Cecelia_Walton_GSEC.doc
Certification and Assignment: GSEC Version 1.4b
Title of Paper: MS SQLSNAKE WORM

## INTRODUCTION:

This paper will discuss the MS SQLSNAKE worm to include the actual connections that were captured from a monitored network utilizing commonly accessible Intrusion Detection Software. The points presented will include the steps a hostile uses to implant the worm on the target system, then, how the hostile can access anything on that system that uses guest name and password for authentication. This paper will break the worm and exploit down by sections and explain whats happening from start to finish. There appears to be no known originator or group that has claimed responsibility for this worm. Additionally, this paper will provide history, vendor recommendations and possible motivations of the intruder. The IPs in this paper have been modified so no correlation can be made between these and actual monitored systems. The network we were responsible for monitoring was violently attacked by the worm during the period of June 02 thru July 02.

## BACKGROUND:

The SQLSNAKE worm takes advantage of the xp_cmdshell exploit in the Microsoft SQL server by building an Active X object containing the commands to be run via xp_cmdshell, and then passes them to the non-password protected default “system administrator” SQL Server administrator account. Operating Systems affected included Microsoft Windows 95, Microsoft Windows 98, Microsoft Windows NT, Microsoft Windows 2000, Microsoft Windows XP and Microsoft Windows ME. SQL Server prior to SQL Server 2000 by default allowed multi-mode authentication. A user could authenticate to the SQL Server using their Windows credentials on their SQL Server account. It’s highly important to note that the default “system administrator” account on these operating systems cannot be disabled and there is no password installed at installation [1]. The SQLSNAKE worm is actually a variant of a similar incident that occurred in November 2001 and is further explained in CERT Incident Note IN-2001-13.HTML. In the previous incident the “KAITEN” Malicious Code compromised systems similar to the SQLSNAKE worm through a “null” password set for the system administrator. Both attacks opened with intensive scans to detect open SQL Servers listening on port 1433/TCP. Additionally both attacks used the extended stored procedure “xp_cmdshell” to execute commands on the target system’s “system administrator “account. There appears to be only three significant modifications between “KAITEN” and the “SQLSNAKE worm”.

1. “KAITEN” employed FTP to install its payload instead of the copy command utilized by the SQLSNAKE.

2. “KAITEN” notified the hostile through an Internet Relay Chat Channel that a particular target had been compromised. While the SQLSNAKE worm utilizes e-mail.

3. “KAITEN” installed a more destruction payload on the compromised system. The payload consisted of denial-of-service tools and scanning tools. The SQLSNAKE worm’s payload was less harmful than “KAITEN”. It installed files that were utilized to infect additional SQL servers and not other targets as “KAITEN” was designed for [11].

The primary target of the SQLSNAKE worm was Microsoft SQL Servers, other targets included systems running Microsoft Data Engine 1.0(MSDE 1.0), Microsoft SQL Server desktop engine (MSDE 2000) and Tumbleweed’s Secure Mail (MMS) versions 4.3, 4.5 and 4.6.

The worm scans across the internet hoping to locate systems that are listening on port 1433/TCP. After a potential victim is identified, the worm tries to login using the Microsoft built-in system administrator account with a null password. If the hostile is successful, the worm that infected the target system then exports the password hacker and other configuration information to a remote email address (ixltd@postone.com)[11].

SQLSNAKE WORM ANATOMY:

The SQLSNAKE worm has a multi threaded scanner which generates a huge amount of network traffic while actually engaged. This embedded scanner probes the target network looking for any and all systems that have TCP port 1433 open. When the worm locates a system with port 1433 open, it attempts to login with a null password. If the hostile is able to do so, it initiates its start-up script by using the xp_cmdshell function. This function is defined as “Execute a given command string as an operating – system command shell and return any output as rows of text and grants nonadministrative users permissions to execute xp_cmdshell” [9]. This is used to execute system commands through SQL queries. This command lets the SQLSNAKE worm install its payload into the %WinDir%\system32, except for a file called service.exe, which is installed in %WinDir%\system32/drivers. When the worm has exploited the system, it executes several JAVASCRIPTS, which activate the built-in guest account and assigns that account to the Local Administrator and Domain Administrator groups. The installed JAVASCRIPTS are also used to collect the network configuration and database configuration. Additionally, the worm executes pwdump2 to obtain the system’s password hashes, which are then systematically emailed along with other accessed information to a remote e-mail account (ixltd@postone.com). PWDUMP2 is defined as “an application which dumps the password hashes(OWFS) from NT’s SAM (Security Account Manager) database, whether or not SYSKEY is enabled on the system. The output follows the same format as the original pwdump(by Jeremy Allison) and can be used as input to

OPHTCRACK, or used with SAMBA [10] .”  After the worm transmits the e-mail it deactivates the guest account and establishes a random four character password to the system administrator account. From there the worm again takes off and continues its probe through the network seeking additional targets to infect.

Some conflicting information exists about the worm's scanning pattern. Mcafee claims that the worm only attacks systems in the following address spaces 10/8 (10.0.0.0- 10.255.255.254), 127/8 (127.0.0.0-127.255.255.254), 172/8 (172.0.0.0 - 172.255.255.255.254), and 192/8 (192.0.0.0 - 192.255.255.255.0). Portions of these address ranges are considered private addresses that are commonly found behind firewalls using Network Address Translation (NAT). ISS asserts that the worm is able to switch between scanning internal NAT addresses and external ones. A SANS analysis of the sqlprocess.js script which controls scanning, shows that external scanning is part of the worm's activities [3] .

The "A" variant appears to scan all but the Class "A" networks 10/8, 127/8, 172/8 and 192/8. The "B" variant scans the public and private ranges. Since the worm uses a mixture of JavaScript, binary and batch file coding, it should be very easy for hostile users to alter this worm to scan their favorite address spaces. In both variants, after the worm has successfully infected ten other hosts, it attempts to remove all traces of itself from the infected machine. This includes removing its files, registry entries, and removing the compromised Guest or sqlagentcmdexec accounts from the Administrators and Domain Admins groups.

SNAKE WORM IMPLEMENTATION PROCESS:

The next section of this paper will cover an actual SQL Snake worm attack. This particular recovered attack was captured by using commercially available Intrusion Detection Software. There were a total of six connections established and will be presented with the connection first and the explanation second.

Connection 1:
(Name of System)1sa00000197’H
G4096 Microsoft Windo128.xxx.xxx.xxxOLEDB
EXPLANATION: The first part of the connection is where the hostile has sent out a scan (the worm files are located and hidden in the %WinDir%\system32 directory) for other targets from that list by sending TCP SYN Packet on port 1433. This connection is where Microsoft Windows returns the target IP address to the system.

Connection 2:
Ph Vdh  (Name of System)1saMicrosoft Windows Scripting Host128.xxx.xxx.xxxLEDBNTLMSSP’
(Name of System)Vexec xp_cmdshell
‘echo 128.xxx.xxx.xxx’
EXPLANATION: The second part of the connection is telling the system to echo the IP address back again.

Connection 3:
P!p Vdh (Name of system)1saMicrosoft Windows Scripting Host128.xxx.xxx.xxxOLEDBNTLMSSP'
(Name of System)bexec xp_cmdshell 'net user guest/active:yes'
EXPLANATION: Once connected the worm has the use of xp_cmdshell. This activates the guest account and the worm issues the Net command to connect to the target system through windows file sharing using the guest account. It also changes options for guest to be able to log in as user guest.

Connection 4:
Ph Vdh  (Name of System)1saMicrosoft Windows Scripting Host128.xxx.xxx.xxxOLEDBNTLMSSP'
(Name of System)\exec xp_cmdshell 'net user guest  d5s1e3v4'
EXPLANATION: Changes system administration account password to the same password as the guest account.

Connection 5:
Ph Vdh  (Name of System)1saMicrosoft Windows Scripting Host128.xxx.xxx.xxxOLEDBNTLMSSP'
(Name of System)~exec xp_cmdshell 'net local group administrators guest/add'
EXPLANATION: Gives the guest account administrator privileges. With these administrator privileges the hostile can do things like listing contents of the system directories.

Connection 6:
Ph Vdh  (Name of System)1saMicrosoft Windows Scripting Host128.xxx.xxx.xxxOLEDBNTLMSSP'
(Name of System)vexec xp_cmdshell 'net group "Domain Admins" guest /add'
EXPLANATION: Guest is added to the Domain Admins. The Domain Admins are security groups that are created automatically when you create an Active Directory Domain. The function of these groups normally consists of controlling access to shared resources and to delegate specific domain-wide administrator roles.

CHECKING AND PROTECTING YOUR ASSETS:

"To determine if your host is running SQL Services, press the start button and from the Programs menu, select "MS-DOS prompt".  Next, type the command: netstat –an [3]. The netstat command displays the contents of various network-related data structures. The A argument tells the command to show the state of all sockets and the N argument tells the command to show network addresses as numbers.

SQL listens on TCP port 1433 and this number might be found to be the last four digits in the local address column. This means the system is using the SQL listening port. The other SQL products (MYSQL) runs on 1433 and you need to perform more checking to make sure that MS SQL or MSDE is installed. (SQL

Server-compatible database engine) MSDE enables developers to build desktop and shared database solutions that easily migrate to SQL Server when the solution must scale [12] . You will see it in the Programs folder as a subfolder and will be called “Microsoft SQL Server”. The MSDE SQL engine has many of the functions of the SQL server, however the databases can only go up to 2 GB in size.

The SQL server system administrators should put a password on the system administrator immediately, (It needs to be a hard password to guess). To avoid brute force tactics [3].

BREAKDOWN OF A AND B VARIANT WORM

The A Variant:

The file that the A variant uses is a binary named sqlexec.exe. The binary is a stand alone MS SQL exploit that was pressed into service by the virus writer. The earlier exploit, which called the xp_cmdshell SQL API, is used to execute external commands on the SQL host. The A variant changes the password in the sqlagentcmdexec account, and uses the same random password that the worm chose for the system administrator account. In the A variant , the sqlagentcmdexec account to the groups of the Administrator and the Domain Admins groups is added in place of the guest account used by the B variant.

The variant files that are found in the same locations as the B variant are the JS/Spida.a. The following are the file names: [4].

sqlexec.exe
clemail.exe
sqlprocess.js
sqlinstall.bat
sqldir.js
run.js
timer.dll
samdump.dll
pwdump2.exe

The below files that the B variant worm uses :

Sqlprocess.js is the main payload of the worm and has the IP addresses used by the services.exe scanner. The ipconfig.exe program is executed by the script to get information from the network and then attaches the data to a file named send.txt. Then the script gets the worm script sqldir.js and attaches the SQL host’s database information to send.txt. Then, a program named pwdump2.exe is ran and adds to send.txt the password hashes. The sqlprocess.js will use its own packaged mail program which is named clemail.exe and then will send the file send.txt to an address named ixltd@postone.com [5]. The script will delete the send.txt when the e-mail is removed. Then the services.exe program is scanned for other servers that are vulnerable. That information is added to rdata.txt and the worm uses it to try to increase with the username “system administrator” and a invalid password. The file sqlprocess.js changes a registry setting to make SQL server use the Winsock library instead of the DBNETLIB library. It also activates the NetDDE server, which lets the SQL server be accessed through the DDE protocol. To get this done, the worm adds the below registry entries:

Hkey_LOCAL_MACHINE\System\CurrentControlSet\Services\NetDDE\ImagePat h where ImagePath equals %COMSPEC% /c start netdde && sqlprocess init and HKEY_LOCAL_MACHINE\System\currentControlSet\Services\NetDDE\Start

Start equals 2

The registry key value will be added

HKEY_LOCAL_MACHINE\software\microsoft\mssqlserver\client\connectto\dsgu ery

dsquery equals dbmssocn

sqlexec.js – a JavaScript script used by sqlprocess.js to execute the xp_cmdshell SQL API. The sqlinstall.bat batch file runs under the xp_cmdshell call.

sqldir.js - Gathers table and row information from databases on the infected system.

run.js - This script passes time information to and from timer.dll.

sqlinstall.bat - A batch file that installs the worm and hides its files. According to Symantec, this batch file activates the guest user account, and sets the guest user account password to eight random characters. It then adds the guest account to the Administrators and Domain Admins groups.

This batch file then searches for the presence of Cscript.exe in the System32 folder. If it finds the file, it then checks if it has previously copied the the 32-bit Registry Editory (Regedt32.exe) file to the root folder of the system disk. This is usually C:\. If these conditions exist, the batch file halts. If this is a new installation, it copies most of its files to the System32 folder. The services.exe file is copied into the System32 Drivers subfolder. For many Windows NT and Windows 2000 systems the system folder is located in the C:\WINNT folder, so the spida worm files would be located in the following locations:

C:\WINNT\System32\Drivers\Services.exe
C:\WINNT\System32\Sqlexec.js
C:\WINNT\System32\Clemail.exe
C:\WINNT\System32\Sqlprocess.js
C:\WINNT\System32\Sqlinstall.bat

C:\WINNT\System32\Sqldir.js
C:\WINNT\System32\Run.js
C:\WINNT\System32\Timer.dll
C:\WINNT\System32\Samdump.dll
C:\WINNT\System32\Pwdump2.exe

After the sqlinstall.bat batch files finishes copying these files, it changes the remote SQL administrator (system administrator) password to a string of four random characters. It then causes the remote computer to start the Sqlprocess.js script.

clemail.exe - A worm-installed mail program used to email out the send.txt file. The outgoing e-mail will be addressed to ixltd@postone.com with the subject line of "SystemData-".

services.exe - A port scanner used by the worm to locate other SQL servers using port 1433/tcp. This information is added to the rdata.txt file. This program is able to scan up to 100 addresses simultaneously. It is supposed to scan internal IP addresses before performing external IP address scans. According to Trend Micro and SANS, this is actually a renamed version of FScan by Foundstone.

pwdump2.exe – Uses samdump.dll to gather password hashes.

samdump.dll – Used to capture Windows password hashes.

timer.dll – Contains timing code used by the worm to control its installation and operation.

DEFENSIVE MEASURES THAT AN ORGANIZATION COULD TAKE TO PROTECT THEMSELVES FROM THIS ATTACK [11].

1. Check to see if any SQL services are running on any Windows NT/2000/XP system within their enterprise. If the audit determines that SQL services are not required on some systems, then those services should be disabled.

2. Do any of the systems identified to be running SQL require Microsoft updates to reduce the potential probability of compromise?

3. Does the account "system administrator" have a strong password? A strong password for this type of account should be greater than 14 characters and include capitals and numbers.

4. Does a corporate policy require the "system administrator" account password to be rotated (e.g. every 30 days)?

5. Microsoft recommends that all users use their own account to login to the SQL Server. These accounts should be members of the "sysadmin" group under SQL.

Does a corporate policy require the database administrator to use another account instead of "system administrator" to perform daily maintenance?

PREVENTIVE MEASURES:

The SQLSNAKE worm tries to discover MS SQL Servers listening on TCP port 1433 in order to try to compromise them. A good thing to do is to change the value of this port to something else. This will make it harder for port scanners to find the SQL Server.

All inbound connections to SMB TCP port 139 should be blocked at the edge router to prevent the SQLSnake worm from accessing shared folders on compromised systems [8].

Use the Integrated Mode Security rather than the Mixed Mode Security to benefit from the Windows authentication mechanisms, such as encryption and password aging. If the Mixed Mode Security is required, system administrators must set a strong password on the system administrator account.

Infected SQL Servers start to scan for other targets, network administrators should block outbound traffic to TCP port 1433 unless for specific needs. This will avoid infected servers from their local network compromising other servers [6].

Test the passwords on the SQL server. A tool called sqlbf can be used to audit the strength of the SQL server passwords [7]. Change weak account passwords.

According to sqlsecurity.com, it is strongly recommended to remove the xp_cmdshell, extended stored procedure if not needed. In the case of the SQLSnake worm, this procedure is used to perform system commands with full privileges on the local machine taking advantage of the unprotected (system administrator) account.

Since the WHS (Windows Scripting Host) is used by viruses and worms to execute their scripts, it is advisable to disable or uninstall the WHS if not needed [2].

AFFECTED HOST REMOVAL INSTRUCTIONS:

If your computer becomes infected, the following removal instructions would apply.

N-stalker has released an SQLSnake removal utility that can be used to detect and remove the SQLSnake worm from a compromised server.

The SQLSnake worm can also be removed manually from a compromised MS SQL Servers:

1. Open a command prompt. Click Start>Run, type COMMAND then hit the Enter key.

2. Disable the guest account, in case the worm has not disabled it. To do this type and execute the following at the command prompt:

```
net user guest/active:no
```

3. Remove the guest user from the local administrators and domain administrators group. To do this, type and execute the following at the command prompt:

```
net localgroup administrators guest/delete
net group “Domain Admins” guest/delete
```

4. Remove TIMER.DLL from memory. To do this, type and execute the following at the command prompt:

```
regsvr32/u TIMER.DLL
```

5. Remove the dropped files from your Systems folder. To do this, type and execute the following at the command prompt:

```
attrib –h %SysDir%\drivers\services.exe
attrib –h %SysDir%\sqlexec.js
attrib –h %SysDir%\clemail.exe
attrib –h %SysDir%\sqlprocess.js
attrib –h %SysDir%\sqlinstall.bat
attrib –h %SysDir%\sqldir.js
attrib –h %SysDir%\run.js
attrib –h %SysDir%\timer.dll
attrib –h %SysDir%\samdump.dll
attrib –h %SysDir%\pwdump2.exe
del %SysDir%\drivers\services.exe
del %SysDir%\sqlexec.js
del %SysDir%\clemail.exe
del %SysDir%\sqlprocess.js
del %SysDir%\sqlinstall.bat
del %SysDir%\sqldir.js
del %SysDir%\run.js
del %SysDir%\timer.dll
del %SysDir%\samdump.dll
del %SysDir%\pwdump2.exe
```

SUMMARY:

Once the worm is on your system and it sends the outgoing e-mail to ixtld@postone.com database of the password for your guest account, the hostile has virtually administrator access to your system and can perform any and all functions set up for the administrator level user. For our network incident reports we were reporting the worm as a malicious logic incident and when the machine was accessed as administrator by the hostile, it became a root level compromise.

RESOURCES:

[1] Tlili, Sirine "Analysis of Sqlsnake worm exploits of MS SQL and Windows VULNERABILITIES " 7April2002
URL:http://www.giac.org/practical/Sirine_Tlili_GCIH.doc

[2] Sygate Security Alerts: SQLsnake *Vulnerability* - Microsoft SQL server "SQLSNAKE Vulnerability in Microsoft SQL server 28May2002
URL:http://www.sygate.com/alerts/SQLsnake_Vulnerability_MS_SQL.htm

[3] VIRUS ALERT "JS/SPIDA Internet Worm Variants" 02June2002
URL:http://www.unl.edu/Security/Virus_Alerts/Spida.htm

[4] Incidents.org – Handlers Diary "SQLsnake Code analysis" 02June2002
URL:http://www.Incidents.org/diary/diary.php?id=157

[5] Cert Incident Note IN-2002-04 "Exploitation of Vulnerabilities in Microsoft SQL SERVER" 22May2002
URL:http://www.cert.org/incident_notes/IN-2002-04.html

[6] Security Focus "SQLsnake" 21May2002
URL: http://www.securityfocus.com/archive/100/273502

[7] MCCLURE, STUART;SCAMBRAY, JOEL; KURTZ,GEORGE."HACKING EXPOSED" (Network Security Secrets and Solutions). Berkeley,Osborne/Mcgraw-Hill, 1999 (page 212).

[8] Garfinkel,Simson;Spafford,Gene "Practical Unix and Internet Security" Oreilly 1996 (925 and 930).

[9] xp_cmdshell (T-SQL) page by Microsoft
URL: http://doc.ddart.net/mssql/sql70/xp_aa-sz.htm

[10] PWDUMP2 (razor.bindview.com) Copywright 1998,2000 Todd Sabin
URL: http://razor.bindview.com/tools/desc/pwdump2_readme.html

[11] SQLSNAKE EXPLOIT ANALYSIS
URL: http://www.giac.org/practical/GCIH/John_Bumgarner_GCIH.pdf

[12] CHOOSING AND USING MSDE 2000 AS THE DATABASE ENGINE FOR YOUR APPLICATION. (Maureen Damery Kirby, Microsoft Corporation) October, 2002