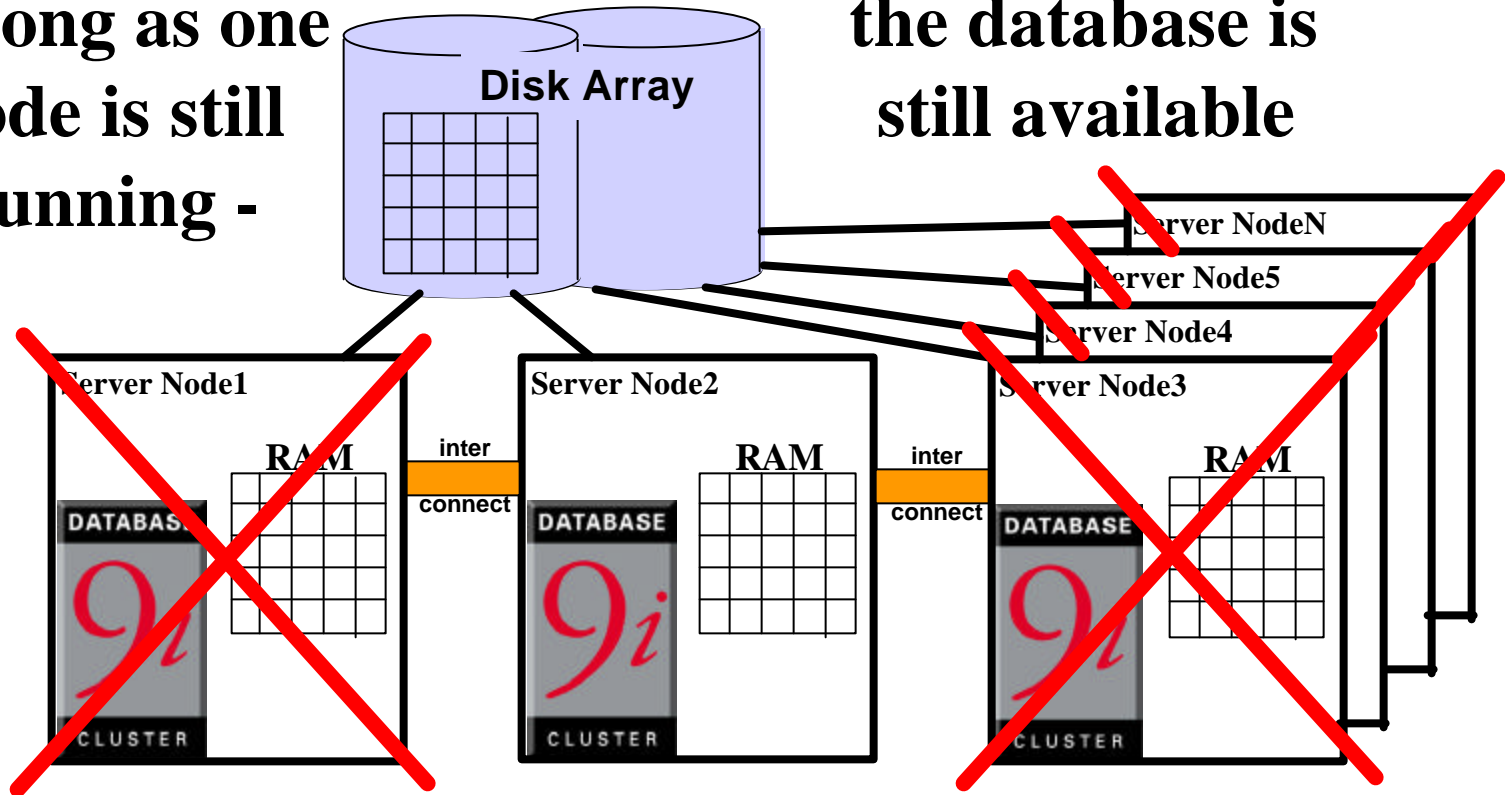


Real Applications Clusters

Localized High-Availability

As long as one
node is still
running -

the database is
still available



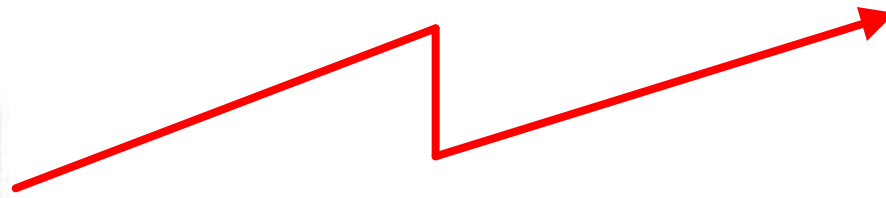
Oracle Data Guard

Distributed for Disaster Recovery

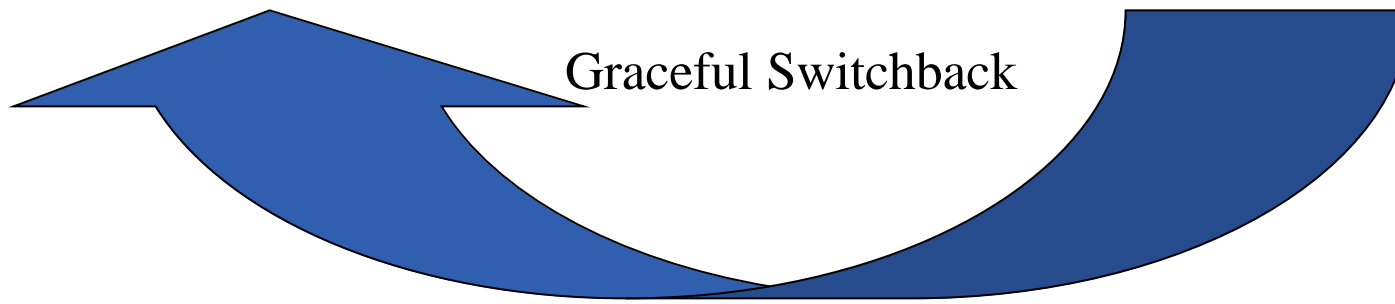
Production



**Delayed and
Zero Loss Modes**



Standby



ORACLE

Information Assurance

1. Ensuring information availability
2. Securing your information assets

**“If you spend more on coffee than on IT security, then you will be hacked
...what's more, you deserve to be hacked!”**

**Richard Clarke, 2002
Special Advisor to the President,
Cyberspace Security**



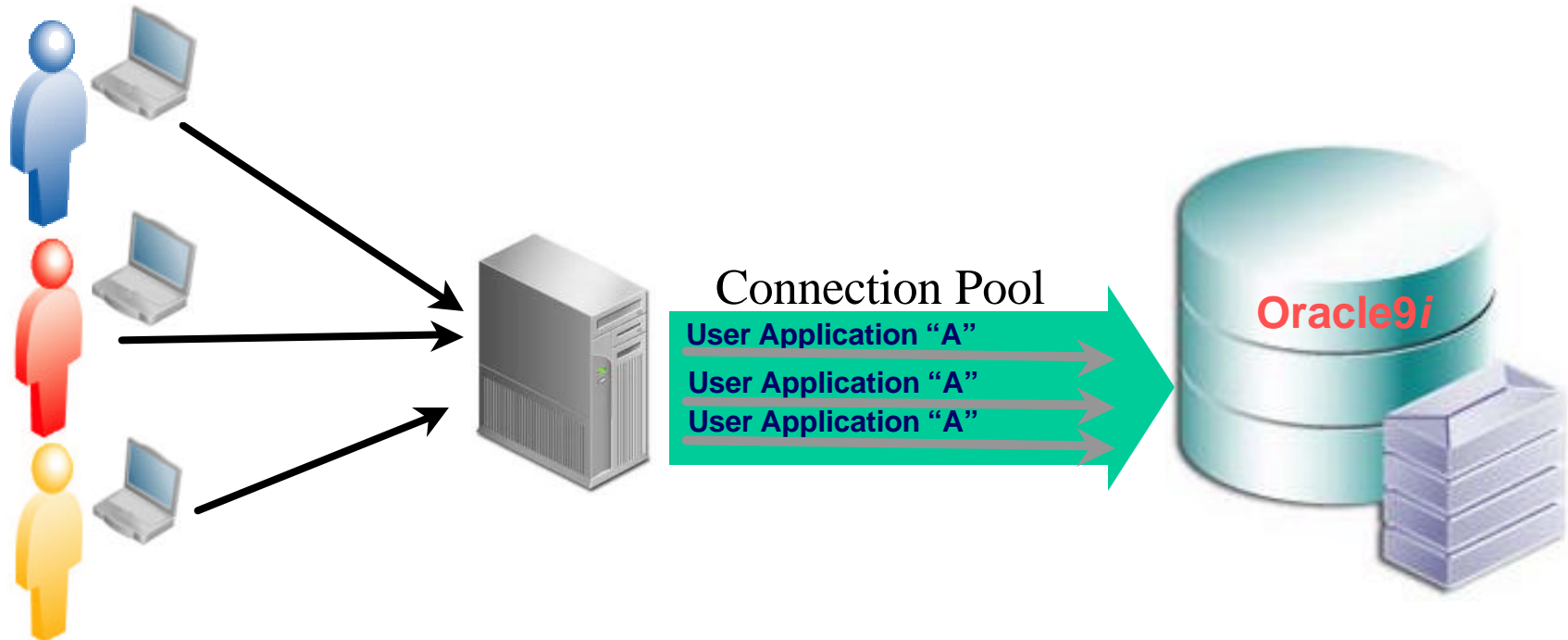
Defense in Depth and Multiple Layers of Security

Introducing the
Oracle⁹ⁱ
DATABASE

- ✍ Identity Preservation
 - Proxy Authentication
- ✍ Authorizations
 - Secure Application Roles
- ✍ Fine-Grained Access Control
 - Virtual Private Database
 - Oracle Label Security
- ✍ Element Level Protections
 - Data Encryption/Digests
- ✍ Accountability
 - 200+ audited events
 - Fine-grained auditing

Typical Authentication Architecture

Security cannot be based on anonymity!



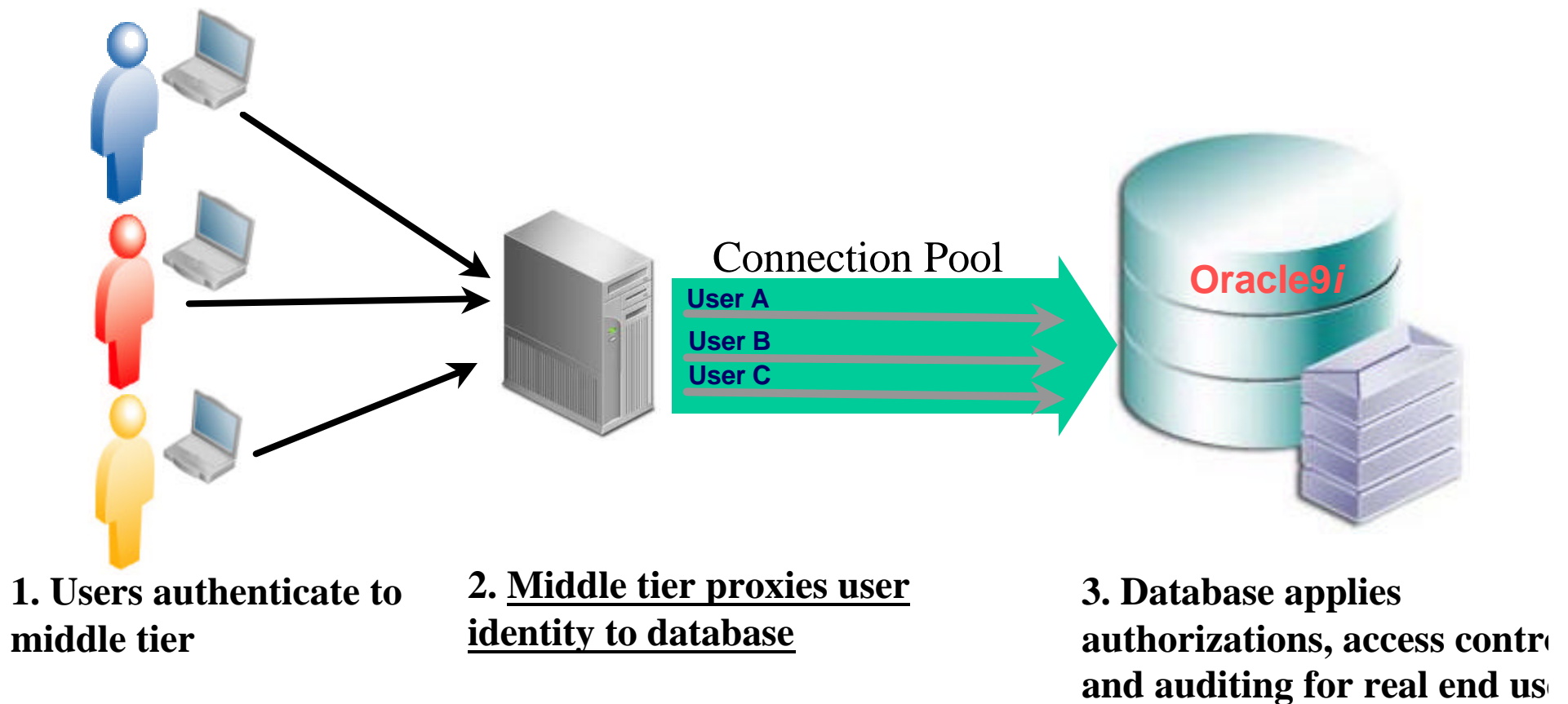
1. Users authenticate to middle tier

2. Middle tier connects to an (anonymous) application account

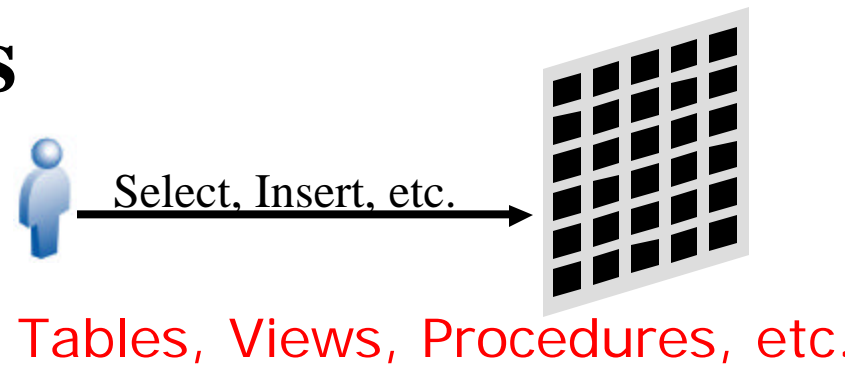
3. Database cannot apply proper authorizations, access control and auditing at the user level

ORACLE

Identity Preservation – Proxy Authentication



Database Authorizations



- ✍ Object or system privileges granted to roles
- ✍ Roles granted to users for ease of administration
- ✍ Roles enabled by default or “turned on” by application
- ✍ Protected by password
- ✍ Challenges
 - Application controls database privileges
 - Sharing password among applications is not easy, practical, or secure

Database Authorizations

Least-privilege access desired

- Users must *only* have the privileges they need to perform a specific task
- Users must *only* be able to access data through an application

Issues

- No way of knowing how data is being accessed
Application? Ad-hoc query? Reports? Direct connection?
- Can't maintain an application-to-privilege binding
Maintain least-privileges for multiple applications with multiple methods accessing same database

Oracle9i Secure Application Role

Enforcing Least-Privilege



- Secure application role is a role enabled by security code
- Application asks database to enable role (can be called transparently)
- Security code performs desired validation before setting role (privileges)

Secure Application Role Benefits

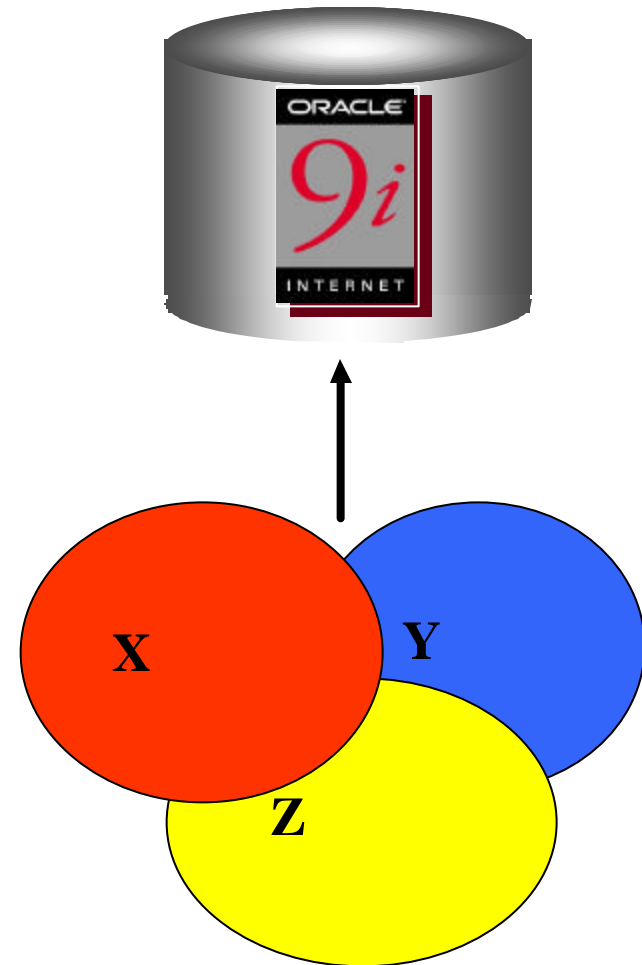
- ✍ Security policy (implementing code) can check anything:
 - time of day
 - day of week
 - IP address/domain
 - Local or remote connection
 - user connected through application
 - X.509 data, etc.
- ✍ Database controls whether privileges are enabled

- ✍ Multiple applications can access database securely
- ✍ No need to maintain secure password store
- ✍ Allows secure handshake between applications and database

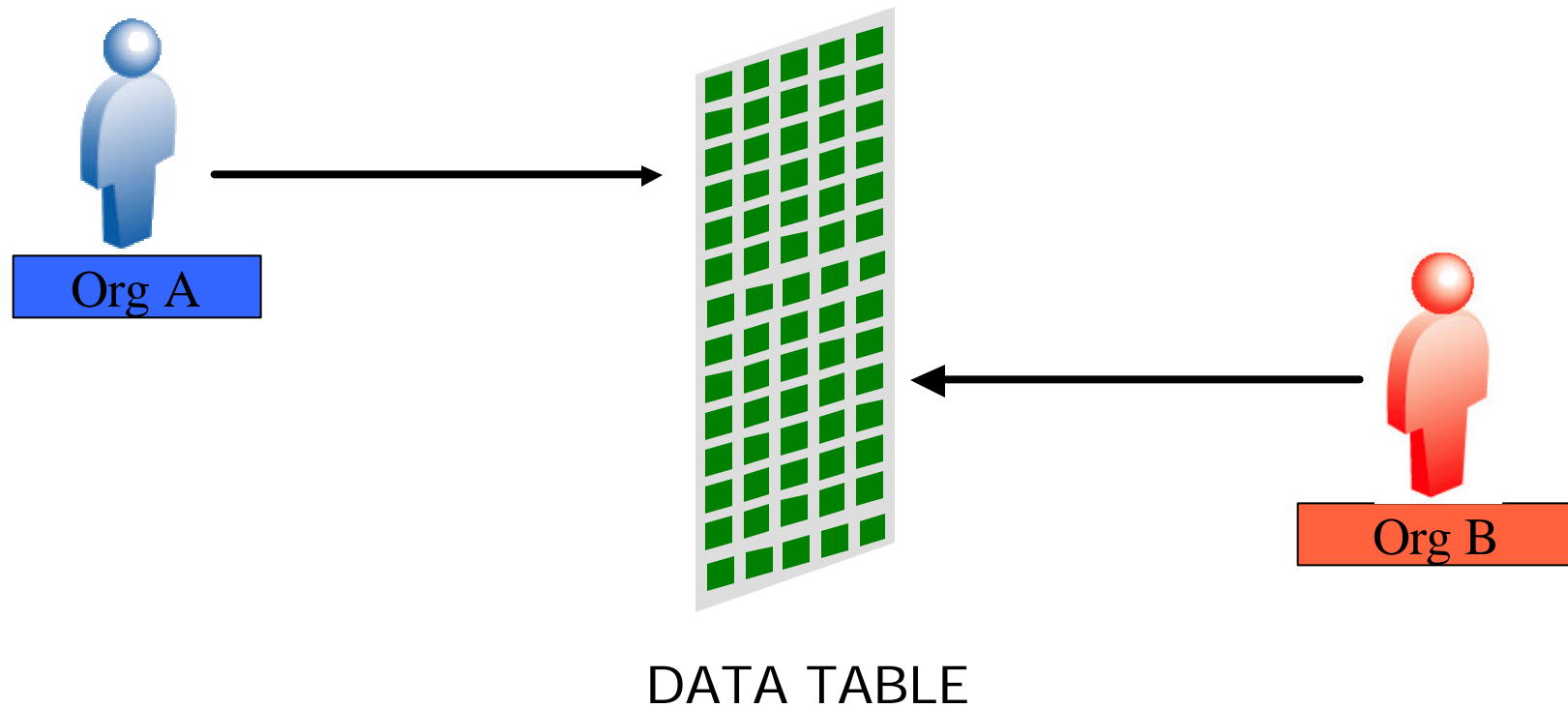
Consolidation and Access Control

Enable enterprise-wide data sharing

- ✍ Consolidated data resides in single repository for speed and manageability
- ✍ Data separation *while* data sharing
- ✍ The importance of security is now increased

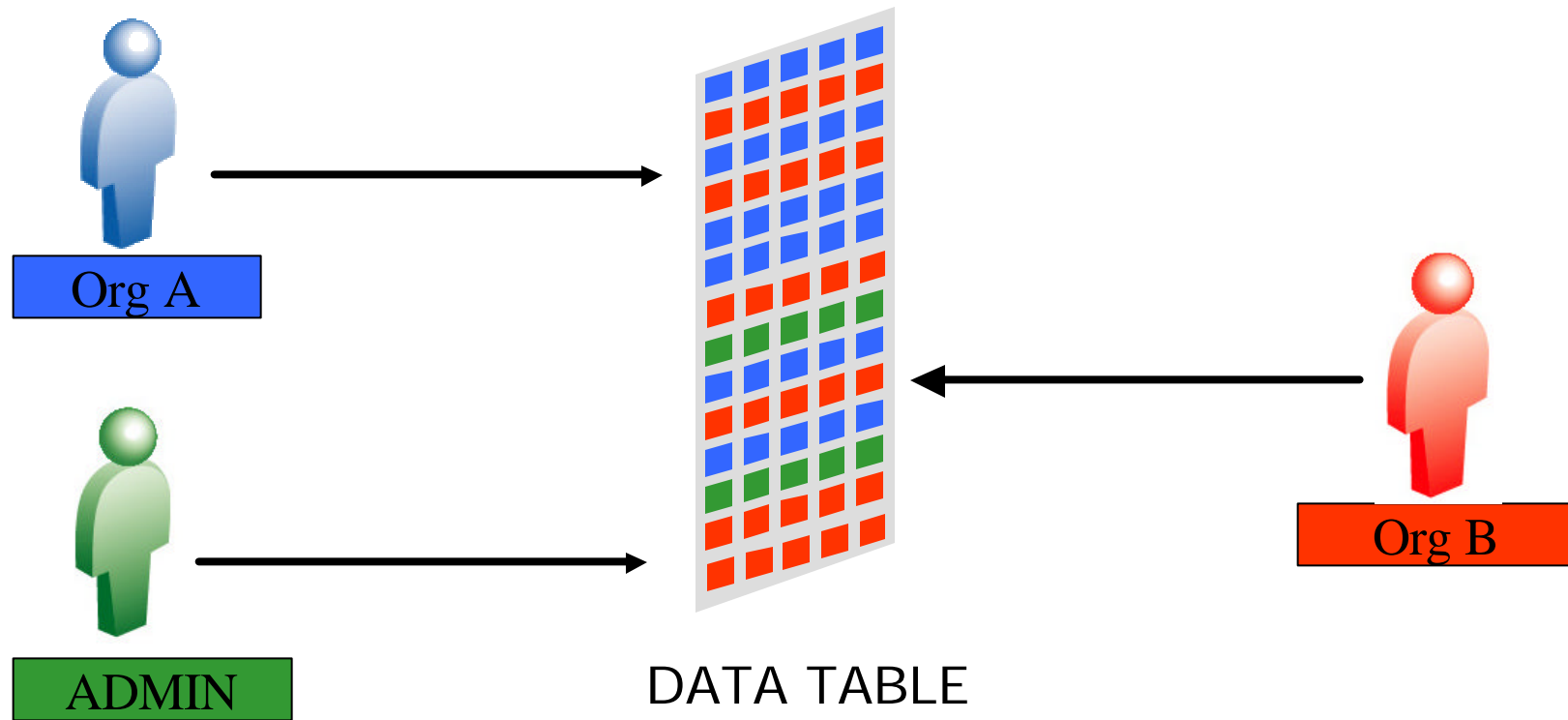


Object Access Control



Fine Grained Access Control

A.K.A. Row-level Security



Fine-Grained Access Control: Enforcement Mechanisms

Application Enforcement

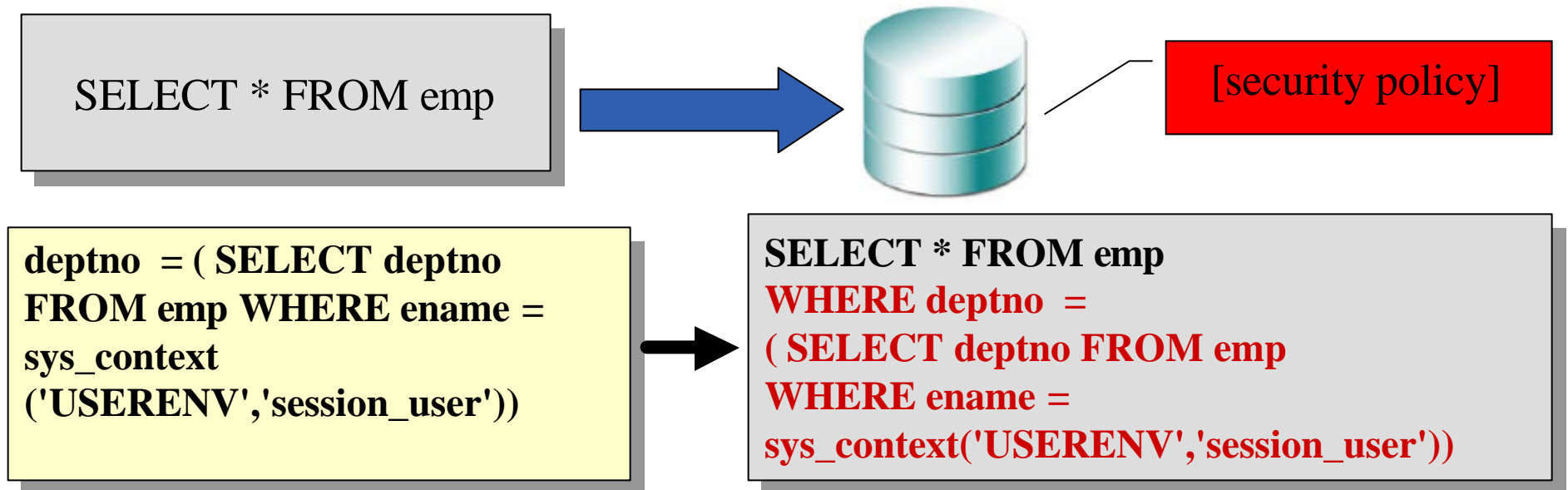
- Performance Suffers
- Enforced within application only
- Requires changes to all applications when policy changes
- Difficult and expensive to maintain

Server Enforcement

- Optimal performance
- Strictly enforced, no exceptions
- No changes to applications when policy changes
- Easy to manage/verify

How it works

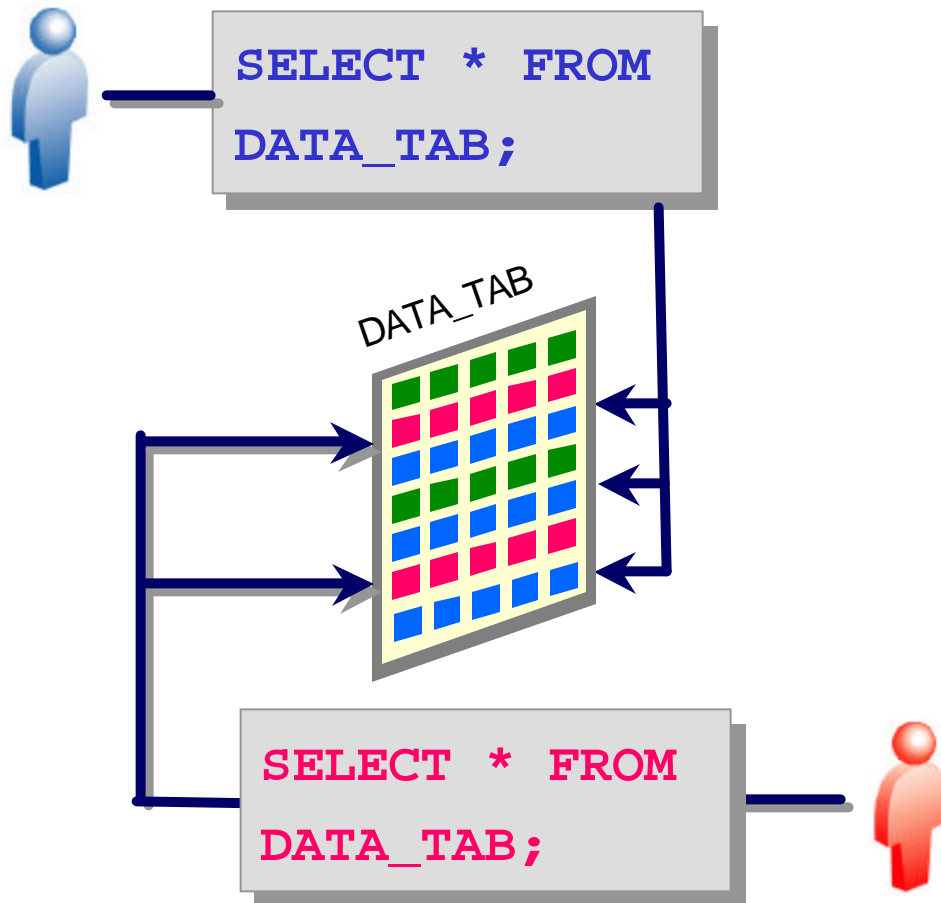
- ✍ Direct or indirect user access to object with an attached policy automatically invokes the policy
- ✍ Policy function returns a predicate (a 'WHERE' condition)
- ✍ Database dynamically rewrites the SQL statement by appending the predicate
- ✍ New statement is optimized and executed



Policy Checks

- ✍ Does NOT require application context
- ✍ IP address
 - `sys_context('userenv','IP_ADDRESS')`
- ✍ Time of day
 - `to_number(to_char(sysdate,'HH24'))`
- ✍ Day of week
 - `to_char(sysdate,'D')` not in ('1','7')
- ✍ User list
 - `USER` in ('SCOTT','SYSTEM','DKNOX')
- ✍ User has role
 - `dba_role_privs => IF (sys_context('MyAppCtx','AppRole') THEN`
- ✍ Environment setup correctly (authentication mode)
- ✍ Part of the data
 - Deptno, owner, etc.

Virtual Private Database (RLS)



- Multiple policies
- Different policies for different operations (CRUD)
- Simplifies application development
- Security cannot be by-passed
- Scalable via secure attribute cache
- Better manageability

Better than Views

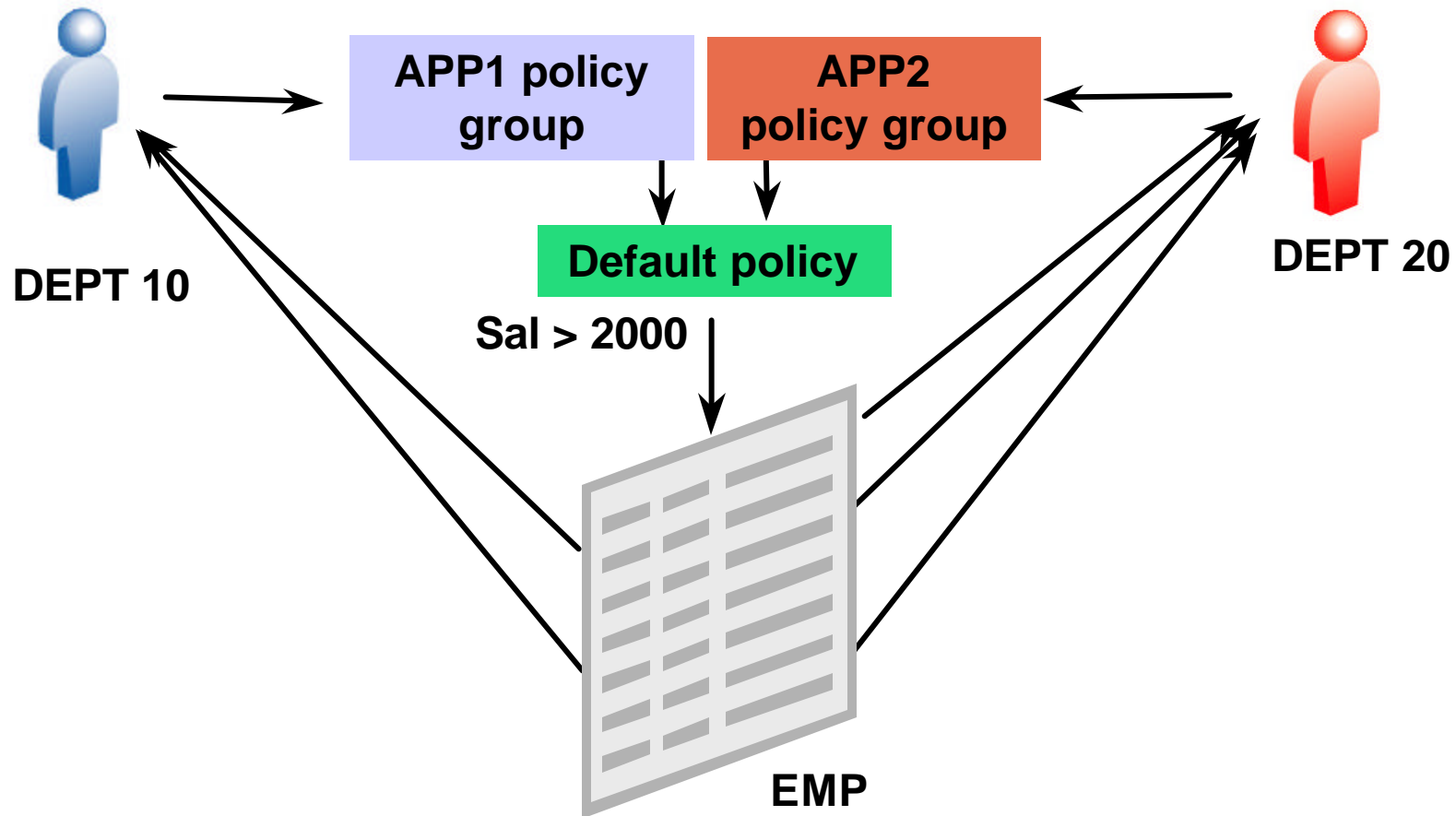
Better than multiple applications

Partitioned FGAC

Application Context: name = APPS_CTX Attribute = ACTIVE_APPS 1

Context set to APP1

Context set to APP2



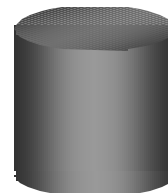
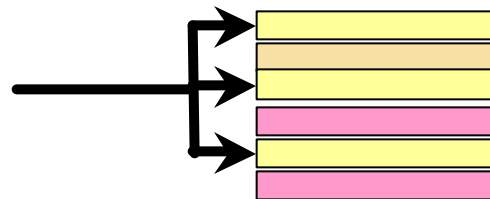
Oracle Label Security

- ✍ Based on VPD, grew out of accredited consulting work and over seven years of MLS efforts
- ✍ Off-the-shelf label based RLS system
- ✍ GUI for administration
 - No coding required

User	Label
Scott	Confidential : Oncology : Patient



Data Rows



Row Label

Confidential	Strategic	Patient
Confidential	Oncology	Research
Secret	Nuclear	POB, SOP
Public	Oncology	POB, SOP
Confidential	Strategic	Research
Confidential	Radiology	Research
TopSecret	Personnel	POB, UK
High Sens	Lab	POB, UK
Confidential	Strategic	Admission
Confidential	Radiology	Admission
TopSecret	Readiness	POB, ER
High Sens	X-ray	POB, ER

Levels

Groups

Compartments

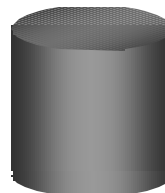
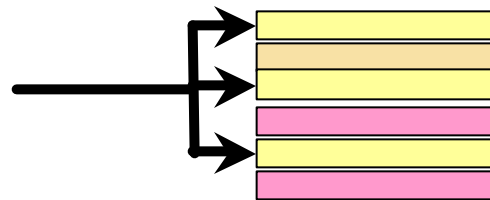
Oracle Label Security

- ✍ Based on VPD, grew out of accredited consulting work and over seven years of MLS efforts
- ✍ Off-the-shelf label based RLS system
- ✍ GUI for administration
 - No coding required

User	Label
Scott	Confidential : Strategic : Coalition



Data Rows



Row Label

Confidential : Strategic : Coalition
Secret : Nuclear : US, UK
Confidential : Strategic : Coalition
TopSecret : Personnel : US, UK
Confidential : Strategic : Coalition
TopSecret : Readiness: US

Levels

Groups

Compartments

PCASSO Project

Patient Centered Access to Secure Systems Online

- ✍ SAIC and UCSD – Patient and health care providers access patients’ complete medical records over the Internet
- ✍ 178,000 patients
- ✍ “In defining those levels, we needed to separately protect highly sensitive information that – by law- requires special protection. ...Label-based access control is ideal for this purpose”
 - Dixie Baker, corporate VP of technology and CTO for SAIC’s healthcare practice

Stored Data Encryption

Requirement

- Selective encryption of sensitive data (e.g., ssn, ccn, diagnosis)
- hackers compromising the operating system and reading database/log files
- malicious DBA

Features

- Data Encryption Standard (DES)
- Triple-DES
- MD5 cryptographic checksum

*Encrypted
SALARY*

KING	10	
SCOTT	20	
BLAKE	30	
SMITH	20	
JAMES	30	
JONES	20	
MILLER	10	

Security Processes: Prevention, Detection and Response

Prevention

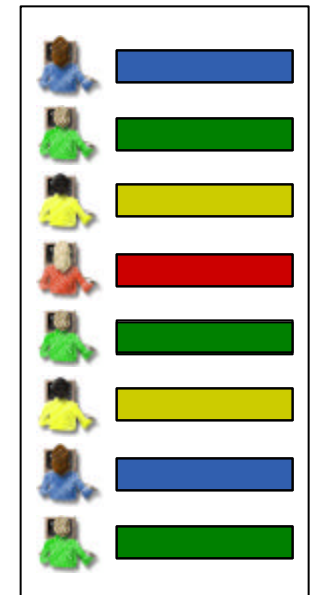
- Authentication, Access Controls

Detection and Response

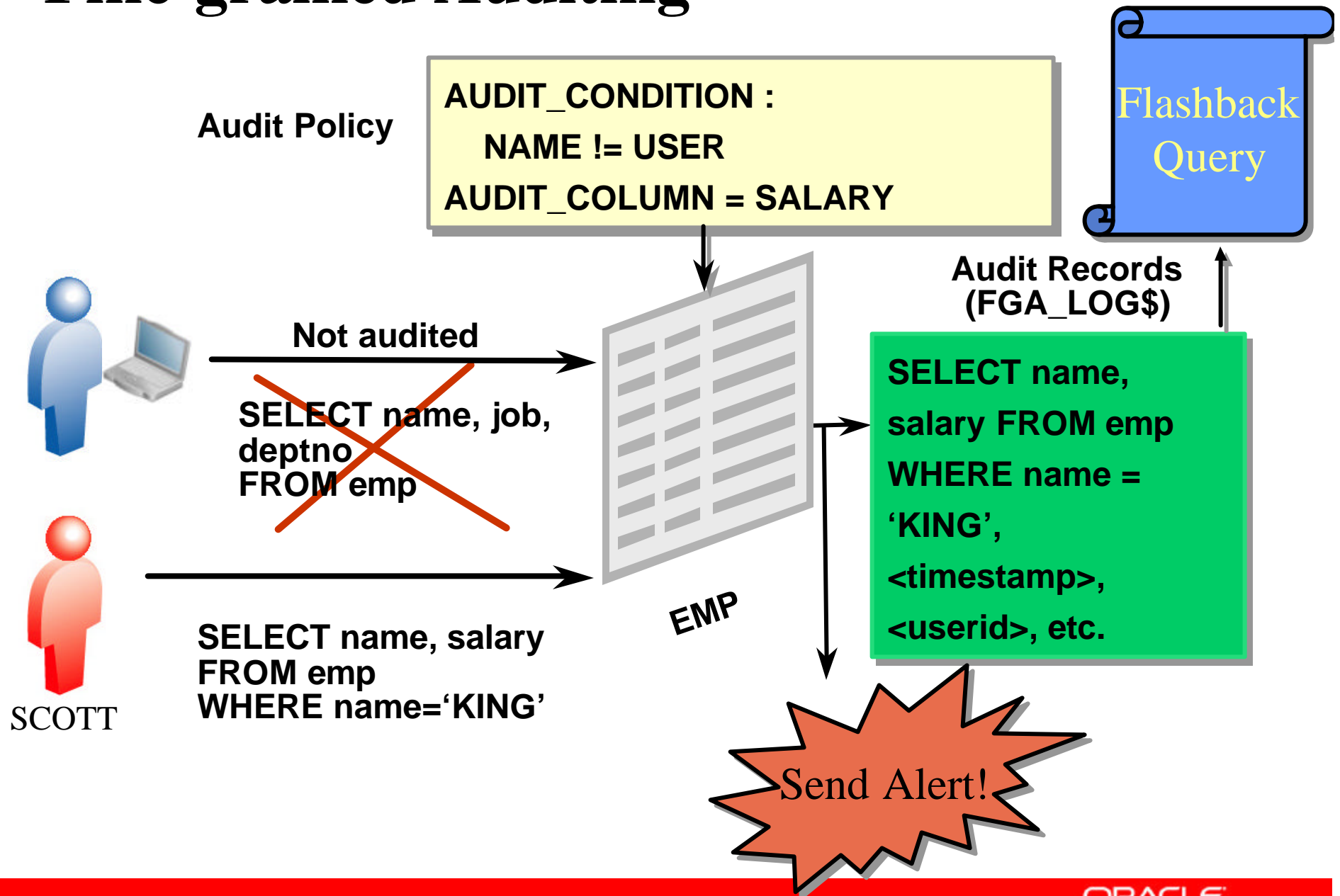
- Database Auditing
- Audit by user, by object, by privilege
- Capture Successful and Unsuccessful actions

Audit Improvements

- Minimize audit data
- Capture user's intent (query)
- See resulting data set



Fine-grained Auditing



Oracle Advanced Security

1. Network encryption & integrity

- ✍ Includes AES and SSL
- ✍ FIPS140-1 level 2 evaluated



2. Strong authentication of end users, clients and servers

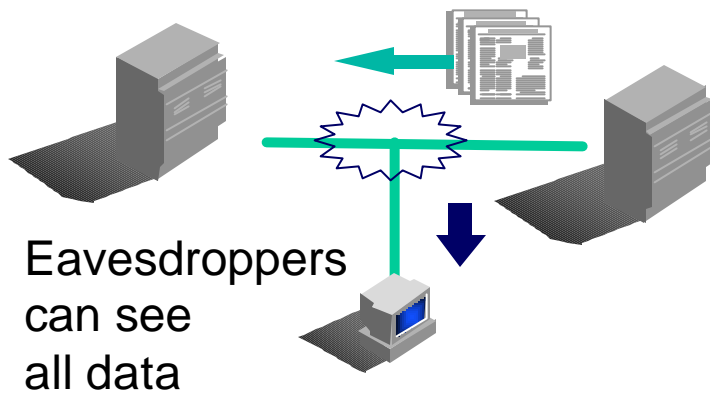
- ✍ Kerberos, biometrics, tokens, RADIUS

3. Identity management/Centralized users

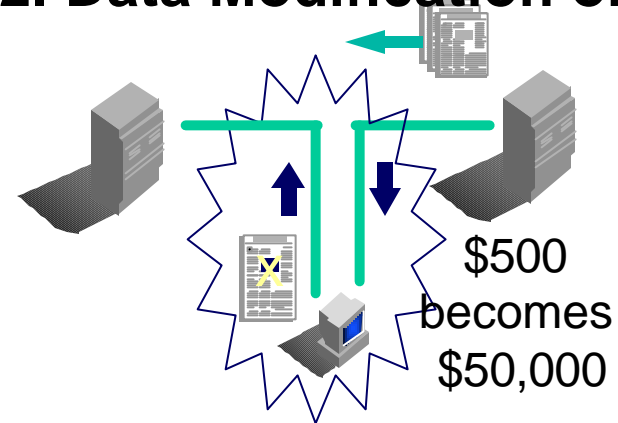
- ✍ Lowers cost of user administration
- ✍ LDAP-standard
- ✍ Extensible
- ✍ Protected by SSL

Threats to Networks and Internet

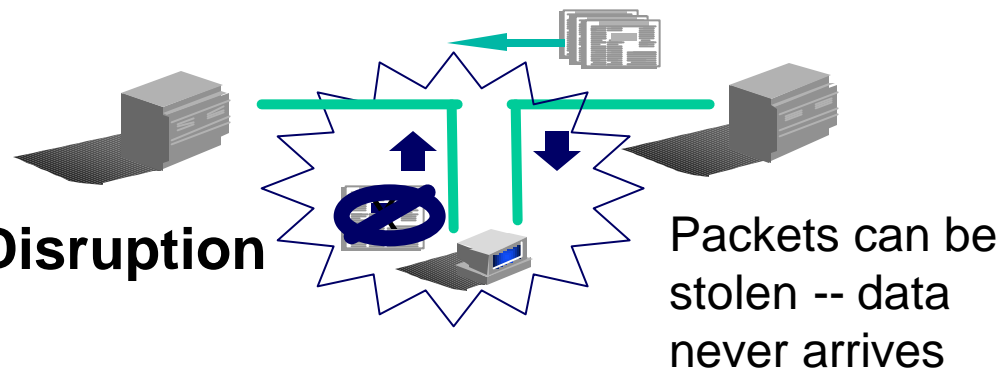
1. Data Theft



2. Data Modification or Replay



3. Data Disruption



Algorithm Negotiation

Flexibility for security and performance

- ✍ Select Encryption
 - for Client as well as Server
- ✍ Choose Encryption Negotiation Options
 - accepted, rejected, requested, required
- ✍ Select the Encryption Algorithms
 - **AES256**
 - RC4 256
 - RC4 128
 - 3DES 168 (3 key)
 - 3DES 112 (2 key)
 - RC4 56
 - DES 56

Thin Java Encryption

- ✍ Secure connections from thin JDBC clients to the database
- ✍ Java classes for the encryption and integrity algorithms
 - RSA RC4 (up to 256 bit key length)
 - DES (40 and 56)
 - MD5
- ✍ Java Crypto code is obfuscated

Data Integrity

- ✍ Includes a sequenced, cryptographic checksum with every packet before it is sent
- ✍ Uses Industry Standard algorithms
 - MD5 and SHA-1
- ✍ Sequenced Cryptographic Checksums - Automatically detects
 - Modifications
 - Replay of packets
 - Missing packets
- ✍ Violations terminate the operation in progress and are logged in server log files

ASO Confidentiality and Integrity

- ✍ Encrypts all communication with the database
 - Oracle Net, thick and thin JDBC, IIOP
- ✍ Sequenced Cryptographic Checksums
 - Prevents replays
 - Prevents data modifications
- ✍ FIPS 140-1 Level 2 certification
- ✍ Applications run UNCHANGED!

Technology
Assurance

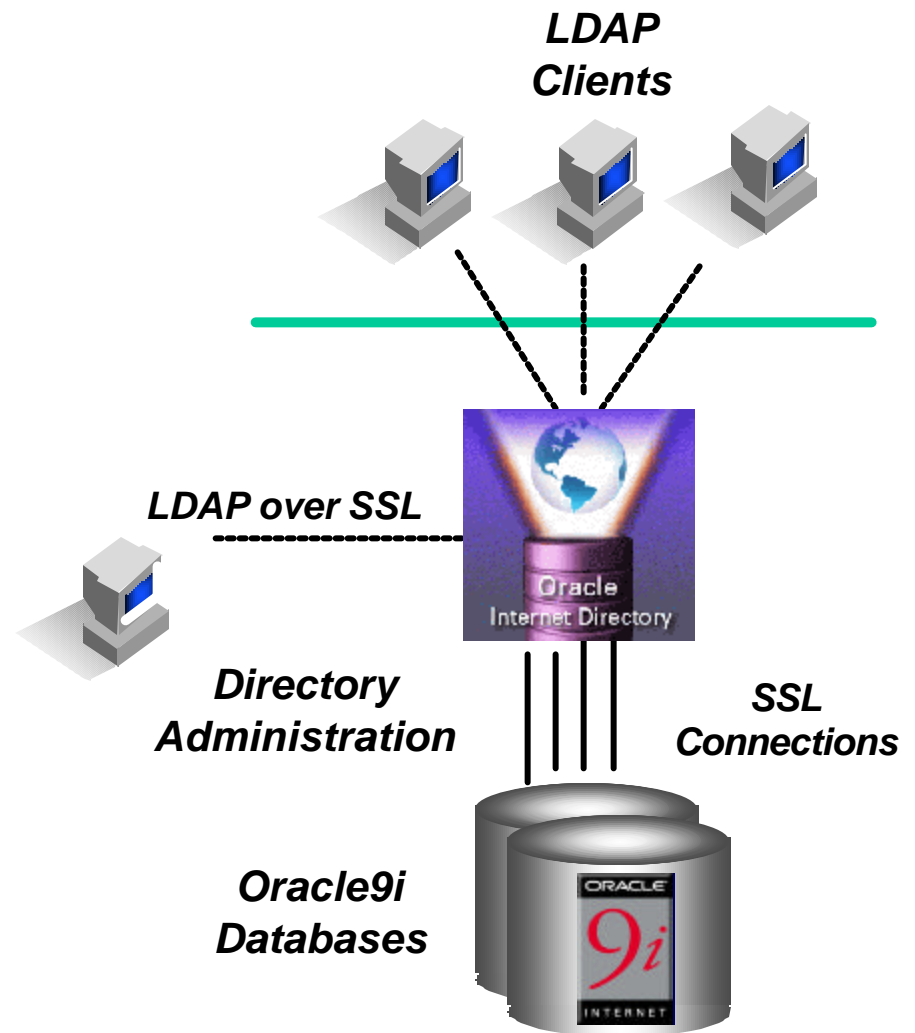
Oracle Advanced Security: Enhanced Authentication for Application Security

- ✍ Strong authentication (client-server, server-server)
- ✍ Single sign-on Kerberos, RADIUS
- ✍ Token Card, Smart Card
- ✍ Public Key Infrastructure

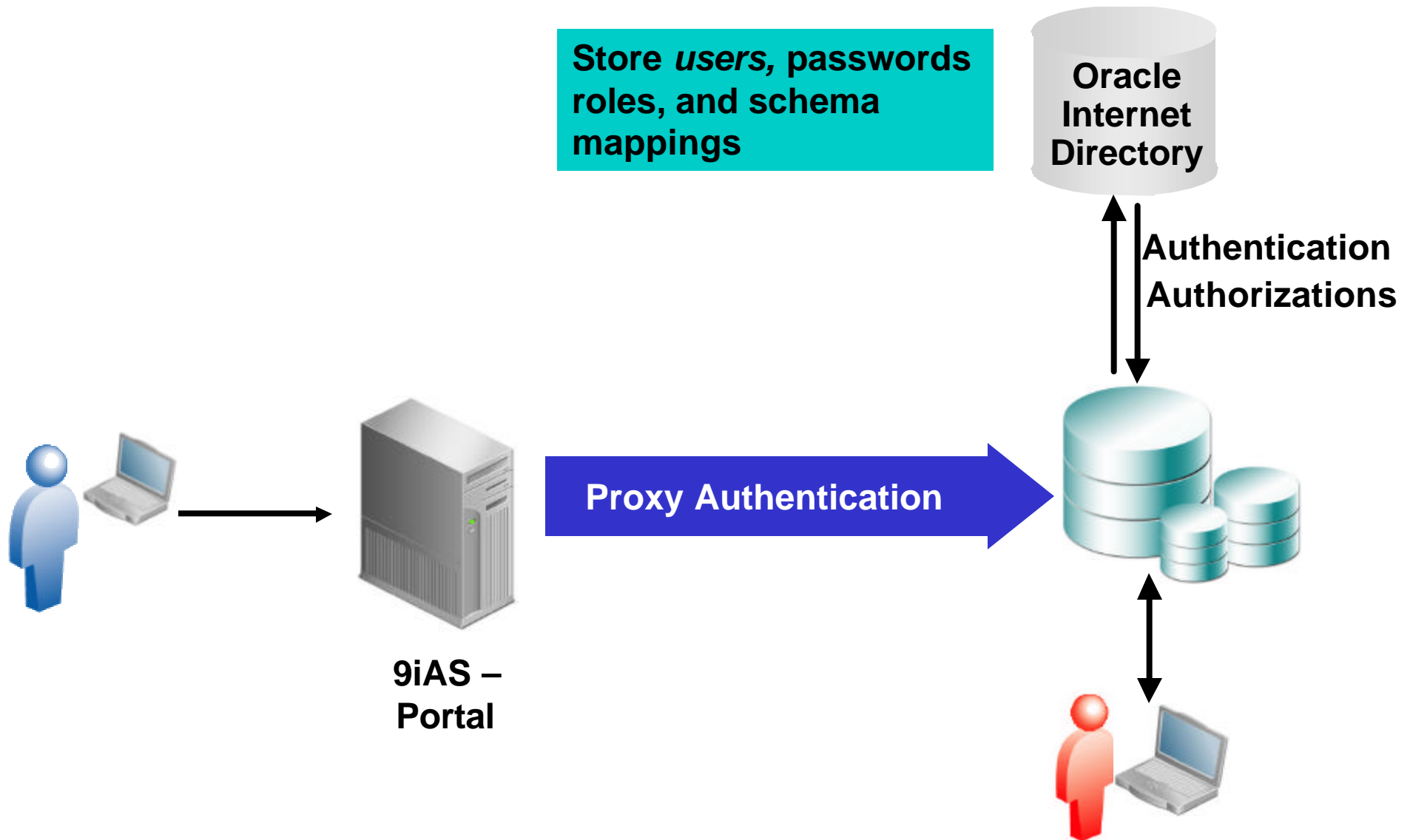


Oracle Internet Directory for Centralized Administration

- ✍ Central repository for user and privilege management
- ✍ Supports LDAP standard protocol and schema
- ✍ Add and delete users in a central location
- ✍ **Enables single sign-on**



Centralized Identity Management – Enterprise Users



Oracle's Web SSO Solution

Unified Web SSO Solution

- For web-based products
- For internal websites
- For hosted websites

Supports Applications and Tools

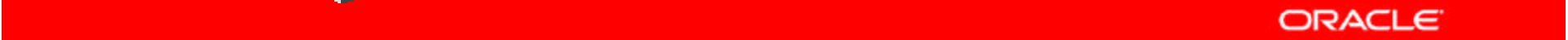
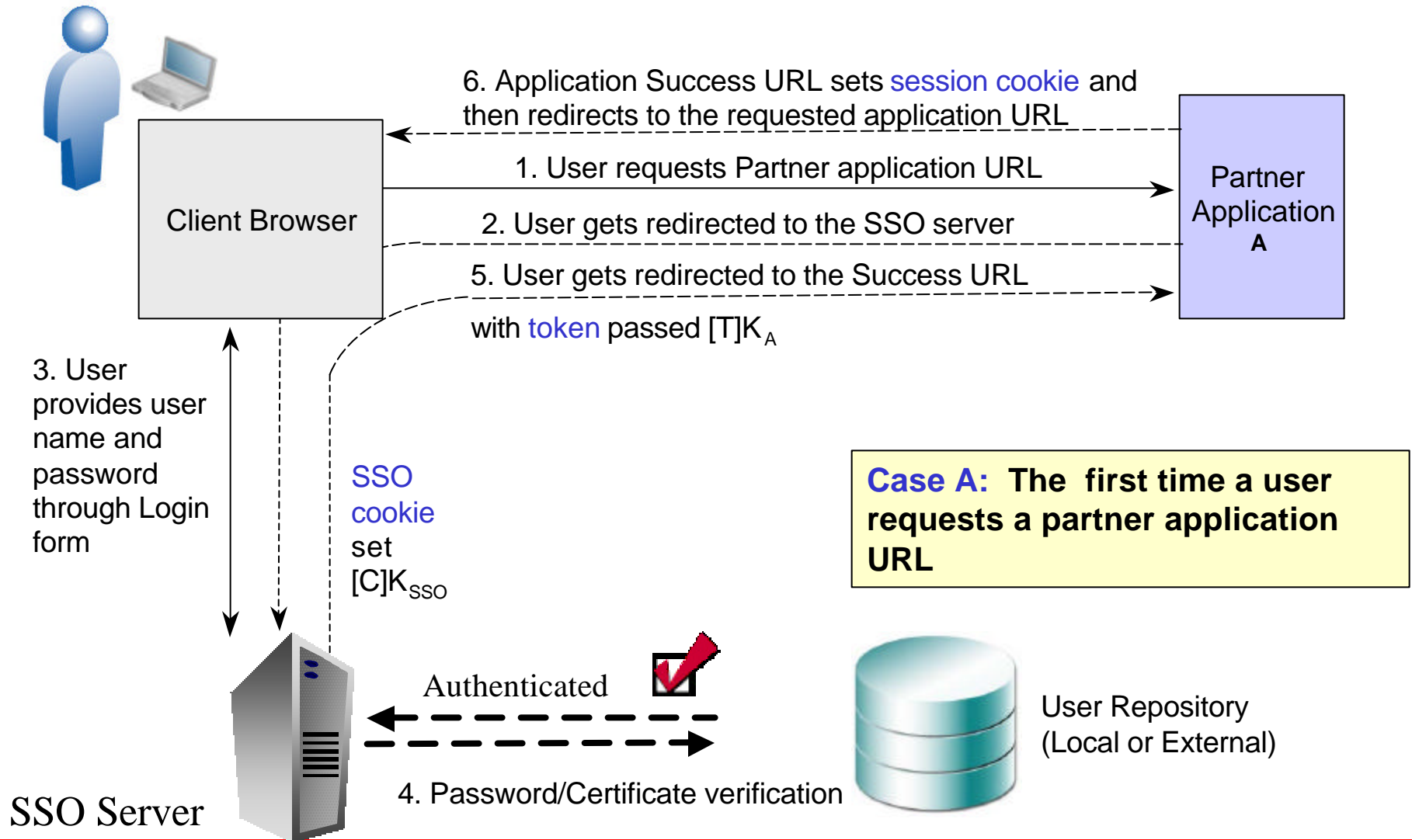
- Joint ST/Oracle Portal/Apps project
- Bundled with Oracle Portal release
- Central login server part of iAS
- SDK for SSO enabling applications

Applications in the SSO framework

- Partner applications
 - Accept authentication by login server
 - Modified to work in SSO framework
 - E.g. Oracle Portal (WebDB), Oracle Applications
- External applications
 - Not modified to work in SSO framework
 - Maintain their own username/password authentication
 - E.g. My Yahoo!, Hotmail

Authentication Flow

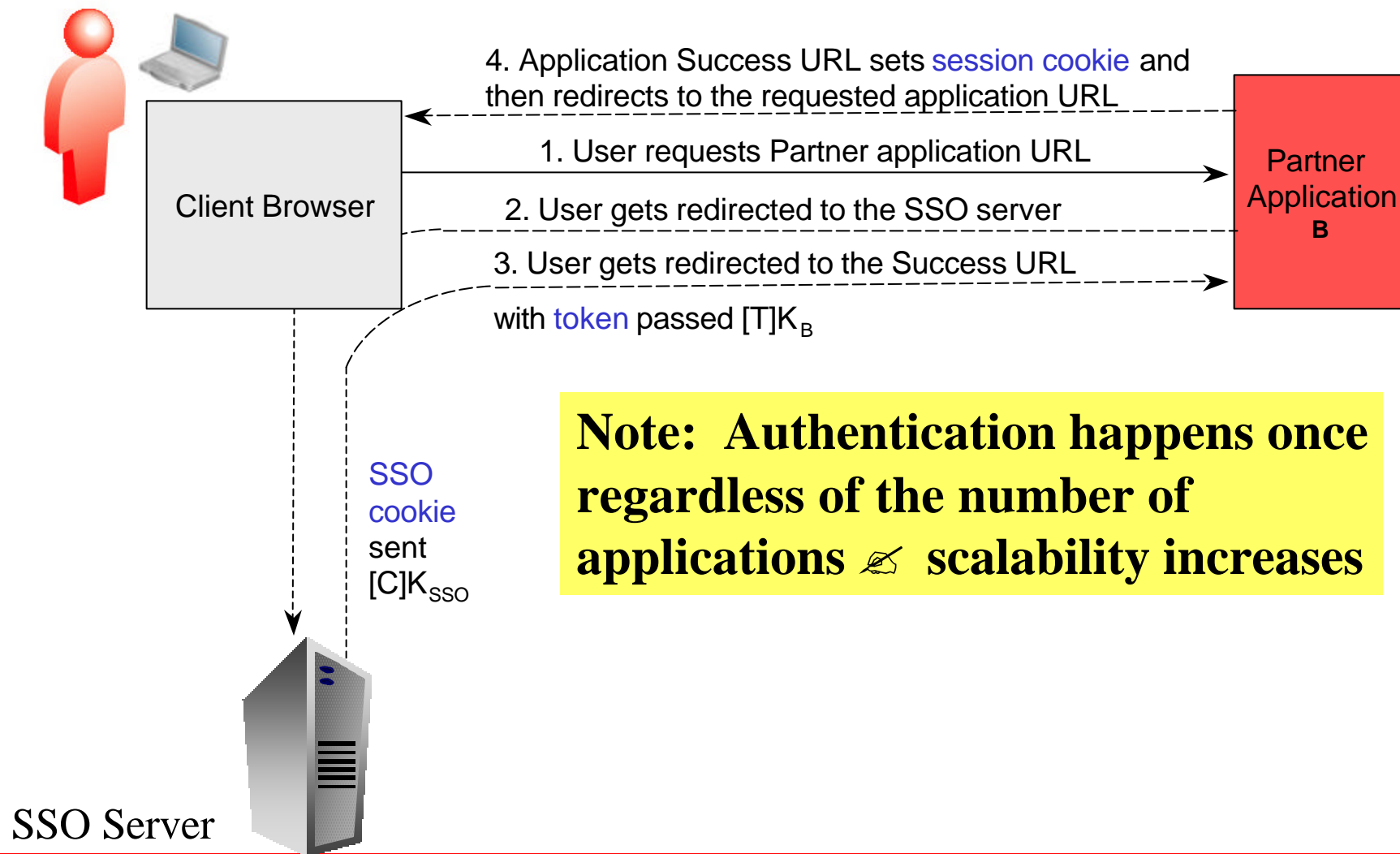
http://server.domain:port/partnerApplicationA



Authentication Flow

http://server.domain:port/partnerApplicationB

Case B: User is already authenticated but goes to another partner application



Note: Authentication happens once regardless of the number of applications ✍ scalability increases

Definition of Terms Used

SSO Cookie

- Contains the identification of the authenticated user
- Accessible only to the Single Sign-On server
- HTTPS is highly recommended for security

Token (parameter to the Success URL)

- Contains SSO user identification

Application Session Cookie

- Managed by applications to keep track of authenticated sessions
- Can have different requirements than SSO cookie

SSO Cookie

- ✎ Identifies the user to the login server
 - Eliminates the need to re-enter credentials -SSO
- ✎ Secure
 - Encrypted and Integrity checked
 - In memory - not saved to cookies.txt
 - Single machine - NOT a domain cookie
 - Has to be “unforgable” and non-replayable
- ✎ Expires after certain amount of time
- ✎ Checks for IP address

Secure transfer of User Identity

Token encrypted with shared key for the partner

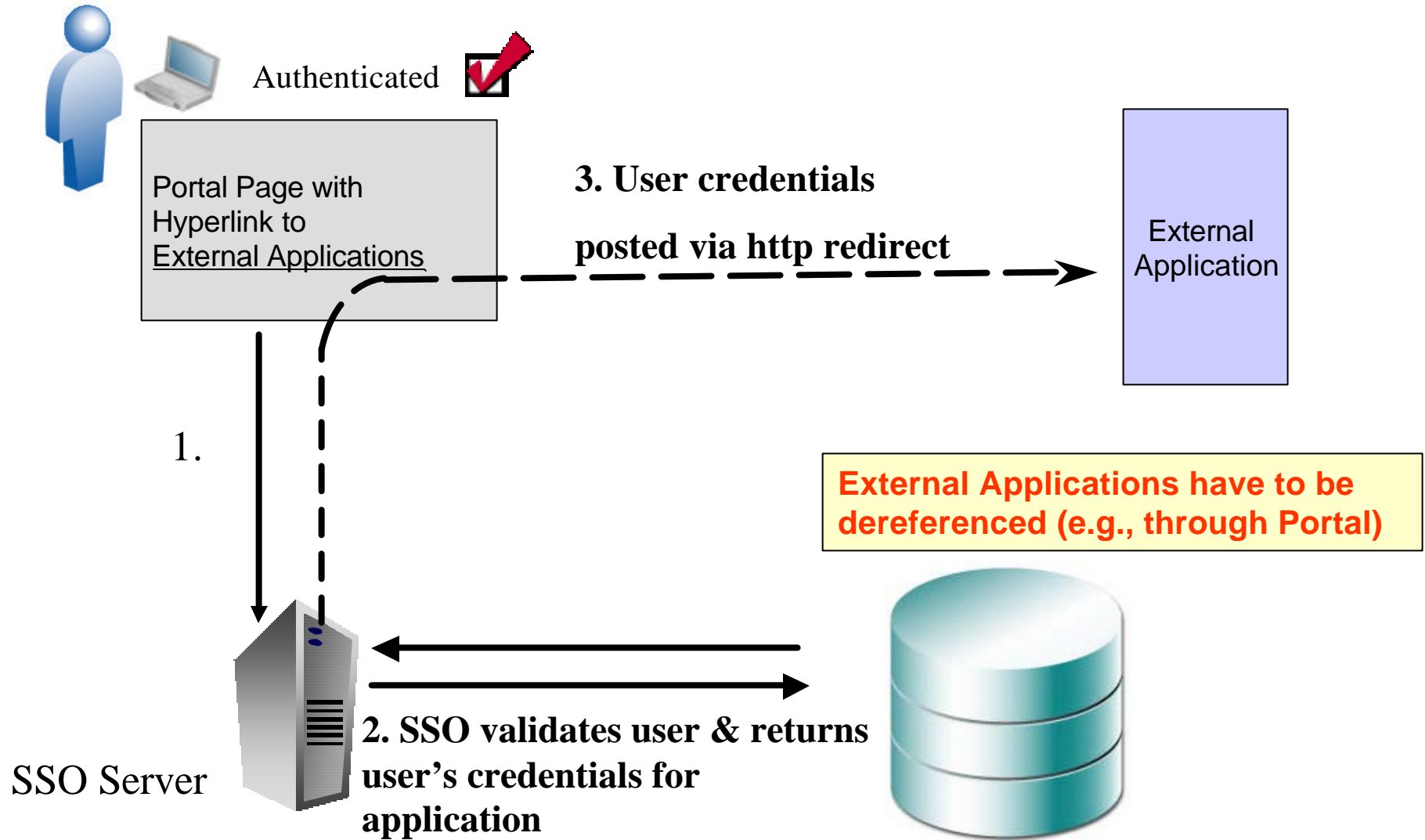
- ✍ Shared symmetric key between login server and partner applications
- ✍ Knowledge of key establishes trust/authentication
- ✍ Different key for each partner

External Application support

- ✎ External applications require username/password
- ✎ Login form constructed and submitted to application via the browser
- ✎ External Username and Password stored encrypted in login server store
- ✎ SSO Cookie checked before access allowed

SSO Support for External Applications

http://server.domain:port/pls/portal/



User authentication

- ✍ Using an external repository
 - Stored in Oracle's OiD (LDAP)
 - User Defined - Specification can be implemented via Password Verifier API
 - Third-Party implementations have included Netegrity's Siteminder and MS AD

Other Useful Features

Single Sign-Off

- Terminate not only SSO session, but all partner sessions too e.g., end of the day logout

Paranoid Application Support

- Forced re-authentication for critical operations and applications with short(er) session lifetimes

Global Inactivity Detection

- Failure to use any partner application in specified time forces re-authentication

PKI enabled

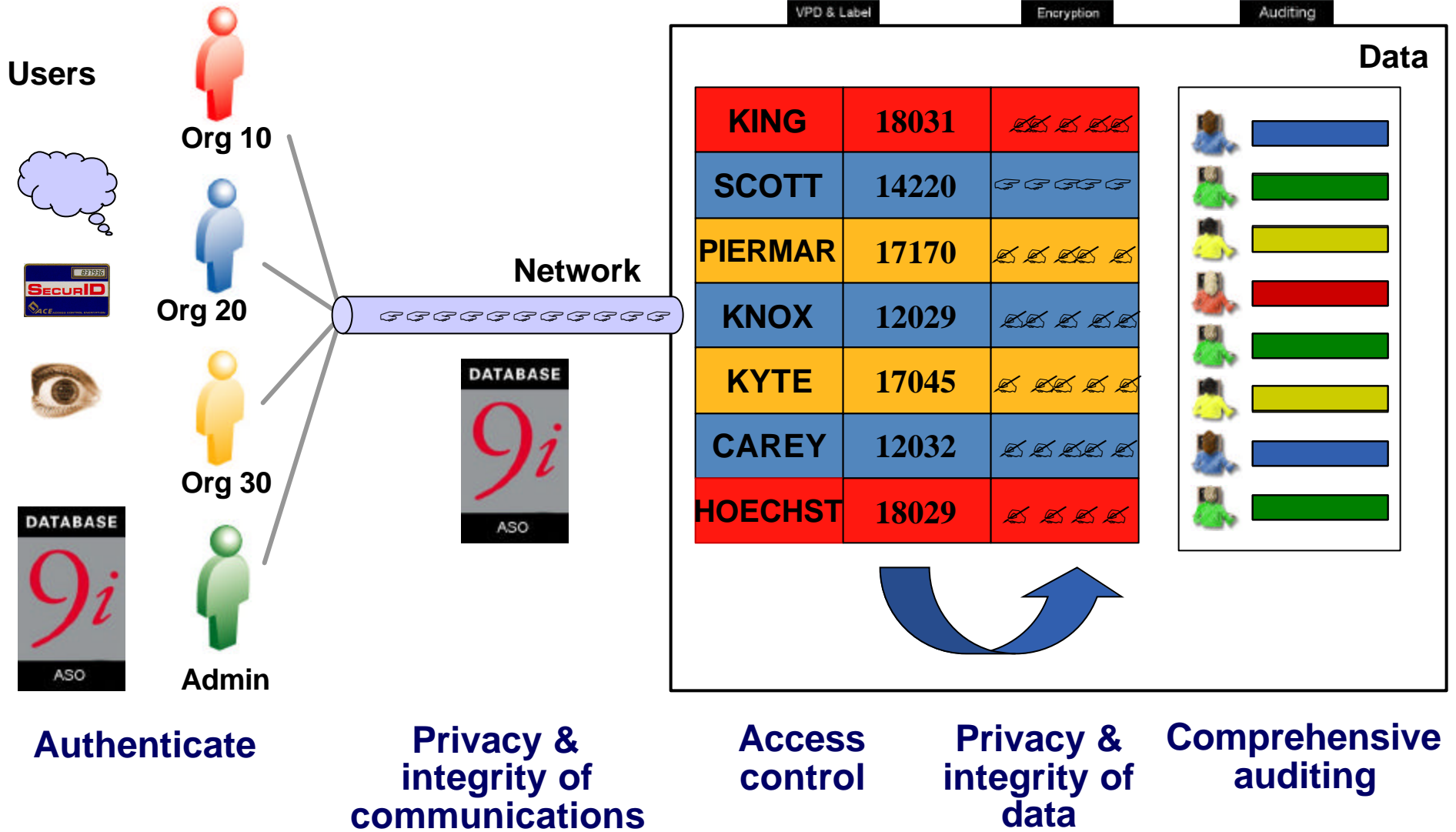
- ✍ Supports X.509 client authentication
- ✍ All partner applications automatically PKI-enabled
- ✍ SSL authentication done once, and cookies used for applications for performance
- ✍ SSL Session Cache in HTTP server
- ✍ Protect files, directories based on X.509 data



Oracle 9iAS Single Sign-On

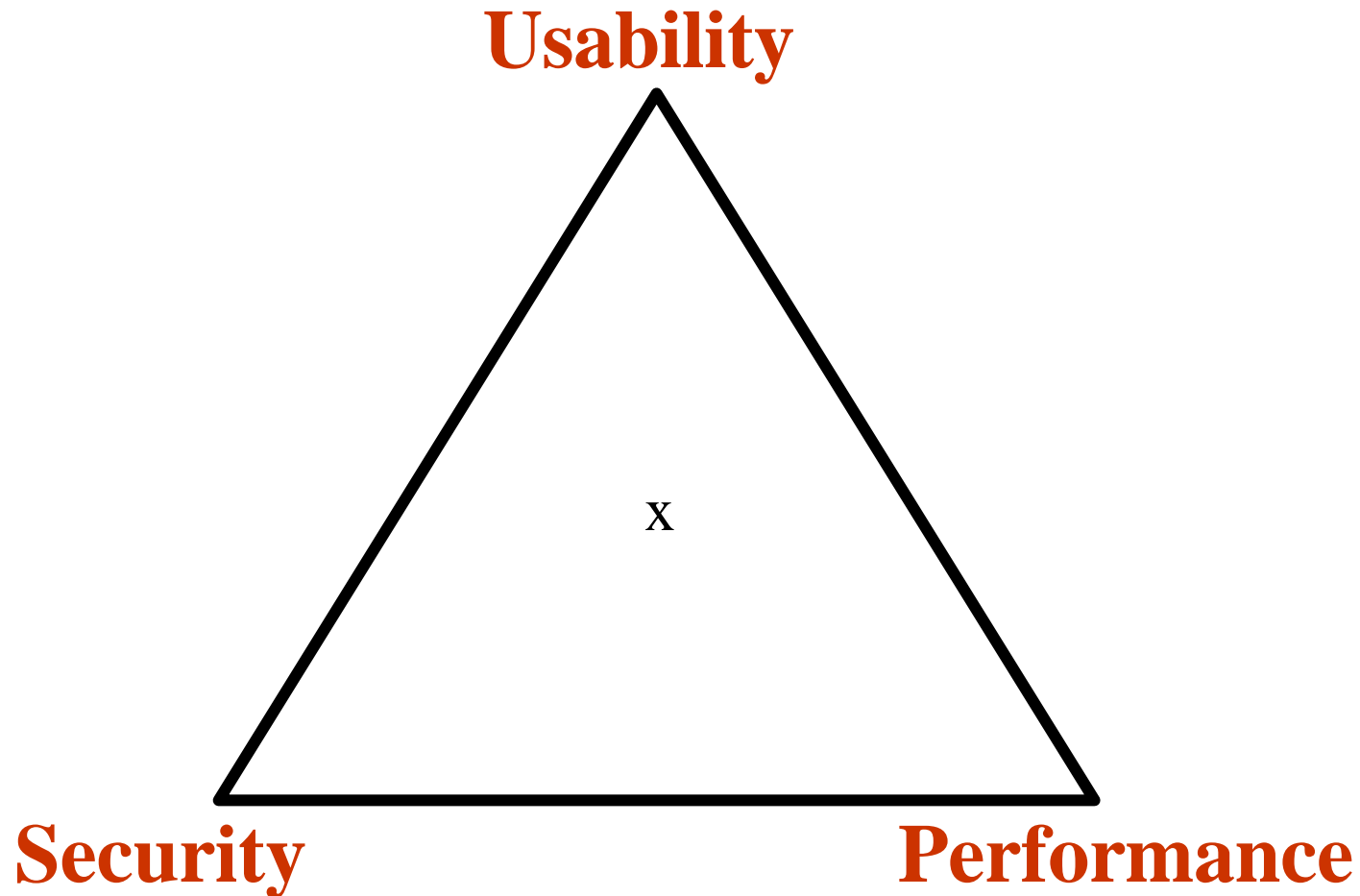
- ✍ Partner and External applications
 - Oracle applications e.g., Oracle Portal
 - Oracle Tools
- ✍ Single Sign-Off
- ✍ Paranoid Application Support
- ✍ Global inactivity detection
- ✍ Supports PKI
- ✍ Three-tier enabled
- ✍ Supports JAAS
- ✍ Delegated administrative services support
- ✍ Encrypted session Cookies for performance
- ✍ Centralized login server
- ✍ Third party repository for credentials
- ✍ Extensible authentication
- ✍ Password rules
 - Password expiry
 - Minimum length, numeric character
- ✍ Auditing
 - Account lockout

Information Assurance



Balancing Requirements

Need flexibility to adjust to current situation



Holistic Approach To Security Reduces Risk

SECURITY

Technology

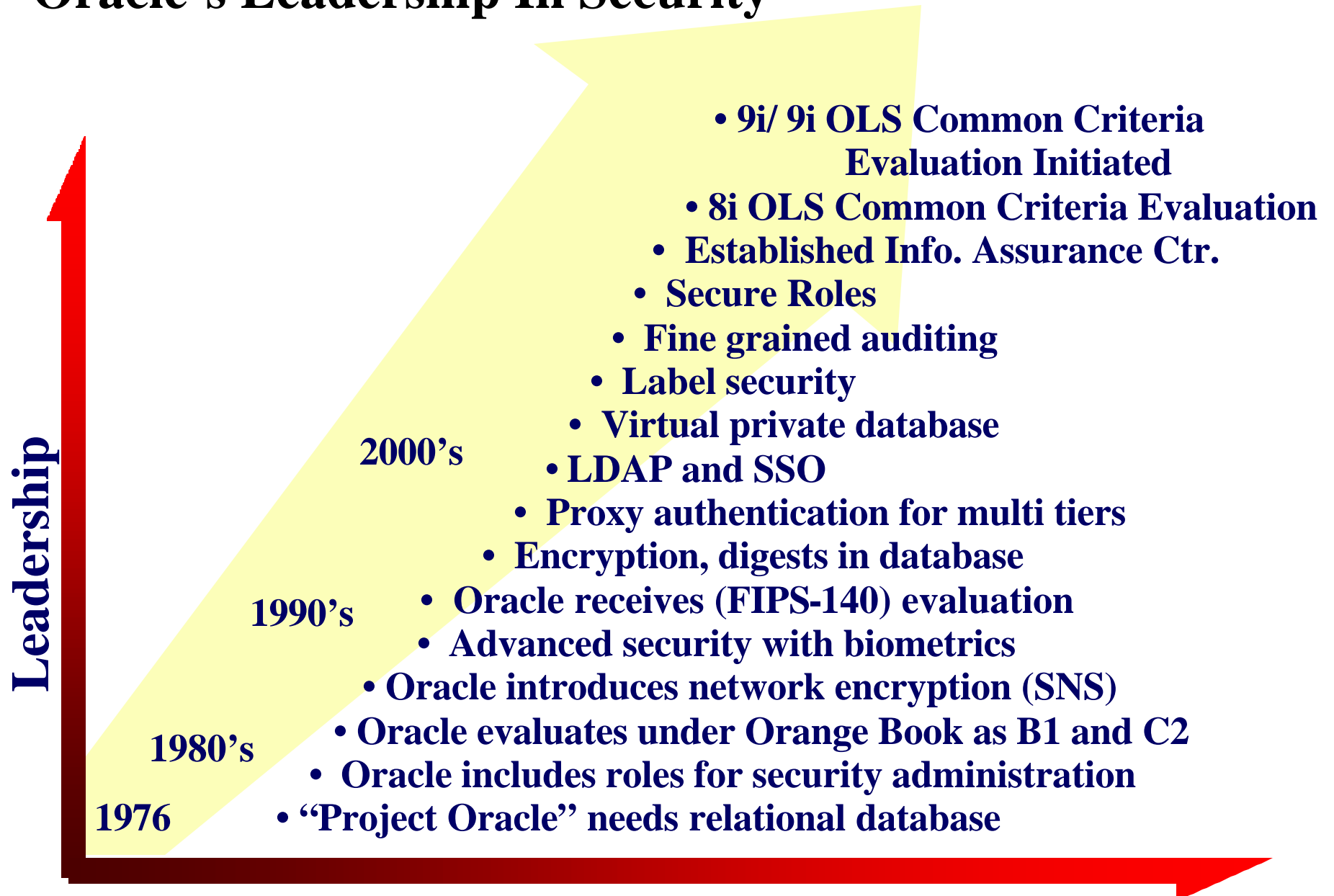
Methodology -
Process

Assurance

ORACLE

© 2003 David Knox, Oracle Corporation

Oracle's Leadership In Security



How do you know it's secure?

Be Aware: DoD 8500.1 (NSTISSP #11), HIPAA

- ✍ Technology + Evaluations = Assurance
 - *Independent* (third party) evaluation and analysis against objective criteria
 - Evaluations: a structured design process, detailed documentation, and extensive testing
- ✍ Assurance provides confidence that vendor security claims are not mere marketing hype
- ✍ Evaluations improve product security in three key ways:
 1. Better development process
 2. Evaluators find security vulnerabilities
 3. Creation of a “culture of security”



**QUESTIONS
ANSWERS**