NCCIC NATIONAL CYBERSECURITY AND COMMUNICATIONS INTEGRATION CENTER | US-CERT UNITED STATES COMPUTER EMERGENCY READINESS TEAM

Malware Analysis Report (MAR) - 10135536-G

2018-02-06

Notification

This report is provided "as is" for informational purposes only. The Department of Homeland Security (DHS) does not provide any warranties of any kind regarding any information contained within. The DHS does not endorse any commercial product or service, referenced in this bulletin or otherwise.

This document is marked TLP:WHITE. Disclosure is not limited. Sources may use TLP:WHITE when information carries minimal or no foreseeable risk of misuse, in accordance with applicable rules and procedures for public release. Subject to standard copyright rules, TLP:WHITE information may be distributed without restriction. For more information on the Traffic Light Protocol, see http://www.us-cert.gov/tlp/.

Summary

Description

This Malware Analysis Report (MAR) is the result of analytic efforts between the Department of Homeland Security (DHS) and the Federal Bureau of Investigation (FBI). Working with U.S. Government partners, DHS and FBI identified Trojan malware variants used by the North Korean government - referred to by the U.S. Government as BADCALL. The U.S. Government refers to malicious cyber activity by the North Korean government as HIDDEN COBRA. For more information on HIDDEN COBRA activity, visit https[:]//www[.]us-cert.gov/hiddencobra.

FBI has high confidence that HIDDEN COBRA actors are using malware variants in conjunction with proxy servers to maintain a presence on victim networks and to further network exploitation. DHS and FBI are distributing this MAR to enable network defense and reduce exposure to North Korean government malicious cyber activity.

This MAR includes malware descriptions related to HIDDEN COBRA, suggested response actions and recommended mitigation techniques. Users or administrators should flag activity associated with the malware, report the activity to the DHS National Cybersecurity and Communications Integration Center (NCCIC) or the FBI Cyber Watch (CyWatch), and give the activity the highest priority for enhanced mitigation.

This report provides analysis of three (3) malicious executable files. The first two (2) files are 32-bit Windows executables that function as proxy servers and implement a "Fake TLS" method similar to the behavior described in a previously published NCCIC report, MAR-10135536-B. The third file is an Android Package Kit (APK) file designed to run on Android platforms as a fully functioning Remote Access Tool (RAT).

The following YARA rule may be used to detect the proxy tools:

```
rule NK_SSL_PROXY{
meta:
Author = "US-CERT Code Analysis Team"
Date = "2018/01/09"
MD5_1 = "C6F78AD187C365D117CACBEE140F6230"
MD5_2 = "C01DC42F65ACAF1C917C0CC29BA63ADC"
Info= "Detects NK SSL PROXY"

strings:
$s0 = {8B4C24088A140880F24780C228881408403BC67CEF5E}
$s1 = {568B74240C33C085F67E158B4C24088A140880EA2880F247881408403BC67CEF5E}
$s2 = {4775401F71343574797536686776686937 5E2524736466}
$s3 = {67686667686A75797566676467667472}
$s4 = {6D2A5E265E676866676534776572}
$s5 = {3171617A5853444332337765}

$s6 = "ghfghjuyufgdgftr"
$s7 = "q45tyu6hgvhi7^%$sdf"
$s8 = "m*^&^ghfge4wer"
```

condition:
($s0 and $s1 and $s2 and $s3 and $s4 and $s5) or ($s6 and $s7 and $s8)

}

Files	
Processed	3 c01dc42f65acaf1c917c0cc29ba63adc (C01DC42F65ACAF1C917C0CC29BA63ADC) c6f78ad187c365d117cacbee140f6230 (C6F78AD187C365D117CACBEE140F6230) d93b6a5c04d392fc8ed30375be17beb4 (D93B6A5C04D392FC8ED30375BE17BEB4)

Files

C6F78AD187C365D117CACBEE140F6230

Details

Name	C6F78AD187C365D117CACBEE140F6230
Size	208896
Type	PE32 executable (GUI) Intel 80386, for MS Windows
MD5	c6f78ad187c365d117cacbee140f6230
SHA1	5116f281c61639b48fd58caaed60018bafdefe7a
ssdeep	1536:X86D0r4QxG5+XCFpaG7+esyzktLYUwnZ7hUOKYUwnZ7hUOaeYUwnZ7hUOKYUwnZr:X8O0IgCvH7+UzktMxzxgRxzx9
Entropy	6.83311979555

Antivirus

Ahnlab	Backdoor/Win32.Akdoor

PE Information

Compiled	2016-02-07T03:17:51Z

PE Sections

Name	MD5	Raw Size	Entropy
(header)	a8f97910c62034b318e17aa17fb97f1c	4096	0.688105697711
.text	08112b571663ff5ed42e331a00ccce0c	53248	6.50896736344
.rdata	ca61927558a4dfe9305eb037a5432960	8192	4.57323662515
.data	bb49b2fb00c1ae88ad440971914711a7	139264	6.94127887342
.sxdata	c58b62cf949e8636ebd5c75f482207c3	4096	0.181138192206

Packers

Name	Version	Entry Point
Microsoft Visual C++ v6.0	NA	NA

Relationships

(F) C6F78AD187C365D117CACBEE140F6230 (c6f78)	Related_To	(S) Figure 1
(F) C6F78AD187C365D117CACBEE140F6230 (c6f78)	Related_To	(S) Figure 2
(F) C6F78AD187C365D117CACBEE140F6230 (c6f78)	Related_To	(S) Figure 3
(F) C6F78AD187C365D117CACBEE140F6230 (c6f78)	Related_To	(S) Figure 4

Description

This file is a malicious 32-bit Windows executable. Analysis indicates this application is designed to force a compromised system to function as a proxy server. When executed, the malware binds and listens for incoming connections on port 8000 of the compromised system. The proxy session traffic is protected by way of a simple cipher based on rotating XOR and ADD. The cypher will XOR each byte sent with 47h and added by 28h. Each byte received by the malware will be XOR'ed by 47h and subtracted by 28h. See Screenshots 1, 2 & 3 for code examples. Notably, this malware attempts to disable the Windows firewall before binding to port 8000 by modifying the following registry key:

--Begin Firewall Reg Key Modified--

SYSTEM\CurrentControlSet\Services\SharedAccess\Parameters\FirewallPolicy\StandardProfileGloballyOpenPorts\\List

--End Firewall Reg Key Modified--

Analysis of this malware indicates it is designed to turn a victim host into a "hop point" by relaying traffic to a remote system. When the adversary initially connects to a victim's machine via port 8000, they must first authenticate (over a session secured with the XOR/ADD cipher described above) by providing the ASCII string "1qazXSDC23we". If the malware does not receive this value, it will terminate the session, responding with the value "m*^&^ghfge4wer".

If the operator authenticates successfully, they can then issue the command "ghfghjuyufgdgftr" which instructs the malware to begin

functioning as a proxy server and respond to the operator with the value "q45tyu6hgvhi7^%$sdf". Next, the malware attempts to create a proxy session between the operator and another server. During this process, the malware will attempt to authenticate with the destination server by sending the value "ghfghjuyufgdgftr" as a challenge. To complete the authentication sequence, the malware expects to receive a response value of "q45tyu6hgvhi7^%$sdf". All challenge & response traffic is encoded using the ADD/XOR cipher described earlier.

Importantly, the connection from this proxy malware to the target proxy system will begin via a "fake TLS" connection attempt, similar to the behavior described in a previously released NCCIC report, MAR-10135536-B. Essentially, the malware initiates the TLS session using one of several public SSL certificates obtained from well known, legitimate internet services and imbedded in the malware. The malware begins a TLS session with the proxy target by issuing calls to the OpenSSL functions SSL_new(), SSL_set_fd, and SSL_connect(). The malware then sends and receives initial data (authentication values) to and from the target proxy system using the OpenSSL functions SSL_read() and SSL_write(). However, the malware never completes the TLS handshake, instead decoding the data upon receipt using the XOR/ADD cipher described earlier. See Figures 1-4 for code examples of this process.

The following is a list of the domains for which the malware contains public SSL certificates, used for initiating the "FAKE TLS" sessions:

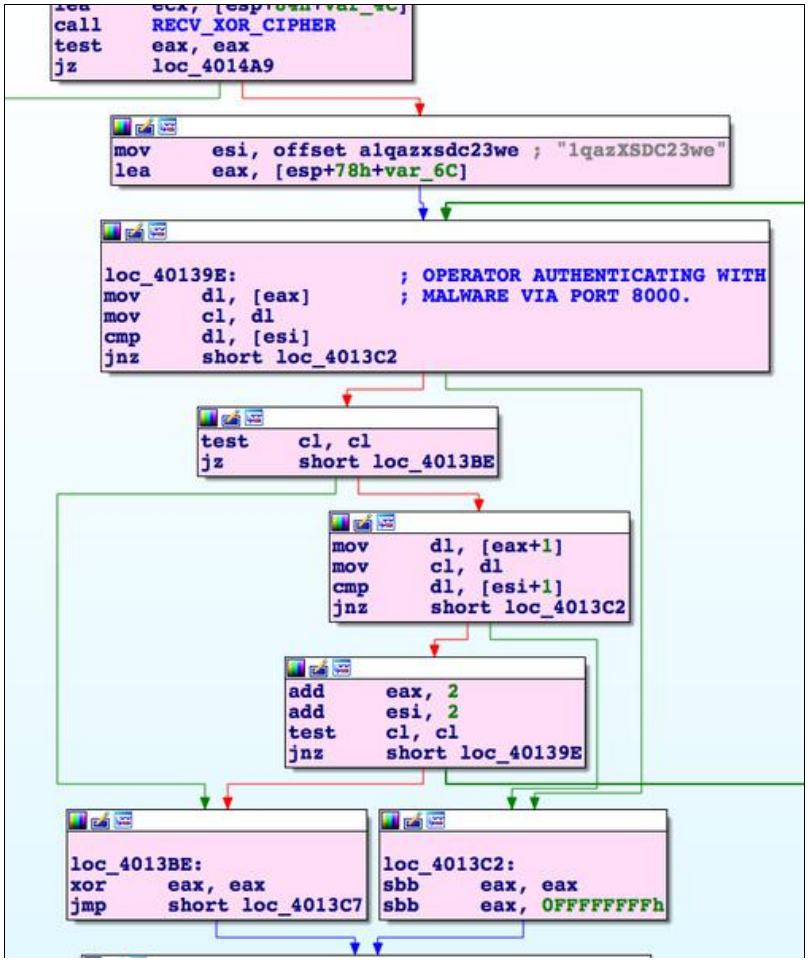
--Begin SSL cert list --

myservice.xbox.com
uk.yahoo.com
web.whatsapp.com
www[.]apple.com
www[.]baidu.com
www[.]bing.com
www[.]bitcoin.org
www[.]comodo.com
www[.]debian.org
www[.]dropbox.com
www[.]facebook.com
www[.]github.com
www[.]google.com
www[.]lenovo.com
www[.]microsoft.com
www[.]paypal.com
www[.]tumblr.com
www[.]twitter.com
www[.]wetransfer.com
www[.]wikipedia.org

-- End SSL cert list--

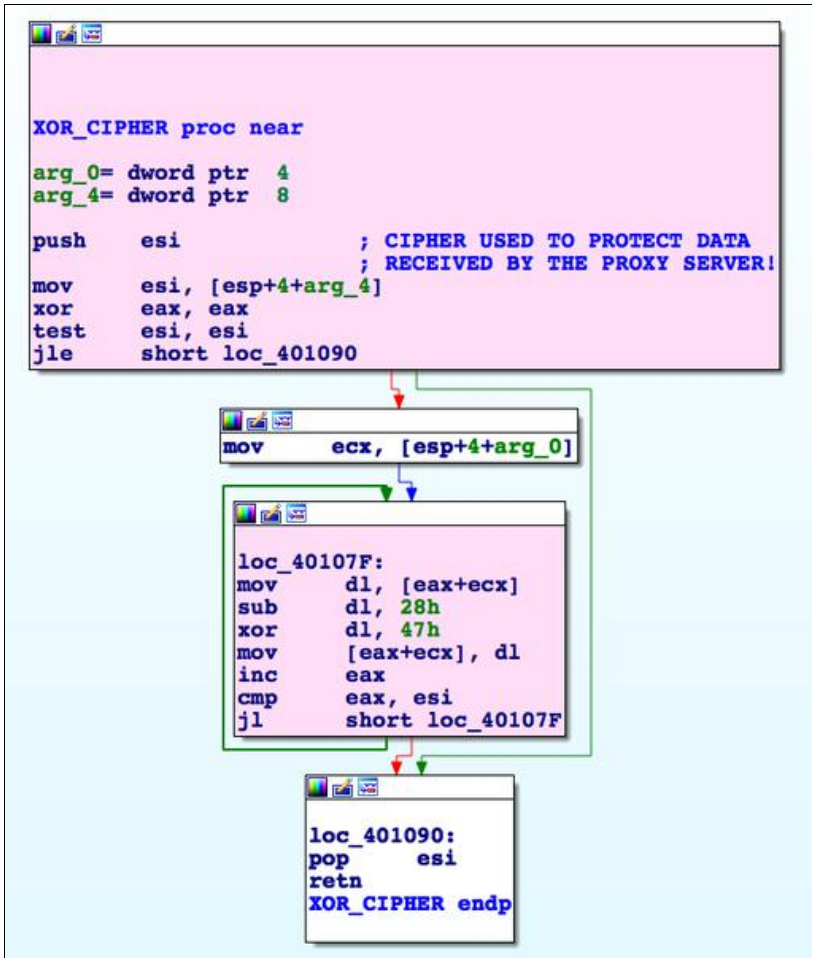
Screenshots

- **Figure 1**



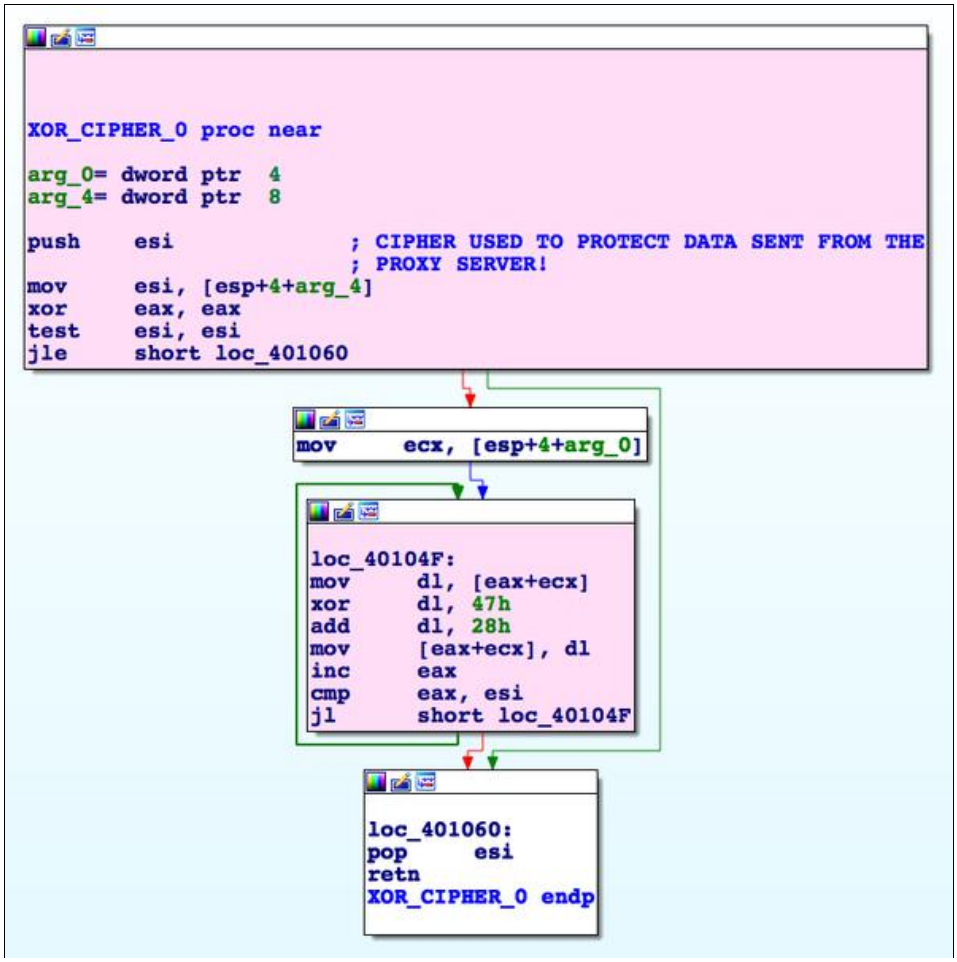
Operator providing command to authenticate with proxy malware.

- **Figure 2**



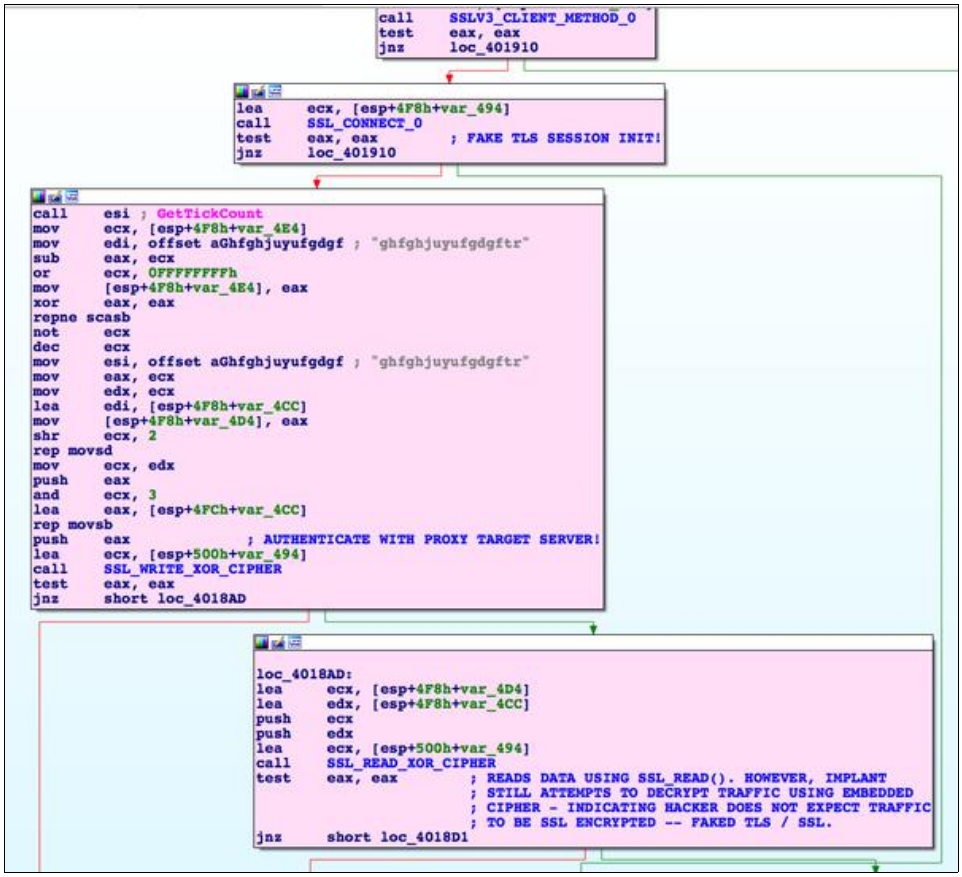
Cipher used to protect the data received by the proxy server. XOR and ADD instructions are used to decode traffic send from the malware.

- **Figure 3**



Cipher used to protect the data sent from the proxy server.

- **Figure 4**



Code demonstrating author's intent to decrypt traffic using imbedded cypher instead of relying on proper implementation of SSL

C01DC42F65ACAF1C917C0CC29BA63ADC

Details

Name	C01DC42F65ACAF1C917C0CC29BA63ADC
Size	233472
Type	PE32 executable (DLL) (GUI) Intel 80386, for MS Windows
MD5	c01dc42f65acaf1c917c0cc29ba63adc
SHA1	d288766fa268bc2534f85fd06a5d52264e646c47
ssdeep	1536:cseScclTQDYY3TSF00sK/LVtKYUwnZ7hUO1YUwnZ7hUOAeYUwnZ7hUO7YUwnZ7hj:cseScjYY3Tyc0LVt9xsxuRxSxzxg0j
Entropy	6.8618428232

Antivirus

nProtect	Trojan/W32.Agent.233472.APN
F-secure	Trojan.Agent.CBEJ
BitDefender	Trojan.Agent.CBEJ
Microsoft Security Essentials	Trojan:Win32/Autophyte.B!dha
Emsisoft	Trojan.Agent.CBEJ (B)
Ahnlab	Backdoor/Win32.Akdoor
Ikarus	Trojan.Agent

PE Information

Compiled	2016-02-05T18:16:54Z

PE Sections

Name	MD5	Raw Size	Entropy
(header)	f0cb80c557b1172362064c51bbb9b271	4096	0.696473380789
.text	e9d0219343e64c8c8aa6f084db44b92c	45056	6.32403974333

.rdata	1092801819f120298e2ddac6a96e3fd0	8192	3.77533292527
.data	5109fb1db61b533c23762d9044579db7	167936	7.04539309174
.reloc	9ce04d3e820fa7056f351dbcfa05b0fb	8192	2.76766633365

Packers

Name	Version	Entry Point
Microsoft Visual C++ 6.0	NA	NA
Microsoft Visual C++ 6.0 DLL (Debug)	NA	NA

Relationships

(F) C01DC42F65ACAF1C917C0CC29BA63ADC (c01dc)	Related_To	(S) Figure 5
(F) C01DC42F65ACAF1C917C0CC29BA63ADC (c01dc)	Related_To	(S) Figure 6
(F) C01DC42F65ACAF1C917C0CC29BA63ADC (c01dc)	Related_To	(S) Figure 7

Description

This file is a malicious 32-bit Windows DLL. Static analysis indicates this application is very similar in structure and function to C6F78AD187C365D117CACBEE140F6230. However, rather than being a PE32 executable this application is a Windows 32-bit DLL, which must be loaded by an external loader. This external loader was not included within this submission.

This DLL is designed to force a compromised system to act as a proxy server. This implant is designed to proxy network traffic from an operator to another software tool that is being operated by the adversary on a remote system. The traffic to and from this proxy server will be protected with the same simple XOR / ADD cipher used by the malware C6F78AD187C365D117CACBEE140F6230.

Analysis of this malware indicates it is designed to bind to and listen for incoming connections on port 443 of a victim's system after disabling the firewall by modifying the following registry key:

--Begin Firewall Reg Key Modified--

SYSTEM\CurrentControlSet\Services\SharedAccess\Parameters\FirewallPolicy\StandardProfileGloballyOpenPorts\\List

--End Firewall Reg Key Modified--

Importantly, analysis indicates this proxy malware expects the incoming system to try to establish a "fake TLS connection" as described in earlier analysis. Static analysis indicates the OpenSSL library is used to implement this TLS/SSL session in such a way to ensure the SSL session fails. For example, the malware attempts to call the OpenSSL function SSL_CTX_use_certificate_file() with the file netconf.dll as the SSL certificate to use (see Figure 5). This application does not drop such a file, therefore this call is likely to fail. Similarly, the malware makes a call to SSL_CTX_use_ProvateKey_file(), designating the file wbemhost.dll as the authentication certificate. This is most likely done intentionally to insure the call will fail.

After connecting to this malware via port 443, the operator must issue the challenge value "qwertyuiop" to authenticate with the implant. This malware also has the added capability of allowing an operator to collect information about the compromised system. This information is collected using the Windows APIs GetComputerNameW, gethostbyname, and GetAdaptersInfo. In order to use this feature, the operator must issue the instruction value "ghfghjuyufgdgftr" after authenticating. As with C6F78AD187C365D117CACBEE140F6230, this malware uses the OpenSLL functions ssl_read() and ssl_write() to exchange data with the operator, however the malware uses a simple SUB/XOR cipher (as earlier described) to decrypt incoming traffic, indicating the operator is aware the traffic will not be encrypted via SSL.

Analysis indicates this malware must also authenticate with the destination server to which the operator wishes to proxy traffic. To do so, this malware first sends that remote server the challenge value "1qazXSDC23we." The malware must then receive the following response from the destination server before it will allow the operator to proxy traffic to it: "m*^&^ghfge4wer" (see Figure 7). The authentication values sent to and from this proxy server will be protected via same XOR / ADD cipher utilized by the malware C6F78AD187C365D117CACBEE140F6230.

The following is a list of the domains for which the malware contains public SSL certificates, used for initiating the "FAKE TLS" sessions:

--Begin SSL cert list--

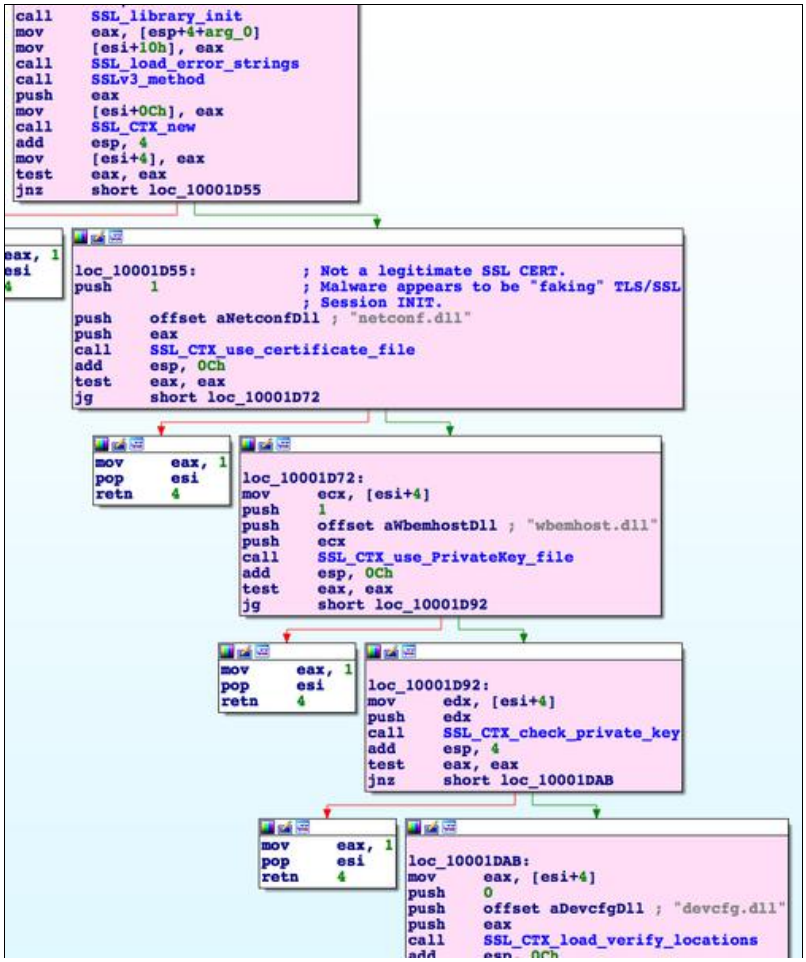
myservice.xbox.com
uk.yahoo.com
web.whatsapp.com
www[.]apple.com
www[.]baidu.com
www[.]bing.com
www[.]bitcoin.org

www[.]comodo.com
www[.]debian.org
www[.]dropbox.com
www[.]facebook.com
www[.]github.com
www[.]google.com
www[.]lenovo.com
www[.]microsoft.com
www[.]paypal.com
www[.]tumblr.com
www[.]twitter.com
www[.]wetransfer.com
www[.]wikipedia.org

--End SSL cert list--

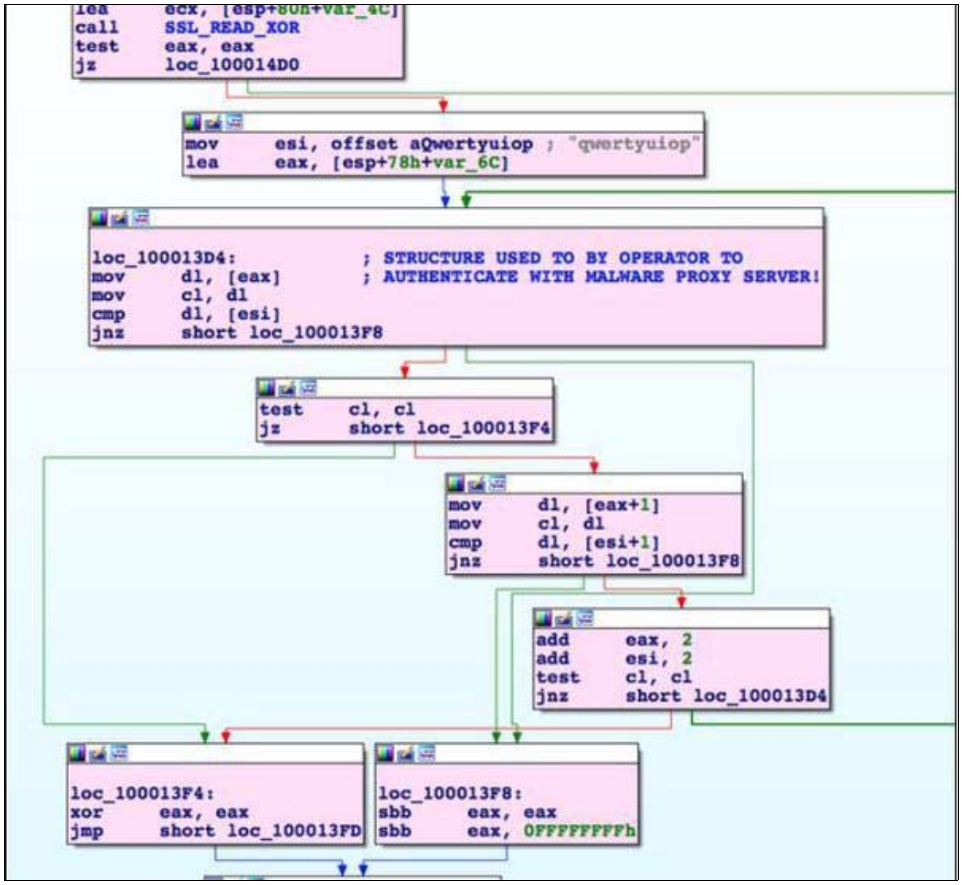
Screenshots

- **Figure 5**



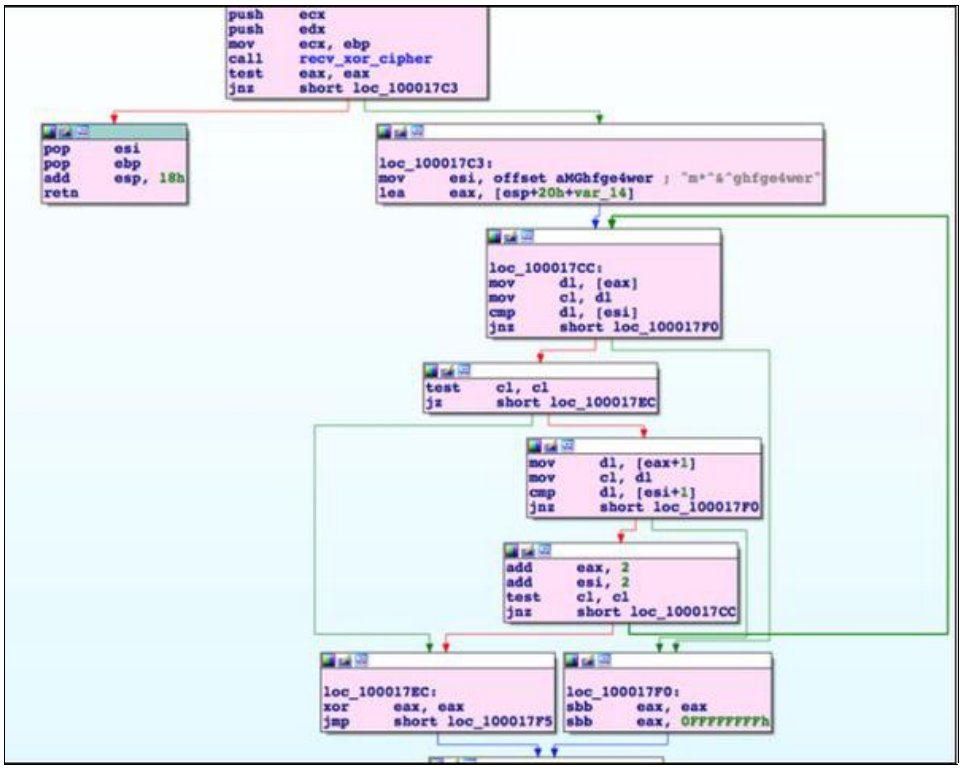
Abnormal calls to SSL_CTX_use_certificate_file() and SSL_CTX_use_PrivateKey_file().

- **Figure 6**



Operator providing command to / authenticating with proxy malware

- **Figure 7**



Malware checking if "m*^&^ghfge4wer" was received from proxy target.

D93B6A5C04D392FC8ED30375BE17BEB4

Details	
Name	D93B6A5C04D392FC8ED30375BE17BEB4
Size	321730
Type	Java archive data (JAR)
MD5	d93b6a5c04d392fc8ed30375be17beb4
SHA1	f862c2899c41a4d1120a7739cdaff561d2490360
ssdeep	6144:1c35mQ6aHY0wxxp/2o0uK1uv8q8lY1pr/Cc800a0sdOQypHIKO9kxZ4:+J5Hlwxmo0Tuv8q8i3+c800NsdFyKKOR
Entropy	7.98967099439

Antivirus	
Sophos	Andr/Spy-ANK
Ikarus	Trojan.AndroidOS.SMForw

Description

This file is a malicious Android APK file. Static analysis indicates it is a Remote Access Tool (RAT), which is designed to listen for incoming connections to a compromised Android device, on port 60000.

Analysis indicates this malware is capable of recording phone calls, taking screenshots using the device's embedded camera, reading data from the contact manager, and downloading and uploading data from the compromised Android device. The application is also capable of executing commands on the compromised system and scanning for open Wi-Fi channels.

Relationship Summary

(F) C6F78AD187C365D117CACBEE140F6230 (c6f78)	Related_To	(S) Figure 1
(F) C6F78AD187C365D117CACBEE140F6230 (c6f78)	Related_To	(S) Figure 2
(F) C6F78AD187C365D117CACBEE140F6230 (c6f78)	Related_To	(S) Figure 3
(F) C6F78AD187C365D117CACBEE140F6230 (c6f78)	Related_To	(S) Figure 4
(S) Figure 1	Related_To	(F) C6F78AD187C365D117CACBEE140F6230 (c6f78)
(S) Figure 2	Related_To	(F) C6F78AD187C365D117CACBEE140F6230 (c6f78)
(S) Figure 3	Related_To	(F) C6F78AD187C365D117CACBEE140F6230 (c6f78)
(S) Figure 4	Related_To	(F) C6F78AD187C365D117CACBEE140F6230 (c6f78)
(F) C01DC42F65ACAF1C917C0CC29BA63ADC (c01dc)	Related_To	(S) Figure 5
(F) C01DC42F65ACAF1C917C0CC29BA63ADC (c01dc)	Related_To	(S) Figure 6
(F) C01DC42F65ACAF1C917C0CC29BA63ADC (c01dc)	Related_To	(S) Figure 7
(S) Figure 5	Related_To	(F) C01DC42F65ACAF1C917C0CC29BA63ADC (c01dc)
(S) Figure 6	Related_To	(F) C01DC42F65ACAF1C917C0CC29BA63ADC (c01dc)
(S) Figure 7	Related_To	(F) C01DC42F65ACAF1C917C0CC29BA63ADC (c01dc)

Mitigation Recommendations

US-CERT would like to remind users and administrators of the following best practices to strengthen the security posture of their organization's systems:

- Maintain up-to-date antivirus signatures and engines.
- Restrict users' ability (permissions) to install and run unwanted software applications.

- Enforce a strong password policy and implement regular password changes.
- Exercise caution when opening e-mail attachments even if the attachment is expected and the sender appears to be known.
- Keep operating system patches up-to-date.
- Enable a personal firewall on agency workstations.
- Disable unnecessary services on agency workstations and servers.
- Scan for and remove suspicious e-mail attachments; ensure the scanned attachment is its "true file type" (i.e., the extension matches the file header).
- Monitor users' web browsing habits; restrict access to sites with unfavorable content.
- Exercise caution when using removable media (e.g., USB thumbdrives, external drives, CDs, etc.).
- Scan all software downloaded from the Internet prior to executing.
- Maintain situational awareness of the latest threats; implement appropriate ACLs.

Contact Information

- 1-888-282-0870
- soc@us-cert.gov (UNCLASS)
- us-cert@dhs.sgov.gov (SIPRNET)
- us-cert@dhs.ic.gov (JWICS)

US-CERT continuously strives to improve its products and services. You can help by answering a very short series of questions about this product at the following URL: https://forms.us-cert.gov/ncsd-feedback/

Document FAQ

What is a MAR? A Malware Analysis Report (MAR) is intended to provide detailed code analysis and insight into specific tactics, techniques, and procedures (TTPs) observed in the malware.

Can I edit this document? This document is not to be edited in any way by recipients. All comments or questions related to this document should be directed to the US-CERT Security Operations Center at 1-888-282-0870 or soc@us-cert.gov.

Can I submit malware to US-CERT? Malware samples can be submitted via three methods. Contact us with any questions.

- Web: https://malware.us-cert.gov
- E-Mail: submit@malware.us-cert.gov
- FTP: ftp.malware.us-cert.gov/malware (anonymous)

US-CERT encourages you to report any suspicious activity, including cybersecurity incidents, possible malicious code, software vulnerabilities, and phishing-related scams. Reporting forms can be found on US-CERT's homepage at www.us-cert.gov.