May 10 2024

NETMANAGEIT

Intelligence Report

Surge of JavaScript Malware in sites with vulnerable versions of LiteSpeed Cache Plugin

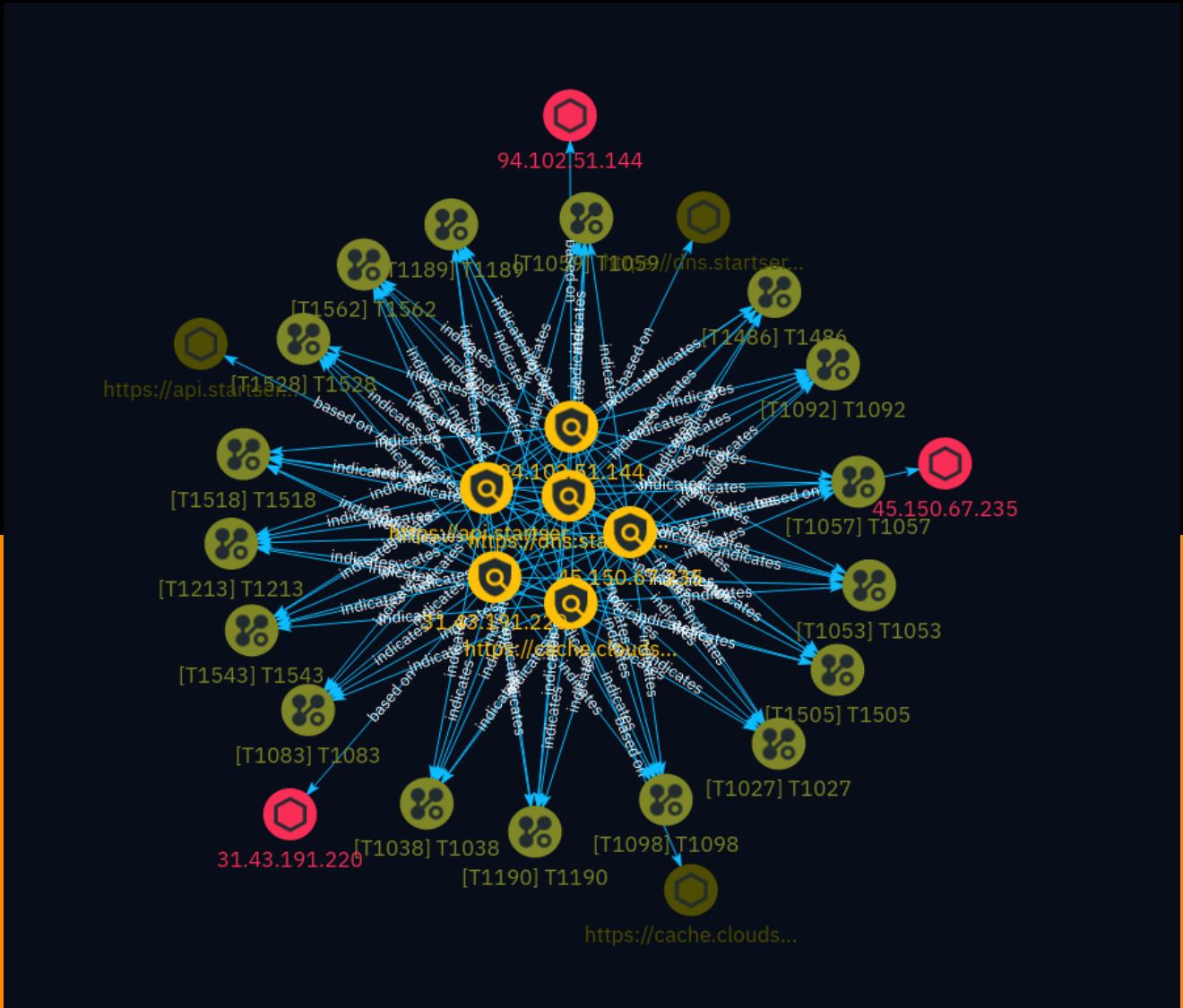

Table of contents

Overview

Entities

Observables

External References

Overview

Description

A recent surge of malicious JavaScript code has been observed targeting websites using vulnerable versions of the LiteSpeed Cache plugin for WordPress. The malware injects code into critical WordPress files or the database, creating unauthorized admin users like 'wpsupp-user.' It exploits the vulnerability in LiteSpeed Cache before version 5.7.0.1, allowing attackers to inject malicious scripts. The malware is often associated with URLs like 'https://dns.startservicefounds.com/service/f.php' and IPs like 45.150.67.235 or 94.102.51.144. Website owners should review installed plugins, update them, and search for suspicious code or users.

Confidence

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

100 / 100

Content

N/A

Indicator

Name

https://dns.startservicefounds.com/service/f.php

Pattern Type

stix

Pattern

[url:value = 'https://dns.startservicefounds.com/service/f.php']

Name

https://cache.cloudswiftcdn.com

Description

- **Unsafe:** False - **Server:** cloudflare - **Domain Rank:** 0 - **DNS Valid:** True - **Parking:** False - **Spamming:** False - **Malware:** False - **Phishing:** False - **Suspicious:** True - **Adult:** False - **Category:** N/A - **Domain Age:** {'human': '6 months ago', 'timestamp': 1699563002, 'iso': '2023-11-09T15:50:02-05:00'} - **IPQS: Domain:** cache.cloudswiftcdn.com - **IPQS: IP Address:** 104.21.59.254

Pattern Type

stix

Pattern

[url:value = 'https://cache.cloudswiftcdn.com']

Name

https://api.startservicefounds.com

Description

- **Unsafe:** True - **Server:** N/A - **Domain Rank:** 0 - **DNS Valid:** True - **Parking:** False - **Spamming:** False - **Malware:** False - **Phishing:** True - **Suspicious:** True - **Adult:** False - **Category:** Web Tracker - **Domain Age:** {'human': '2 months ago', 'timestamp': 1709060343, 'iso': '2024-02-27T13:59:03-05:00'} - **IPQS: Domain:** api.startservicefounds.com - **IPQS: IP Address:** 127.0.0.127

Pattern Type

stix

Pattern

[url:value = 'https://api.startservicefounds.com']

Name

45.150.67.235

Description

ISP: STARK INDUSTRIES SOLUTIONS LTD **OS:** - ---------- Services: **21:** ``` 220 Welcome! Please note that all activity is logged. 530 Login incorrect. 530 Please login with USER and PASS. 211-Features: UTF8 EPRT EPSV MDTM PASV PBSZ PROT REST STREAM SIZE TVFS 211 End ``` ---------- **22:** ``` SSH-2.0-OpenSSH_8.2p1 Key type: ssh-rsa Key: AAAAB3NzaC1yc2EAAAADAQABAAABgQDgUxYl/AI7fOFvNEGz4xt4ov7StifsKyC74oRP4U4B130Y DE+m9E8CXBEfMQSYHoI6X4Mx53glQEp/OpCY0VLJrYAAMoAoR5KTuJkuExVW0yLlgL23VBJRCyGN NempODBN09EHX3s57DFv5OcNwL/

wzX5Aj/yhbc/ezANjK8mbKswyNOBgZC+LtPOXyMlnknRyNLtB 0e9zEkJ0toWi1DPqZfD+/AElCqvToaFg+AtKpwk6u1QAsLn7ihJwDQXcHEbfq/6hQgRHPIjumTXskU8FRlb0pryVxeSUGpHmnCi0rKERruvyl+sGIHBHlsyXLNVl+eoKuJpsklSrd/obw6bo2HXVlpDKfypoHR4BHO/x8pUa+/66NiOpgfqNWS1xaDHxPMoqW2AP8IK5sRwdRc/G3a8ZX7ldMZXtYQSh1Zt2 9i4wb3o8Go44SsyDSAfznc/3YVFOfzG7uEgXIJhlUtPhFOW8fgwjYXsFWbg583SI1o82600t+OrK jGDU8pO3kbE= Fingerprint: 33:da:56:81:7c:03:53:b3:36:51:aa:71:b1:cc:0e:2a Kex Algorithms: curve25519-sha256 curve25519-sha256@libssh.org ecdh-sha2-nistp256 ecdh-sha2-nistp384 ecdh-sha2-nistp521 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 ssh-rsa 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 Server: nginx Date: Thu, 09 May 2024 02:05:16 GMT Content-Type: text/html; charset=utf-8 Content-Length: 2588 Connection: keep-alive Vary: Accept-Encoding Last-Modified: Tue, 27 Feb 2024 19:12:45 GMT ETag: "a1c-61261cfafebd0" Accept-Ranges: bytes Vary: Accept-Encoding ``` ---------- **443:** ``` HTTP/1.1 301 Moved Permanently Server: nginx Date: Wed, 08 May 2024 13:19:48 GMT Content-Type: text/html Content-Length: 162 Connection: keep-alive Location: http://45.150.67.235/ ``` HEARTBLEED: 2024/05/08 13:20:01 45.150.67.235:443 - SAFE ---------- **8083:** ``` HTTP/1.1 200 OK Server: nginx Date: Wed, 08 May 2024 06:25:59 GMT Content-Type: text/html; charset=UTF-8 Transfer-Encoding: chunked Connection: keep-alive Vary: Accept-Encoding Set-Cookie: PHPSESSID=6pr6pl4b8df4varjs73plbh53t; path=/; secure; HttpOnly Expires: Thu, 19 Nov 1981 08:52:00 GMT Cache-Control: no-store, no-cache, must-revalidate Pragma: no-cache X-Content-Type-Options: nosniff X-Frame-Options: SAMEORIGIN X-XSS-Protection: 1; mode=block ``` HEARTBLEED: 2024/05/08 06:26:10 45.150.67.235:8083 - SAFE ----------

Pattern Type

stix

Pattern

[ipv4-addr:value = '45.150.67.235']

Name

31.43.191.220

Description

ISP: Telkom Internet LTD **OS:** - ---------- Services: **80:** ``` HTTP/1.1 200 OK Server: nginx/1.25.3 Date: Thu, 02 May 2024 14:17:37 GMT Content-Type: text/html Content-Length: 612 Last-Modified: Sat, 18 Nov 2023 06:29:54 GMT Connection: keep-alive ETag: "655859e2-264" Accept-Ranges: bytes ``` ----------

Pattern Type

stix

Pattern

[ipv4-addr:value = '31.43.191.220']

Name

94.102.51.144

Description

- **Zip Code:** N/A - **ISP:** IP Volume - **ASN:** 202425 - **Organization:** IP Volume - **Is Crawler:** False - **Timezone:** Europe/Amsterdam - **Mobile:** False - **Host:** 94.102.51.144 - **Proxy:** True - **VPN:** False - **TOR:** False - **Active VPN:** False - **Active TOR:** False - **Recent Abuse:** True - **Bot Status:** False - **Connection Type:** Premium required. - **Abuse Velocity:** Premium required. - **Country Code:** NL - **Region:** North Holland - **City:** Amsterdam - **Latitude:** 52.34999847 - **Longitude:** 4.92000008

Pattern Type

stix

Pattern

```
[ipv4-addr:value = '94.102.51.144']
```

Attack-Pattern

Name
T1038
ID
T1038
Name
T1189
ID
T1189
Description
Adversaries may gain access to a system through a user visiting a website over the normal course of browsing. With this technique, the user's web browser is typically targeted for exploitation, but adversaries may also use compromised websites for non-exploitation behavior such as acquiring [Application Access Token](https://attack.mitre.org/techniques/T1550/001). Multiple ways of delivering exploit code to a browser exist (i.e., [Drive-by Target](https://attack.mitre.org/techniques/T1608/004)), including: * A legitimate website is compromised where adversaries have injected some form of malicious code such as JavaScript, iFrames, and cross-site scripting * Script files served to a legitimate website from a publicly writeable cloud storage bucket are modified by an adversary * Malicious ads are paid for and served through legitimate ad providers (i.e., [Malvertising](https://attack.mitre.org/techniques/T1583/008)) * Built-in web application interfaces are

leveraged for the insertion of any other kind of object that can be used to display web content or contain a script that executes on the visiting client (e.g. forum posts, comments, and other user controllable web content). Often the website used by an adversary is one 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 often referred to a strategic web compromise or watering hole attack. There are several known examples of this occurring.(Citation: Shadowserver Strategic Web Compromise) Typical drive-by compromise process: 1. A user visits a website that is used to host the adversary controlled content. 2. Scripts automatically execute, typically searching versions of the browser and plugins for a potentially vulnerable version. * The user may be required to assist in this process by enabling scripting or active website components and ignoring warning dialog boxes. 3. Upon finding a vulnerable version, exploit code is delivered to the browser. 4. If exploitation is successful, then it will give the adversary code execution on the user's system unless other protections are in place. * In some cases a second visit to the website after the initial scan is required before exploit code is delivered. Unlike [Exploit Public-Facing Application](https://attack.mitre.org/techniques/T1190), the focus of this technique is to exploit software on a client endpoint upon visiting a website. This will commonly give an adversary access to systems on the internal network instead of external systems that may be in a DMZ. Adversaries may also use compromised websites to deliver a user to a malicious application designed to [Steal Application Access Token](https://attack.mitre.org/techniques/T1528)s, like OAuth tokens, to gain access to protected applications and information. These malicious applications have been delivered through popups on legitimate websites.(Citation: Volexity OceanLotus Nov 2017)

Name

T1505

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

T1213

ID

T1213

Description

Adversaries may leverage information repositories to mine valuable information. Information repositories are tools that allow for storage of information, typically to facilitate collaboration or information sharing between users, and can store a wide variety of data that may aid adversaries in further objectives, or direct access to the target information. Adversaries may also abuse external sharing features to share sensitive documents with recipients outside of the organization. The following is a brief list of example information that may hold potential value to an adversary and may also be found on an information repository: * Policies, procedures, and standards * Physical / logical network diagrams * System architecture diagrams * Technical system documentation * Testing / development credentials * Work / project schedules * Source code snippets * Links to network shares and other internal resources Information stored in a repository may vary based on the specific instance or environment. Specific common information repositories include web-based platforms such as [Sharepoint](https://attack.mitre.org/techniques/T1213/002) and [Confluence](https://attack.mitre.org/techniques/T1213/001), specific services such as Code Repositories, IaaS databases, enterprise databases, and other storage infrastructure such as SQL Server.

Name

T1486

ID

T1486

Description

Adversaries may encrypt data on target systems or on large numbers of systems in a network to interrupt availability to system and network resources. They can attempt to render stored data inaccessible by encrypting files or data on local and remote drives and withholding access to a decryption key. This may be done in order to extract monetary compensation from a victim in exchange for decryption or a decryption key (ransomware) or to render data permanently inaccessible in cases where the key is not saved or transmitted.(Citation: US-CERT Ransomware 2016)(Citation: FireEye WannaCry 2017)(Citation: US-CERT NotPetya 2017)(Citation: US-CERT SamSam 2018) In the case of ransomware, it is typical that common user files like Office documents, PDFs, images, videos, audio, text, and source code files will be encrypted (and often renamed and/or tagged with specific file markers). Adversaries may need to first employ other behaviors, such as [File and Directory Permissions Modification](https://attack.mitre.org/techniques/T1222) or [System Shutdown/Reboot](https://attack.mitre.org/techniques/T1529), in order to unlock and/or gain access to manipulate these files.(Citation: CarbonBlack Conti July 2020) In some cases, adversaries may encrypt critical system files, disk partitions, and the MBR.(Citation: US-CERT NotPetya 2017) To maximize impact on the target organization, malware designed for encrypting data may have worm-like features to propagate across a network by leveraging other attack techniques like [Valid Accounts](https://attack.mitre.org/techniques/T1078), [OS Credential Dumping](https://attack.mitre.org/techniques/T1003), and [SMB/Windows Admin Shares](https://attack.mitre.org/techniques/T1021/002).(Citation: FireEye WannaCry 2017)(Citation: US-CERT NotPetya 2017) Encryption malware may also leverage [Internal Defacement](https://attack.mitre.org/techniques/T1491/001), such as changing victim wallpapers, or otherwise intimidate victims by sending ransom notes or other messages to connected printers (known as "print bombing").(Citation: NHS Digital Egregor Nov 2020) In cloud environments, storage objects within compromised accounts may also be encrypted.(Citation: Rhino S3 Ransomware Part 1)

Name

T1098

ID

T1098

Description

Adversaries may manipulate accounts to maintain and/or elevate access to victim systems. Account manipulation may consist of any action that preserves or modifies adversary access to a compromised account, such as modifying credentials or permission groups.

(Citation: FireEye SMOKEDHAM June 2021) These actions could also include account activity designed to subvert security policies, such as performing iterative password updates to bypass password duration policies and preserve the life of compromised credentials. In order to create or manipulate accounts, the adversary must already have sufficient permissions on systems or the domain. However, account manipulation may also lead to privilege escalation where modifications grant access to additional roles, permissions, or higher-privileged [Valid Accounts](https://attack.mitre.org/techniques/T1078).

Name

T1528

ID

T1528

Description

Adversaries can steal application access tokens as a means of acquiring credentials to access remote systems and resources. Application access tokens are used to make authorized API requests on behalf of a user or service and are commonly used as a way to access resources in cloud and container-based applications and software-as-a-service (SaaS).(Citation: Auth0 - Why You Should Always Use Access Tokens to Secure APIs Sept 2019) Adversaries who steal account API tokens in cloud and containerized environments may be able to access data and perform actions with the permissions of these accounts, which can lead to privilege escalation and further compromise of the environment. For example, in Kubernetes environments, processes running inside a container may communicate with the Kubernetes API server using service account tokens. If a container is compromised, an adversary may be able to steal the container's token and thereby gain access to Kubernetes API commands.(Citation: Kubernetes Service Accounts) Similarly, instances within continuous-development / continuous-integration (CI/CD) pipelines will often use API tokens to authenticate to other services for testing and deployment. (Citation: Cider Security Top 10 CICD Security Risks) If these pipelines are compromised, adversaries may be able to steal these tokens and leverage their privileges. Token theft can also occur through social engineering, in which case user action may be required to grant access. OAuth is one commonly implemented framework that issues tokens to users for access to systems. An application desiring access to cloud-based services or protected APIs can gain entry using OAuth 2.0 through a variety of authorization protocols. An example commonly-used sequence is Microsoft's Authorization Code Grant flow.(Citation: Microsoft Identity Platform Protocols May 2019)(Citation: Microsoft - OAuth Code Authorization flow - June 2019) An OAuth access token enables a third-party application to

interact with resources containing user data in the ways requested by the application without obtaining user credentials. Adversaries can leverage OAuth authorization by constructing a malicious application designed to be granted access to resources with the target user's OAuth token.(Citation: Amnesty OAuth Phishing Attacks, August 2019)(Citation: Trend Micro Pawn Storm OAuth 2017) The adversary will need to complete registration of their application with the authorization server, for example Microsoft Identity Platform using Azure Portal, the Visual Studio IDE, the command-line interface, PowerShell, or REST API calls.(Citation: Microsoft - Azure AD App Registration - May 2019) Then, they can send a [Spearphishing Link](https://attack.mitre.org/techniques/T1566/002) to the target user to entice them to grant access to the application. Once the OAuth access token is granted, the application can gain potentially long-term access to features of the user account through [Application Access Token](https://attack.mitre.org/techniques/T1550/001). (Citation: Microsoft - Azure AD Identity Tokens - Aug 2019) Application access tokens may function within a limited lifetime, limiting how long an adversary can utilize the stolen token. However, in some cases, adversaries can also steal application refresh tokens(Citation: Auth0 Understanding Refresh Tokens), allowing them to obtain new access tokens without prompting the user.

Name

T1057

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. Administrator or otherwise elevated access may provide better process details. 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

T1083

ID

T1083

Description

Adversaries may enumerate files and directories or may search in specific locations of a host or network share for certain information within a file system. Adversaries may use the information from [File and Directory Discovery](https://attack.mitre.org/techniques/T1083) during automated discovery to shape follow-on behaviors, including whether or not the adversary fully infects the target and/or attempts specific actions. Many command shell utilities can be used to obtain this information. Examples include `dir`, `tree`, `ls`, `find`, and `locate`.(Citation: Windows Commands JPCERT) Custom tools may also be used to gather file and directory information and interact with the [Native API](https://attack.mitre.org/techniques/T1106). Adversaries may also leverage a [Network Device CLI](https://attack.mitre.org/techniques/T1059/008) on network devices to gather file and directory information (e.g. `dir`, `show flash`, and/or `nvram`).(Citation: US-CERT-TA18-106A) Some files and directories may require elevated or specific user permissions to access.

Name

T1059

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)

Name

T1027

ID

T1027

Description

Adversaries may attempt to make an executable or file difficult to discover or analyze by encrypting, encoding, or otherwise obfuscating its contents on the system or in transit. This is common behavior that can be used across different platforms and the network to evade defenses. Payloads may be compressed, archived, or encrypted in order to avoid detection. These payloads may be used during Initial Access or later to mitigate detection. Sometimes a user's action may be required to open and [Deobfuscate/Decode Files or Information](https://attack.mitre.org/techniques/T1140) for [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) Adversaries may also use compressed or

archived scripts, such as JavaScript. Portions of files can also be encoded to hide the plain-text strings that would otherwise help defenders with discovery. (Citation: Linux/Cdorked.A We Live Security Analysis) Payloads may also be split into separate, seemingly benign files that only reveal malicious functionality when reassembled. (Citation: Carbon Black Obfuscation Sept 2016) Adversaries may also abuse [Command Obfuscation](https://attack.mitre.org/techniques/T1027/010) to obscure commands executed from payloads or directly via [Command and Scripting Interpreter](https://attack.mitre.org/techniques/T1059). Environment variables, aliases, characters, and other platform/language specific semantics can be used to evade signature based detections and application control mechanisms. (Citation: FireEye Obfuscation June 2017) (Citation: FireEye Revoke-Obfuscation July 2017)(Citation: PaloAlto EncodedCommand March 2017)

Name

T1518

ID

T1518

Description

Adversaries may attempt to get a listing of software and software versions that are installed on a system or in a cloud environment. Adversaries may use the information from [Software Discovery](https://attack.mitre.org/techniques/T1518) during automated discovery to shape follow-on behaviors, including whether or not the adversary fully infects the target and/or attempts specific actions. Such software may be deployed widely across the environment for configuration management or security reasons, such as [Software Deployment Tools](https://attack.mitre.org/techniques/T1072), and may allow adversaries broad access to infect devices or move laterally. Adversaries may attempt to enumerate software for a variety of reasons, such as figuring out what security measures are present or if the compromised system has a version of software that is vulnerable to [Exploitation for Privilege Escalation](https://attack.mitre.org/techniques/T1068).

Name

T1092

ID

T1092

Description

Adversaries can perform command and control between compromised hosts on potentially disconnected networks using removable media to transfer commands from system to system.(Citation: ESET Sednit USBStealer 2014) Both systems would need to be compromised, with the likelihood that an Internet-connected system was compromised first and the second through lateral movement by [Replication Through Removable Media](https://attack.mitre.org/techniques/T1091). Commands and files would be relayed from the disconnected system to the Internet-connected system to which the adversary has direct access.

Name

T1190

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) or [Exploitation for Client Execution](https://attack.mitre.org/techniques/T1203). 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

T1562

ID

T1562

Description

Adversaries may maliciously modify components of a victim environment in order to hinder or disable defensive mechanisms. This not only involves impairing preventative defenses, such as firewalls and anti-virus, but also detection capabilities that defenders can use to audit activity and identify malicious behavior. This may also span both native defenses as well as supplemental capabilities installed by users and administrators. Adversaries may also impair routine operations that contribute to defensive hygiene, such as blocking users from logging out of a computer or stopping it from being shut down. These restrictions can further enable malicious operations as well as the continued propagation of incidents.(Citation: Emotet shutdown) Adversaries could also target event aggregation and analysis mechanisms, or otherwise disrupt these procedures by altering other system components.

Name

T1053

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

T1543

ID

T1543

Description

Adversaries may create or modify system-level processes to repeatedly execute malicious payloads as part of persistence. When operating systems boot up, they can start processes that perform background system functions. On Windows and Linux, these system processes are referred to as services.(Citation: TechNet Services) On macOS, launchd processes known as [Launch Daemon](https://attack.mitre.org/techniques/T1543/004) and [Launch Agent](https://attack.mitre.org/techniques/T1543/001) are run to finish system initialization and load user specific parameters.(Citation: AppleDocs Launch Agent Daemons) Adversaries may install new services, daemons, or agents that can be configured to execute at startup or a repeatable interval in order to establish persistence. Similarly, adversaries may modify existing services, daemons, or agents to achieve the same effect. Services, daemons, or agents may be created with administrator privileges but executed under root/SYSTEM privileges. Adversaries may leverage this functionality to create or modify system processes in order to escalate privileges.(Citation: OSX Malware Detection)

Url

Value
https://dns.startservicefounds.com/service/f.php
https://cache.cloudswiftcdn.com
https://api.startservicefounds.com

IPv4-Addr

Value
45.150.67.235
31.43.191.220
94.102.51.144

External References

- https://wpscan.com/blog/surge-of-javascript-malware-in-sites-with-vulnerable-versions-of-litespeed-cache-plugin/
- https://otx.alienvault.com/pulse/663ce6fb65626ccc9e6e833f