Feb 20 2024

NETMANAGEIT

Intelligence Report

Migo - a Redis Miner with Novel System Weakening Techniques

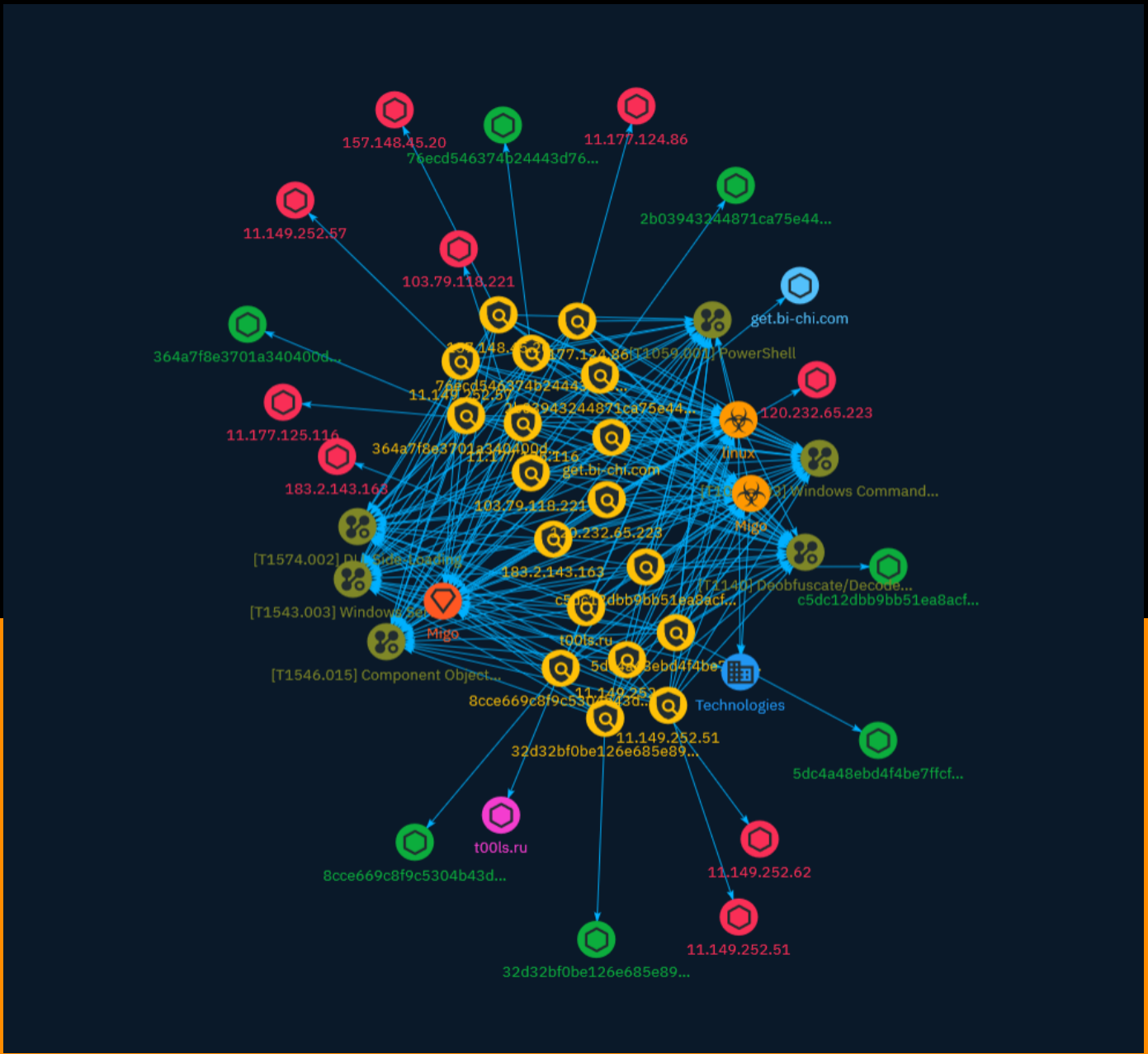

Table of contents

Overview

Entities

Observables

External References

Overview

Description

A novel malware campaign targeting Redis for initial access was recently encountered. The malware, named Migo, aims to compromise Redis servers for cryptocurrency mining on the underlying Linux host. It utilizes novel Redis system weakening commands to exploit Redis. Migo is delivered as an obfuscated Golang ELF binary with the ability to persist on Linux hosts. A modified rootkit is deployed to hide processes and artifacts.

Confidence

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

15 / 100

Content

N/A

Indicator

Name
get.bi-chi.com
Pattern Type
stix
Pattern
[hostname:value = 'get.bi-chi.com']
Name
t00ls.ru
Pattern Type
stix
Pattern
[domain-name:value = 't00ls.ru']
Name
11.177.125.116

Pattern Type

stix

Pattern

[ipv4-addr:value = '11.177.125.116']

Name

183.2.143.163

Description

ISP: CHINANET-BACKBONE **OS:** - ---------- Services: **80:** ``` HTTP/1.1 400 Bad Request Server: stgw Date: Mon, 29 Jan 2024 06:05:00 GMT Content-Type: text/html Content-Length: 649 Connection: close ``` ---------- **443:** ``` HTTP/1.1 403 Forbidden Date: Fri, 16 Feb 2024 07:35:43 GMT Content-Type: application/json; charset=utf-8 Content-Length: 47 Connection: keep-alive Server: openresty ``` HEARTBLEED: 2024/02/16 07:36:31 183.2.143.163:443 - SAFE ---------- **8080:** ``` HTTP/1.1 400 Bad Request Server: stgw Date: Sun, 11 Feb 2024 22:26:44 GMT Content-Type: text/html Content-Length: 649 Connection: close ``` ---------- **9080:** ``` HTTP/1.1 400 Bad Request Server: stgw Date: Fri, 16 Feb 2024 03:51:45 GMT Content-Type: text/html Content-Length: 649 Connection: close ``` ----------

Pattern Type

stix

Pattern

[ipv4-addr:value = '183.2.143.163']

Name

120.232.65.223

Description

ISP: China Mobile Communications Group Co., Ltd. **OS:** - ---------- Services: **80:** ``` HTTP/1.1 400 Bad Request Server: stgw Date: Mon, 19 Feb 2024 09:58:37 GMT Content-Type: text/html Content-Length: 649 Connection: close ``` ---------- **443:** ``` HTTP/1.1 403 Forbidden Date: Fri, 16 Feb 2024 04:46:17 GMT Content-Type: application/json; charset=utf-8 Content-Length: 47 Connection: keep-alive Server: openresty ``` HEARTBLEED: 2024/02/16 04:46:41 120.232.65.223:443 - SAFE ---------- **8080:** ``` HTTP/1.1 400 Bad Request Server: stgw Date: Mon, 19 Feb 2024 11:15:35 GMT Content-Type: text/html Content-Length: 649 Connection: close ``` ---------- **9080:** ``` HTTP/1.1 400 Bad Request Server: stgw Date: Sat, 17 Feb 2024 21:31:41 GMT Content-Type: text/html Content-Length: 649 Connection: close ``` ----------

Pattern Type

stix

Pattern

[ipv4-addr:value = '120.232.65.223']

Name

11.177.124.86

Pattern Type

stix

Pattern

[ipv4-addr:value = '11.177.124.86']

Name

11.149.252.62

Pattern Type

stix

Pattern

[ipv4-addr:value = '11.149.252.62']

Name

11.149.252.57

Pattern Type

stix

Pattern

[ipv4-addr:value = '11.149.252.57']

Name

c5dc12dbb9bb51ea8acf93d6349d5bc7fe5ee11b68d6371c1bbb098e21d0f685

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = 'c5dc12dbb9bb51ea8acf93d6349d5bc7fe5ee11b68d6371c1bbb098e21d0f685']

Name

8cce669c8f9c5304b43d6e91e6332b1cf1113c81f355877dabd25198c3c3f208

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '8cce669c8f9c5304b43d6e91e6332b1cf1113c81f355877dabd25198c3c3f208']

Name

76ecd546374b24443d76c450cb8ed7226db84681ee725482d5b9ff4ce3273c7f

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '76ecd546374b24443d76c450cb8ed7226db84681ee725482d5b9ff4ce3273c7f']

Name

5dc4a48ebd4f4be7ffcf3d2c1e1ae4f2640e41ca137a58dbb33b0b249b68759e

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '5dc4a48ebd4f4be7ffcf3d2c1e1ae4f2640e41ca137a58dbb33b0b249b68759e']

Name

364a7f8e3701a340400d77795512c18f680ee67e178880e1bb1fcda36ddbc12c

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '364a7f8e3701a340400d77795512c18f680ee67e178880e1bb1fcda36ddbc12c']

Name

2b03943244871ca75e44513e4d20470b8f3e0f209d185395de82b447022437ec

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '2b03943244871ca75e44513e4d20470b8f3e0f209d185395de82b447022437ec']

Name

32d32bf0be126e685e898d0ac21d93618f95f405c6400e1c8b0a8a72aa753933

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '32d32bf0be126e685e898d0ac21d93618f95f405c6400e1c8b0a8a72aa753933']

Name

157.148.45.20

Description

ISP: China Unicom Guangdong IP network **OS:** - ---------- Services: **80:** ``` HTTP/1.1 400 Bad Request Server: stgw Date: Fri, 16 Feb 2024 12:08:34 GMT Content-Type: text/html Content-Length: 649 Connection: close ``` ---------- **443:** ``` HTTP/1.1 403 Forbidden Date: Mon, 19 Feb 2024 05:25:05 GMT Content-Type: application/json; charset=utf-8 Content-Length: 47 Connection: keep-alive Server: openresty ``` HEARTBLEED: 2024/02/19 05:25:21 157.148.45.20:443 - SAFE ---------- **8080:** ``` HTTP/1.1 400 Bad Request Server: stgw Date: Sun, 18 Feb 2024 23:47:28 GMT Content-Type: text/html Content-Length: 649 Connection: close ``` ---------- **9080:** ``` HTTP/1.1 400 Bad Request Server: stgw Date: Tue, 13 Feb 2024 01:37:54 GMT Content-Type: text/html Content-Length: 649 Connection: close ``` ----------

Pattern Type

stix

Pattern

[ipv4-addr:value = '157.148.45.20']

Name

11.149.252.51

Pattern Type

stix

Pattern

[ipv4-addr:value = '11.149.252.51']

Name

103.79.118.221

Description

ISP: QuadraNet Enterprises LLC **OS:** Ubuntu ---------- Services: **22:** ``` SSH-2.0-OpenSSH_8.9p1 Ubuntu-3 Key type: ssh-rsa Key: AAAAB3NzaC1yc2EAAAADAQABAAABgQC3Bero6d9QzAiS4GueLAD3YIb71O+JFaFjJYUdr2fXhzquU5IaldpgICacciu9ANp0VepfgWjsx1ts+9Bph4LcLcfvh/QpGGAwAKA3WKhqRi69DhheSPklz9Q4+waGLir+2TkcQEeEFllnjeWhbck4zmzknA+9QKbc7AK7uSvSPSBSedZCTOp0gZIGkvBFMKCpMHw83sDu52TRvsK4qhKayIl+oCyzn6sSuwE89hSuhWju130Xrud2e6FWg4Hjafx+7e+zqSeuV4pPhDLxC2EQM+pFRHGNLyo5+MfAn+TLtNd1sUuTPspexBoBga5pD8Ra/vyjU8RFPPO8tAuYZVFMO9WSkpSB OgMQuC0anm+SDMQEHpCzSUpYSE0/ksUC6WJ1idS3xR1WaNmSuHqlGp7XZH0hSXmP2F0yXozxIdYH ptumbt5cTIGlozw1TVWHeqU/9GNFoUXfu0fCykThkrbxHmdbFGNs7XCXqqWmJJmc9ntMLMr+fK0U +/bSp/LFgcU= Fingerprint: df:c7:87:0b:d4:9b:74:9f:58:5d:a4:61:4a:bf:f9:88 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 Server Host Key Algorithms: rsa-sha2-512 rsa-sha2-256 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 ``` ----------

Pattern Type

stix
Pattern
[ipv4-addr:value = '103.79.118.221']

Malware

Name
Migo

Name
linux

Intrusion-Set

Name
Migo

Attack-Pattern

Name
DLL Side-Loading
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
Windows Command Shell
ID

T1059.003

Description

Adversaries may abuse the Windows command shell for execution. The Windows command shell ([cmd](https://attack.mitre.org/software/S0106)) is the primary command prompt on Windows systems. The Windows command prompt can be used to control almost any aspect of a system, with various permission levels required for different subsets of commands. The command prompt can be invoked remotely via [Remote Services](https://attack.mitre.org/techniques/T1021) such as [SSH](https://attack.mitre.org/techniques/T1021/004).(Citation: SSH in Windows) Batch files (ex: .bat or .cmd) also provide the shell with a list of sequential commands to run, as well as normal scripting operations such as conditionals and loops. Common uses of batch files include long or repetitive tasks, or the need to run the same set of commands on multiple systems. Adversaries may leverage [cmd](https://attack.mitre.org/software/S0106) to execute various commands and payloads. Common uses include [cmd](https://attack.mitre.org/software/S0106) to execute a single command, or abusing [cmd](https://attack.mitre.org/software/S0106) interactively with input and output forwarded over a command and control channel.

Name

Component Object Model Hijacking

ID

T1546.015

Description

Adversaries may establish persistence by executing malicious content triggered by hijacked references to Component Object Model (COM) objects. COM is a system within Windows to enable interaction between software components through the operating system.(Citation: Microsoft Component Object Model) References to various COM objects are stored in the Registry. Adversaries can use the COM system to insert malicious code that can be executed in place of legitimate software through hijacking the COM references and relationships as a means for persistence. Hijacking a COM object requires a change in the Registry to replace a reference to a legitimate system component which may cause that component to not work when executed. When that system component is executed through normal system operation the adversary's code will be executed instead.(Citation:

GDATA COM Hijacking) An adversary is likely to hijack objects that are used frequently enough to maintain a consistent level of persistence, but are unlikely to break noticeable functionality within the system as to avoid system instability that could lead to detection.

Name

PowerShell

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

Deobfuscate/Decode Files or Information

ID

T1140

Description

Adversaries may use [Obfuscated Files or Information](https://attack.mitre.org/techniques/T1027) to hide artifacts of an intrusion from analysis. They may require separate mechanisms to decode or deobfuscate that information depending on how they intend to use it. Methods for doing that include built-in functionality of malware or by using utilities present on the system. One such example is the use of [certutil](https://attack.mitre.org/software/S0160) to decode a remote access tool portable executable file that has been hidden inside a certificate file.(Citation: Malwarebytes Targeted Attack against Saudi Arabia) Another example is using the Windows `copy /b` command to reassemble binary fragments into a malicious payload.(Citation: Carbon Black Obfuscation Sept 2016) Sometimes a user's action may be required to open it for deobfuscation or decryption as part of [User Execution](https://attack.mitre.org/techniques/T1204). The user may also be required to input a password to open a password protected compressed/encrypted file that was provided by the adversary. (Citation: Volexity PowerDuke November 2016)

Name

Windows Service

ID

T1543.003

Description

Adversaries may create or modify Windows services to repeatedly execute malicious payloads as part of persistence. When Windows boots up, it starts programs or applications called services that perform background system functions.(Citation: TechNet Services) Windows service configuration information, including the file path to the service's executable or recovery programs/commands, is stored in the Windows Registry. Adversaries may install a new service or modify an existing service to execute at startup in order to persist on a system. Service configurations can be set or modified using system utilities (such as sc.exe), by directly modifying the Registry, or by interacting directly with the Windows API. Adversaries may also use services to install and execute malicious drivers. For example, after dropping a driver file (ex: `.sys`) to disk, the payload can be

loaded and registered via [Native API](https://attack.mitre.org/techniques/T1106) functions such as `CreateServiceW()` (or manually via functions such as `ZwLoadDriver()` and `ZwSetValueKey()`), by creating the required service Registry values (i.e. [Modify Registry](https://attack.mitre.org/techniques/T1112)), or by using command-line utilities such as `PnPUtil.exe`.(Citation: Symantec W.32 Stuxnet Dossier)(Citation: Crowdstrike DriveSlayer February 2022)(Citation: Unit42 AcidBox June 2020) Adversaries may leverage these drivers as [Rootkit](https://attack.mitre.org/techniques/T1014)s to hide the presence of malicious activity on a system. Adversaries may also load a signed yet vulnerable driver onto a compromised machine (known as "Bring Your Own Vulnerable Driver" (BYOVD)) as part of [Exploitation for Privilege Escalation](https://attack.mitre.org/techniques/T1068).(Citation: ESET InvisiMole June 2020)(Citation: Unit42 AcidBox June 2020) Services may be created with administrator privileges but are executed under SYSTEM privileges, so an adversary may also use a service to escalate privileges. Adversaries may also directly start services through [Service Execution](https://attack.mitre.org/techniques/T1569/002). To make detection analysis more challenging, malicious services may also incorporate [Masquerade Task or Service](https://attack.mitre.org/techniques/T1036/004) (ex: using a service and/or payload name related to a legitimate OS or benign software component).

Sector

Name
Technologies
Description
Private entities related to the research, development, manufacturing and distribution of electronics, softwares, computers and products related to information technologies.

Hostname

Value
get.bi-chi.com

Domain-Name

Value
t00ls.ru

IPv4-Addr

Value
183.2.143.163
120.232.65.223
11.177.125.116
11.177.124.86
11.149.252.62
11.149.252.57
103.79.118.221
11.149.252.51
157.148.45.20

StixFile

Value
c5dc12dbb9bb51ea8acf93d6349d5bc7fe5ee11b68d6371c1bbb098e21d0f685
8cce669c8f9c5304b43d6e91e6332b1cf1113c81f355877dabd25198c3c3f208
76ecd546374b24443d76c450cb8ed7226db84681ee725482d5b9ff4ce3273c7f
5dc4a48ebd4f4be7ffcf3d2c1e1ae4f2640e41ca137a58dbb33b0b249b68759e
364a7f8e3701a340400d77795512c18f680ee67e178880e1bb1fcda36ddbc12c
2b03943244871ca75e44513e4d20470b8f3e0f209d185395de82b447022437ec
32d32bf0be126e685e898d0ac21d93618f95f405c6400e1c8b0a8a72aa753933

External References

- https://www.cadosecurity.com/migo-a-redis-miner-with-novel-system-weakening-techniques/
- https://otx.alienvault.com/pulse/65d4de3f0bebe853a542f3b2