

Real-life Database Security Mistakes

Stephen Kost
Integrigy Corporation
Session # 383

Introduction

- **Stephen Kost**

- Chief Technology Officer of Integrigy Corporation
- 12 years experience with Oracle Database as database administrator, architect, and security researcher
- Found more than 50 security bugs fixed in CPUs

- **Integrigy Corporation**

- Dedicated to Oracle Security
- Services – Oracle Security Assessments
- Products – AppSentry and AppDefend

Agenda

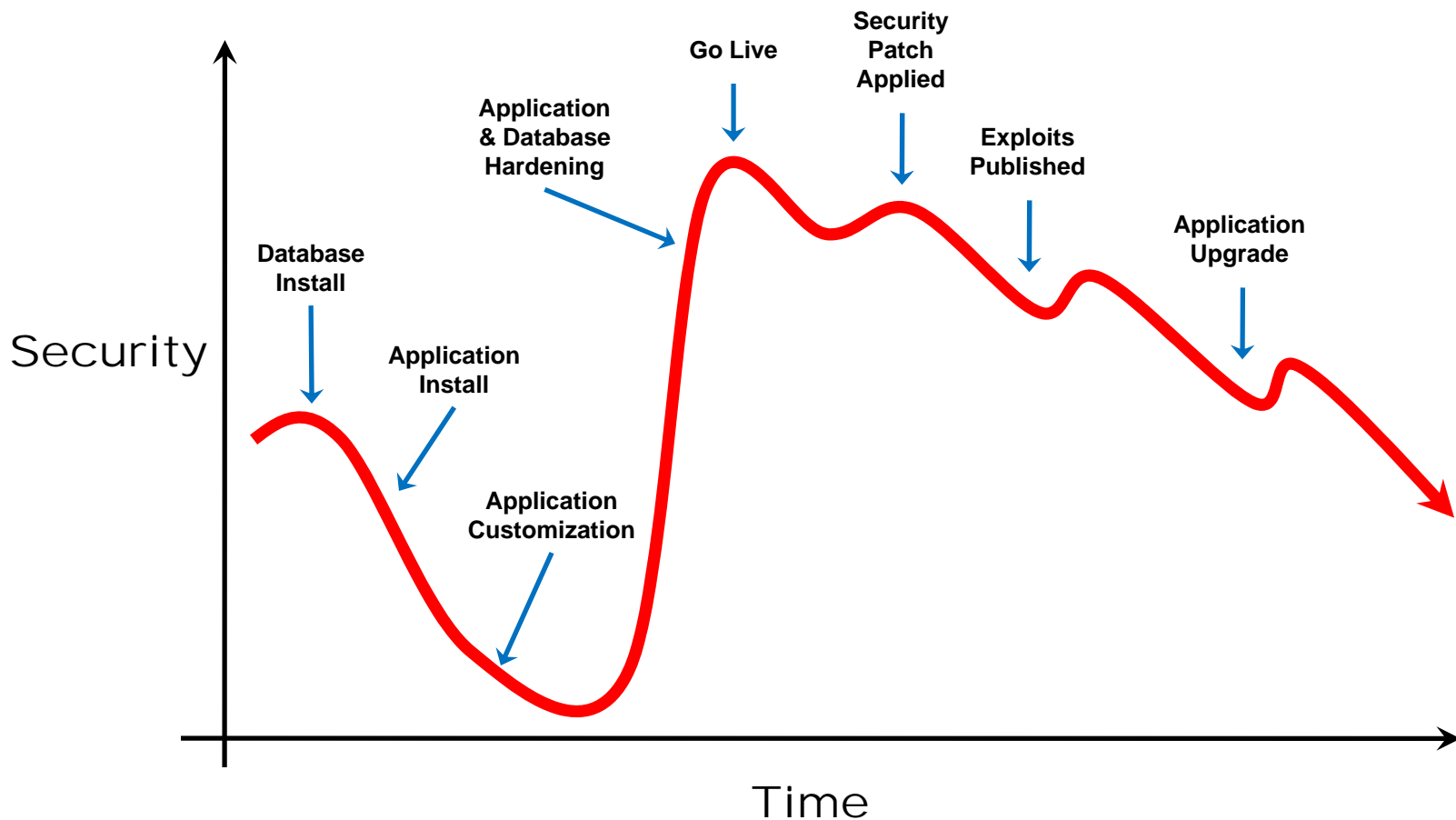
- Root Cause Analysis
- Default Passwords
- System Privileges
- Roles
- Invoker and Definer Rights
- Oracle Critical Patch Updates
- Questions

Caveats

- The information contained here is generalized across Oracle Database versions platforms and may not be applicable for a specific version or platform
- The scenarios and situations described are from actual customers, although the customers processes and actions may not always be considered best practice

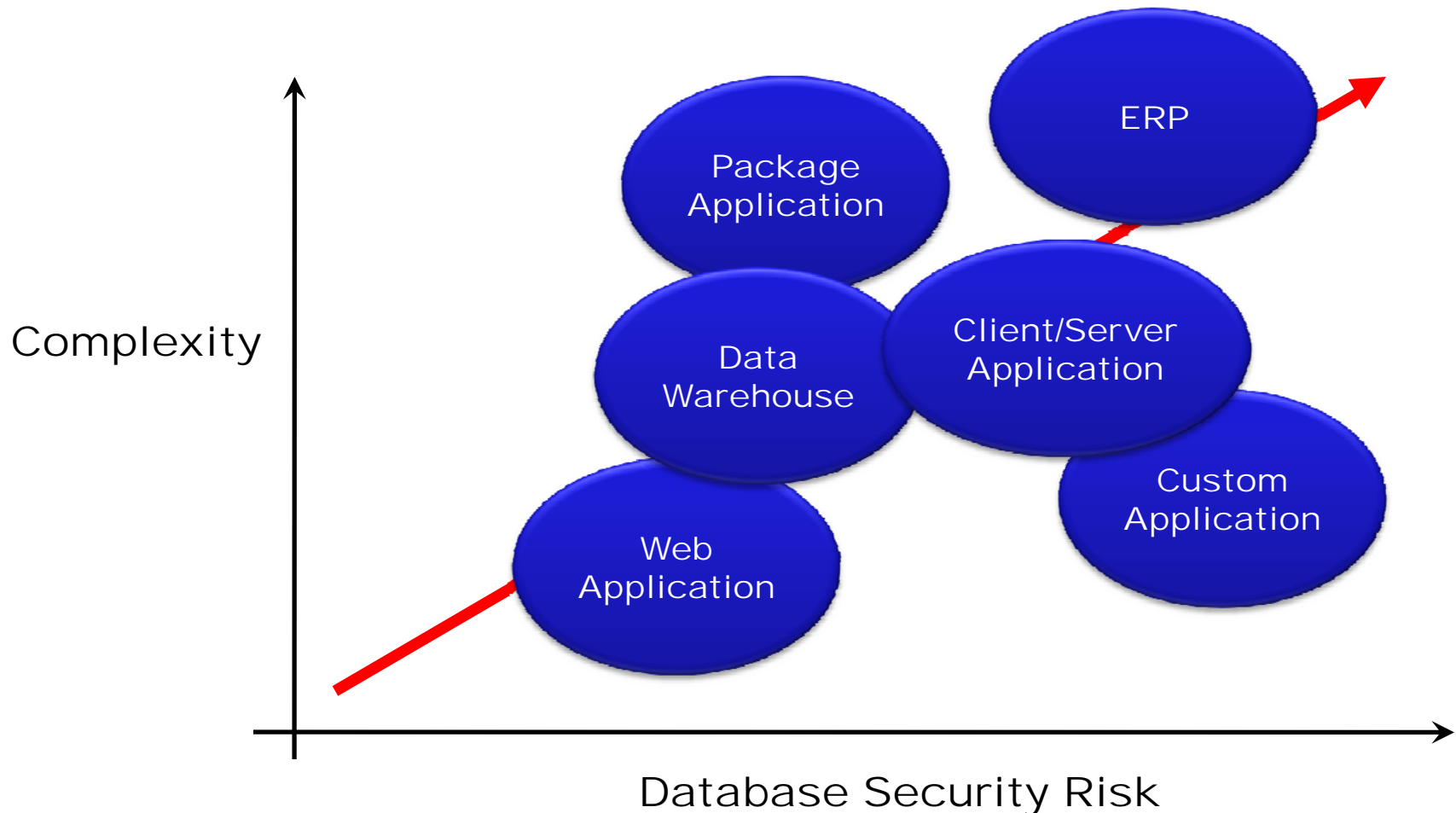
Database Security Decay

Database security decays over time due to complexity, usage, application changes, upgrades, published security exploits, etc.



Complexity and Security are Opposed

The more complex a database and application environment are, the less secure the entire environment will be.



#1

Default Passwords and More Default Passwords

1a

The completely and totally
obvious answer –

You never changed the
default passwords!

1b – DB Creation DBCA vs. Script

- Using Database Creation Assistance (DBCA), no default passwords are set and/or the accounts are locked/expired
- DBCA can be used to generate a create script in order to recreate the database
 - Depending on version, default passwords may be set
- Manual database creation using custom scripts may set default database passwords

1c – “How did that get reset!?”

- Default database account passwords like CTXSYS and OUTLN often “magically” get reset to default values
 - As part of application installation or maintenance (i.e., Oracle Applications)
 - As part of database maintenance (see April 2008 CPU DB13 for DBMS_STATS and OUTLN)

1d – “I checked all the passwords!”

- **Oracle Password Scanner** only checks 683 known accounts for a single password
 - Oracle Metalink Note ID 361482.1
 - Included as part of 11g – dba_users_with_defpwd
 - Oracle’s password list has a number of important omissions/errors and does not include many common application accounts
- **Use a tool that checks all password hashes for all accounts against common dictionary words**
 - See <http://www.petefinnigan.com/tools.htm> for a list of password checking tools
- Use database auditing with “AUDIT USER;” to capture new database accounts and password changes

Oracle Database Passwords

- Standard Oracle passwords are a limited character set
 - A...Z, 0...9, and _ # \$
 - Passwords must start with an alpha character
 - More complex passwords can be set by enclosing the password in double quotes, however, many programs do not support these types of passwords
- Oracle Password algorithm is published on the Internet
 - Algorithm uses two cycles of DES encryption with the Username to produce a one-way hash of the password
 - Hash is unique to the username, but common across all versions and platforms of the Oracle database
 - APPS/APPS is always D728438E8A5925E0 in every database

Cracking Database Passwords

- A number of efficient and quick password cracking programs exist for Oracle
 - Speed is around 1 million passwords per second
 - Speed improvements up to 100 times due to technical advances
 - Only the hash and username are required
 - Estimated time to crack a password of x length –

Length	Permutations	Time
1	26 (26)	0 seconds
2	1,040 (26 x 39)	0 seconds
3	40,586 (26 x 39 x 39)	0 seconds
4	1,582,880	1.5 seconds
5	61,732,346	2 minute
6	2,407,561,520	40 minutes
7	93,894,899,306	1 day
8	3,661,901,072,960	42 days
9	142,814,141,845,466	1,600 days
10	5,569,751,531,973,200	64,000 days

#2

System Privileges

2a

Create Public Synonym

**“Probably not the best, but
somewhat innocent, right?”**

CREATE object privilege

equals

CREATE AND REPLACE

DBMS_SQL, DBMS_OBFUSCATION_TOOLKIT, DBMS_CRYPTO, ...

Other System Privileges

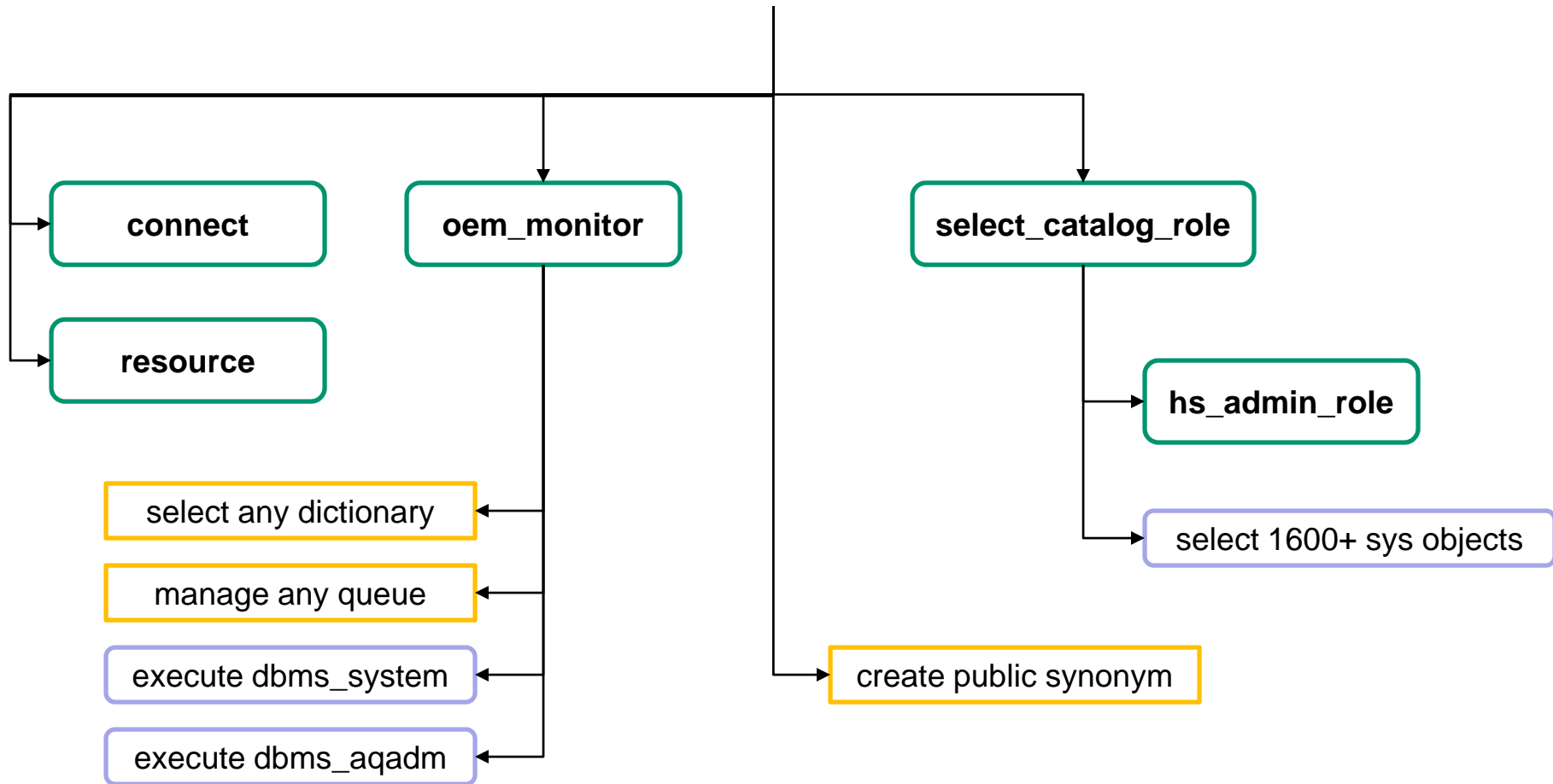
- CREATE ANY <object>
- CREATE PUBLIC DATABASE LINK
- EXECUTE ANY PROCEDURE
- CREATE ANY JOB
- CREATE EXTERNAL JOB

#3

Roles

STEVE (User) ← OLAP_USER (Role)

STEVE (User) ← OLAP_USER (Role)



Roles

- Roles can be deceptive especially when dealing with the hierarchical granting of roles
 - No standard way to determine all the privileges assigned to a specific role through the role hierarchy
 - See www.petefinnigan.com for find_all_privs.sql to see hierarchy of roles and privileges

#4

Invoker and Definer Rights

Dear PL/SQL developers,

Definer Rights is Default

By default, stored procedures and SQL methods execute with the privileges of their **owner**, not their current user.

```
CREATE TYPE Num AUTHID CURRENT_USER AS OBJECT (x NUMBER,  
    STATIC PROCEDURE new_num (  
        n NUMBER, schema_name VARCHAR2, table_name VARCHAR2)  
    );
```

```
CREATE TYPE BODY Num AS  
    STATIC PROCEDURE new_num (  
        n NUMBER, schema_name VARCHAR2, table_name VARCHAR2) IS  
        sql_stmt VARCHAR2(200);  
BEGIN  
    sql_stmt := 'INSERT INTO ' || schema_name || '.'  
        || table_name || ' VALUES (blake.Num(:1))';  
    EXECUTE IMMEDIATE sql_stmt USING n;  
END;  
END;
```

```
GRANT EXECUTE ON Num TO scott;
```

Definer and Invoker Rights

- Definer/Invoker rights impact packages, procedures, functions, java, types, views (definer), and triggers (definer)
- Must understand access to and execution of packages/procedures/functions
 - Easier for developers use definer rights since no need for specific grants to objects in the code schema
 - For every code object creation and grants on the object, need to think through the impact
 - Should have a development standard if definer or invoker rights should always be used
 - Impact is mostly for databases where end-users have direct access to code objects through direct SQL*Net connections or ad-hoc query tools

#5

Critical Patch Updates

5a – CPU Forgotten Steps

- CPU is two parts –
 - OPatch to update files in the ORACLE_HOME
 - catcpu.sql to update database objects
- Some CPUs require additional manual steps –
 - January 2008 CPU requires all views to be recompiled due view/SQL compiler bugs in July 2007 CPU
- Query `SYS.REGISTRY$HISTORY` to verify CPU row is present
 - An indicator CPU patch was successfully applied

5b – CPU Database Upgrades

- Scenario
 - Latest CPU patch is applied (January 2008)
 - Upgrade database to new version or patchset (10.2.0.2 to 10.2.0.3)
- Do I have to reapply the latest CPU after the database upgrade?
 - Yes, you must apply 10.2.0.3 January 2008 patch

5c – CPU Oracle Home vs. DB

- Scenario
 - Latest CPU patch is applied (January 2008) to Oracle Home
 - Install a new database from the patched Oracle Home
- Do I have to run the catcpu.sql from the January 2008 CPU?
 - Yes, since some of the SQL statements in the catcpu.sql do not exist as files in the Oracle Home
 - catcpu.sql does perform some drops and grants

5d – CPU Oracle Home vs. DBCA

- Scenario
 - Latest CPU patch is applied (January 2008) to Oracle Home
 - Install a new database from the patched Oracle Home using DBCA and a seeded database
- Do I have to run the catcpu.sql from the January 2008 CPU?
 - Yes, since the seeded database files are pre-loaded with packages and none of the vulnerable packages would be updated without running catcpu.sql

Stephen Kost
Chief Technology Officer
Integrigy Corporation

E-mail: skost@integrigy.com
Phone: 312-961-0215

Integrigy Corporation
P.O. Box 81545
Chicago, Illinois 60681
888/542-4802

Website: www.integrigy.com
Sales: sales@integrigy.com
Development: development@integrigy.com
Support: support@integrigy.com
Security Alerts: alerts@integrigy.com

Questions?