

# NETMANAGEIT

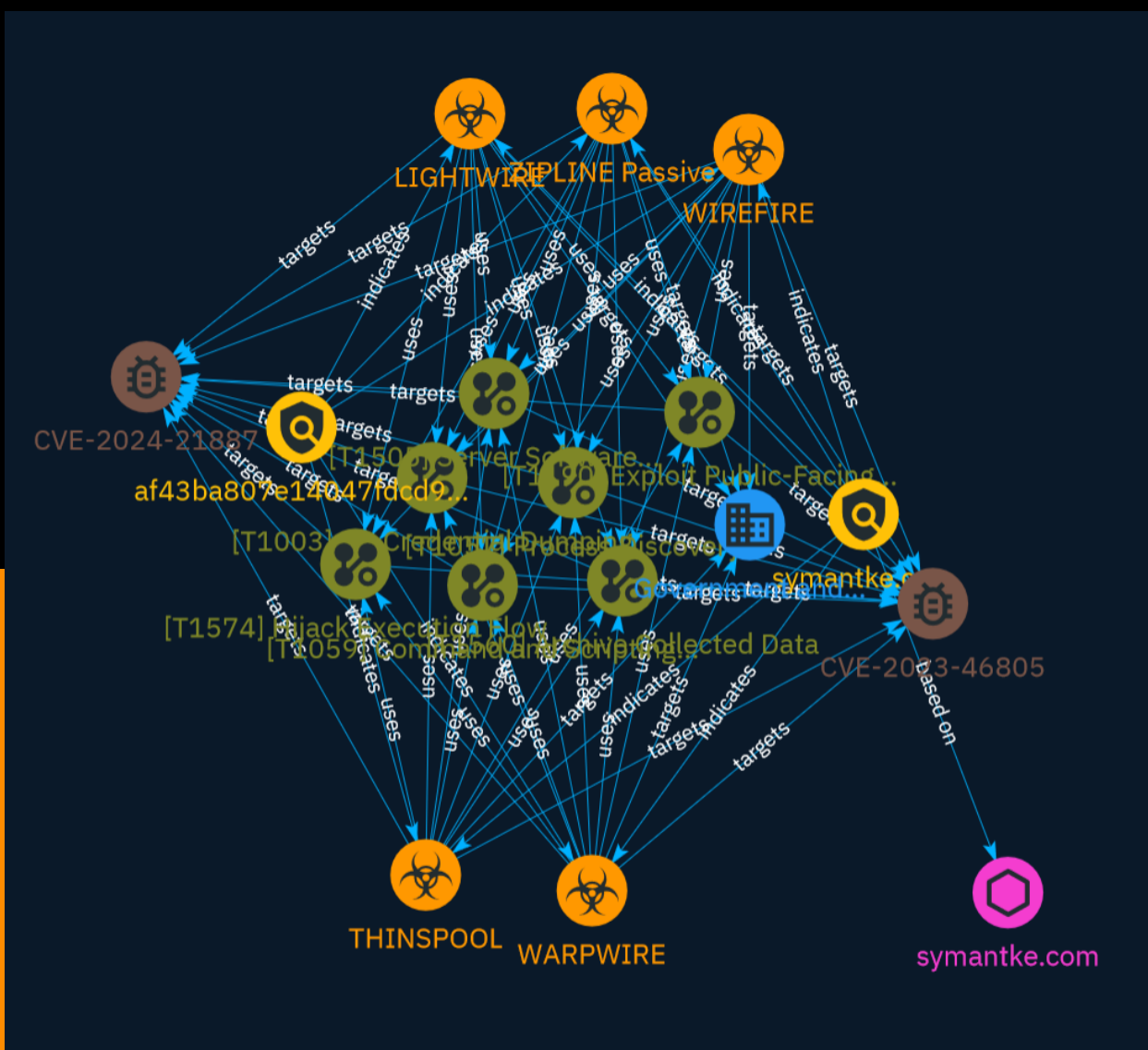
## Intelligence Report

### Suspected APT Targets

### Ivanti Connect Secure VPN

### in New Zero-Day

### Exploitation



# Table of contents

---

## Overview

---

● Description	4
● Confidence	4
● Content	5

---

## Entities

---

● Attack-Pattern	6
● Sector	11
● Indicator	12
● Malware	14
● Vulnerability	15

---

## Observables

---

● Domain-Name	16
---------------	----



## External References

- 
- External References

17

# Overview

## Description

Mandiant is sharing details of five malware families associated with the exploitation of CS and PS devices. These families allow the threat actors to circumvent authentication and provide backdoor access to these devices. Additional post-exploitation tools have also been identified in our investigation and are highlighted further in this post.

## Confidence

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

15 / 100

# Content

N/A

# Attack-Pattern

## Name

Process Discovery

## ID

T1057

## Description

Adversaries may attempt to get information about running processes on a system. Information obtained could be used to gain an understanding of common software/ applications running on systems within the network. Adversaries may use the information from [Process Discovery](https://attack.mitre.org/techniques/T1057) during automated discovery to shape follow-on behaviors, including whether or not the adversary fully infects the target and/or attempts specific actions. In Windows environments, adversaries could obtain details on running processes using the [Tasklist](https://attack.mitre.org/software/S0057) utility via [cmd](https://attack.mitre.org/software/S0106) or `Get-Process` via [PowerShell](https://attack.mitre.org/techniques/T1059/001). Information about processes can also be extracted from the output of [Native API](https://attack.mitre.org/techniques/T1106) calls such as `CreateToolhelp32Snapshot`. In Mac and Linux, this is accomplished with the `ps` command. Adversaries may also opt to enumerate processes via `/proc`. On network devices, [Network Device CLI](https://attack.mitre.org/techniques/T1059/008) commands such as `show processes` can be used to display current running processes.(Citation: US-CERT-TA18-106A)(Citation: show\_processes\_cisco\_cmd)

## Name

OS Credential Dumping

**ID**

T1003

**Description**

Adversaries may attempt to dump credentials to obtain account login and credential material, normally in the form of a hash or a clear text password, from the operating system and software. Credentials can then be used to perform [Lateral Movement](<https://attack.mitre.org/tactics/TA0008>) and access restricted information. Several of the tools mentioned in associated sub-techniques may be used by both adversaries and professional security testers. Additional custom tools likely exist as well.

**Name**

Server Software Component

**ID**

T1505

**Description**

Adversaries may abuse legitimate extensible development features of servers to establish persistent access to systems. Enterprise server applications may include features that allow developers to write and install software or scripts to extend the functionality of the main application. Adversaries may install malicious components to extend and abuse server applications.(Citation: volexity\_0day\_sophos\_FW)

**Name**

Archive Collected Data

**ID**

T1560

**Description**

An adversary may compress and/or encrypt data that is collected prior to exfiltration. Compressing the data can help to obfuscate the collected data and minimize the amount of data sent over the network. Encryption can be used to hide information that is being exfiltrated from detection or make exfiltration less conspicuous upon inspection by a defender. Both compression and encryption are done prior to exfiltration, and can be performed using a utility, 3rd party library, or custom method.

**Name**

Hijack Execution Flow

**ID**

T1574

**Description**

Adversaries may execute their own malicious payloads by hijacking the way operating systems run programs. Hijacking execution flow can be for the purposes of persistence, since this hijacked execution may reoccur over time. Adversaries may also use these mechanisms to elevate privileges or evade defenses, such as application control or other restrictions on execution. There are many ways an adversary may hijack the flow of execution, including by manipulating how the operating system locates programs to be executed. How the operating system locates libraries to be used by a program can also be intercepted. Locations where the operating system looks for programs/resources, such as file directories and in the case of Windows the Registry, could also be poisoned to include malicious payloads.

**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**

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)

# Sector

**Name**

Government and administrations

**Description**

Civilian government institutions and administrations of the executive and legislative branches. The diplomatic and judicial branches are not included.

# Indicator

## Name

af43ba807e14047fdcd92519016c0e53e0f124fd

## Description

This rule detects unique strings in ZIPLINE, a passive ELF backdoor that waits for incoming TCP connections to receive commands from the threat actor.

## Pattern Type

yara

## Pattern

```
rule M_Hunting_Backdoor_ZIPLINE_1 { meta: author = "Mandiant" description = "This rule detects unique strings in ZIPLINE, a passive ELF backdoor that waits for incoming TCP connections to receive commands from the threat actor." strings: $s1 = "SSH-2.0-OpenSSH_0.3xx" ascii $s2 = "$(exec $installer $@)" ascii $t1 = "./installer/do-install" ascii $t2 = "./installer/bom_files/" ascii $t3 = "/tmp/data/root/etc/ld.so.preload" ascii $t4 = "/tmp/data/root/home/etc/manifest/exclusion_list" ascii condition: uint32(0) == 0x464c457f and filesize < 5MB and ((1 of ($s*)) or (3 of ($t*))) }
```

## Name

symantke.com

## Description

UTA0178 domain used to collect credentials from compromised devices

**Pattern Type**

stix

**Pattern**

[domain-name:value = 'symantke.com']

# Malware

**Name**

WARPWIRE

**Name**

LIGHTWIRE

**Name**

WIREFIRE

**Name**

THINSPOOL

**Name**

ZIPLINE Passive

# Vulnerability

## Name

CVE-2023-46805

## Description

Ivanti Connect Secure (ICS, formerly known as Pulse Connect Secure) and Ivanti Policy Secure gateways contain an authentication bypass vulnerability in the web component that allows an attacker to access restricted resources by bypassing control checks. This vulnerability can be leveraged in conjunction with CVE-2024-21887, a command injection vulnerability.

## Name

CVE-2024-21887

## Description

Ivanti Connect Secure (ICS, formerly known as Pulse Connect Secure) and Ivanti Policy Secure contain a command injection vulnerability in the web components of these products, which can allow an authenticated administrator to send crafted requests to execute code on affected appliances. This vulnerability can be leveraged in conjunction with CVE-2023-46805, an authenticated bypass issue.

# Domain-Name

## Value

symantke.com



# External References

- 
- <https://www.mandiant.com/resources/blog/suspected-apt-targets-ivanti-zero-day>
- 
- <https://otx.alienvault.com/pulse/65a50f3d68cb69771e53fd45>