Apr 06 2024

NETMANAGEIT

Intelligence Report

Phishing Deception - Suspended Domains Reveal Malicious Payload for Latin American Region

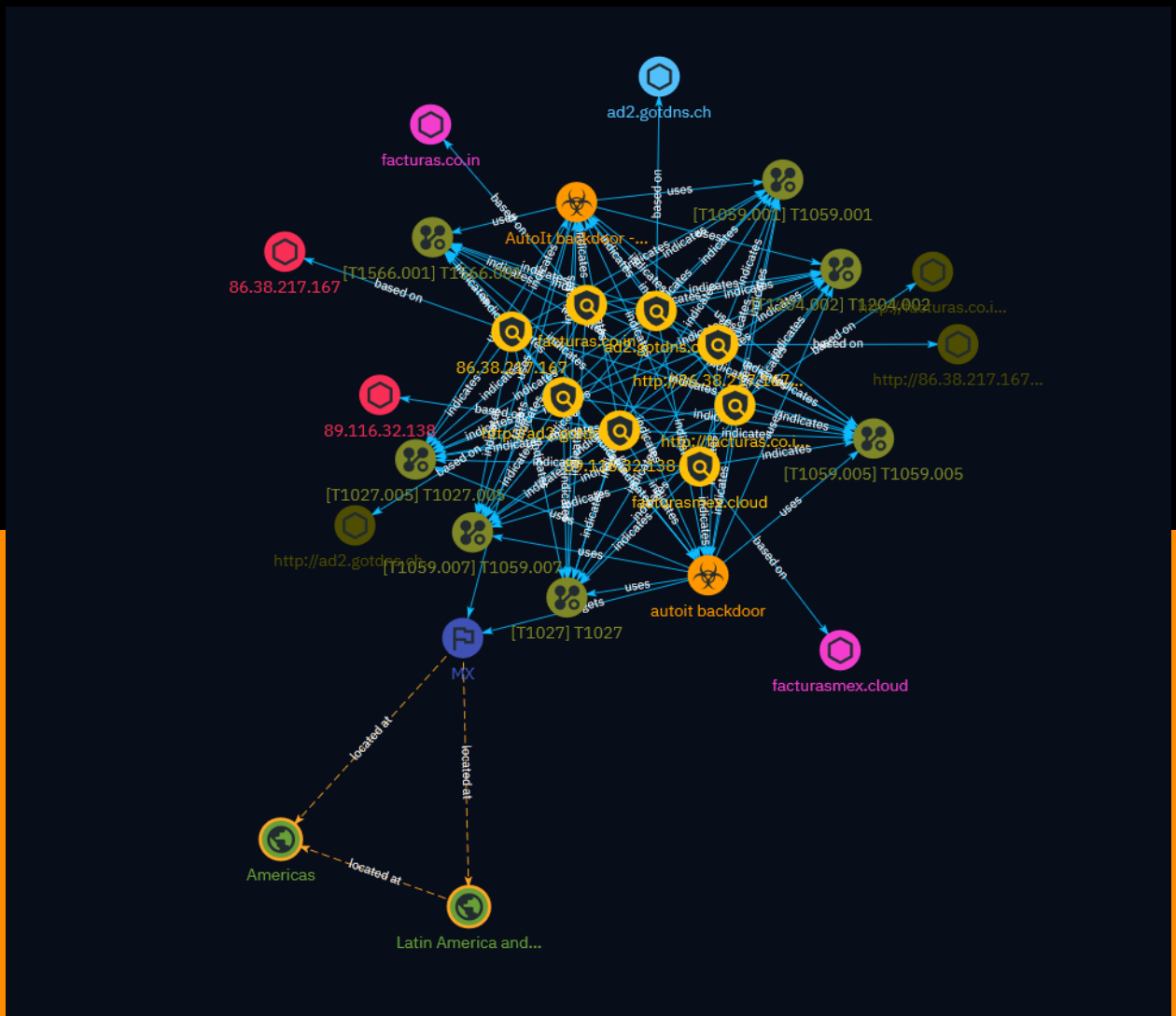

Table of contents

Overview

Entities

Observables

External References

Overview

Description

This analysis examines a phishing campaign targeting the Latin American region, where the malicious email contains a ZIP file attachment with an HTML file that leads to a malicious file download disguised as an invoice. The campaign employs techniques like using newly created domains, geolocation-based redirection, and obfuscated code to evade detection and deliver malware payloads, exhibiting similarities to previous 'Horabot' campaigns.

Confidence

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

100 / 100

Content

N/A

Indicator

Name

ad2.gotdns.ch

Pattern Type

stix

Pattern

[hostname:value = 'ad2.gotdns.ch']

Name

facturasmex.cloud

Description

- **Unsafe:** True - **Server:** - **Domain Rank:** 0 - **DNS Valid:** False - **Parking:** False - **Spamming:** False - **Malware:** True - **Phishing:** True - **Suspicious:** True - **Adult:** False - **Category:** Web Tracker - **Domain Age:** {'human': '1 month ago', 'timestamp': 1709063156, 'iso': '2024-02-27T14:45:56-05:00'} - **IPQS: Domain:** facturasmex.cloud - **IPQS: IP Address:** N/A

Pattern Type

stix

Pattern

[domain-name:value = 'facturasmex.cloud']

Name

facturas.co.in

Description

- **Unsafe:** True - **Server:** cloudflare - **Domain Rank:** 0 - **DNS Valid:** True - **Parking:** False - **Spamming:** False - **Malware:** True - **Phishing:** True - **Suspicious:** True - **Adult:** False - **Category:** Web Tracker - **Domain Age:** {'human': '2 months ago', 'timestamp': 1706528054, 'iso': '2024-01-29T06:34:14-05:00'} - **IPQS: Domain:** facturas.co.in - **IPQS: IP Address:** 104.21.5.78

Pattern Type

stix

Pattern

[domain-name:value = 'facturas.co.in']

Name

http://facturas.co.in/index.php?va

Description

- **Unsafe:** True - **Server:** cloudflare - **Domain Rank:** 0 - **DNS Valid:** True - **Parking:** False - **Spamming:** False - **Malware:** True - **Phishing:** True - **Suspicious:** True - **Adult:** False - **Category:** Web Tracker - **Domain Age:** {'human': '2 months ago', 'timestamp': 1706528054, 'iso': '2024-01-29T06:34:14-05:00'} - **IPQS: Domain:** facturas.co.in - **IPQS: IP Address:** 172.67.133.47

Pattern Type

stix

Pattern

[url:value = 'http://facturas.co.in/index.php?va']

Name

http://ad2.gotdns.ch/22/22

Description

- **Unsafe:** False - **Server:** - **Domain Rank:** 39858 - **DNS Valid:** True - **Parking:** False - **Spamming:** True - **Malware:** False - **Phishing:** False - **Suspicious:** True - **Adult:** False - **Category:** Dynamic dns - **Domain Age:** {'human': '3 years ago', 'timestamp': 1616279682, 'iso': '2021-03-20T18:34:42-04:00'} - **IPQS: Domain:** ad2.gotdns.ch - **IPQS: IP Address:** 158.247.7.206

Pattern Type

stix

Pattern

[url:value = 'http://ad2.gotdns.ch/22/22']

Name

http://86.38.217.167/ps/index.php

Description

- **Unsafe:** False - **Server:** Apache/2.4.52 (Ubu - **Domain Rank:** 0 - **DNS Valid:** False - **Parking:** False - **Spamming:** False - **Malware:** False - **Phishing:** False -

Suspicious: True - **Adult:** False - **Category:** N/A - **Domain Age:** {'human': 'N/A', 'timestamp': None, 'iso': None} - **IPQS: Domain:** 86.38.217.167 - **IPQS: IP Address:** N/A

Pattern Type

stix

Pattern

[url:value = 'http://86.38.217.167/ps/index.php']

Name

89.116.32.138

Description

ISP: Hostinger International Limited **OS:** - ---------- Services: **22:** ``` SSH-2.0-OpenSSH_8.9p1 Ubuntu-3ubuntu0.6 Key type: ecdsa-sha2-nistp256 Key: AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBJAOUI/ggP9JAe0bc9Fd8mAiMjdeBKvpO9rXMFNsDO5WCdLQW3WhXSRqUAeH8caDheaoaj4f7Mvg1Dv6JwDjI2k= Fingerprint: 3c:89:c5:a3:cc:03:ff:f2:55:af:80:55:bc:2a:4c:4d 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 ``` ---------- **80:** ``` HTTP/1.1 200 OK Date: Mon, 18 Mar 2024 20:08:56 GMT Server: Apache/2.4.52 (Ubuntu) Last-Modified: Wed, 11 Oct 2023 14:03:10 GMT ETag: "da6-60771488a8380" Accept-Ranges: bytes Content-Length: 3494 Vary: Accept-Encoding Content-Type: text/html ``` ---------- **443:** ``` HTTP/1.1 200 OK Date: Wed, 20 Mar 2024 10:32:10 GMT Server: Apache/2.4.52 (Ubuntu) Strict-Transport-Security: max-age=31536000 Content-Security-Policy: upgrade-insecure-requests Last-Modified: Wed, 11 Oct 2023 14:03:10 GMT ETag: "da6-60771488a8380" Accept-Ranges: bytes Content-Length:

3494 Vary: Accept-Encoding Content-Type: text/html ``` HEARTBLEED: 2024/03/20 10:32:31 89.116.32.138:443 - SAFE ----------

Pattern Type

stix

Pattern

[ipv4-addr:value = '89.116.32.138']

Name

86.38.217.167

Description

ISP: Hostinger International Limited **OS:** - ---------- Services: **22:** ``` SSH-2.0-OpenSSH_8.9p1 Ubuntu-3ubuntu0.6 Key type: ecdsa-sha2-nistp256 Key: AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBC1f5jovA66yM9sC3mscmQXK clz0F6JMe7f+SxuGm3d+NniL/Gw3a2HZYpH/IGVP6Tq+AyyCWnM7xvj3/n0k3qY= Fingerprint: 02:97:d9:45:61:7b:4c:49:dc:ed:65:3e:fe:c5:6c:cd 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 ``` ---------- **80:** ``` HTTP/1.1 200 OK Date: Thu, 21 Mar 2024 12:24:32 GMT Server: Apache/2.4.52 (Ubuntu) Last-Modified: Wed, 11 Oct 2023 14:03:10 GMT ETag: "da6-60771488a8380" Accept-Ranges: bytes Content-Length: 3494 Vary: Accept-Encoding Content-Type: text/html ``` ---------- **443:** ``` HTTP/1.1 200 OK Date: Tue, 12 Mar 2024 01:53:05 GMT Server: Apache/2.4.52 (Ubuntu) Strict-Transport-Security: max-age=31536000 Content-Security-Policy: upgrade-insecure-requests Last-Modified: Wed, 11 Oct 2023 14:03:10 GMT ETag: "da6-60771488a8380" Accept-Ranges: bytes Content-Length:

3494 Vary: Accept-Encoding Content-Type: text/html ``` HEARTBLEED: 2024/03/12 01:53:19 86.38.217.167:443 - SAFE ----------

Pattern Type

stix

Pattern

[ipv4-addr:value = '86.38.217.167']

Malware

Name
AutoIt backdoor - S0129
Name
autoit backdoor
Description
[AutoIt backdoor](https://attack.mitre.org/software/S0129) is malware that has been used by the actors responsible for the MONSOON campaign. The actors frequently used it in weaponized .pps files exploiting CVE-2014-6352. (Citation: Forcepoint Monsoon) This malware makes use of the legitimate scripting language for Windows GUI automation with the same name.

Attack-Pattern

Name
T1059.005
ID
T1059.005
Description
Adversaries may abuse Visual Basic (VB) for execution. VB is a programming language created by Microsoft with interoperability with many Windows technologies such as [Component Object Model](https://attack.mitre.org/techniques/T1559/001) and the [Native API](https://attack.mitre.org/techniques/T1106) through the Windows API. Although tagged as legacy with no planned future evolutions, VB is integrated and supported in the .NET Framework and cross-platform .NET Core.(Citation: VB .NET Mar 2020)(Citation: VB Microsoft) Derivative languages based on VB have also been created, such as Visual Basic for Applications (VBA) and VBScript. VBA is an event-driven programming language built into Microsoft Office, as well as several third-party applications.(Citation: Microsoft VBA) (Citation: Wikipedia VBA) VBA enables documents to contain macros used to automate the execution of tasks and other functionality on the host. VBScript is a default scripting language on Windows hosts and can also be used in place of [JavaScript](https://attack.mitre.org/techniques/T1059/007) on HTML Application (HTA) webpages served to Internet Explorer (though most modern browsers do not come with VBScript support). (Citation: Microsoft VBScript) Adversaries may use VB payloads to execute malicious commands. Common malicious usage includes automating execution of behaviors with VBScript or embedding VBA content into [Spearphishing Attachment](https://attack.mitre.org/techniques/T1566/001) payloads (which may also involve [Mark-of-the-Web Bypass](https://attack.mitre.org/techniques/T1553/005) to enable execution).(Citation: Default VBS macros Blocking)

Name

T1027.005

ID

T1027.005

Description

Adversaries may remove indicators from tools if they believe their malicious tool was detected, quarantined, or otherwise curtailed. They can modify the tool by removing the indicator and using the updated version that is no longer detected by the target's defensive systems or subsequent targets that may use similar systems. A good example of this is when malware is detected with a file signature and quarantined by anti-virus software. An adversary who can determine that the malware was quarantined because of its file signature may modify the file to explicitly avoid that signature, and then re-use the malware.

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

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

T1566.001

ID

T1566.001

Description

Adversaries may send spearphishing emails with a malicious attachment in an attempt to gain access to victim systems. Spearphishing attachment is a specific variant of spearphishing. Spearphishing attachment is different from other forms of spearphishing in that it employs the use of malware attached to an email. All forms of spearphishing are electronically delivered social engineering targeted at a specific individual, company, or industry. In this scenario, adversaries attach a file to the spearphishing email and usually rely upon [User Execution](https://attack.mitre.org/techniques/T1204) to gain execution. Spearphishing may also involve social engineering techniques, such as posing as a trusted source. There are many options for the attachment such as Microsoft Office documents, executables, PDFs, or archived files. Upon opening the attachment (and potentially clicking past protections), the adversary's payload exploits a vulnerability or directly executes on the user's system. The text of the spearphishing email usually tries to give a plausible reason why the file should be opened, and may explain how to bypass system protections in order to do so. The email may also contain instructions on how to decrypt an attachment, such as a zip file password, in order to evade email boundary defenses. Adversaries frequently manipulate file extensions and icons in order to make attached executables appear to be document files, or files exploiting one application appear to be a file for a different one.

Name

T1204.002

ID

T1204.002

Description

An adversary may rely upon a user opening a malicious file in order to gain execution. Users may be subjected to social engineering to get them to open a file that will lead to code execution. This user action will typically be observed as follow-on behavior from [Spearphishing Attachment](https://attack.mitre.org/techniques/T1566/001). Adversaries may use several types of files that require a user to execute them, including .doc, .pdf, .xls, .rtf, .scr, .exe, .lnk, .pif, and .cpl. Adversaries may employ various forms of [Masquerading](https://attack.mitre.org/techniques/T1036) and [Obfuscated Files or Information](https://attack.mitre.org/techniques/T1027) to increase the likelihood that a user will open and successfully execute a malicious file. These methods may include using a familiar naming convention and/or password protecting the file and supplying instructions to a user on how to open it.(Citation: Password Protected Word Docs) While [Malicious File](https://attack.mitre.org/techniques/T1204/002) frequently occurs shortly after Initial Access it may occur at other phases of an intrusion, such as when an adversary places a file in a shared directory or on a user's desktop hoping that a user will click on it. This activity may also be seen shortly after [Internal Spearphishing](https://attack.mitre.org/techniques/T1534).

Name

T1059.007

ID

T1059.007

Description

Adversaries may abuse various implementations of JavaScript for execution. JavaScript (JS) is a platform-independent scripting language (compiled just-in-time at runtime) commonly associated with scripts in webpages, though JS can be executed in runtime environments outside the browser.(Citation: NodeJS) JScript is the Microsoft implementation of the same scripting standard. JScript is interpreted via the Windows Script engine and thus integrated with many components of Windows such as the [Component Object Model](https://attack.mitre.org/techniques/T1559/001) and Internet Explorer HTML Application (HTA) pages.(Citation: JScrip May 2018)(Citation: Microsoft JScript 2007)(Citation: Microsoft Windows Scripts) JavaScript for Automation (JXA) is a macOS scripting language based on JavaScript, included as part of Apple’s Open Scripting Architecture (OSA), that was introduced in OSX 10.10. Apple’s OSA provides scripting capabilities to control applications, interface with the operating system, and bridge access into the rest of Apple’s internal APIs. As of OSX 10.10, OSA only supports two languages, JXA

and [AppleScript](https://attack.mitre.org/techniques/T1059/002). Scripts can be executed via the command line utility `osascript`, they can be compiled into applications or script files via `osacompile`, and they can be compiled and executed in memory of other programs by leveraging the OSAKit Framework.(Citation: Apple About Mac Scripting 2016) (Citation: SpecterOps JXA 2020)(Citation: SentinelOne macOS Red Team)(Citation: Red Canary Silver Sparrow Feb2021)(Citation: MDSec macOS JXA and VSCode) Adversaries may abuse various implementations of JavaScript to execute various behaviors. Common uses include hosting malicious scripts on websites as part of a [Drive-by Compromise](https://attack.mitre.org/techniques/T1189) or downloading and executing these script files as secondary payloads. Since these payloads are text-based, it is also very common for adversaries to obfuscate their content as part of [Obfuscated Files or Information](https://attack.mitre.org/techniques/T1027).

Country

Name
MX

Region

Name
Latin America and the Caribbean

Name
Americas

Hostname

Value
ad2.gotdns.ch

Domain-Name

Value
facturasmex.cloud
facturas.co.in

Url

Value
http://facturas.co.in/index.php?va
http://ad2.gotdns.ch/22/22
http://86.38.217.167/ps/index.php

IPv4-Addr

Value
89.116.32.138
86.38.217.167

External References

- https://www.trustwave.com/en-us/resources/blogs/spiderlabs-blog/phishing-deception-suspended-domains-reveal-malicious-payload-for-latin-american-region/
- https://otx.alienvault.com/pulse/66103234fe8bf33acad72700