Sep 25 2023

NETMANAGEIT

Intelligence Report

Multi-year Chinese APT Campaign Targets South Korean Academic, Government, and Political Entities

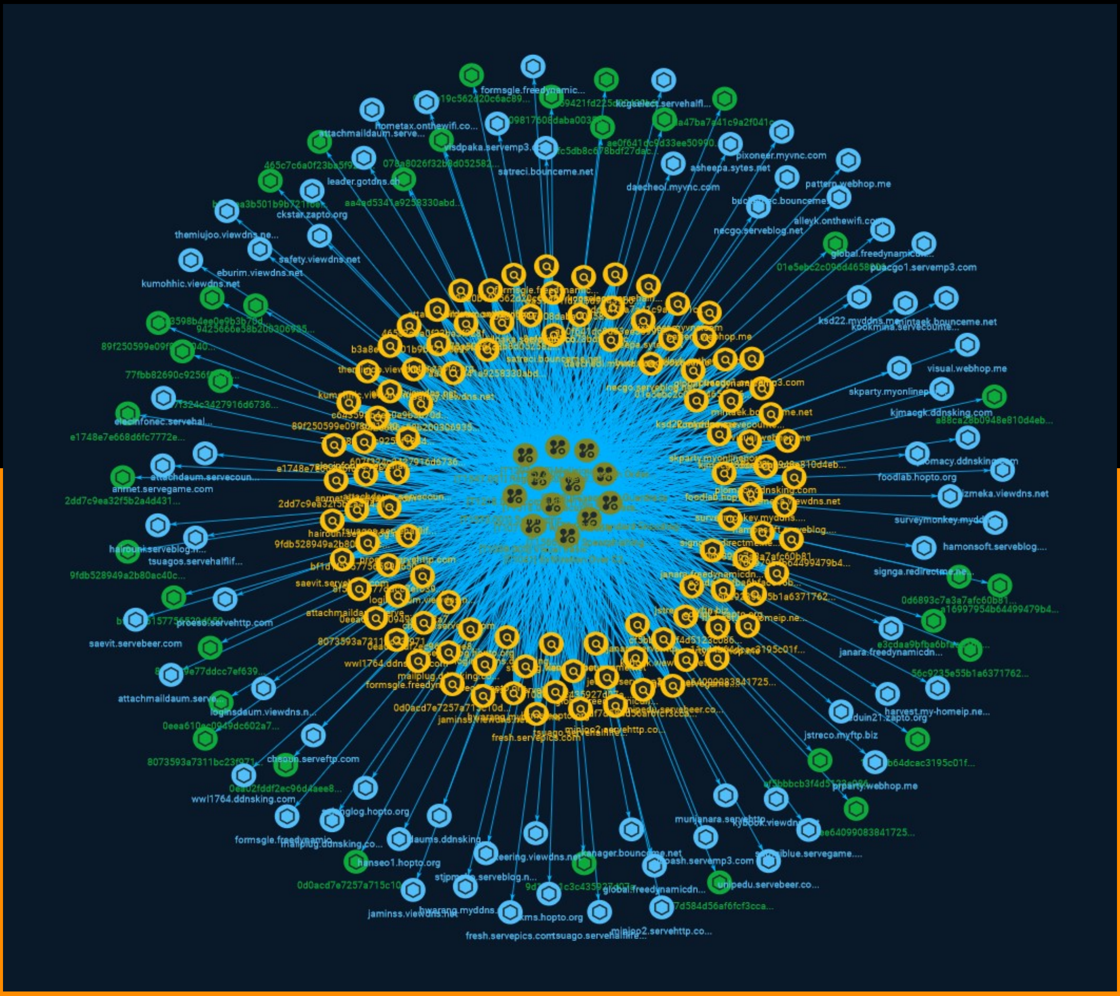

Table of contents

Overview

- Description 3
- Confidence 3

Entities

- Attack-Pattern 4
- Indicator 13

Observables

- StixFile 51
- Hostname 54

External References

- External References 59

Overview

Description

Recorded Future's Insikt Group has conducted an analysis of a prolonged cyber-espionage campaign known as TAG-74, which is attributed to Chinese state-sponsored actors. TAG-74 primarily focuses on infiltrating South Korean academic, political, and government organizations. This group has been linked to Chinese military intelligence and poses a significant threat to academic, aerospace and defense, government, military, and political entities in South Korea, Japan, and Russia. TAG-74’s targeting of South Korean academic institutions aligns with China's broader espionage efforts aimed at intellectual property theft and expanding its influence within higher education worldwide.

Confidence

This value represents the confidence in the correctness of the data contained within this report.

15 / 100

Attack-Pattern

Name

Security Software Discovery

ID

T1518.001

Description

Adversaries may attempt to get a listing of security software, configurations, defensive tools, and sensors that are installed on a system or in a cloud environment. This may include things such as firewall rules and anti-virus. Adversaries may use the information from [Security Software Discovery](https://attack.mitre.org/techniques/T1518/001) during automated discovery to shape follow-on behaviors, including whether or not the adversary fully infects the target and/or attempts specific actions. Example commands that can be used to obtain security software information are [netsh](https://attack.mitre.org/software/S0108), `reg query` with [Reg](https://attack.mitre.org/software/S0075), `dir` with [cmd](https://attack.mitre.org/software/S0106), and [Tasklist](https://attack.mitre.org/software/S0057), but other indicators of discovery behavior may be more specific to the type of software or security system the adversary is looking for. It is becoming more common to see macOS malware perform checks for LittleSnitch and KnockKnock software. Adversaries may also utilize cloud APIs to discover the configurations of firewall rules within an environment.(Citation: Expel IO Evil in AWS) For example, the permitted IP ranges, ports or user accounts for the inbound/outbound rules of security groups, virtual firewalls established within AWS for EC2 and/or VPC instances, can be revealed by the `DescribeSecurityGroups` action with various request parameters. (Citation: DescribeSecurityGroups - Amazon Elastic Compute Cloud)

Name

Web Protocols

ID

T1071.001

Description

Adversaries may communicate using application layer protocols associated with web traffic to avoid detection/network filtering by blending in with existing traffic. Commands to the remote system, and often the results of those commands, will be embedded within the protocol traffic between the client and server. Protocols such as HTTP/S(Citation: CrowdStrike Putter Panda) and WebSocket(Citation: Brazking-Websockets) that carry web traffic may be very common in environments. HTTP/S packets have many fields and headers in which data can be concealed. An adversary may abuse these protocols to communicate with systems under their control within a victim network while also mimicking normal, expected traffic.

Name

Malicious File

ID

T1204.002

Description

An adversary may rely upon a user opening a malicious file in order to gain execution. Users may be subjected to social engineering to get them to open a file that will lead to code execution. This user action will typically be observed as follow-on behavior from [Spearphishing Attachment](https://attack.mitre.org/techniques/T1566/001). Adversaries may use several types of files that require a user to execute them, including .doc, .pdf, .xls, .rtf, .scr, .exe, .lnk, .pif, and .cpl. Adversaries may employ various forms of [Masquerading](https://attack.mitre.org/techniques/T1036) and [Obfuscated Files or Information](https://attack.mitre.org/techniques/T1027) to increase the likelihood that a user will open and successfully execute a malicious file. These methods may include using a familiar naming convention and/or password protecting the file and supplying instructions to a user on how to open it.(Citation: Password Protected Word Docs) While [Malicious File](https://attack.mitre.org/techniques/T1204/002) frequently occurs shortly after Initial Access it may occur at other phases of an intrusion, such as when an adversary places a file in a shared directory or on a user's desktop hoping that a user will click on it. This activity may also be seen shortly after [Internal Spearphishing](https://attack.mitre.org/techniques/T1534).

Name

Spearphishing Attachment

ID

T1566.001

Description

Adversaries may send spearphishing emails with a malicious attachment in an attempt to gain access to victim systems. Spearphishing attachment is a specific variant of spearphishing. Spearphishing attachment is different from other forms of spearphishing in that it employs the use of malware attached to an email. All forms of spearphishing are electronically delivered social engineering targeted at a specific individual, company, or industry. In this scenario, adversaries attach a file to the spearphishing email and usually rely upon [User Execution](https://attack.mitre.org/techniques/T1204) to gain execution. Spearphishing may also involve social engineering techniques, such as posing as a trusted source. There are many options for the attachment such as Microsoft Office documents, executables, PDFs, or archived files. Upon opening the attachment (and potentially clicking past protections), the adversary's payload exploits a vulnerability or directly executes on the user's system. The text of the spearphishing email usually tries to give a plausible reason why the file should be opened, and may explain how to bypass system protections in order to do so. The email may also contain instructions on how to decrypt an attachment, such as a zip file password, in order to evade email boundary defenses. Adversaries frequently manipulate file extensions and icons in order to make attached executables appear to be document files, or files exploiting one application appear to be a file for a different one.

Name

Compiled HTML File

ID

T1218.001

Description

Adversaries may abuse Compiled HTML files (.chm) to conceal malicious code. CHM files are commonly distributed as part of the Microsoft HTML Help system. CHM files are compressed compilations of various content such as HTML documents, images, and scripting/web related programming languages such VBA, JScript, Java, and ActiveX. (Citation: Microsoft HTML Help May 2018) CHM content is displayed using underlying components of the Internet Explorer browser (Citation: Microsoft HTML Help ActiveX) loaded by the HTML Help executable program (hh.exe). (Citation: Microsoft HTML Help Executable Program) A custom CHM file containing embedded payloads could be delivered to a victim then triggered by [User Execution](https://attack.mitre.org/techniques/T1204). CHM execution may also bypass application application control on older and/or unpatched systems that do not account for execution of binaries through hh.exe. (Citation: MsitPros CHM Aug 2017) (Citation: Microsoft CVE-2017-8625 Aug 2017)

Name

Execution Guardrails

ID

T1480

Description

Adversaries may use execution guardrails to constrain execution or actions based on adversary supplied and environment specific conditions that are expected to be present on the target. Guardrails ensure that a payload only executes against an intended target and reduces collateral damage from an adversary's campaign.(Citation: FireEye Kevin Mandia Guardrails) Values an adversary can provide about a target system or environment to use as guardrails may include specific network share names, attached physical devices, files, joined Active Directory (AD) domains, and local/external IP addresses.(Citation: FireEye Outlook Dec 2019) Guardrails can be used to prevent exposure of capabilities in environments that are not intended to be compromised or operated within. This use of guardrails is distinct from typical [Virtualization/Sandbox Evasion](https://

attack.mitre.org/techniques/T1497). While use of [Virtualization/Sandbox Evasion](https://attack.mitre.org/techniques/T1497) may involve checking for known sandbox values and continuing with execution only if there is no match, the use of guardrails will involve checking for an expected target-specific value and only continuing with execution if there is such a match.

Name

Visual Basic

ID

T1059.005

Description

Adversaries may abuse Visual Basic (VB) for execution. VB is a programming language created by Microsoft with interoperability with many Windows technologies such as [Component Object Model](https://attack.mitre.org/techniques/T1559/001) and the [Native API](https://attack.mitre.org/techniques/T1106) through the Windows API. Although tagged as legacy with no planned future evolutions, VB is integrated and supported in the .NET Framework and cross-platform .NET Core.(Citation: VB .NET Mar 2020)(Citation: VB Microsoft) Derivative languages based on VB have also been created, such as Visual Basic for Applications (VBA) and VBScript. VBA is an event-driven programming language built into Microsoft Office, as well as several third-party applications.(Citation: Microsoft VBA)(Citation: Wikipedia VBA) VBA enables documents to contain macros used to automate the execution of tasks and other functionality on the host. VBScript is a default scripting language on Windows hosts and can also be used in place of [JavaScript](https://attack.mitre.org/techniques/T1059/007) on HTML Application (HTA) webpages served to Internet Explorer (though most modern browsers do not come with VBScript support).(Citation: Microsoft VBScript) Adversaries may use VB payloads to execute malicious commands. Common malicious usage includes automating execution of behaviors with VBScript or embedding VBA content into [Spearphishing Attachment](https://attack.mitre.org/techniques/T1566/001) payloads (which may also involve [Mark-of-the-Web Bypass](https://attack.mitre.org/techniques/T1553/005) to enable execution).(Citation: Default VBS macros Blocking)

Name

Registry Run Keys / Startup Folder

ID

T1547.001

Description

Adversaries may achieve persistence by adding a program to a startup folder or referencing it with a Registry run key. Adding an entry to the "run keys" in the Registry or startup folder will cause the program referenced to be executed when a user logs in. (Citation: Microsoft Run Key) These programs will be executed under the context of the user and will have the account's associated permissions level. Placing a program within a startup folder will also cause that program to execute when a user logs in. There is a startup folder location for individual user accounts as well as a system-wide startup folder that will be checked regardless of which user account logs in. The startup folder path for the current user is `C:\Users\\[Username]\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup`. The startup folder path for all users is `C:\ProgramData\Microsoft\Windows\Start Menu\Programs\StartUp`. The following run keys are created by default on Windows systems: * `HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Run` * `HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\RunOnce` * `HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Run` * `HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\RunOnce` Run keys may exist under multiple hives.(Citation: Microsoft Wow6432Node 2018)(Citation: Malwarebytes Wow6432Node 2016) The `HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\RunOnceEx` is also available but is not created by default on Windows Vista and newer. Registry run key entries can reference programs directly or list them as a dependency.(Citation: Microsoft Run Key) For example, it is possible to load a DLL at logon using a "Depend" key with RunOnceEx: `reg add HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\RunOnceEx\0001\Depend /v 1 /d "C:\temp\evil[.]dll"` (Citation: Oddvar Moe RunOnceEx Mar 2018) The following Registry keys can be used to set startup folder items for persistence: * `HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\User Shell Folders` * `HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\Shell Folders` * `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\Shell Folders` * `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\User

Shell Folders` The following Registry keys can control automatic startup of services during boot: * `HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\RunServicesOnce` * `HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\RunServicesOnce` * `HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\RunServices` * `HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\RunServices` Using policy settings to specify startup programs creates corresponding values in either of two Registry keys: * `HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Policies\Explorer\Run` * `HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Policies\Explorer\Run` The Winlogon key controls actions that occur when a user logs on to a computer running Windows 7. Most of these actions are under the control of the operating system, but you can also add custom actions here. The `HKEY_LOCAL_MACHINE\Software\Microsoft\Windows NT\CurrentVersion\Winlogon\Userinit` and `HKEY_LOCAL_MACHINE\Software\Microsoft\Windows NT\CurrentVersion\Winlogon\Shell` subkeys can automatically launch programs. Programs listed in the load value of the registry key `HKEY_CURRENT_USER\Software\Microsoft\Windows NT\CurrentVersion\Windows` run when any user logs on. By default, the multistring `BootExecute` value of the registry key `HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\Session Manager` is set to `autocheck autochk *`. This value causes Windows, at startup, to check the file-system integrity of the hard disks if the system has been shut down abnormally. Adversaries can add other programs or processes to this registry value which will automatically launch at boot. Adversaries can use these configuration locations to execute malware, such as remote access tools, to maintain persistence through system reboots. Adversaries may also use [Masquerading](https://attack.mitre.org/techniques/T1036) to make the Registry entries look as if they are associated with legitimate programs.

Name

Standard Encoding

ID

T1132.001

Description

Adversaries may encode data with a standard data encoding system to make the content of command and control traffic more difficult to detect. Command and control (C2) information can be encoded using a standard data encoding system that adheres to existing protocol specifications. Common data encoding schemes include ASCII, Unicode, hexadecimal, Base64, and MIME.(Citation: Wikipedia Binary-to-text Encoding)(Citation: Wikipedia Character Encoding) Some data encoding systems may also result in data compression, such as gzip.

Name

Symmetric Cryptography

ID

T1573.001

Description

Adversaries may employ a known symmetric encryption algorithm to conceal command and control traffic rather than relying on any inherent protections provided by a communication protocol. Symmetric encryption algorithms use the same key for plaintext encryption and ciphertext decryption. Common symmetric encryption algorithms include AES, DES, 3DES, Blowfish, and RC4.

Name

DLL Search Order Hijacking

ID

T1574.001

Description

Adversaries may execute their own malicious payloads by hijacking the search order used to load DLLs. Windows systems use a common method to look for required DLLs to load into a program. (Citation: Microsoft Dynamic Link Library Search Order)(Citation: FireEye Hijacking July 2010) Hijacking DLL loads may be for the purpose of establishing persistence as well as elevating privileges and/or evading restrictions on file execution. There are many ways an adversary can hijack DLL loads. Adversaries may plant trojan dynamic-link library files (DLLs) in a directory that will be searched before the location of a legitimate library that will be requested by a program, causing Windows to load their malicious library when it is called for by the victim program. Adversaries may also perform DLL preloading, also called binary planting attacks, (Citation: OWASP Binary Planting) by placing a malicious DLL with the same name as an ambiguously specified DLL in a location that Windows searches before the legitimate DLL. Often this location is the current working directory of the program.(Citation: FireEye fxsst June 2011) Remote DLL preloading attacks occur when a program sets its current directory to a remote location such as a Web share before loading a DLL. (Citation: Microsoft Security Advisory 2269637) Adversaries may also directly modify the search order via DLL redirection, which after being enabled (in the Registry and creation of a redirection file) may cause a program to load a different DLL.(Citation: Microsoft Dynamic-Link Library Redirection)(Citation: Microsoft Manifests)(Citation: FireEye DLL Search Order Hijacking) If a search order-vulnerable program is configured to run at a higher privilege level, then the adversary-controlled DLL that is loaded will also be executed at the higher level. In this case, the technique could be used for privilege escalation from user to administrator or SYSTEM or from administrator to SYSTEM, depending on the program. Programs that fall victim to path hijacking may appear to behave normally because malicious DLLs may be configured to also load the legitimate DLLs they were meant to replace.

Name

Exfiltration Over C2 Channel

ID

T1041

Description

Adversaries may steal data by exfiltrating it over an existing command and control channel. Stolen data is encoded into the normal communications channel using the same protocol as command and control communications.

Indicator

Name
foodlab.hopto.org

Pattern Type
stix

Pattern
[hostname:value = 'foodlab.hopto.org']

Name
chsoun.serveftp.com

Pattern Type
stix

Pattern
[hostname:value = 'chsoun.serveftp.com']

Name
aa4ad5341a9258330abd732cbab3721d76764f1ff21a8f960622661d701a1a71

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = 'aa4ad5341a9258330abd732cbab3721d76764f1ff21a8f960622661d701a1a71']

Name

jstreco.myftp.biz

Pattern Type

stix

Pattern

[hostname:value = 'jstreco.myftp.biz']

Name

attachdaum.servecounterstrike.com

Pattern Type

stix

Pattern

[hostname:value = 'attachdaum.servecounterstrike.com']

Name

necgo.serveblog.net

Pattern Type

stix

Pattern

[hostname:value = 'necgo.serveblog.net']

Name

steering.viewdns.net

Pattern Type

stix

Pattern

[hostname:value = 'steering.viewdns.net']

Name

9425666e58b200306935c36301d66a4bf2c831ad41ea0ee8984f056257b86eb6

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '9425666e58b200306935c36301d66a4bf2c831ad41ea0ee8984f056257b86eb6']

Name
signga.redirectme.net

Pattern Type
stix

Pattern
[hostname:value = 'signga.redirectme.net']

Name
proeso.servehttp.com

Pattern Type
stix

Pattern
[hostname:value = 'proeso.servehttp.com']

Name
satreci.bounceme.net

Pattern Type
stix

Pattern
[hostname:value = 'satreci.bounceme.net']

Name

cf5bbbcb3f4d5123c08635c8fd398e55e516893b902a33cd6f478e8797eea962

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = 'cf5bbbcb3f4d5123c08635c8fd398e55e516893b902a33cd6f478e8797eea962']

Name

anrnet.servegame.com

Pattern Type

stix

Pattern

[hostname:value = 'anrnet.servegame.com']

Name

skparty.myonlineportal.org

Pattern Type

stix

Pattern

[hostname:value = 'skparty.myonlineportal.org']
Name
logindaums.ddnsking.com
Pattern Type
stix
Pattern
[hostname:value = 'logindaums.ddnsking.com']
Name
leader.gotdns.ch
Pattern Type
stix
Pattern
[hostname:value = 'leader.gotdns.ch']
Name
e3cdaa9bfba6bfac616b7f275c1e888b8910efcb8a3df071f68ad1e83710bd61
Pattern Type
stix
Pattern

[file:hashes.'SHA-256' = 'e3cdaa9bfba6bfac616b7f275c1e888b8910efcb8a3df071f68ad1e83710bd61']

Name

loginsdaum.viewdns.net

Pattern Type

stix

Pattern

[hostname:value = 'loginsdaum.viewdns.net']

Name

asheepa.sytes.net

Pattern Type

stix

Pattern

[hostname:value = 'asheepa.sytes.net']

Name

munjanara.servehttp.com

Pattern Type

stix

Pattern

[hostname:value = 'munjanara.servehttp.com']

Name

eburim.viewdns.net

Pattern Type

stix

Pattern

[hostname:value = 'eburim.viewdns.net']

Name

surveymonkey.myddns.me

Pattern Type

stix

Pattern

[hostname:value = 'surveymonkey.myddns.me']

Name

sejonglog.hopto.org

Pattern Type

stix

Pattern
[hostname:value = 'sejonglog.hopto.org']

Name
8efc5db8c678bdf27dacbf033842c2ef676c979afdc4561cb8d315d2d488491f

Pattern Type
stix

Pattern
[file:hashes.'SHA-256' = '8efc5db8c678bdf27dacbf033842c2ef676c979afdc4561cb8d315d2d488491f']

Name
eduin21.zapto.org

Pattern Type
stix

Pattern
[hostname:value = 'eduin21.zapto.org']

Name
hairouni.serveblog.net

Pattern Type

stix

Pattern

[hostname:value = 'hairouni.serveblog.net']

Name

01e5ebc2c096d465800660a0ad6d62208a5b2b675e3700f3734fac225b1d38bd

Description

ALF:Trojan:Win32/Cassini_ade36583!ibt

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '01e5ebc2c096d465800660a0ad6d62208a5b2b675e3700f3734fac225b1d38bd']

Name

kanager.bounceme.net

Pattern Type

stix

Pattern

[hostname:value = 'kanager.bounceme.net']

Name
safety.viewdns.net
Pattern Type
stix
Pattern
[hostname:value = 'safety.viewdns.net']
Name
janara.freedynamicdns.org
Pattern Type
stix
Pattern
[hostname:value = 'janara.freedynamicdns.org']
Name
0eea610ec0949dc602a7178f25f316c4db654301e7389ee414c9826783fd64c0
Pattern Type
stix
Pattern

[file:hashes.'SHA-256' =
'0eea610ec0949dc602a7178f25f316c4db654301e7389ee414c9826783fd64c0']

Name

daecheol.myvnc.com

Pattern Type

stix

Pattern

[hostname:value = 'daecheol.myvnc.com']

Name

jeoash.servemp3.com

Pattern Type

stix

Pattern

[hostname:value = 'jeoash.servemp3.com']

Name

607f324c3427916d67369e40af72aa441f3ca7be1e0ec6c53c3558fc7a1c4186

Description

Win.Worm.Mytob-270

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '607f324c3427916d67369e40af72aa441f3ca7be1e0ec6c53c3558fc7a1c4186']

Name

kookmina.servecounterstrike.com

Pattern Type

stix

Pattern

[hostname:value = 'kookmina.servecounterstrike.com']

Name

stjpmsko.serveblog.net

Pattern Type

stix

Pattern

[hostname:value = 'stjpmsko.serveblog.net']

Name

078a8026f32b8d05258285dc527408388c651f6c3eaebc45f8bb3f4b42248631

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '078a8026f32b8d05258285dc527408388c651f6c3eaebc45f8bb3f4b42248631']

Name

harvest.my-homeip.net

Pattern Type

stix

Pattern

[hostname:value = 'harvest.my-homeip.net']

Name

tsuagos.servehalflife.com

Pattern Type

stix

Pattern

[hostname:value = 'tsuagos.servehalflife.com']

Name
bucketnec.bounceme.net
Pattern Type
stix
Pattern
[hostname:value = 'bucketnec.bounceme.net']
Name
global.freedynamicdns.net
Pattern Type
stix
Pattern
[hostname:value = 'global.freedynamicdns.net']
Name
ckstar.zapto.org
Pattern Type
stix
Pattern
[hostname:value = 'ckstar.zapto.org']

Name

plomacy.ddnsking.com

Pattern Type

stix

Pattern

[hostname:value = 'plomacy.ddnsking.com']

Name

0ea02fddf2ec96d4aee8adaffda2dd5fab0ea989b0c3f8c1577a1be22ee9153a

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '0ea02fddf2ec96d4aee8adaffda2dd5fab0ea989b0c3f8c1577a1be22ee9153a']

Name

visdpaka.servemp3.com

Pattern Type

stix

Pattern

[hostname:value = 'visdpaka.servemp3.com']

Name

kybook.viewdns.net

Pattern Type

stix

Pattern

[hostname:value = 'kybook.viewdns.net']

Name

77fbb82690c9256f18544e26bb6e306a3f878d3e9ab5966457ac39631dfd2cb0

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '77fbb82690c9256f18544e26bb6e306a3f878d3e9ab5966457ac39631dfd2cb0']

Name

ksd22.myddns.me

Pattern Type

stix

Pattern
[hostname:value = 'ksd22.myddns.me']

Name
9d10de1c3c435927d07a1280390faf82c5d7d5465d772f6e1206751400072261

Pattern Type
stix

Pattern
[file:hashes.'SHA-256' = '9d10de1c3c435927d07a1280390faf82c5d7d5465d772f6e1206751400072261']

Name
hometax.onthewifi.com

Pattern Type
stix

Pattern
[hostname:value = 'hometax.onthewifi.com']

Name
0d6893c7a3a7afc60b81c136b1dcdfb24b35efab01aac165fe0083b9b981da7c

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '0d6893c7a3a7afc60b81c136b1dcdfb24b35efab01aac165fe0083b9b981da7c']

Name

8f50f49e77ddcc7ef639a76217b2eb25c48f9ce21ae8341050d0da49b89b7b34

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '8f50f49e77ddcc7ef639a76217b2eb25c48f9ce21ae8341050d0da49b89b7b34']

Name

kcgselect.servehalflife.com

Pattern Type

stix

Pattern

[hostname:value = 'kcgselect.servehalflife.com']

Name

minjoo2.servehttp.com

Pattern Type

stix

Pattern

[hostname:value = 'minjoo2.servehttp.com']

Name

c643598b4ee0e9b3b70dae19437bbec01e881a1ad3b2ec1f6f5c335e552e5d6e

Description

ConventionEngine_Term_Desktop

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = 'c643598b4ee0e9b3b70dae19437bbec01e881a1ad3b2ec1f6f5c335e552e5d6e']

Name

kjmacgk.ddnsking.com

Pattern Type

stix

Pattern

[hostname:value = 'kjmacgk.ddnsking.com']

Name

mailplug.ddnsking.com

Pattern Type

stix

Pattern

[hostname:value = 'mailplug.ddnsking.com']

Name

beb09817608daba003589292a6cca2f724c52f756df2ef0e230380345d702716

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = 'beb09817608daba003589292a6cca2f724c52f756df2ef0e230380345d702716']

Name

prparty.webhop.me

Pattern Type

stix

Pattern
[hostname:value = 'prparty.webhop.me']

Name
9fdb528949a2b80ac40cb7d3333bdff5d504294cc3d90cf353db72b8beffd2b2

Pattern Type
stix

Pattern
[file:hashes.'SHA-256' = '9fdb528949a2b80ac40cb7d3333bdff5d504294cc3d90cf353db72b8beffd2b2']

Name
alleyk.onthewifi.com

Pattern Type
stix

Pattern
[hostname:value = 'alleyk.onthewifi.com']

Name
hwarang.myddns.me

Pattern Type

stix

Pattern

[hostname:value = 'hwarang.myddns.me']

Name

b3a8ea3b501b9b721f6e371dd57025dc14d117c29ce8ee955b240d4a17bc2127

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = 'b3a8ea3b501b9b721f6e371dd57025dc14d117c29ce8ee955b240d4a17bc2127']

Name

hamonsoft.serveblog.net

Pattern Type

stix

Pattern

[hostname:value = 'hamonsoft.serveblog.net']

Name

wwl1764.ddnsking.com

Pattern Type

stix

Pattern

[hostname:value = 'wwl1764.ddnsking.com']

Name

ae0f641dc9d33ee50990971104ef1c598e216693700be6b74bb1e9ef373af97c

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = 'ae0f641dc9d33ee50990971104ef1c598e216693700be6b74bb1e9ef373af97c']

Name

samgiblue.servegame.com

Pattern Type

stix

Pattern

[hostname:value = 'samgiblue.servegame.com']

Name

89f250599e09f8631040e73cd9ea5e515d87e3d1d989f484686893bececla9bc

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '89f250599e09f8631040e73cd9ea5e515d87e3d1d989f484686893bececla9bc']

Name

mintaek.bounceme.net

Pattern Type

stix

Pattern

[hostname:value = 'mintaek.bounceme.net']

Name

hanseo1.hopto.org

Pattern Type

stix

Pattern

[hostname:value = 'hanseo1.hopto.org']

Name

attachmaildaum.servecounterstrike.com

Pattern Type

stix

Pattern

[hostname:value = 'attachmaildaum.servecounterstrike.com']

Name

56c9235e55b1a6371762159619e949686d8de2b45a348aeb4fd5bed6a126f66a

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '56c9235e55b1a6371762159619e949686d8de2b45a348aeb4fd5bed6a126f66a']

Name

saevit.servebeer.com

Pattern Type

stix

Pattern

[hostname:value = 'saevit.servebeer.com']

Name

pattern.webhop.me

Pattern Type

stix

Pattern

[hostname:value = 'pattern.webhop.me']

Name

tsuago.servehalflife.com

Pattern Type

stix

Pattern

[hostname:value = 'tsuago.servehalflife.com']

Name

kumohhic.viewdns.net

Pattern Type

stix

Pattern

[hostname:value = 'kumohhic.viewdns.net']

Name

a88ca28b0948e810d4eb519db7b72a40cfe7907ce4c6a881a192880278f3c8b5

Description

ALF:Trojan:Win32/Korlia.F!dha

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = 'a88ca28b0948e810d4eb519db7b72a40cfe7907ce4c6a881a192880278f3c8b5']

Name

formsgle.freedynamicdns.org

Pattern Type

stix

Pattern

[hostname:value = 'formsgle.freedynamicdns.org']

Name

formsgle.freedynamicdns.net

Pattern Type
stix
Pattern
[hostname:value = 'formsgle.freedynamicdns.net']
Name
ba07ee6409908384172511563e6b9059cf84121fcb42c54d45c76ec67cb36d7c
Pattern Type
stix
Pattern
[file:hashes.'SHA-256' = 'ba07ee6409908384172511563e6b9059cf84121fcb42c54d45c76ec67cb36d7c']
Name
2dd7c9ea32f5b2a4d431fc54aa68cd76837f80bb324ef2e4e1e5134e467e35af
Pattern Type
stix
Pattern
[file:hashes.'SHA-256' = '2dd7c9ea32f5b2a4d431fc54aa68cd76837f80bb324ef2e4e1e5134e467e35af']
Name

global.freedynamicdns.org

Pattern Type

stix

Pattern

[hostname:value = 'global.freedynamicdns.org']

Name

fresh.servepics.com

Pattern Type

stix

Pattern

[hostname:value = 'fresh.servepics.com']

Name

465c7c6a0f23ba5f928fc0d0cdc4d9f6ec89e03dcedafc3d72b3b3c01a54a00c

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '465c7c6a0f23ba5f928fc0d0cdc4d9f6ec89e03dcedafc3d72b3b3c01a54a00c']

Name

a16997954b64499479b4721c9f742b5d2875496f2035e1c654b06694981041b2

Description

ASPackv212AlexeySolodovnikov

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = 'a16997954b64499479b4721c9f742b5d2875496f2035e1c654b06694981041b2']

Name

attachmaildaum.serveblog.net

Pattern Type

stix

Pattern

[hostname:value = 'attachmaildaum.serveblog.net']

Name

dda47ba7a41c9a2f041cc10f9b058a78e0019315c51cc98d0f356e2054209ae5

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' =
'dda47ba7a41c9a2f041cc10f9b058a78e0019315c51cc98d0f356e2054209ae5']

Name

df7d584d56af6fcf3cca31ed0d3a4d34abd2c1019b8d223a230f8a78075a7d9a

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' =
'df7d584d56af6fcf3cca31ed0d3a4d34abd2c1019b8d223a230f8a78075a7d9a']

Name

e1748e7e668d6fc7772e95c08d32f41ad340f4a9acf0e2f933f3cbeba7323afa

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' =
'e1748e7e668d6fc7772e95c08d32f41ad340f4a9acf0e2f933f3cbeba7323afa']

Name

visual.webhop.me

Pattern Type

stix

Pattern

[hostname:value = 'visual.webhop.me']

Name

0d0acd7e7257a715c10dded76acb233adc8fdfe32857eda060bd1448e8b54585

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '0d0acd7e7257a715c10dded76acb233adc8fdfe32857eda060bd1448e8b54585']

Name

0ea0b19c562d20c6ac89a1f2db06eedcb147cde2281e79bb0497cef62094b514

Description

korlia

Pattern Type

stix

Pattern
[file:hashes.'SHA-256' = '0ea0b19c562d20c6ac89a1f2db06eedcb147cde2281e79bb0497cef62094b514']
Name
11cd4b64dcac3195c01ffc937ae1eb77aa2f98d560a75347036d54a1cf69a5fd
Pattern Type
stix
Pattern
[file:hashes.'SHA-256' = '11cd4b64dcac3195c01ffc937ae1eb77aa2f98d560a75347036d54a1cf69a5fd']
Name
bf1d1f5157756529d650719cc531ec2de94edb66ae1dabd00ed6f4b90a336d9c
Description
Win.Worm.Mytob-270
Pattern Type
stix
Pattern
[file:hashes.'SHA-256' = 'bf1d1f5157756529d650719cc531ec2de94edb66ae1dabd00ed6f4b90a336d9c']

Name
pixoneer.myvnc.com
Pattern Type
stix
Pattern
[hostname:value = 'pixoneer.myvnc.com']
Name
likms.hopto.org
Pattern Type
stix
Pattern
[hostname:value = 'likms.hopto.org']
Name
themiujoo.viewdns.net
Pattern Type
stix
Pattern
[hostname:value = 'themiujoo.viewdns.net']

Name

bizmeka.viewdns.net

Pattern Type

stix

Pattern

[hostname:value = 'bizmeka.viewdns.net']

Name

8073593a7311bc23f971352c85ce2034c01d3d3fbbe4f99a8f3825292e8f9f77

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' =
'8073593a7311bc23f971352c85ce2034c01d3d3fbbe4f99a8f3825292e8f9f77']

Name

6a59421fd225d90439b6a933458718cf43dbe518c63979e8980bc070c070558a

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '6a59421fd225d90439b6a933458718cf43dbe518c63979e8980bc070c070558a']

Name

unipedu.servebeer.com

Pattern Type

stix

Pattern

[hostname:value = 'unipedu.servebeer.com']

Name

elecinfonec.servehalflife.com

Pattern Type

stix

Pattern

[hostname:value = 'elecinfonec.servehalflife.com']

Name

puacgo1.servemp3.com

Pattern Type

stix

Pattern
[hostname:value = 'puacgo1.servemp3.com']
Name
jaminss.viewdns.net
Pattern Type
stix
Pattern
[hostname:value = 'jaminss.viewdns.net']

StixFile

Value
ba07ee6409908384172511563e6b9059cf84121fcb42c54d45c76ec67cb36d7c
8f50f49e77ddcc7ef639a76217b2eb25c48f9ce21ae8341050d0da49b89b7b34
56c9235e55b1a6371762159619e949686d8de2b45a348aeb4fd5bed6a126f66a
cf5bbbcb3f4d5123c08635c8fd398e55e516893b902a33cd6f478e8797eea962
e1748e7e668d6fc7772e95c08d32f41ad340f4a9acf0e2f933f3cbeba7323afa
607f324c3427916d67369e40af72aa441f3ca7be1e0ec6c53c3558fc7a1c4186
a88ca28b0948e810d4eb519db7b72a40cfe7907ce4c6a881a192880278f3c8b5
11cd4b64dcac3195c01ffc937ae1eb77aa2f98d560a75347036d54a1cf69a5fd
a16997954b64499479b4721c9f742b5d2875496f2035e1c654b06694981041b2
8efc5db8c678bdf27dacbf033842c2ef676c979afdc4561cb8d315d2d488491f
b3a8ea3b501b9b721f6e371dd57025dc14d117c29ce8ee955b240d4a17bc2127
dda47ba7a41c9a2f041cc10f9b058a78e0019315c51cc98d0f356e2054209ae5
e3cdaa9bfba6bfac616b7f275c1e888b8910efcb8a3df071f68ad1e83710bd61

6a59421fd225d90439b6a933458718cf43dbe518c63979e8980bc070c070558a

0ea0b19c562d20c6ac89a1f2db06eedcb147cde2281e79bb0497cef62094b514

0ea02fddf2ec96d4aee8adaffda2dd5fab0ea989b0c3f8c1577a1be22ee9153a

aa4ad5341a9258330abd732cbab3721d76764f1ff21a8f960622661d701a1a71

078a8026f32b8d05258285dc527408388c651f6c3eaebc45f8bb3f4b42248631

89f250599e09f8631040e73cd9ea5e515d87e3d1d989f484686893becec1a9bc

0d0acd7e7257a715c10dded76acb233adc8fdfe32857eda060bd1448e8b54585

2dd7c9ea32f5b2a4d431fc54aa68cd76837f80bb324ef2e4e1e5134e467e35af

ae0f641dc9d33ee50990971104ef1c598e216693700be6b74bb1e9ef373af97c

9d10de1c3c435927d07a1280390faf82c5d7d5465d772f6e1206751400072261

0d6893c7a3a7afc60b81c136b1dcdfb24b35efab01aac165fe0083b9b981da7c

c643598b4ee0e9b3b70dae19437bbec01e881a1ad3b2ec1f6f5c335e552e5d6e

df7d584d56af6fcf3cca31ed0d3a4d34abd2c1019b8d223a230f8a78075a7d9a

9425666e58b200306935c36301d66a4bf2c831ad41ea0ee8984f056257b86eb6

beb09817608daba003589292a6cca2f724c52f756df2ef0e230380345d702716

9fdb528949a2b80ac40cb7d3333bdff5d504294cc3d90cf353db72b8beffd2b2

8073593a7311bc23f971352c85ce2034c01d3d3fbbe4f99a8f3825292e8f9f77

01e5ebc2c096d465800660a0ad6d62208a5b2b675e3700f3734fac225b1d38bd

465c7c6a0f23ba5f928fc0d0cdc4d9f6ec89e03dcedafc3d72b3b3c01a54a00c
77fbb82690c9256f18544e26bb6e306a3f878d3e9ab5966457ac39631dfd2cb0
bf1d1f5157756529d650719cc531ec2de94edb66ae1dabd00ed6f4b90a336d9c
0eea610ec0949dc602a7178f25f316c4db654301e7389ee414c9826783fd64c0

Hostname

Value
kjmacgk.ddnsking.com
skparty.myonlineportal.org
satreci.bounceme.net
mailplug.ddnsking.com
themiujoo.viewdns.net
global.freedynamicdns.org
mintaek.bounceme.net
visdpaka.servemp3.com
necgo.serveblog.net
hometax.onthewifi.com
proeso.servehttp.com
signga.redirectme.net
jstreco.myftp.biz

puacgo1.servemp3.com
global.freedynamicdns.net
logindaums.ddnsking.com
tsuagos.servehalflife.com
jeoash.servemp3.com
steering.viewdns.net
leader.gotdns.ch
formsgle.freedynamicdns.org
attachmaildaum.serveblog.net
harvest.my-homeip.net
tsuago.servehalflife.com
kookmina.servecounterstrike.com
ksd22.myddns.me
kumohhic.viewdns.net
daecheol.myvnc.com
unipedu.servebeer.com
jaminss.viewdns.net
kybook.viewdns.net

chsoun.serveftp.com

elecinfonec.servehalflife.com

hairouni.serveblog.net

eburim.viewdns.net

sejonglog.hopto.org

janara.freedynamicdns.org

asheepa.sytes.net

minjoo2.servehttp.com

bizmeka.viewdns.net

kanager.bounceme.net

attachmaildaum.servecounterstrike.com

attachdaum.servecounterstrike.com

pattern.webhop.me

saevit.servebeer.com

formsgle.freedynamicdns.net

safety.viewdns.net

ckstar.zapto.org

samgiblue.servegame.com

likms.hopto.org
bucketnec.bounceme.net
hamonsoft.serveblog.net
alleyk.onthewifi.com
plomacy.ddnsking.com
surveymonkey.myddns.me
kcgselect.servehalflife.com
foodlab.hopto.org
visual.webhop.me
munjanara.servehttp.com
loginsdaum.viewdns.net
prparty.webhop.me
stjpmsko.serveblog.net
anrnet.servegame.com
pixoneer.myvnc.com
wwl1764.ddnsking.com
hwarang.myddns.me
eduin21.zapto.org

fresh.servepics.com
hanseo1.hopto.org

External References

- https://otx.alienvault.com/pulse/6512042de044dd134bcfb416
- https://www.recordedfuture.com/multi-year-chinese-apt-campaign-targets-south-korean-academic-government-political-entities
- https://go.recordedfuture.com/hubfs/reports/cta-2023-0919.pdf