Oct 29 2023

NETMANAGEIT

Intelligence Report

WatchDog Mining Organization's Activity Analysis

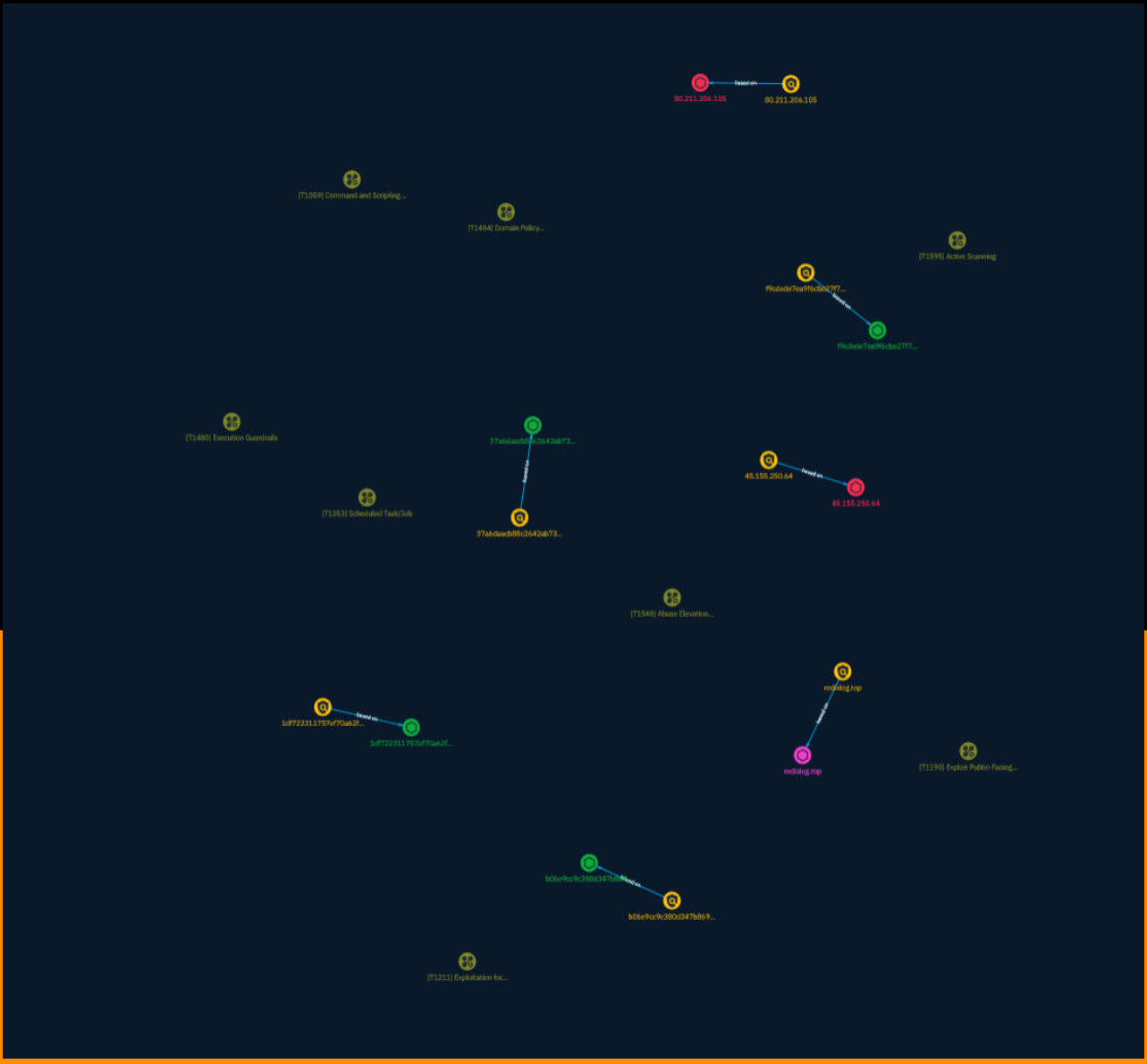

Table of contents

Overview

Entities

Observables

External References

Overview

Description

Recently, Antian CERT has captured a number of active WatchDog data samples. The organization mainly uses exposed Docker Engine API endpoints and Redis servers to launch attacks. The WatchDog cryptojacking org has been discovered since January 2019 and is still active.

Confidence

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

15 / 100

Content

N/A

Attack-Pattern

Name
Abuse Elevation Control Mechanism
ID
T1548
Description
Adversaries may circumvent mechanisms designed to control elevate privileges to gain higher-level permissions. Most modern systems contain native elevation control mechanisms that are intended to limit privileges that a user can perform on a machine. Authorization has to be granted to specific users in order to perform tasks that can be considered of higher risk. An adversary can perform several methods to take advantage of built-in control mechanisms in order to escalate privileges on a system.
Name
Scheduled Task/Job
ID
T1053
Description
Adversaries may abuse task scheduling functionality to facilitate initial or recurring execution of malicious code. Utilities exist within all major operating systems to schedule

programs or scripts to be executed at a specified date and time. A task can also be scheduled on a remote system, provided the proper authentication is met (ex: RPC and file and printer sharing in Windows environments). Scheduling a task on a remote system typically may require being a member of an admin or otherwise privileged group on the remote system.(Citation: TechNet Task Scheduler Security) Adversaries may use task scheduling to execute programs at system startup or on a scheduled basis for persistence. These mechanisms can also be abused to run a process under the context of a specified account (such as one with elevated permissions/privileges). Similar to [System Binary Proxy Execution](https://attack.mitre.org/techniques/T1218), adversaries have also abused task scheduling to potentially mask one-time execution under a trusted system process. (Citation: ProofPoint Serpent)

Name

Domain Policy Modification

ID

T1484

Description

Adversaries may modify the configuration settings of a domain to evade defenses and/or escalate privileges in domain environments. Domains provide a centralized means of managing how computer resources (ex: computers, user accounts) can act, and interact with each other, on a network. The policy of the domain also includes configuration settings that may apply between domains in a multi-domain/forest environment. Modifications to domain settings may include altering domain Group Policy Objects (GPOs) or changing trust settings for domains, including federation trusts. With sufficient permissions, adversaries can modify domain policy settings. Since domain configuration settings control many of the interactions within the Active Directory (AD) environment, there are a great number of potential attacks that can stem from this abuse. Examples of such abuse include modifying GPOs to push a malicious [Scheduled Task](https://attack.mitre.org/techniques/T1053/005) to computers throughout the domain environment(Citation: ADSecurity GPO Persistence 2016)(Citation: Wald0 Guide to GPOs) (Citation: Harmj0y Abusing GPO Permissions) or modifying domain trusts to include an adversary controlled domain where they can control access tokens that will subsequently be accepted by victim domain resources.(Citation: Microsoft - Customer Guidance on Recent Nation-State Cyber Attacks) Adversaries can also change configuration settings within the AD environment to implement a [Rogue Domain Controller](https://attack.mitre.org/techniques/T1207). Adversaries may temporarily modify domain policy,

carry out a malicious action(s), and then revert the change to remove suspicious indicators.

Name

Exploit Public-Facing Application

ID

T1190

Description

Adversaries may attempt to exploit a weakness in an Internet-facing host or system to initially access a network. The weakness in the system can be a software bug, a temporary glitch, or a misconfiguration. Exploited applications are often websites/web servers, but can also include databases (like SQL), standard services (like SMB or SSH), network device administration and management protocols (like SNMP and Smart Install), and any other system with Internet accessible open sockets.(Citation: NVD CVE-2016-6662)(Citation: CIS Multiple SMB Vulnerabilities)(Citation: US-CERT TA18-106A Network Infrastructure Devices 2018)(Citation: Cisco Blog Legacy Device Attacks)(Citation: NVD CVE-2014-7169) Depending on the flaw being exploited this may also involve [Exploitation for Defense Evasion](https://attack.mitre.org/techniques/T1211). If an application is hosted on cloud-based infrastructure and/or is containerized, then exploiting it may lead to compromise of the underlying instance or container. This can allow an adversary a path to access the cloud or container APIs, exploit container host access via [Escape to Host](https://attack.mitre.org/techniques/T1611), or take advantage of weak identity and access management policies. Adversaries may also exploit edge network infrastructure and related appliances, specifically targeting devices that do not support robust host-based defenses.(Citation: Mandiant Fortinet Zero Day)(Citation: Wired Russia Cyberwar) For websites and databases, the OWASP top 10 and CWE top 25 highlight the most common web-based vulnerabilities. (Citation: OWASP Top 10)(Citation: CWE top 25)

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

Active Scanning

ID

T1595

Description

Adversaries may execute active reconnaissance scans to gather information that can be used during targeting. Active scans are those where the adversary probes victim infrastructure via network traffic, as opposed to other forms of reconnaissance that do not involve direct interaction. Adversaries may perform different forms of active scanning depending on what information they seek to gather. These scans can also be performed in various ways, including using native features of network protocols such as ICMP.(Citation: Botnet Scan)(Citation: OWASP Fingerprinting) Information from these scans may reveal opportunities for other forms of reconnaissance (ex: [Search Open Websites/Domains](https://attack.mitre.org/techniques/T1593) or [Search Open Technical Databases](https://

attack.mitre.org/techniques/T1596)), establishing operational resources (ex: [Develop Capabilities](https://attack.mitre.org/techniques/T1587) or [Obtain Capabilities](https://attack.mitre.org/techniques/T1588)), and/or initial access (ex: [External Remote Services](https://attack.mitre.org/techniques/T1133) or [Exploit Public-Facing Application](https://attack.mitre.org/techniques/T1190)).

Name

Exploitation for Defense Evasion

ID

T1211

Description

Adversaries may exploit a system or application vulnerability to bypass security features. Exploitation of a software vulnerability occurs when an adversary takes advantage of a programming error in a program, service, or within the operating system software or kernel itself to execute adversary-controlled code. Vulnerabilities may exist in defensive security software that can be used to disable or circumvent them. Adversaries may have prior knowledge through reconnaissance that security software exists within an environment or they may perform checks during or shortly after the system is compromised for [Security Software Discovery](https://attack.mitre.org/techniques/T1518/001). The security software will likely be targeted directly for exploitation. There are examples of antivirus software being targeted by persistent threat groups to avoid detection.

Name

Command and Scripting Interpreter

ID

T1059

Description

Adversaries may abuse command and script interpreters to execute commands, scripts, or binaries. These interfaces and languages provide ways of interacting with computer systems and are a common feature across many different platforms. Most systems come with some built-in command-line interface and scripting capabilities, for example, macOS and Linux distributions include some flavor of [Unix Shell](https://attack.mitre.org/techniques/T1059/004) while Windows installations include the [Windows Command Shell](https://attack.mitre.org/techniques/T1059/003) and [PowerShell](https://attack.mitre.org/techniques/T1059/001). There are also cross-platform interpreters such as [Python](https://attack.mitre.org/techniques/T1059/006), as well as those commonly associated with client applications such as [JavaScript](https://attack.mitre.org/techniques/T1059/007) and [Visual Basic](https://attack.mitre.org/techniques/T1059/005). Adversaries may abuse these technologies in various ways as a means of executing arbitrary commands. Commands and scripts can be embedded in [Initial Access](https://attack.mitre.org/tactics/TA0001) payloads delivered to victims as lure documents or as secondary payloads downloaded from an existing C2. Adversaries may also execute commands through interactive terminals/shells, as well as utilize various [Remote Services](https://attack.mitre.org/techniques/T1021) in order to achieve remote Execution. (Citation: Powershell Remote Commands)(Citation: Cisco IOS Software Integrity Assurance - Command History)(Citation: Remote Shell Execution in Python)

Indicator

Name

45.155.250.64

Description

ISP: GleSYS AB **OS:** None ---------- Hostnames: ---------- Domains: ---------- Services: **22:** ``` SSH-2.0-OpenSSH_8.9p1 Ubuntu-3ubuntu0.3 Key type: ecdsa-sha2-nistp256 Key: AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBGFC9wDOrS6TX5+A2jm6v7V3 hXmHB42aF4XfjIba2r9rErrKE/iiYRf7tf/CoL94VVslmh+HIuevNJ9pXqgmYm8= Fingerprint: 4a:a0:4b:33:6e:7a:a8:98:c3:1a:13:6c:8f:13:03:a0 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 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 ``` ---------- **80:** ``` HTTP/1.1 200 OK Date: Thu, 19 Oct 2023 09:19:25 GMT Server: Apache/2.4.52 (Ubuntu) Last-Modified: Wed, 12 Apr 2023 22:12:48 GMT ETag: "a5-5f92ae88abb48" Accept-Ranges: bytes Content-Length: 165 Vary: Accept-Encoding Content-Type: text/html ``` ---------- **443:** ``` HTTP/1.1 200 OK Date: Tue, 24 Oct 2023 09:58:54 GMT Server: Apache/2.4.52 (Ubuntu) Last-Modified: Wed, 12 Apr 2023 22:12:48 GMT ETag: "a5-5f92ae88abb48" Accept-Ranges: bytes Content-Length: 165 Vary: Accept-Encoding Content-Type: text/html ``` ----------

Pattern Type

stix

Pattern

[ipv4-addr:value = '45.155.250.64']

Name

80.211.206.105

Description

ISP: INTERNET CZ, a.s. **OS:** None ---------- Hostnames: - 105.206.forpsi.net ---------- Domains: - forpsi.net ---------- Services: **80:** ``` HTTP/1.1 308 Permanent Redirect Connection: close Location: https://80.211.206.105/ Server: Caddy Date: Tue, 24 Oct 2023 22:10:24 GMT Content-Length: 0 ``` ---------- **5555:** ``` HTTP/1.1 200 OK Content-Type: text/plain Content-Length: 18 ``` ---------- **6666:** ``` HTTP/1.1 200 OK Content-Type: text/plain Content-Length: 18 ``` ---------- **7777:** ``` HTTP/1.1 200 OK Content-Type: text/plain Content-Length: 18 ``` ---------- **8888:** ``` HTTP/1.1 200 OK Content-Type: text/plain Content-Length: 18 ``` ---------- **9000:** ``` ``` ----------

Pattern Type

stix

Pattern

[ipv4-addr:value = '80.211.206.105']

Name

37a6daacb88c2642ab736d72256257945f8ab2b53203f00c704650da5efa721e

Description

is__elf SHA256 of bdb81ac3eb3a8ac27e11f3ab7703783d

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '37a6daacb88c2642ab736d72256257945f8ab2b53203f00c704650da5efa721e']

Name

f9cdede7ea9f6cbe27f7905bf9f830d75dd84d3916026f37b97b47805e2d0d36

Description

Unix.Dropper.Btcmine-9819414-0 SHA256 of 159d5ab60f9f7897cd9f0922d8318460

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = 'f9cdede7ea9f6cbe27f7905bf9f830d75dd84d3916026f37b97b47805e2d0d36']

Name

b06e9cc9c380d347b8696bdecb85f1bca6c4c0db2e6b79432828e3b07eebad3a

Description

Win.Trojan.Miner-6958808-0 SHA256 of 2ec4ae1aaabc5ba4b804706b72f8ce9b

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = 'b06e9cc9c380d347b8696bdecb85f1bca6c4c0db2e6b79432828e3b07eebad3a']

Name

1df722311757ef70a62fca6a68ad21d7f0d7b198aba755dd433b14183b198957

Description

Unix.Exploit.Exploitscan-9789947-0 SHA256 of 878a551c08da641024d87dc91ed92067

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '1df722311757ef70a62fca6a68ad21d7f0d7b198aba755dd433b14183b198957']

Name

redislog.top

Pattern Type

stix

Pattern

```
[domain-name:value = 'redislog.top']
```

Domain-Name

Value
redislog.top

StixFile

Value
f9cdede7ea9f6cbe27f7905bf9f830d75dd84d3916026f37b97b47805e2d0d36
b06e9cc9c380d347b8696bdecb85f1bca6c4c0db2e6b79432828e3b07eebad3a
37a6daacb88c2642ab736d72256257945f8ab2b53203f00c704650da5efa721e
1df722311757ef70a62fca6a68ad21d7f0d7b198aba755dd433b14183b198957

IPv4-Addr

Value
80.211.206.105
45.155.250.64

External References

- https://otx.alienvault.com/pulse/6537f2a810d99a320ca30f7d
- https://www.antiy.cn/research/notice&report/research_report/WatchDogTrojans_Analysis.html