Resecurity | The New Version of JsOutProx is Attacking Financial Institutions in APAC and MENA via GitLab Abuse

Published: 2024-04-03 · Archived: 2026-04-05 17:28:05 UTC

Intro

Resecurity has detected a new version of **JSOutProx**, targeting financial services and organizations in the APAC and MENA regions. JSOutProx is a sophisticated attack framework utilizing both JavaScript and .NET. It employs the .NET (de)serialization feature to interact with a core JavaScript module running on the victim's machine. Once executed, the malware enables the framework to load various plugins, which conduct additional malicious activities on the target. This malware was first identified in 2019 and was initially attributed to **SOLAR SPIDER**'s phishing campaigns, which delivered the JSOutProx RAT to financial institutions across Africa, the Middle East, South Asia, and Southeast Asia.

Notable Pattern - From GitHub to GitHab Abuse

The spike in this activity was identified around February 8, 2024, when a major system integrator based in the Kingdom of Saudi Arabia reported an incident targeting customers of one of their major banks regional banks. Resecurity assisted multiple victims in acquiring relevant malicious code artifacts due to Digital Forensics & Incident Response (DFIR) engagement and helped recover the payload. Multiple banking customers were targeted via an impersonation attack using the "**mike.will@my[.]com**" email account. The actors employed a fake SWIFT payment notification (for enterprise customers) and a Moneygram template (for private customers), using misleading notifications to confuse victims and execute malicious code.

Transaction_Ref_jpg.zip
d22f76e60a786f0c92fa20af1a1619b2

Transaction_Ref_jpg.js
89a088cd92b7ed59fd3bcc7786075130

MoneyGram_Global_Compliance_pdf.zip
9c9df8fbcef8acd1a5265be5fd8fdce9

MoneyGram_Global_Compliance_pdf.js
66514548cdffab50d1ea75772a08df3d

Swift_Copy_jpg.zip
81b9e7deb17e3371d417ad94776b2a26

Swift_Copy_jpg.js / TRXN-00000087312_pdf.js
bea8cf1f983120b68204f2fa9448526e

MoneyGram_AML_Compliance_review.pdf.zip
72461c94bd27e5b001265bbccc931534

MoneyGram_AML_Compliance_review.pdf.js
1bd7ce64f1a7cf7dc94b912ceb9533d0

Transaction_details_jpg.zip
f1858438a353d38e3e19109bf0a5e1be

Transaction_details_jpg.js
6764dbc4df70e559b2a59e913d940d4b

Transaction_Ref_01302024_jpg.zip
3a2104953478d1e60927aa6def17e8e7

Transaction_Ref_01302024_jpg.js
3d46a462f262818cada6899634354138

Most of the identified payloads were hosted on **GitHub** repositories. Notably, independent cybersecurity researchers first reported some of these payloads around **November 14, 2023**. Solar Spider is employing the classic Masquerading technique (T1036), disguising its code as a PDF file rather than JS code.

hxxps://github[.]com/agbusi/ikeketeorie/blob/main/Transaction_Ref_jpg.zip ->
hxxps://raw.githubusercontent[.]com/agbusi/ikeketeorie/main/Transaction_Ref_jpg.zip

hxxps://github[.]com/agbusi/compliance/blob/main/MoneyGram_Global_Compliance_pdf.zip ->
hxxps://raw.githubusercontent[.]com/agbusi/compliance/main/MoneyGram_Global_Compliance_pdf.zip

hxxps://github[.]com/agbusi/Singapore/blob/main/Swift_Copy_jpg.zip ->
hxxps://raw.githubusercontent[.]com/agbusi/Singapore/main/Swift_Copy_jpg.zip

hxxps://github[.]com/vectorvector11/transaction/blob/main/MoneyGram_AML_Compliance_review.pdf.zip ->
hxxps://raw.githubusercontent[.]com/vectorvector11/transaction/main/MoneyGram_AML_Compliance_review.pdf.zip

hxxps://github[.]com/Conel10/deal/raw/main/Transaction_details_jpg.zip ->
hxxps://raw.githubusercontent[.]com/Conel10/deal/main/Transaction_details_jpg.zip

hxxps://github[.]com/winners101/admin/raw/main/Transaction_Ref_01302024_jpg.zip ->
hxxps://raw.githubusercontent[.]com/winners101/admin/main/Transaction_Ref_01302024_jpg.zip

In the result of the multi-stage infection chain, the actors drop multiple JS-based obfuscated payloads to collect sensitive information and plant a proxy server to connect remotely to the victim.

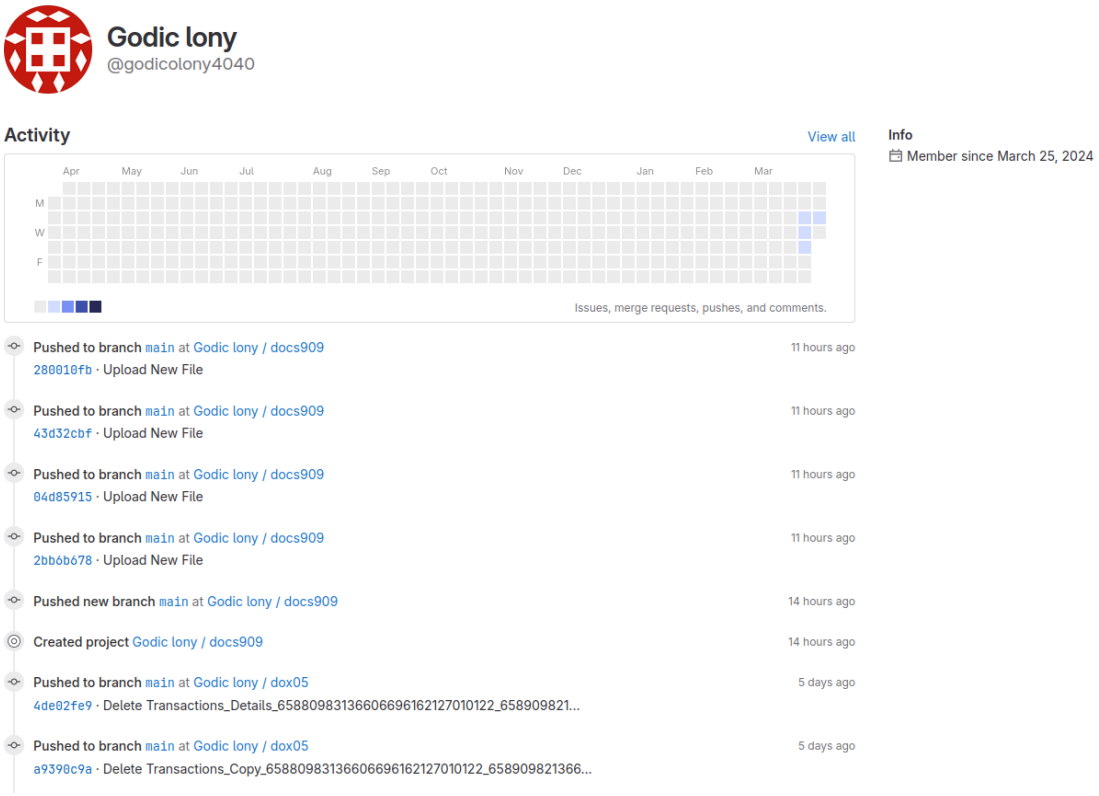
March 27, 2024 - Resecurity became aware of a new malware sample attributed to the same group. The notable difference was in the act of using **GitLab** (instead of GitHub) in a multi-stage infection chain:

hxxps://gitlab[.]com/godicolony4040/dox05/-
/raw/main/Transactions_Copy_65880983136606696162127010122_65890982136606

696162127010102.zip

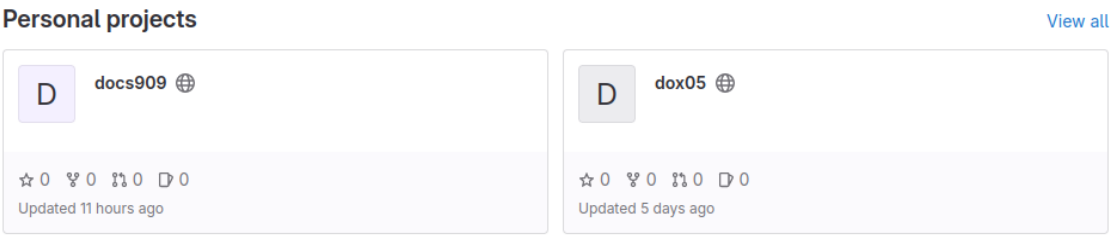
hxxps://gitlab[.]com/godicolony4040/dox05/-/raw/b540e3682457f2499b687fa0cd213b03ba77290c/Transactions_Copy_65880983136606696162127010122_658909821366066696162127010102.zip

The actor registered multiple accounts on GitLab around **March 25, 2024**, and used them to deploy repositories containing malicious payloads.



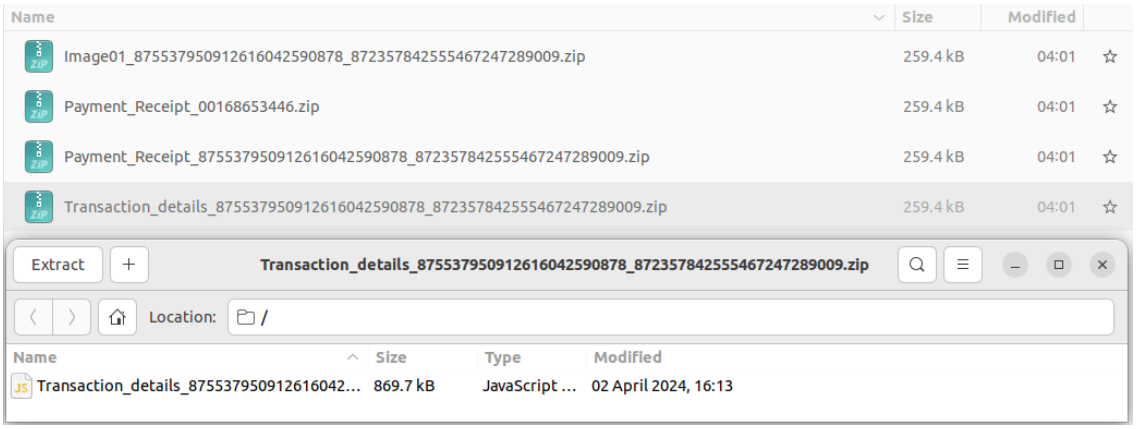
The identified repositories controlled by the actor were:

- docs909 (created April 2, 2024)
- dox05 (created March 26, 2024)



Once the malicious code has been successfully delivered, the actor removes the repository and creates a new one. This tactic is likely related to the actor uses to manage multiple malicious payloads and differentiate targets.

Resecurity acquired the most recent malware payloads uploaded by the actor on **April 2, 2024**:



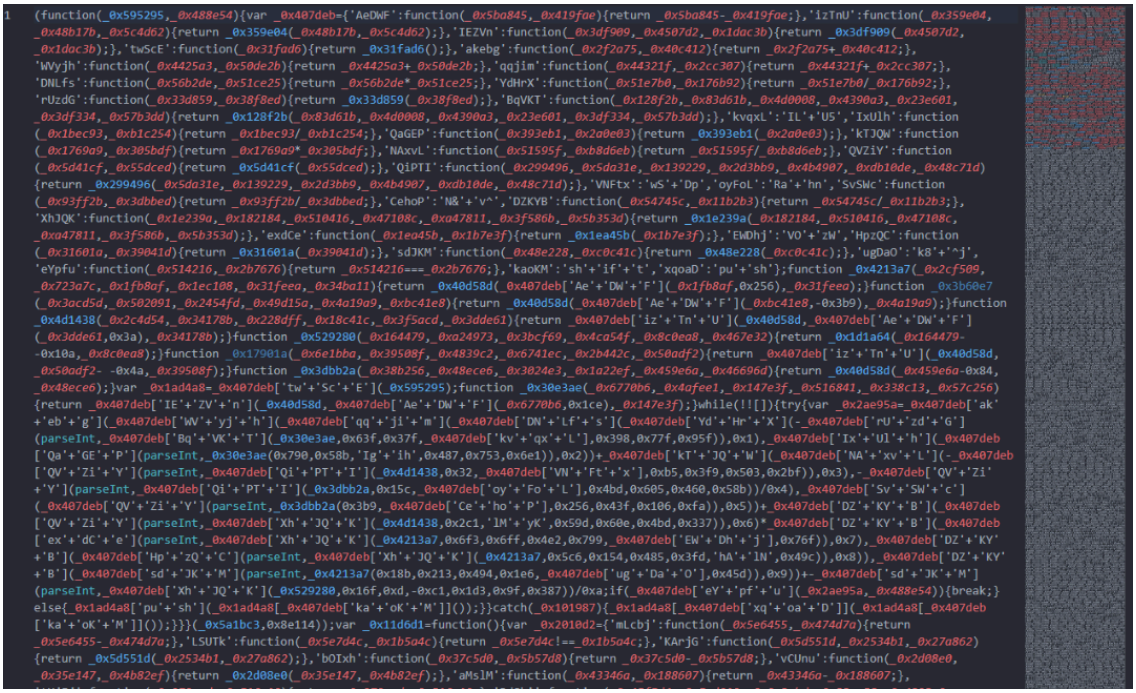
Reverse Engineering

The analysis of this malware, as provided below, can be used to detect, mitigate, and prevent threat campaigns utilizing JSOutProx.

The JSOutProx RAT malware features complex obfuscation within its JavaScript backdoor structure. It boasts a modular plugin architecture, which enables it to execute shell commands, handle file uploads and downloads, run files, modify the file system, ensure persistence, capture screenshots, and control keyboard and mouse actions. A distinctive aspect of this malware is its utilization of the Cookie header field in its command and control (C2) communications.

We downloaded the implants and unzipped them from their archives. They were then obfuscated using obfuscator.io. After deobfuscating them, we obtained the decoded JavaScript code.

Obfuscated implant:



Deobfuscated implant:

```
(function (_0x595295, _0x488e54) {
  var _0x1ad4a8 = _0x595295();
  while (true) {
    try {
      var _0x2ae95a = -parseInt(_0x40d58d(1137, "ILU5")) / 0x1 * (parseInt(_0x40d58d(1474, "Igih")) / 0x2) + -parseInt(_0x40d58d(645, "wSDp")) /
      if (_0x2ae95a === _0x488e54) {
        break;
      } else {
        _0x1ad4a8.push(_0x1ad4a8.shift());
      }
    } catch (_0x101987) {
      _0x1ad4a8.push(_0x1ad4a8.shift());
    }
  }
})(_0x5a1bc3, 0x8e114);
var _0x11d6d1 = function () {
  if (typeof WScript !== "undefined") {
    return WScript;
  }
  return undefined;
};
var _0x24abd2 = function () {
  return undefined;
};
var _0xc51b03 = function () {
  return undefined;
};
var _0x5f3726 = function () {
  return false;
};
function _0x4ab51f(_0x2c8123, _0x12aa84, _0x5365b5, _0x269ea6, _0x4e4310, _0x53cf49) {
  return _0x1d1a64(_0x4e4310 + 0x76, _0x53cf49);
}
var _0x512412 = function () {
  return typeof _0x11d6d1() !== "undefined";
};
```

The 1st stage implant supports the following commands:

- pat – update implant
- uss.s – set proxy and update sleep time
- uss.g – set proxy and set sleep time to C2
- upd – update and restart implant
- l32 – start x86 process
- l64 – start x64 process
- dcn – exit
- ejs – evaluate javascript code
- int.g – send sleep time to C2
- int.s – update sleep time

```
switch (_0x47ac9f[0x0]) {
  case "pat":
    _0x288844(_0xfe749a(), _0x3eb80a(_0x47ac9f));
    break;
  case "uss.s":
    _0x30b026 = _0x47ac9f[0x1] === '1';
    _0x1591f8 = parseInt(_0x47ac9f[0x2]);
    _0x1dae6c = _0x47ac9f[0x3] === '1';
    break;
  case "uss.g":
    _0xebfa44(_0x47ac9f[0x0] + "_|_" + (_0x30b026 ? '1' : '0') + "_=_" + _0x1591f8.toString() + "_=
    break;
  case "upd":
    if (_0x288844(_0xfe749a(), _0x3eb80a(_0x47ac9f))) {
      _0x287d37();
    }
    break;
  case "l32":
    _0x432f3c(_0xfe749a(), true);
    break;
  case "l64":
    _0x432f3c(_0xfe749a(), false);
    break;
  case "dcn":
    _0x452962();
    break;
  case "ejs":
    eval(_0x3eb80a(_0x47ac9f));
    break;
  case "int.g":
    _0xebfa44(_0x47ac9f[0x0], _0x1591f8.toString(), false);
    break;
  case "int.s":
    _0x1591f8 = parseInt(_0x47ac9f[0x1]);
    break;
  default:
    if (_0x47ac9f[0x0].startsWith('cn')) {
      _0x280bac(_0x47ac9f);
    }
```

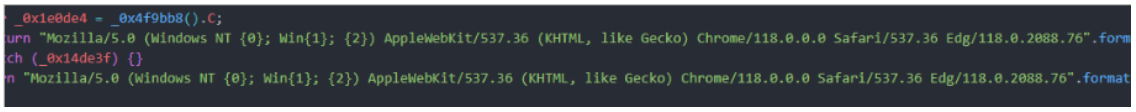
The script interacts with Windows Script Host (WSH) objects, such as ActiveXObject, to perform operations typical for automation or administration tasks, but for malicious purposes. For example, it uses WinHttp.WinHttpRequest.5.1 for HTTP requests, WScript.Shell for executing commands, and Scripting.FileSystemObject for file system access. Additionally, WMI is utilized to retrieve information about the system.

```
var _0x919780 = null;
try {
  _0x919780 = new ActiveXObject("WinHttp.WinHttpRequest.5.1");
  _0x919780.setTimeouts(0x0, 0x0, 0x0, 0x0);
} catch (_0x128aec) {}
return _0x919780;
```

Using WMI, the implant collects information about the victim's environment:

The implant uses the following static User Agent, which could potentially be used for malware tracking:

Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/118.0.0.0 Safari/537.36 Edg/118.0.2088.76



The implant communicates with command and control (C2C) servers deployed using Dynamic DNS, for example:

- http://mdytreudsgurifedei.ddns.net:9708/
- http://kiftpuseridsfryiri.ddns.net:8907/
- http://hudukpgdgfytpddswq.ddns.net:8843/
- http://ykderpgdgopopfuvgt.ddns.net:7891/

One unique feature of the malware is its use of the Cookie header field in its command and control (C2C) communication. During its initialization routine, the malware gathers various types of information. These information values are separated by the delimiter "_|_", concatenated, hex-encoded, and then set in the Cookie header field.

The 2nd-stage implant supports the following additional plugins:

ActivityPlugin	Enables the RAT to be in an Online or Offline state. When the state is online, it creates a adodb.stream object to save downloaded/collected data on disk.

CensorMiniPlugin	Enables/disables proxy settings on user machine by modifying registry key “Software\Microsoft\Windows\CurrentVersion\Internet Settings\ProxyEnable”
AdminConsolePlugin	
CensorPlugin	
ClipboardPlugin	It is used to copy the clipboard data and send it to C2. It can also modify clipboard data.
DnsPlugin	Used to set DNS path. Add or modify new path in C:\Windows\System32\drivers\etc\hosts.
LibraryPlugin	Sends list of dotnet versions installed on the machine to C2.
OutlookPlugin	It accesses the outlook account details and contacts list.
PriviledgePlugin	In this, the option “UAC” allows to write in registry location “SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System\” by setting value 0 for keys EnableLUA and ConsentPromptBehaviorAdmin. The option “elevateScript” executes the script using wscript.exe with the batch mode option. The option “elevateCommand” executes the command using Wsh with ‘runas’ flag. It also has options for using UAC bypass techniques like fodhelper.exe, Slui File Handler Hijacking, CompMgmtLauncher, EventViewer.exe etc.
PromptPlugin	
ProxyPlugin	Sets DNS path. Add or modify new path in C:\Windows\System32\drivers\etc\hosts.
ShortcutPlugin	Create a shortcut file for a given executable. Execute the shortcut file. Get the target of a shortcut file or dump the content of the file.
RecoveryPlugin	
TokensPlugin	Steal OTP received from SymantecVIP application.

Industrial Implant with Chinese Character

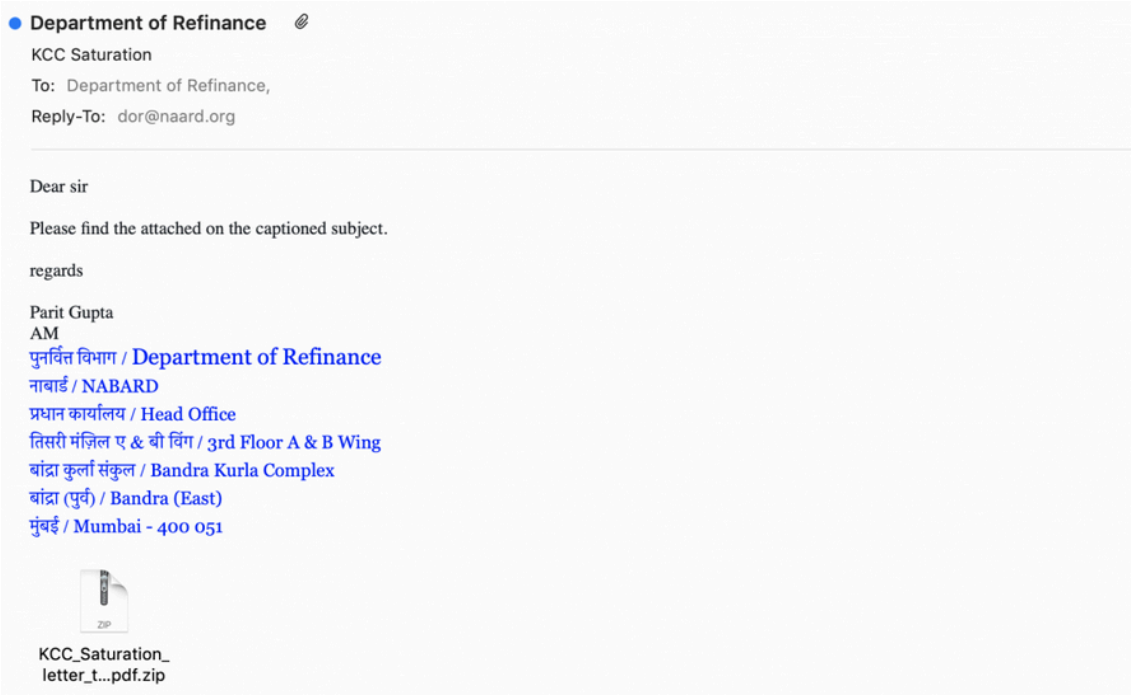
Before the newly identified campaign targeting multiple financial institutions in the APAC and MENA regions, JSOutProx was actively used in targeted attacks against Indian Cooperative Banks and Finance Companies.

In April 2020, ZScaler observed several targeted attacks on Indian government establishments and the banking sector. Organizations such as the Reserve Bank of India (RBI), IDBI Bank, and the Department of Refinance (DOR) within the National Bank for Agriculture and Rural Development (NABARD) in India received emails with archive file attachments. These attachments contained JavaScript and Java-based backdoors.

Further analysis of the JavaScript-based backdoor allowed us to correlate it with the JSOutProx RAT. This RAT was first used by a threat actor in December 2019, as mentioned by Yoroi. The Java-based RAT in this attack

provided functionalities similar to the JavaScript-based backdoor.

In one such campaign, the actors leveraging JSOutProx targeted government officials in NABARD (The National Bank for Agriculture and Rural Development, a national financial institution in India), using a malicious archive file attachment.



The actors used specific naming conventions for malicious files relevant to the government sector. Examples include:

- Nodal_Police_Stations_furnished_MHA_GOI_New_Delhi_xlsx.hta
- Slip_RTGS_IDBI_To_HDFC_pdf.hta
- 2685-Vishwambharlal_Kanahiyalal_Bhoot_Attachment_Order_pdf.hta
- NPCI_Compliance_Form_pdf.hta

Based on the analysis of the most recent campaign, the following victims have been identified::

- government organizations in India
- government organizations in Taiwan
- financial organizations in the Philippines
- financial organizations in Laos
- financial organizations in Singapore
- financial organizations in Malaysia
- financial organizations in India
- financial organizations in KSA

Considering the malware's significant sophistication, the profile of the targets, and the geography of past attacks, it can be suggested with a moderate level of confidence that JSOutProx may have been developed by actor(s) from China or those affiliated with it.

The malware was initially identified around 2019 and has been constantly improved, which may indicate an organized and continuous effort in its development.

Indicators of Compromise (IOCs)

The following indicators of compromise (IOCs) are associated with the recent JSOutProx malware campaigns, as described above, from November 14, 2023, March 27, 2024, and April 2, 2024:

- Transaction_Ref_jpg.zip
  d22f76e60a786f0c92fa20af1a1619b2
- Transaction_Ref_jpg.js
  89a088cd92b7ed59fd3bcc7786075130
- MoneyGram_Global_Compliance_pdf.zip
  9c9df8fbcef8acd1a5265be5fd8fdce9
- MoneyGram_Global_Compliance_pdf.js
  66514548cdffab50d1ea75772a08df3d
- Swift_Copy_jpg.zip
  81b9e7deb17e3371d417ad94776b2a26
- Swift_Copy_jpg.js / TRXN-00000087312_pdf.js
  bea8cf1f983120b68204f2fa9448526e
- MoneyGram_AML_Compliance_review.pdf.zip
  72461c94bd27e5b001265bbccc931534
- MoneyGram_AML_Compliance_review.pdf.js
  1bd7ce64f1a7cf7dc94b912ceb9533d0
- Transaction_details_jpg.zip
  f1858438a353d38e3e19109bf0a5e1be
- Transaction_details_jpg.js
  6764dbc4df70e559b2a59e913d940d4b
- Transaction_Ref_01302024_jpg.zip
  3a2104953478d1e60927aa6def17e8e7
- Transaction_Ref_01302024_jpg.js
  3d46a462f262818cada6899634354138
- Transactions_Copy_6588098313660669616212701012 2_6589098213660669616212701010 2.zip
  efad51e48d585b639d974fcf39f7ee07
- Transactions_Copy_658809831366066961621270101 22,658909821366066961621270101 02.js
  118b6673bd06c8eb082296a7b35f8fa5

C2C Communications

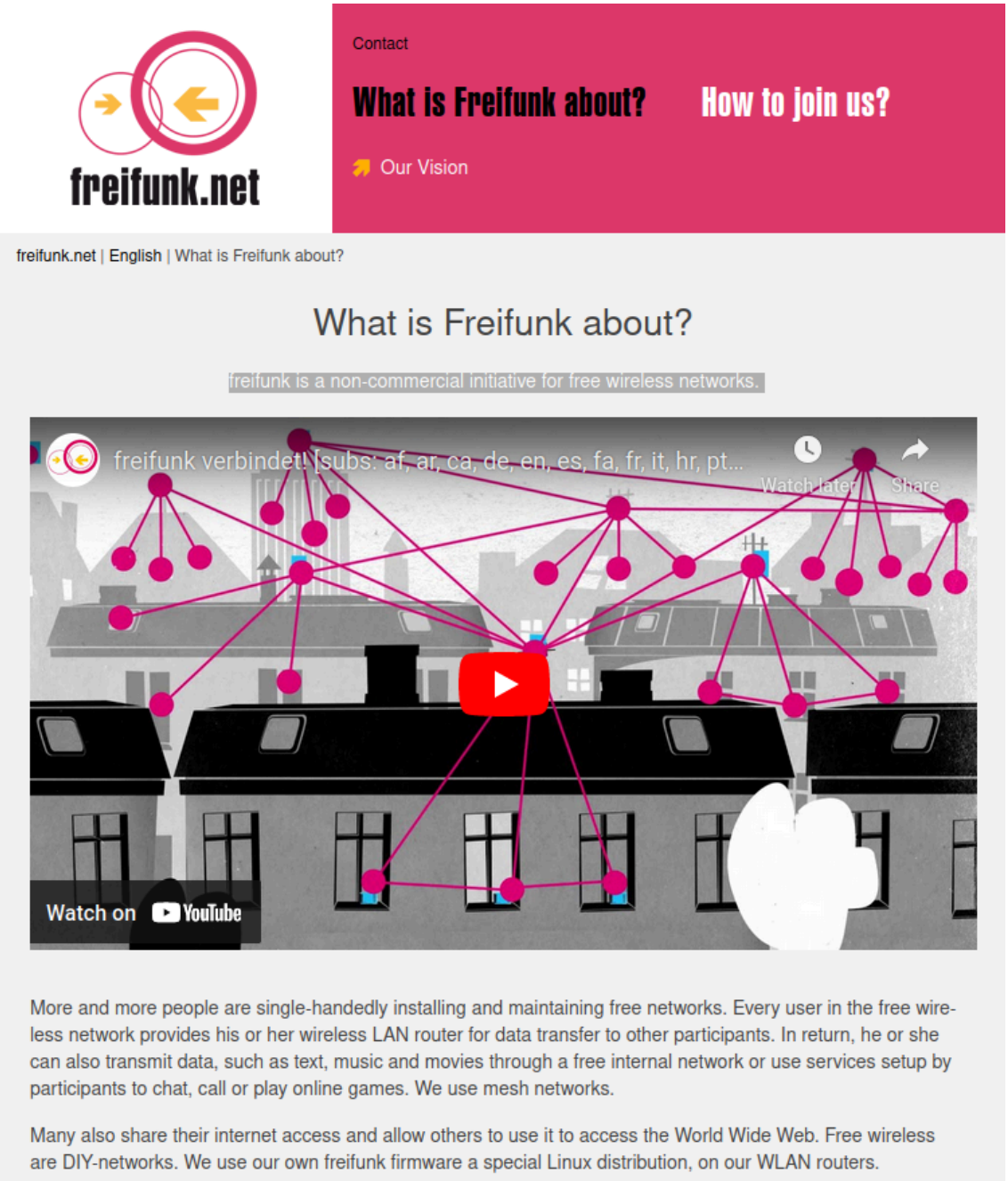
- suedxcapuertggando.ddns[.]net:8843/ (185.244.30[.]218)
- mdytreudsgurifedei.ddns[.]net:9708/ (offline)
- kiftpuseridsfryiri.ddns[.]net:8907/ (offline)
- hudukpgdgfytpddswq.ddns[.]net:8843/ (offline)
- ykderpgdgopopfuvgt.ddns[.]net:7891/ (offline)

- mdytreudsgurifedei.ddns[.]net (79.134.225[.]17)
- mdytreudsgurifedei.ddns[.]net (79.134.225[.]17)
- kiftpuseridsfryiri.ddns[.]net (79.134.225[.]17)
- eopgupgdpopopfuupi.ddns[.]net (103.212.81[.]155)
- ykderpgdgopopfuvgt.ddns[.]net (103.212.81[.]157)
- hudukpgdgfytpddswq.ddns[.]net (185.244.30[.]218)

Notably, some of the IP addresses identified in the most recent campaign from April 2, 2024, such as 185.244.30[.]218, were related to the Freemesh project.

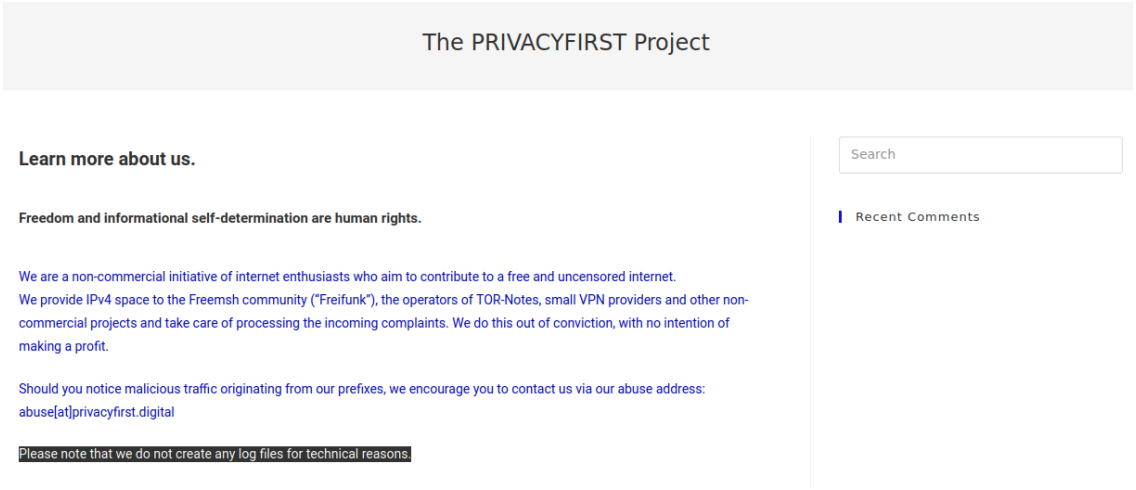
IP Address	185.244.30.218
Country	Poland [PL]
Region	Mazowieckie
City	Jozefow
Coordinates of City	52.137070, 21.235890 (52°8'13"N 21°14'9"E)
ISP	Freemesh - Sieci Niekomercyjne
Local Time	03 Apr, 2024 01:22 PM (UTC +01:00)
Domain	freemesh.net
Net Speed	(T1) Data Center/Transit

Freemesh redirects to a website dedicated to a non-commercial initiative for free wireless networks.

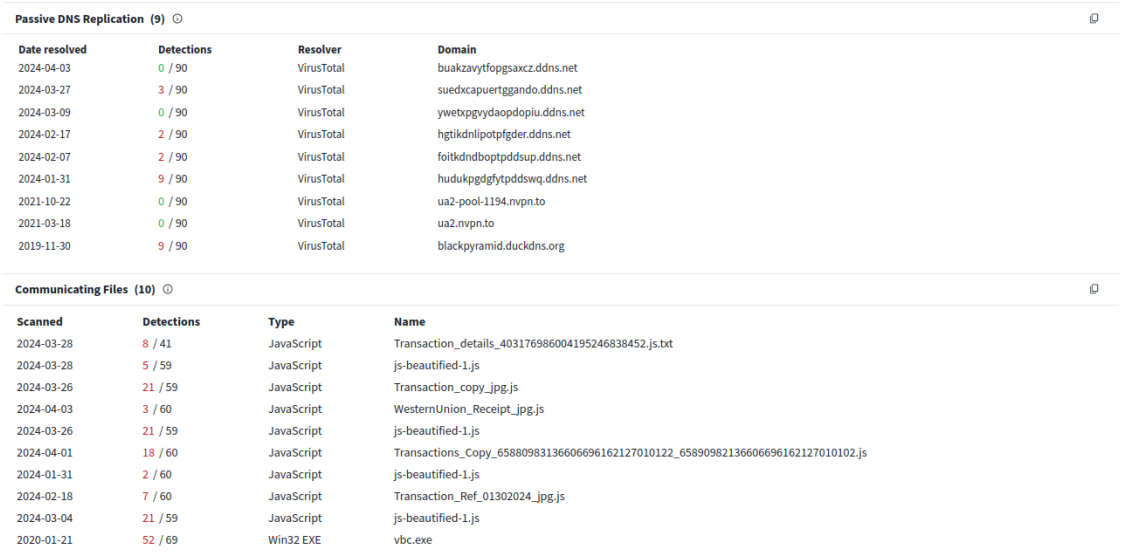


It is possible that the actor purposely deployed the command and control (C2C) host in such infrastructure to abuse it and conceal malicious network activity.

In fact, 185.244.30[.]218 is related to "**The Privacy First Project**", a non-profit which claims to provide "*IPv4 space to the Freemesh community ("Freifunk"), the operators of TOR nodes, small VPN providers and take care of processing the incoming complaints*". The project also states having no log files.



Based on Virustotal historical data, this host has an extensive malicious activity history and multiple subdomains tied to the JsOutProx infrastructure specifically.



Passive DNS Replication (9)

Date resolved	Detections	Resolver	Domain
2024-04-03	0 / 90	VirusTotal	buakzavytfopgsaxcz.ddns.net
2024-03-27	3 / 90	VirusTotal	suedxcapuertggando.ddns.net
2024-03-09	0 / 90	VirusTotal	ywetxpgvydaopdopiu.ddns.net
2024-02-17	2 / 90	VirusTotal	hgtikdnlipotpfgder.ddns.net
2024-02-07	2 / 90	VirusTotal	foitkdndboptpddsup.ddns.net
2024-01-31	9 / 90	VirusTotal	hudukpgdgfytpddswq.ddns.net
2021-10-22	0 / 90	VirusTotal	ua2-pool-1194.nvpn.to
2021-03-18	0 / 90	VirusTotal	ua2.nvpn.to
2019-11-30	9 / 90	VirusTotal	blackpyramid.duckdns.org

Communicating Files (10)

Scanned	Detections	Type	Name
2024-03-28	8 / 41	JavaScript	Transaction_details_403176986004195246838452.js.txt
2024-03-28	5 / 59	JavaScript	js-beautified-1.js
2024-03-26	21 / 59	JavaScript	Transaction_copy_jpg.js
2024-04-03	3 / 60	JavaScript	WesternUnion_Receipt_jpg.js
2024-03-26	21 / 59	JavaScript	js-beautified-1.js
2024-04-01	18 / 60	JavaScript	Transactions_Copy_6588098313660669616212701012 2_6589098213660669616212701010 2.js
2024-01-31	2 / 60	JavaScript	js-beautified-1.js
2024-02-18	7 / 60	JavaScript	Transaction_Ref_01302024_jpg.js
2024-03-04	21 / 59	JavaScript	js-beautified-1.js
2020-01-21	52 / 69	Win32 EXE	vbc.exe

Resecurity has reached out to both operators to learn more about this activity. Our team has arranged successful takedowns of multiple C2C servers to disrupt the new JsOutProx campaign.

References

- Solar Spider (Threat Actor) https://www.crowdstrike.com/adversaries/solar-spider/
- Financial Institutions in the Sight of New JsOutProx Attack Waves https://yoroi.company/en/research/financial-institutions-in-the-sight-of-new-jsoutprox-attack-waves/
- Multi-Staged JSOutProx RAT Targets Indian Co-Operative Banks and Finance Companies https://www.seqrite.com/documents/en/white-papers/whitepaper-multi-staged-jsoutprox-rat-target-india...
- Unveiling JsOutProx: A New Enterprise Grade Implant https://securityaffairs.com/95438/malware/jsoutprox-enterprise-grade-implant.html
- Adversary Playbook: JavaScript RAT Looking for that Government Cheese https://www.fortinet.com/blog/threat-research/adversary-playbook-javascript-rat-looking-for-that-gov...

Conclusion

The increasing abuse of Public Cloud and Web 3.0 Services is a favored tactic among threat actors to distribute malicious code. In February 2024, Resecurity highlighted this trend in a comprehensive threat research publication. This report underscored the continuous evolution of cybercriminals' arsenals and their innovative strategies to escalate global malicious campaigns.

The discovery of the new version of JSOutProx, coupled with the exploitation of platforms like GitHub and GitLab, emphasizes these malicious actors' relentless efforts and sophisticated consistency. First detected in 2019, JSOutProx remains a significant and evolving threat, particularly to financial institution customers. This year, in a worrying expansion of scope, these threat actors have broadened their horizons in the MENA region, intensifying their cybercriminal footprint.

As these threats escalate in complexity and reach, Resecurity remains vigilant in its pursuit of tracking JSOutProx and safeguarding financial institutions and their customers globally from such nefarious activities.
