Jun 28 2024

NETMANAGEIT

Intelligence Report

Supposed Grasshopper: operators impersonate Israeli government and private companies to deploy open-source malware

Table of contents

Overview

Entities

Observables

External References

Overview

Description

A long-running campaign was identified involving malicious actors impersonating Israeli entities and private companies. The operators delivered payloads through crafted WordPress sites, employing a mix of custom code and open-source malware like Donut and Sliver. While the motivations remain unclear, the activities illustrate the challenges of distinguishing legitimate penetration testing from malicious operations, especially when targeting government bodies. The investigation highlights the increasing adoption of publicly available attack tools and the need for greater transparency in the cybersecurity industry.

Confidence

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

100 / 100

Content

N/A

Attack-Pattern

Name
T1070.004
ID
T1070.004
Description
Adversaries may delete files left behind by the actions of their intrusion activity. Malware, tools, or other non-native files dropped or created on a system by an adversary (ex: [Ingress Tool Transfer](https://attack.mitre.org/techniques/T1105)) may leave traces to indicate to what was done within a network and how. Removal of these files can occur during an intrusion, or as part of a post-intrusion process to minimize the adversary's footprint. There are tools available from the host operating system to perform cleanup, but adversaries may use other tools as well.(Citation: Microsoft SDelete July 2016) Examples of built-in [Command and Scripting Interpreter](https://attack.mitre.org/techniques/T1059) functions include `del` on Windows and `rm` or `unlink` on Linux and macOS.
Name
T1608.004
ID
T1608.004

Description

Adversaries may prepare an operational environment to infect systems that visit a website over the normal course of browsing. Endpoint systems may be compromised through browsing to adversary controlled sites, as in [Drive-by Compromise](https://attack.mitre.org/techniques/T1189). In such cases, the user's web browser is typically targeted for exploitation (often not requiring any extra user interaction once landing on the site), but adversaries may also set up websites for non-exploitation behavior such as [Application Access Token](https://attack.mitre.org/techniques/T1550/001). Prior to [Drive-by Compromise](https://attack.mitre.org/techniques/T1189), adversaries must stage resources needed to deliver that exploit to users who browse to an adversary controlled site. Drive-by content can be staged on adversary controlled infrastructure that has been acquired ([Acquire Infrastructure](https://attack.mitre.org/techniques/T1583)) or previously compromised ([Compromise Infrastructure](https://attack.mitre.org/techniques/T1584)). Adversaries may upload or inject malicious web content, such as [JavaScript](https://attack.mitre.org/techniques/T1059/007), into websites.(Citation: FireEye CFR Watering Hole 2012)(Citation: Gallagher 2015) This may be done in a number of ways, including: * Inserting malicious scripts into web pages or other user controllable web content such as forum posts * Modifying script files served to websites from publicly writeable cloud storage buckets * Crafting malicious web advertisements and purchasing ad space on a website through legitimate ad providers (i.e., [Malvertising](https://attack.mitre.org/techniques/T1583/008)) In addition to staging content to exploit a user's web browser, adversaries may also stage scripting content to profile the user's browser (as in [Gather Victim Host Information](https://attack.mitre.org/techniques/T1592)) to ensure it is vulnerable prior to attempting exploitation.(Citation: ATT ScanBox) Websites compromised by an adversary and used to stage a drive-by may be ones visited by a specific community, such as government, a particular industry, or region, where the goal is to compromise a specific user or set of users based on a shared interest. This kind of targeted campaign is referred to a strategic web compromise or watering hole attack. Adversaries may purchase domains similar to legitimate domains (ex: homoglyphs, typosquatting, different top-level domain, etc.) during acquisition of infrastructure ([Domains](https://attack.mitre.org/techniques/T1583/001)) to help facilitate [Drive-by Compromise](https://attack.mitre.org/techniques/T1189).

Name

T1071.001

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

T1059.001

ID

T1059.001

Description

Adversaries may abuse PowerShell commands and scripts for execution. PowerShell is a powerful interactive command-line interface and scripting environment included in the Windows operating system.(Citation: TechNet PowerShell) Adversaries can use PowerShell to perform a number of actions, including discovery of information and execution of code. Examples include the `Start-Process` cmdlet which can be used to run an executable and the `Invoke-Command` cmdlet which runs a command locally or on a remote computer (though administrator permissions are required to use PowerShell to connect to remote systems). PowerShell may also be used to download and run executables from the Internet, which can be executed from disk or in memory without touching disk. A number of PowerShell-based offensive testing tools are available, including [Empire](https://attack.mitre.org/software/S0363), [PowerSploit](https://attack.mitre.org/software/S0194), [PoshC2](https://attack.mitre.org/software/S0378), and PSAttack.(Citation: Github PSAttack) PowerShell commands/scripts can also be executed without directly invoking the `powershell.exe` binary through interfaces to PowerShell's underlying `System.Management.Automation` assembly DLL exposed through the .NET framework and

Windows Common Language Interface (CLI).(Citation: Sixdub PowerPick Jan 2016)(Citation: SilentBreak Offensive PS Dec 2015)(Citation: Microsoft PSfromCsharp APR 2014)

Name

T1053.005

ID

T1053.005

Description

Adversaries may abuse the Windows Task Scheduler to perform task scheduling for initial or recurring execution of malicious code. There are multiple ways to access the Task Scheduler in Windows. The [schtasks](https://attack.mitre.org/software/S0111) utility can be run directly on the command line, or the Task Scheduler can be opened through the GUI within the Administrator Tools section of the Control Panel. In some cases, adversaries have used a .NET wrapper for the Windows Task Scheduler, and alternatively, adversaries have used the Windows netapi32 library to create a scheduled task. The deprecated [at](https://attack.mitre.org/software/S0110) utility could also be abused by adversaries (ex: [At](https://attack.mitre.org/techniques/T1053/002)), though `at.exe` can not access tasks created with `schtasks` or the Control Panel. An adversary may use Windows Task Scheduler to execute programs at system startup or on a scheduled basis for persistence. The Windows Task Scheduler can also be abused to conduct remote Execution as part of Lateral Movement and/or to run a process under the context of a specified account (such as SYSTEM). Similar to [System Binary Proxy Execution](https://attack.mitre.org/techniques/T1218), adversaries have also abused the Windows Task Scheduler to potentially mask one-time execution under signed/trusted system processes.(Citation: ProofPoint Serpent) Adversaries may also create "hidden" scheduled tasks (i.e. [Hide Artifacts](https://attack.mitre.org/techniques/T1564)) that may not be visible to defender tools and manual queries used to enumerate tasks. Specifically, an adversary may hide a task from `schtasks /query` and the Task Scheduler by deleting the associated Security Descriptor (SD) registry value (where deletion of this value must be completed using SYSTEM permissions).(Citation: SigmaHQ)(Citation: Tarrask scheduled task) Adversaries may also employ alternate methods to hide tasks, such as altering the metadata (e.g., `Index` value) within associated registry keys.(Citation: Defending Against Scheduled Task Attacks in Windows Environments)

Name

T1105

ID

T1105

Description

Adversaries may transfer tools or other files from an external system into a compromised environment. Tools or files may be copied from an external adversary-controlled system to the victim network through the command and control channel or through alternate protocols such as [ftp](https://attack.mitre.org/software/S0095). Once present, adversaries may also transfer/spread tools between victim devices within a compromised environment (i.e. [Lateral Tool Transfer](https://attack.mitre.org/techniques/T1570)). On Windows, adversaries may use various utilities to download tools, such as `copy`, `finger`, [certutil](https://attack.mitre.org/software/S0160), and [PowerShell](https://attack.mitre.org/techniques/T1059/001) commands such as `IEX(New-Object Net.WebClient).downloadString()` and `Invoke-WebRequest`. On Linux and macOS systems, a variety of utilities also exist, such as `curl`, `scp`, `sftp`, `tftp`, `rsync`, `finger`, and `wget`. (Citation: t1105_lolbas) Adversaries may also abuse installers and package managers, such as `yum` or `winget`, to download tools to victim hosts. Adversaries have also abused file application features, such as the Windows `search-ms` protocol handler, to deliver malicious files to victims through remote file searches invoked by [User Execution](https://attack.mitre.org/techniques/T1204) (typically after interacting with [Phishing](https://attack.mitre.org/techniques/T1566) lures).(Citation: T1105: Trellix_search-ms) Files can also be transferred using various [Web Service](https://attack.mitre.org/techniques/T1102)s as well as native or otherwise present tools on the victim system.(Citation: PTSecurity Cobalt Dec 2016) In some cases, adversaries may be able to leverage services that sync between a web-based and an on-premises client, such as Dropbox or OneDrive, to transfer files onto victim systems. For example, by compromising a cloud account and logging into the service's web portal, an adversary may be able to trigger an automatic syncing process that transfers the file onto the victim's machine.(Citation: Dropbox Malware Sync)

Name

T1574.002

ID

T1574.002

Description

Adversaries may execute their own malicious payloads by side-loading DLLs. Similar to [DLL Search Order Hijacking](https://attack.mitre.org/techniques/T1574/001), side-loading involves hijacking which DLL a program loads. But rather than just planting the DLL within the search order of a program then waiting for the victim application to be invoked, adversaries may directly side-load their payloads by planting then invoking a legitimate application that executes their payload(s). Side-loading takes advantage of the DLL search order used by the loader by positioning both the victim application and malicious payload(s) alongside each other. Adversaries likely use side-loading as a means of masking actions they perform under a legitimate, trusted, and potentially elevated system or software process. Benign executables used to side-load payloads may not be flagged during delivery and/or execution. Adversary payloads may also be encrypted/packed or otherwise obfuscated until loaded into the memory of the trusted process.(Citation: FireEye DLL Side-Loading)

Name

T1562.001

ID

T1562.001

Description

Adversaries may modify and/or disable security tools to avoid possible detection of their malware/tools and activities. This may take many forms, such as killing security software processes or services, modifying / deleting Registry keys or configuration files so that tools do not operate properly, or other methods to interfere with security tools scanning or reporting information. Adversaries may also disable updates to prevent the latest security patches from reaching tools on victim systems.(Citation: SCADAfence_ransomware) Adversaries may also tamper with artifacts deployed and utilized by security tools. Security tools may make dynamic changes to system components in order to maintain visibility into specific events. For example, security products may load their own modules and/or modify those loaded by processes to facilitate data collection. Similar to [Indicator Blocking](https://attack.mitre.org/techniques/T1562/006), adversaries may unhook or

otherwise modify these features added by tools (especially those that exist in userland or are otherwise potentially accessible to adversaries) to avoid detection.(Citation: OutFlank System Calls)(Citation: MDSec System Calls) Adversaries may also focus on specific applications such as Sysmon. For example, the “Start” and “Enable” values in `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\WMI\Autologger\EventLog-Microsoft-Windows-Sysmon-Operational` may be modified to tamper with and potentially disable Sysmon logging.(Citation: disable_win_evt_logging) On network devices, adversaries may attempt to skip digital signature verification checks by altering startup configuration files and effectively disabling firmware verification that typically occurs at boot.(Citation: Fortinet Zero-Day and Custom Malware Used by Suspected Chinese Actor in Espionage Operation)(Citation: Analysis of FG-IR-22-369) In cloud environments, tools disabled by adversaries may include cloud monitoring agents that report back to services such as AWS CloudWatch or Google Cloud Monitor. Furthermore, although defensive tools may have anti-tampering mechanisms, adversaries may abuse tools such as legitimate rootkit removal kits to impair and/or disable these tools.(Citation: chasing_avaddon_ransomware)(Citation: dharma_ransomware)(Citation: demystifying_ryuk)(Citation: doppelpaymer_crowdstrike) For example, adversaries have used tools such as GMER to find and shut down hidden processes and antivirus software on infected systems.(Citation: demystifying_ryuk) Additionally, adversaries may exploit legitimate drivers from anti-virus software to gain access to kernel space (i.e. [Exploitation for Privilege Escalation](https://attack.mitre.org/techniques/T1068)), which may lead to bypassing anti-tampering features.(Citation: avoslocker_ransomware)

Name

T1562.004

ID

T1562.004

Description

Adversaries may disable or modify system firewalls in order to bypass controls limiting network usage. Changes could be disabling the entire mechanism as well as adding, deleting, or modifying particular rules. This can be done numerous ways depending on the operating system, including via command-line, editing Windows Registry keys, and Windows Control Panel. Modifying or disabling a system firewall may enable adversary C2 communications, lateral movement, and/or data exfiltration that would otherwise not be allowed. For example, adversaries may add a new firewall rule for a well-known protocol (such as RDP) using a non-traditional and potentially less securitized port (i.e. [Non-

Standard Port](https://attack.mitre.org/techniques/T1571)).(Citation: change_rdp_port_conti) Adversaries may also modify host networking settings that indirectly manipulate system firewalls, such as interface bandwidth or network connection request thresholds.(Citation: Huntress BlackCat) Settings related to enabling abuse of various [Remote Services](https://attack.mitre.org/techniques/T1021) may also indirectly modify firewall rules.

Indicator

Name
carlsberg.site
Pattern Type
stix
Pattern
[domain-name:value = 'carlsberg.site']
Name
c21ad804c22a67ddb62adf5f6153a99268f0b26e359b842ebeabcada824c277f
Pattern Type
stix
Pattern
[file:hashes.'SHA-256' = 'c21ad804c22a67ddb62adf5f6153a99268f0b26e359b842ebeabcada824c277f']
Name

carls.employers-view.com

Pattern Type

stix

Pattern

[hostname:value = 'carls.employers-view.com']

Name

d891f4339354d3f4c4b834e781fa4eaca2b59c6a8ee9340cc489ab0023e034c8

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = 'd891f4339354d3f4c4b834e781fa4eaca2b59c6a8ee9340cc489ab0023e034c8']

Name

economy-gov-il.com

Pattern Type

stix

Pattern

[domain-name:value = 'economy-gov-il.com']

Name

portal.carlsberg.site

Pattern Type

stix

Pattern

[hostname:value = 'portal.carlsberg.site']

Name

d7a66f8529f1c32342c4ed06c4a4750a93bd44161f578e5b94d6d30f7cc41581

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = 'd7a66f8529f1c32342c4ed06c4a4750a93bd44161f578e5b94d6d30f7cc41581']

Name

login.carlsberg.site

Pattern Type

stix

Pattern

[hostname:value = 'login.carlsberg.site']

Name

auth.economy-gov-il.com

Pattern Type

stix

Pattern

[hostname:value = 'auth.economy-gov-il.com']

Name

157.90.153.59

Description

ISP: Hetzner Online GmbH **OS:** - ---------- Services: **22:** ``` SSH-2.0-OpenSSH_8.9p1 Ubuntu-3ubuntu0.6 Key type: ecdsa-sha2-nistp256 Key: AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBL0PUMoz31EWXhB4o8p39l GL nuekUS7j+lyC/bHgm/sr2N32Nuz26hk8WGb6BpT4U2yT0l47ECxjAnfcxWQUmdM= Fingerprint: 27:a9:dd:08:b5:4b:74:94:04:25:e1:04:7d:57:1b:ec Kex Algorithms: curve25519-sha256 curve25519-sha256@libssh.org ecdh-sha2-nistp256 ecdh-sha2-nistp384 ecdh-sha2-nistp521 sntrup761x25519-sha512@openssh.com diffie-hellman-group-exchange-sha256 diffie-hellman-group16-sha512 diffie-hellman-group18-sha512 diffie-hellman-group14-sha256 kex-strict-s-v00@openssh.com Server Host Key Algorithms: rsa-sha2-512 rsa-sha2-256 ecdsa-sha2-nistp256 ssh-ed25519 Encryption Algorithms: chacha20-poly1305@openssh.com aes128-ctr aes192-ctr aes256-ctr aes128-gcm@openssh.com aes256-gcm@openssh.com MAC Algorithms: umac-64-etm@openssh.com umac-128-etm@openssh.com hmac-sha2-256-etm@openssh.com hmac-sha2-512-etm@openssh.com hmac-sha1-etm@openssh.com umac-64@openssh.com umac-128@openssh.com hmac-sha2-256 hmac-sha2-512 hmac-sha1 Compression Algorithms: none zlib@openssh.com ``` ---------- **443:** ``` SSL Error: TLSV1_ALERT_PROTOCOL_VERSION ``` HEARTBLEED: 2024/06/24 17:20:11 157.90.153.59:443 - ERROR: remote error: protocol version not supported ----------

Pattern Type

stix

Pattern

[ipv4-addr:value = '157.90.153.59']

Name

portal.operative-sintecmedia.com

Pattern Type

stix

Pattern

[hostname:value = 'portal.operative-sintecmedia.com']

Name

6fb531839410b65be4f4833d73f02429b4dba8ed56fa236cce76750b9a1be23b

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '6fb531839410b65be4f4833d73f02429b4dba8ed56fa236cce76750b9a1be23b']

Name

2070dd30e87c492e6f44ebb0a37bcae7cb309de61e1c4e6223df090bb26b3cd7

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '2070dd30e87c492e6f44ebb0a37bcae7cb309de61e1c4e6223df090bb26b3cd7']

Name

login.operative-sintecmedia.com

Pattern Type

stix

Pattern

[hostname:value = 'login.operative-sintecmedia.com']

Name

a8948dd8e4e4961da537b40bf7e313f0358510f93e25dea1a2fafd522bfd0e84

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = 'a8948dd8e4e4961da537b40bf7e313f0358510f93e25dea1a2fafd522bfd0e84']

Name
employees.carlsberg.site
Pattern Type
stix
Pattern
[hostname:value = 'employees.carlsberg.site']
Name
login.microsofonlline.com
Pattern Type
stix
Pattern
[hostname:value = 'login.microsofonlline.com']
Name
www.economy-gov-il.com
Pattern Type
stix
Pattern
[hostname:value = 'www.economy-gov-il.com']

Malware

Name
Sliver

Name
Donut

indicates

Name

Name

Name

Name

Name

Name

Name

Name

Name

Name
Name
Name
Name
Name
Name
Name
Name
Name
Name
Name
Name

Name
Name
Name
Name
Name
Name
Name
Name
Name
Name
Name
Name

Name
Name
Name
Name
Name
Name
Name
Name
Name
Name
Name
Name

Name
Name
Name
Name

uses

Name

Name

based-on

Name
Name
Name
Name
Name

Domain-Name

Value
carlsberg.site
economy-gov-il.com

StixFile

Value
6fb531839410b65be4f4833d73f02429b4dba8ed56fa236cce76750b9a1be23b
a8948dd8e4e4961da537b40bf7e313f0358510f93e25dea1a2fafd522bfd0e84
c21ad804c22a67ddb62adf5f6153a99268f0b26e359b842ebeabcada824c277f
d7a66f8529f1c32342c4ed06c4a4750a93bd44161f578e5b94d6d30f7cc41581
d891f4339354d3f4c4b834e781fa4eaca2b59c6a8ee9340cc489ab0023e034c8
2070dd30e87c492e6f44ebb0a37bcae7cb309de61e1c4e6223df090bb26b3cd7

Hostname

Value
login.carlsberg.site
employees.carlsberg.site
auth.economy-gov-il.com
carls.employers-view.com
portal.carlsberg.site
portal.operative-sintecmedia.com
login.operative-sintecmedia.com
www.economy-gov-il.com
login.microsofonlline.com

IPv4-Addr

Value
157.90.153.59

External References

- https://harfanglab.io/en/insidethelab/supposed-grasshopper-operators-impersonate-israeli-gov-private-companies-deploy-open-source-malware/
- https://otx.alienvault.com/pulse/667ecf871c3ec2d1d3f4715b