

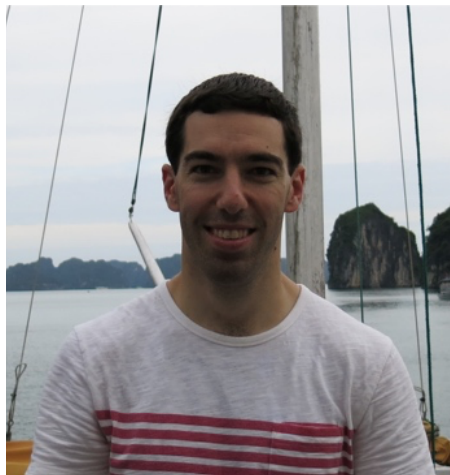
Hunting in the Dark

Ryan Kazanciyan,
Chief Security Architect

October 5, 2016



whoami



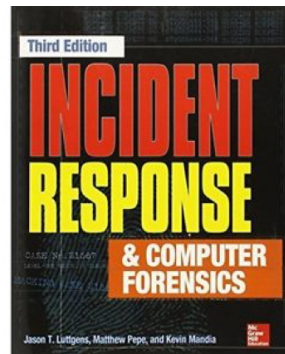
2004 - 2009



2009 - 2015



2015 - Present



Examining an environment, on a
proactive or reactive basis, for
evidence of malicious activity –
without specific investigative leads

Goals and success criteria

- Functional at enterprise-scale
- Complementary to IOC & threat feed detection
- Repeatable over time

My focus for this presentation

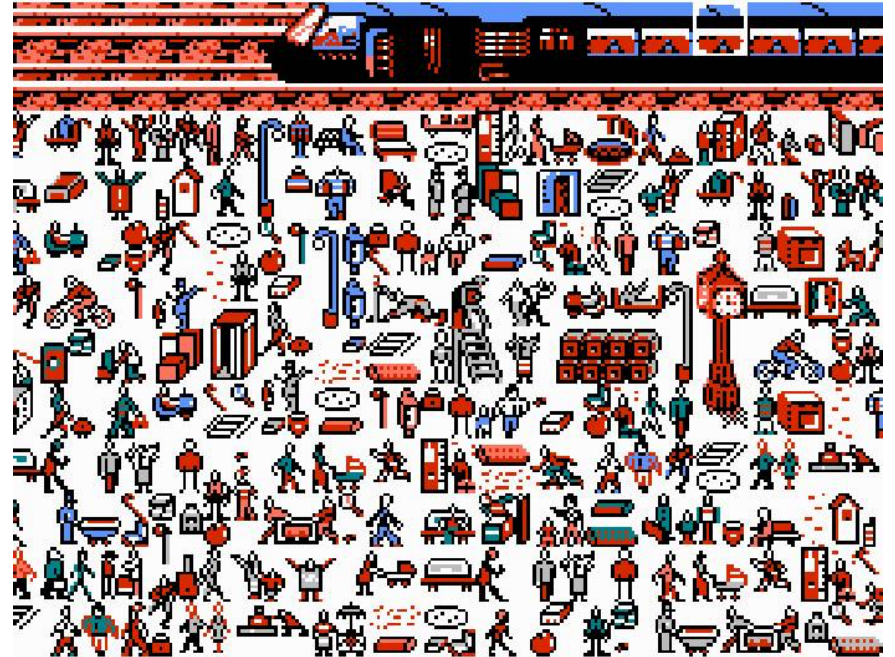
- Endpoint-centric
- Widely-available data
- Techniques, not specific tools

Common Pitfalls



**Distinguishing normal,
interesting, and bad**

Analysts often
radically
underestimate
the noise level
of an enterprise
environment



Your applications are noisy

“How many unique PE files (EXEs, DLLs, drivers) have been loaded across all my systems?”

- Different OS versions and add-ons
- User-installed applications
- Random / GUID file names & paths
- Temporary artifacts of software installers
- Updates & patches

Your users are noisy

“How often do my privileged accounts authenticate across the environment?”

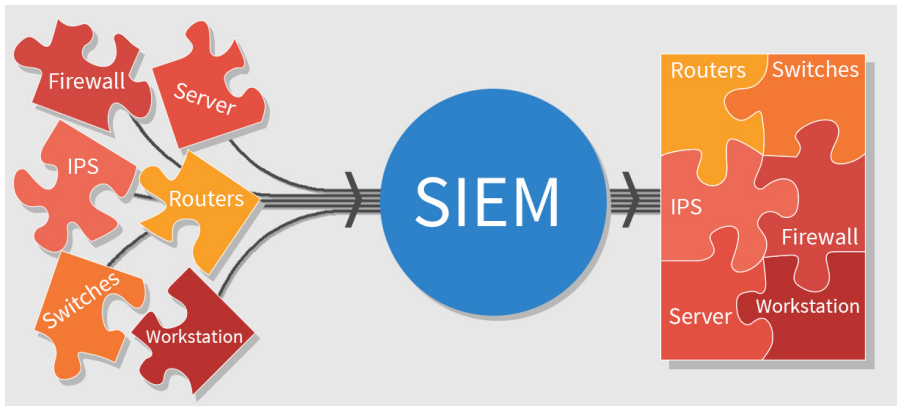
- Maintenance and administration scripts
- Ad-hoc troubleshooting
- Service and application accounts
- Misunderstood native OS behavior

**Overwhelming yourself
with data, “just in case”...**

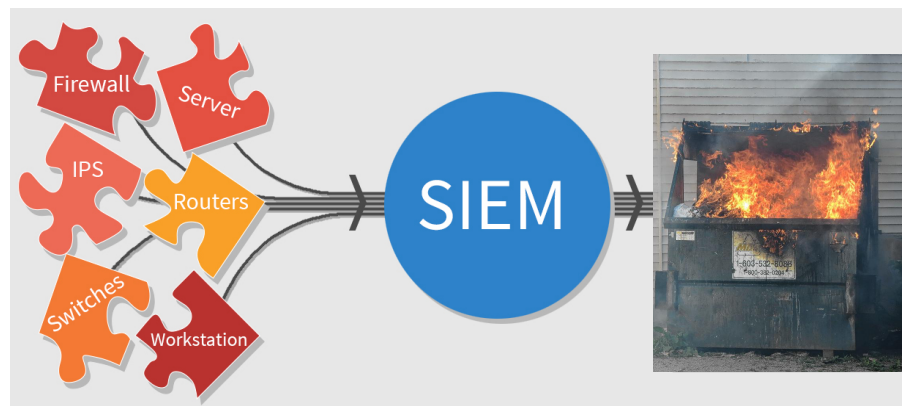
You cannot capture everything, constantly

- OS-level telemetry
- Application-level telemetry
- Data at-rest
- Volatile memory

We've been through this already...



Expectation



Reality

**Falling victim to tunnel vision
on “important assets”**

Defender bias

“...what may be critical to you...may not be the ‘crown jewels’ from the perspective of the adversary...”

“You'll find yourself hunkered down in your Maginot Line bunkers, awaiting that final assault, only to be mystified when it never seems to come.”

– Harlan Carvey

Source: <http://windowsir.blogspot.com>)

Hacking is graph traversal



John Lambert

@JohnLaTwC



Following

Your network is a directed graph of credentials. Logon rights define ingress edges. Dumped creds define egress. Hacking is graph traversal.



RETWEETS

28

FAVORITES

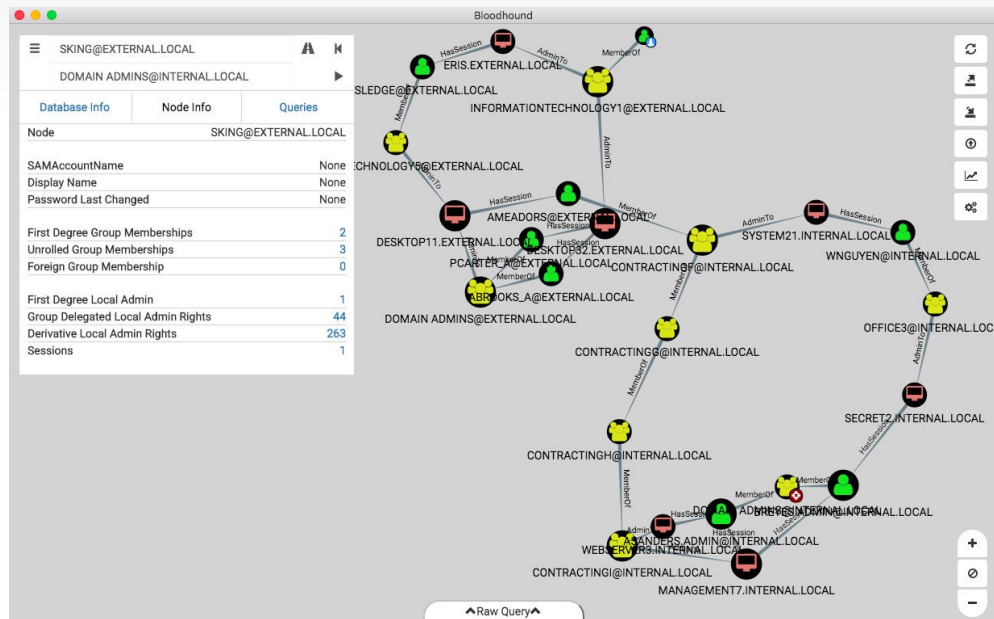
29



12:21 AM - 27 Mar 2014

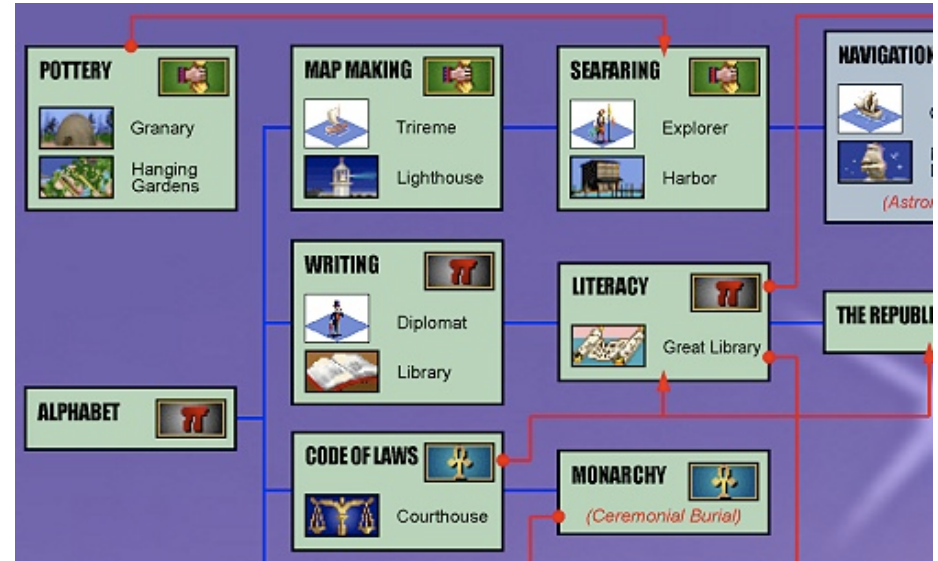
Practical example: BloodHound

- Graph analysis of AD relationships
- Identify pathways to privilege escalation



<https://github.com/adaptivethreat/BloodHound>

Developing a sustainable hunting strategy



Structuring the process



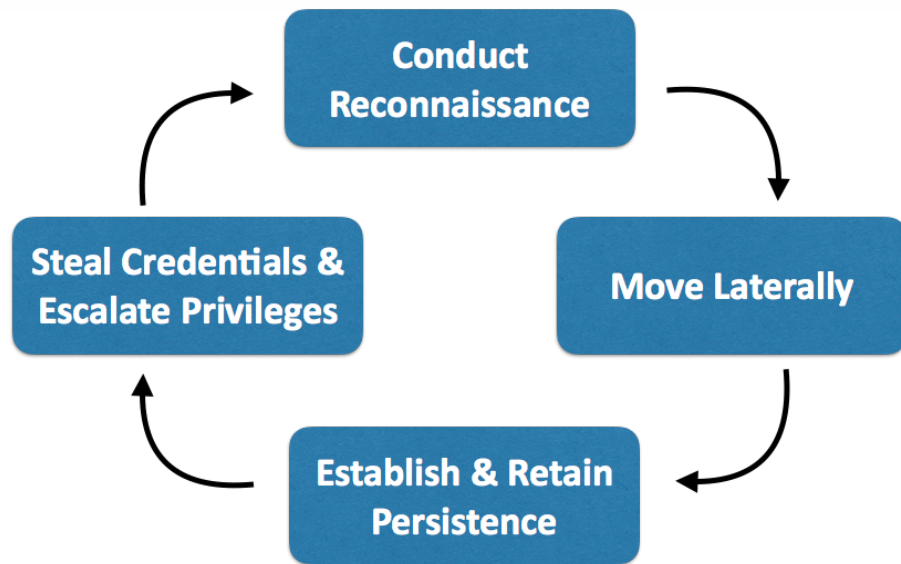
Success criteria

- Establish useful baselines
- Implement repeatable & scalable tasks
- Track “hit rate” vs. level of effort
- Drive towards automation

Focusing on High-Value Data

Prioritizing your efforts

- What are the “**lowest common denominators**” across intrusions?
- What evidence do they leave behind?
- What easily-observable outlier conditions do they create?



MITRE's "ATT^CK" framework

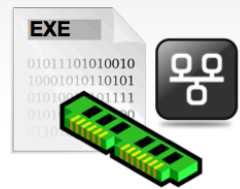
Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Execution	Collection	Exfiltration	Command and Control
Accessibility Features	Accessibility Features	Binary Padding	Brute Force	Account Discovery	Application Deployment Software	Command-Line Interface	Automated Collection	Automated Exfiltration	Commonly Used Port
AppInit DLLs	AppInit DLLs	Bypass User Account Control	Credential Dumping	Application Window Discovery	Exploitation of Vulnerability	Execution through API	Clipboard Data	Data Compressed	Communication Through Removable Media
Basic Input/Output System	Bypass User Account Control	Code Signing	Credential Manipulation	File and Directory Discovery	Logon Scripts	Graphical User Interface	Data Staged	Data Encrypted	Connection Proxy
Bootkit	DLL Injection	Component Firmware	Credentials in Files	Local Network Configuration Discovery	Pass the Hash	InstallUtil	Data from Local System	Data Transfer Size Limits	Custom Command and Control

https://attack.mitre.org/wiki/Technique_Matrix

Collecting and Managing the Data

Assess your visibility

- What's available?
- At what scale?
- How much post-processing?
- What's available ad-hoc?
- What may require “data lakes”?



Current Activity
(Processes, Network
Connections, Memory)



Historical Activity
(Telemetry, logs, alerts,
historical data)



Data at Rest
(Files on disk, registry)

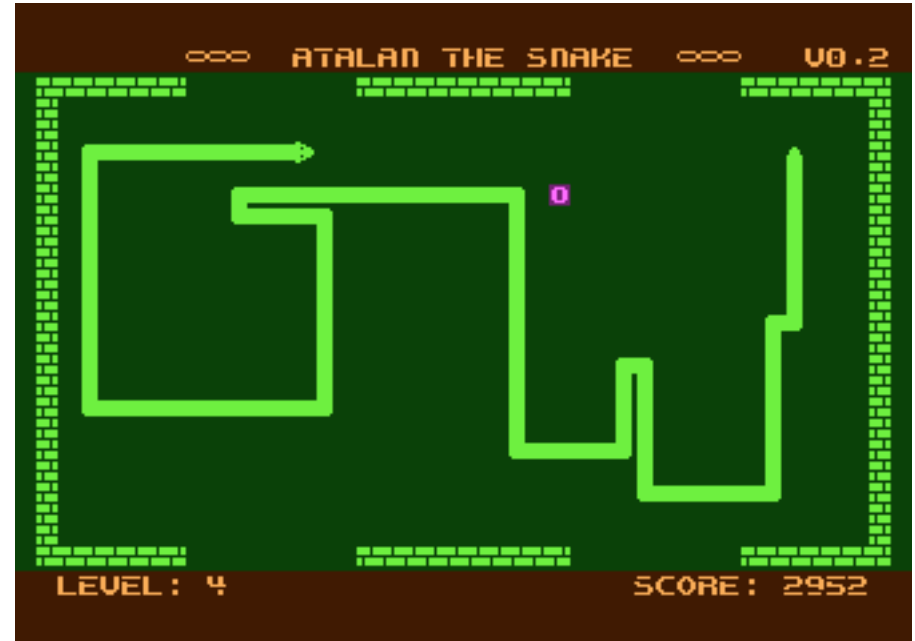


Workstations



Servers

Mitigating the “Long Tail”



Group your systems, group your data

Question: Get AutoRun Program Details from all machines with Operating System containing "windows"

Search

Save this question | Copy to Question Builder

View:   

Question: Get AutoRun Program Details from all machines with Installed Applications containing "sql server"

Search

Save this question | Copy to Question Builder

View:   

Items:
71 (147 total)

Filter by Computer Group

Contains

Filter by Text



Live Updates: On | 100%

Clear Sort

Text Wrap: ☐

Merge



AutoRun Program Details

Entry	Category	Description	Publisher	Image Path	Version	MD5 Hash	Count ↓
☒ mlx4_bus	Drivers	MLX4 Bus Driver	Mellanox	c:\windows\system32\drivers\mlx4_bus.sys	4.4.13905.0	13340C572F24BB6AFAD3AE034BEC	3
☐ megasr	Drivers	LSI MegaRAID Software RAID	LSI Corporation, Inc.	c:\windows\system32\drivers\megasr.sys	15.2.2013.129	F6F13533196DE7A582D422B0241E4	3
☐ LSI_SAS3	Drivers	LSI SAS Gen3 Driver (StorPort)	LSI Corporation	c:\windows\system32\drivers\lsi_sas3.sys	2.50.65.1	04D1274BB9BBCCF12BD12374002A	3
☐ LSI_SSS	Drivers	LSI SSS PCIe/Flash Driver (StorPort)	LSI Corporation	c:\windows\system32\drivers\lsi_sss.sys	2.10.61.81	327469EEF3833D0C584B7E88A76A	3
☐ LSI_SAS2	Drivers	LSI SAS Gen2 Driver (StorPort)	LSI Corporation	c:\windows\system32\drivers\lsi_sas2.sys	2.0.60.82	ADAC09CBE7A2040B7F68B5E5C9A	3
☐ bxois	Drivers	iSCSI offload x64 FREE	Broadcom Corporation	c:\windows\system32\drivers\bxois.sys	7.4.4.0	4DFA44593FAFB909D261898461D6E	3
☐ ibbus	Drivers	InfiniBand Fabric Bus Driver	Mellanox	c:\windows\system32\drivers\ibbus.sys	4.4.13905.0	BF00494818FD9E0B3E841B93A1847	3
☐ mvumis	Drivers	Marvell Flash Controller Driver	Marvell Semiconductor, Inc.	c:\windows\system32\drivers\mvumis.sys	1.0.5.1015	B8C35C94DCB2DFAF03BB42131F	3
☐ BALLOON	Drivers	Red Hat VirtIO Balloon driver	Red Hat Inc.	c:\windows\system32\drivers\balloon.sys	62.71.104.9600	C975501DD5B43ADE3717722B81AC	3

Practical Examples

How to draw an Owl.

"A fun and creative guide for beginners"



Fig 1. Draw two circles



Fig 2. Draw the rest of the damn Owl

Hunting for Lateral Command Execution

Lateral command execution

Common Methods



Scheduled
Tasks



WinRM &
PowerShell



PsExec,
WMIC



Sources of Evidence



Logon & service
events



Process
history



Other forensic
artifacts

Analysis Criteria

Who?



Accounts used

Where?



Source & target
systems

When?



Time & frequency

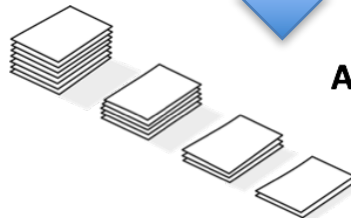
What?



Executed commands,
dropped files, etc.

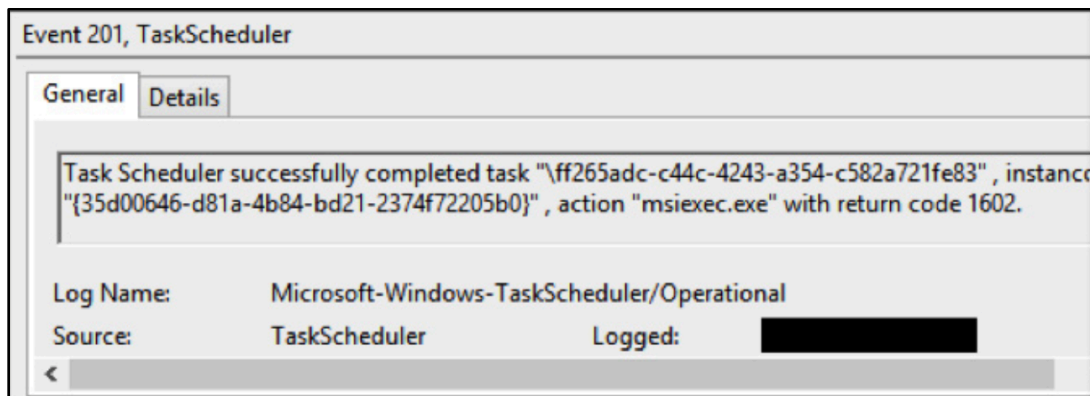


Assess outliers



Example: Duqu 2.0 and Scheduled Tasks

“In addition to creating services to infect other computers in the LAN, attackers can also use the Task Scheduler to start ‘msiexec.exe’ remotely. The usage of Task Scheduler during Duqu infections for lateral movement was also observed with the 2011 version...”



Source:

[https://securelist.com/files/2015/06/The Mystery of Duqu 2 0 a sophisticated cyber espionage actor returns.pdf](https://securelist.com/files/2015/06/The_Mystery_of_Duqu_2_0_a_sophisticated_cyber_espionage_actor_returns.pdf)

What was the shared IOC?

Authored: 06/10/2015 by Kaspersky Lab

Description: Indicators of compromise for the Duqu 2.0 <https://securelist.com/mystery-of-duqu-2-0-a-sophisticated-cyberespionage-actor/52000/>

OR

- FileItem/Md5sum is 089a14f69a31ea5e9a5b375dc0c46e45
- FileItem/Md5sum is 16ed790940a701c813e0943b5a27c6c1
- FileItem/Md5sum is 26c48a03a5f3218b4a10f2d3d9420b97
- FileItem/Md5sum is a6dcae1c11c0d4dd146937368050f655
- FileItem/Md5sum is acbf2d1f8a419528814b2efa9284ea8b
- FileItem/Md5sum is c04724afdb6063b640499b52623f09b5
- FileItem/Md5sum is e8eaec1f021a564b82b824af1dbe6c4d
- FileItem/Md5sum is 10e16e36fe459f6f2899a8cea1303f06
- FileItem/Md5sum is 48fb0166c5e2248b665f480deac9f5e1
- FileItem/Md5sum is 520cd9ee4395ee85ccb073a00649602
- FileItem/Md5sum is 7699d7e0c7d6b2822992ad485caacb3e

- FileItem/Md5sum is 84c2e7ff26e6dd500ec007d6d5d2255e
- FileItem/Md5sum is 856752482c29bd93a5c2b62ff50df2f0
- FileItem/Md5sum is 85f5feeed15b75cacb63f9935331cf4e
- FileItem/Md5sum is 8783ac3cc0168ebaef9c448fbe7e937f
- FileItem/Md5sum is 966953034b7d7501906d8b4cd3f90f6b
- FileItem/Md5sum is a14a6fb62d7efc114b99138a80b6dc7d
- FileItem/Md5sum is a6b2ac3ee683be6fbbbab0fa12d88f73
- FileItem/Md5sum is cc68fcc0a4fab798763632f9515b3f92
- FileItem/Md5sum is 3f52ea949f2bd98f1e6ee4ea1320e80d
- FileItem/Md5sum is c7c647a14cb1b8bc141b089775130834
- PortItem/remotelP contains 182.253.220.29
- PortItem/remotelP contains 186.226.56.103

How could we do better?

Hunting tasks with event logs

Microsoft-Windows-TaskScheduler/Operational.evtx

Event IDs to harvest

- 106 (**Task Registered**)
- 129 (**Created Task Process**)
- 200 (**Action Started**)
- 201 (**Action Completed**)

Fields to filter & stack

- **TaskName** (What's the task called?)
- **UserContext** (Who registered it?)
- **ActionName** (What did it run?)

Stack and search workflow

Stack by User

```
User, Count
...
CORP\ITAdmin,9761
CORP\InfosecAdmin,2240
CORP\DomainAdmin,77
CORP\EBachman,5
AlicePC\Administrator,3
ACMEDC01\Administrator,2
CORP\RHendricks,1
...
```

Stack by ActionName

```
Path, Count
...
C:\Program Files (x86)\Google\Update\GoogleUpdate.exe,1862
c:\temp\patchFix.exe,310
...
c:\test.bat,15
c:\dump.bat,7
c:\temp\1.bat,3
...
```

Summarized "At" Task Events

```
Hostname, Timestamp, EID, UserContext, ActionName
...
AlicePC,2015-08-15 16:44:33,106,AlicePC\Administrator,N/A
AlicePC,2015-08-15 16:45:00,200,N/A,c:\test.bat
...
BobPC,2015-08-15 16:47:18,106,CORP\ITAdmin,N/A
BobPC,2015-08-15 16:49:00,200,N/A,c:\dump.exe
...
AcmeDC01,2015-06-15 09:44:00,200,N/A,c:\temp\patchFix.exe
AcmeDB02,2015-06-15 09:51:00,200,N/A,c:\temp\patchFix.exe
AcmeEX02,2015-06-15 09:37:00,200,N/A,c:\temp\patchFix.exe
...
```

Remember: must match EID 106 with EID 200 events
by proximity and sequencing – no correlation ID!

Stack and search workflow

Stack by User

User, Count

...

CORP\ITAdmin,9761

CORP\InfosecAdmin,2240

CORP\DomainAdmin,77

CORP\EBachman,5

AlicePC\Administrator,3

ACMEDC01\Administrator,2

CORP\RHendricks,1

...

Stack by ActionName

Path, Count

Hostna

...

AliceP

AliceP

...

BobPC,

BobPC,

...

AcmeDC

AcmeDB

AcmeEX

```
Aticorp\Administrator,5  
ACMEDC01\Administrator,2  
CORP\RHendricks,1  
...
```

Stack by ActionName

```
Path, Count
```

```
...  
C:\Program Files (x86)\Google\Update\GoogleUpdate.exe,1862  
c:\temp\patchFix.exe,310  
...  
c:\test.bat,15  
c:\dump.bat,7  
c:\temp\1.bat,3  
...
```

```
Aticorp\Administrator,5  
ACMEDC01\Administrator,2  
CORP\RHendricks,1  
...  
BobF...  
BobF...  
...  
Acme...  
Acme...  
Acme...  
...
```

Summarized "At" Task Events

Hostname, Timestamp, EID, UserContext, ActionName

...

AlicePC,2015-08-15 16:44:33,106,AlicePC\Administrator,N/A

AlicePC,2015-08-15 16:45:00,200,N/A,c:\test.bat

...

BobPC,2015-08-15 16:47:18,106,CORP\ITAdmin,N/A

BobPC,2015-08-15 16:49:00,200,N/A,c:\dump.exe

...

AcmeDC01,2015-06-15 09:44:00,200,N/A,c:\temp\patchFix.exe

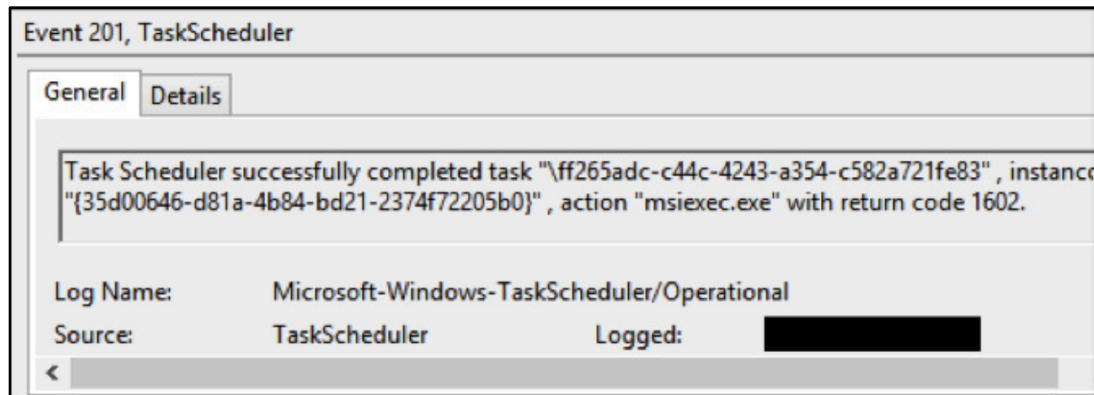
AcmeDB02,2015-06-15 09:51:00,200,N/A,c:\temp\patchFix.exe

AcmeEX02,2015-06-15 09:37:00,200,N/A,c:\temp\patchFix.exe

...

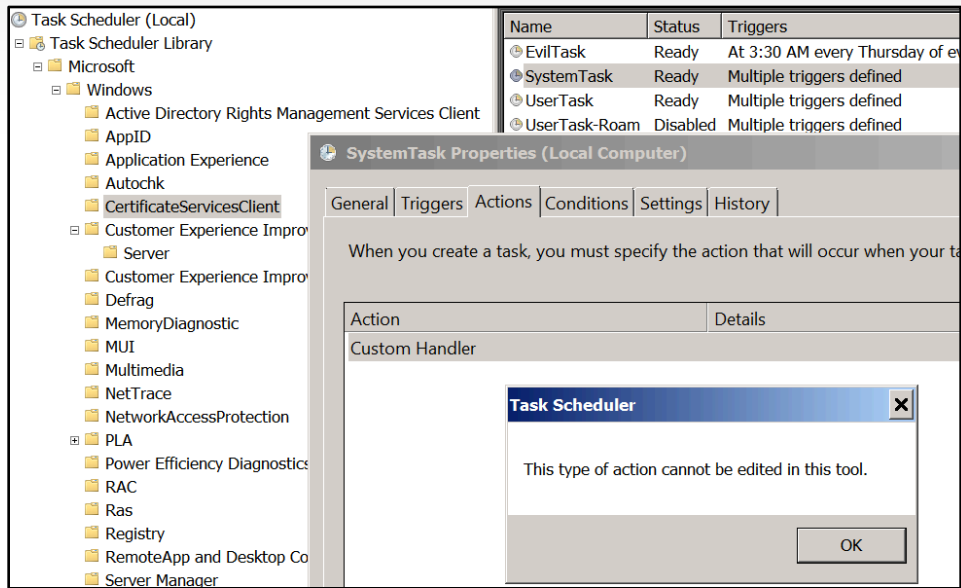
Revisiting our example: Duqu 2.0

- How common are remotely registered tasks with `ActionName="msiexec.exe"`
- By user? By time? By endpoints?
- Could you have found this proactively, without any leads?



Interesting blind spot: COM handler tasks

- Cannot examine or edit in Task Viewer UI
- ActionName is a descriptive string
- How do you know what this loads?



EID 200

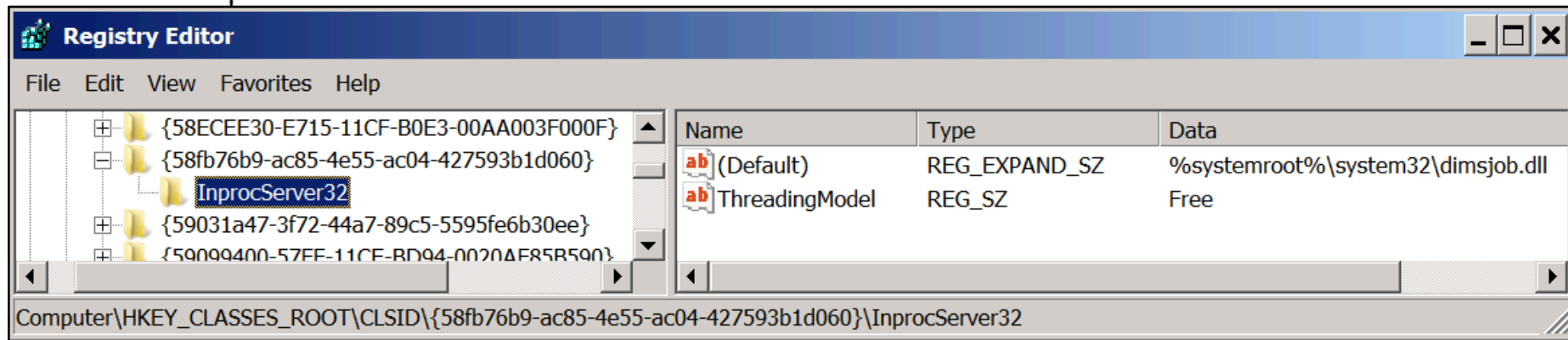
```
<EventData Name="ActionStart">
  <Data Name="TaskName">\Microsoft\Windows\CertificateServicesClient\SystemTask</Data>
  <Data Name="ActionName">Certificate Services Client Task Handler</Data>
  <Data Name="TaskInstanceId">{A9D9ED23-7D59-4D64-817F-206F18CF6A75}</Data>
</EventData>
```

Mapping COM handler to associated DLL

SystemTask.job

```
<Task version="1.3"
xmlns="http://schemas.microsoft.com/windows/2004/02/mit/task">
...
  <Actions Context="LocalSystem">
    <ComHandler>
      <ClassId>{58fb76b9-ac85-4e55-ac04-427593b1d060}</ClassId>
      <Data>SYSTEM</Data>
    </ComHandler>
  </Actions>
```

CLSID →



Attacker limitations

- Must import task configuration XML file if using COM

```
schtasks /Create /XML c:\EvilTask.xml  
/TN Microsoft\Windows\CertificateServicesClient\EvilTask
```

- Cannot modify existing tasks without breaking hash
 - Stored in the registry
 - Stuxnet exploited weak task hash algorithm in older versions of Windows

Other approaches to scheduled task analysis

- Examine configurations for “at-rest” tasks
- Detect anomalous tasks as they are created



Stacking “current” tasks: Command lines

Task Name	Command	Count ↑
At1	cmd.exe /c c:\drop.exe	1
Pulse_DiskMemCPU_AlwaysOn	Pulse_DiskMemCPU_AlwaysOn	1
Tanium_Pulse	logman start Tanium_Pulse	1
At2	cmd.exe /c c:\drop.exe	1
Adobe Acrobat Update Task	C:\Program Files\Common Files\Adobe\ARM\1.0\AdobeARM.exe	1
Adobe Flash Player Updater	C:\Windows\system32\Macromed\Flash\FlashPlayerUpdateService.exe	2
GoogleUpdateTaskMachineUA	C:\Program Files\Google\Update\GoogleUpdate.exe /ua /installsource scheduler	3
GoogleUpdateTaskMachineCore	C:\Program Files\Google\Update\GoogleUpdate.exe /c	3

Stacking “current” tasks: Full metadata

Description	Publisher	Image Path	Launch String	ASEP Hash	Count ↓
Microsoft® Windows Backup	Microsoft Corporation	c:\windows\system32\sdclt.exe	%systemroot%\System32\sdclt.exe /CONFIGNC	8D4E7579031F773DFC73D54C72E865CCCDAC	183
Adobe® Flash® Player Update Service 16.0 r0	Adobe Systems Incorporated	c:\windows\system32\macromed\flash	C:\Windows\system32\Macromed\Flash\FlashPla	CDBD0CE61D2CD73F985C374A9C73CA803A6E	43
Adobe® Flash® Player Update Service 11.7 r700	Adobe Systems Incorporated	c:\windows\system32\macromed\flash	C:\Windows\system32\Macromed\Flash\FlashPla	F5929E2BF7D646A839AAE822878707EFDA95A	41
Adobe® Flash® Player Update Service 13.0 r0	Adobe Systems Incorporated	c:\windows\system32\macromed\flash	C:\Windows\system32\Macromed\Flash\FlashPla	CF84E2B7E0A2DDB92D4CE66151D418B0E189	39
Adobe® Flash® Player Update Service 12.0 r0	Adobe Systems Incorporated	c:\windows\system32\macromed\flash	C:\Windows\system32\Macromed\Flash\FlashPla	6759B187498344AEC0546543D09451411190A	28
Microsoft Windows Backup	Microsoft Corporation	c:\windows\system32\sdclt.exe	%systemroot%\System32\sdclt.exe /CONFIGNC	8D4E7579031F773DFC73D54C72E865CCCDAC	26
Windows Rights Management client	Microsoft Corporation	c:\windows\system32\msdrm.dll	HKCR\CLSID\{BF5CB148-7C77-4D8A-A53E-D8}	D9A8F587560D2066629FA235FAC0969F09DD3E	9
Digital TV Tuner device registration application.	Microsoft Corporation	c:\windows\ehome\ehprivjob.exe	%SystemRoot%\ehome\ehPrivJob.exe /wait:7 /F	B4C123C7D17A3C44E74880B4108766E8CE4B6	9
MsCtfMonitor DLL	Microsoft Corporation	c:\windows\system32\msctfmonitor.dll	HKCR\CLSID\{01575CFE-9A55-4003-A5E1-F38}	C8F5413362A3F81EA33CF4298D44B303C8D5	9
Windows Media Center Store Update Manager	Microsoft Corporation	c:\windows\ehome\mcupdate.exe	%SystemRoot%\ehome\mcupdate.exe \$(Arg0)	96AF86DC7CDD73E5FC93AF05D53CC1AF972E	9
Windows Media Center Store Update Manager	Microsoft Corporation	c:\windows\ehome\mcupdate.exe	%SystemRoot%\ehome\mcupdate.exe -SqlLiteF	96AF86DC7CDD73E5FC93AF05D53CC1AF972E	9
MUI Language pack cleanup	Microsoft Corporation	c:\windows\system32\lpremove.exe	%windir%\system32\lpremove.exe	6BBA8EB7F665954DCCE33E0C0E4810B130E6	9
Digital TV Tuner device registration application.	Microsoft Corporation	c:\windows\ehome\ehprivjob.exe	%SystemRoot%\ehome\ehPrivJob.exe /PBDAD	B4C123C7D17A3C44E74880B4108766E8CE4B6	9
Windows Media Player Network Sharing Service	Microsoft Corporation	c:\program files\windows media player	%ProgramFiles%\Windows Media Player\wmpns	57F5B40476BC6E501FBD7CF2E075B05C0337E	9

“In-motion” remote task creation

Trace Executed Processes[1424906688340 1551153240000, 0, 1, 0, 0, , , at \, , ,]						
Endpoint Name ↑	Start Time	Parent Process Path	Process Path	Command Line	MD5	Domain
Win7-x64-Client	Fri, 26 Feb 2016 00:06:32 +0000	C:\Windows\System32\cmd.exe	C:\Windows\System32\at.exe	at \\127.0.0.1 12:01 cmd.exe /c "c:\drop.exe";	ED4CD10A13080E51F550F560F7	WIN7-X64-Client

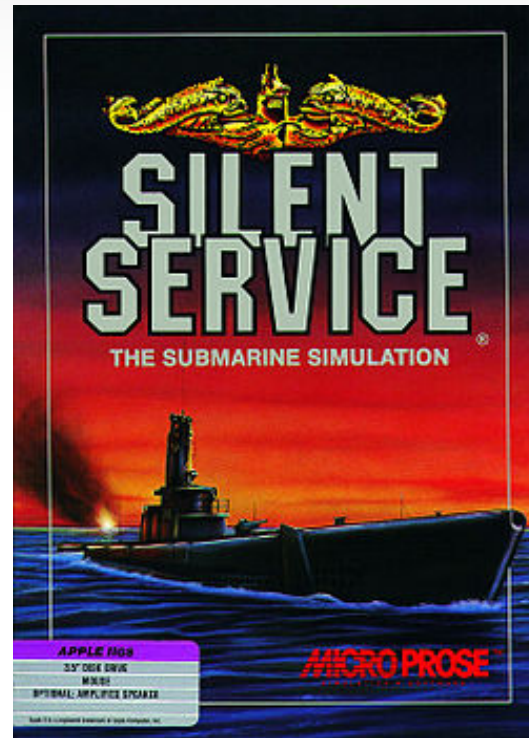
Command Line

at \\127.0.0.1 12:01 cmd.exe /c

Hunting for Services

Windows Services

- Common persistence mechanism for long-running malware
- Loading mechanism for short-lived tools
- Installation and usage leaves behind evidence
 - Registry
 - Service control manager
 - Event logs



Examples from the wild

A service was installed in the system.

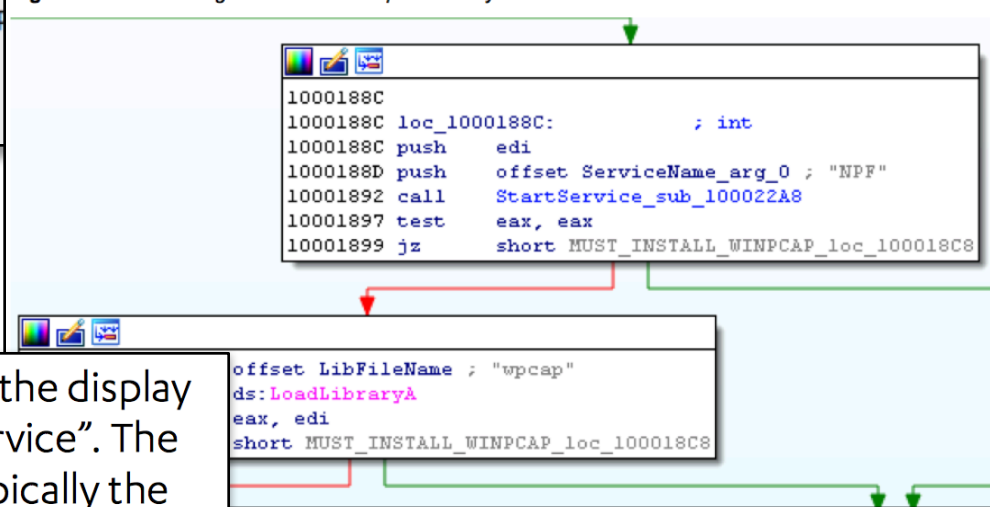
Service Name: svcmsi_32@
Service File Name: msixec /i %systemroot%\temp\tmp129deb0.tmp /q P
Service Type: user mode service
Service Start Type: demand start
Service Account: LocalSystem

Duqu 2.0 (Kaspersky)

The service typically has name javamtsup, and the display name is "Java(TM) Virtual Machine Support Service". The size of the service binary on disk varies, but typically the real size is 5120 bytes (based on PE headers) and the SHA1 value is 7803f160af428bcfb4b9ea2aba07886f232cde4e.

CosmicDuke (F-Secure)

Figure 4: Checking to see if WinPcap is already installed



BlackEnergy installing WinPCAP (Arbor Networks)

Service activity in event logs

EID 7045



Log Name: System
Source: Service Control Manager
Date: 8/26/2015 9:19:13 AM
Event ID: 7045
Task Category: None
Level: Information
Keywords: Classic
User: AlicePC\Administrator
Computer: AlicePC.acme.com
Description: A service was installed in the system.



```
...  
<EventData>  
  <Data Name="ServiceName">TestService</Data>  
  <Data Name="ImagePath">c:\windows\system32\evil.exe</Data>  
  <Data Name="ServiceType">user mode service</Data>  
  <Data Name="StartType">auto start</Data>  
  <Data Name="AccountName">LocalSystem</Data>  
</EventData>  
...
```

Stacking service creation events

- “Who created which services?”
- “When and where?”
- `ServiceName` + `ImagePath` + `User` from EID 7045
- Use time and hostname to further sub-filter

Example / Case Study: Harvesting PsExec service events

Blind spots and noise

- Attackers can install services without calling `CreateService`
 - Avoids generating event log entry
 - Still may leave evidence in registry
- Many 3rd party applications install services
- Noise over time

Stack analysis of current services

Service Module Details with Hash[sha256]						Count ↓
Name ↑	Caption	Image Path	Binary Path	Hash of Binary		
protectedstorage	Protected Storage	C:\Windows\system32\lsass.exe	C:\Windows\system32\lsass.exe	29b0a8889857cebfa6cbd795d5ee		233
samss	Security Accounts Manage	C:\Windows\system32\lsass.exe	C:\Windows\system32\lsass.exe	29b0a8889857cebfa6cbd795d5ee		233
vaultsvc	Credential Manager	C:\Windows\system32\lsass.exe	C:\Windows\system32\lsass.exe	29b0a8889857cebfa6cbd795d5ee		233
storsvc	Storage Service	C:\Windows\System32\svchost.exe -k Loc	C:\Windows\system32\storsvc.dll	83eebacde4f69a2866b69caa633f5		231
watadminsfc	Windows Activation Techno	C:\Windows\system32\Wat\WatAdminSvc	C:\Windows\system32\Wat\WatAdmin	ffae53b6b53aefc9e8a10bf27480e0		228
tanium client	Tanium Client	"C:\Program Files\Tanium\Tanium Client\Te	C:\Program Files\Tanium\Tanium Clie	c6286a449e53e0d66dff31629854c		202
adobearmservice	Adobe Acrobat Update Ser	"C:\Program Files\Common Files\AdobeA	C:\Program Files\Common Files\Adob	69514e5177a0aea89c27c2234712		81
adobearmservice	Adobe Acrobat Update Ser	"C:\Program Files\Common Files\AdobeA	C:\Program Files\Common Files\Adob	f419e159d3e428a3929a1a983142e		67
adobeflashplayerup	Adobe Flash Player Update	C:\Windows\system32\Macromed\Flash\F	C:\Windows\system32\Macromed\Flas	75808821fbc732d0504795b8f8585		55
adobeflashplayerup	Adobe Flash Player Update	C:\Windows\system32\Macromed\Flash\F	C:\Windows\system32\Macromed\Flas	68b4338da3042dbb891a8583ee32		51
adobeflashplayerup	Adobe Flash Player Update	C:\Windows\system32\Macromed\Flash\F	C:\Windows\system32\Macromed\Flas	fc34b8cbded5a63325b251077b19		45
adobeflashplayerup	Adobe Flash Player Update	C:\Windows\system32\Macromed\Flash\F	C:\Windows\system32\Macromed\Flas	11c452d77d0a8a5e430d0d0c9949		34

Further persistence analysis

Items: **2,490** (95,262 total)

Live Updates: **Off** |  **100%** (100% available)

Items: **355** (15,152 total) Filter by Computer

Live Updates: **Off** |  **100%** (100% available)

Items: **225** (8,298 total) Filter by Computer Group

Live Updates: **Off** |  **100%** (100% available)

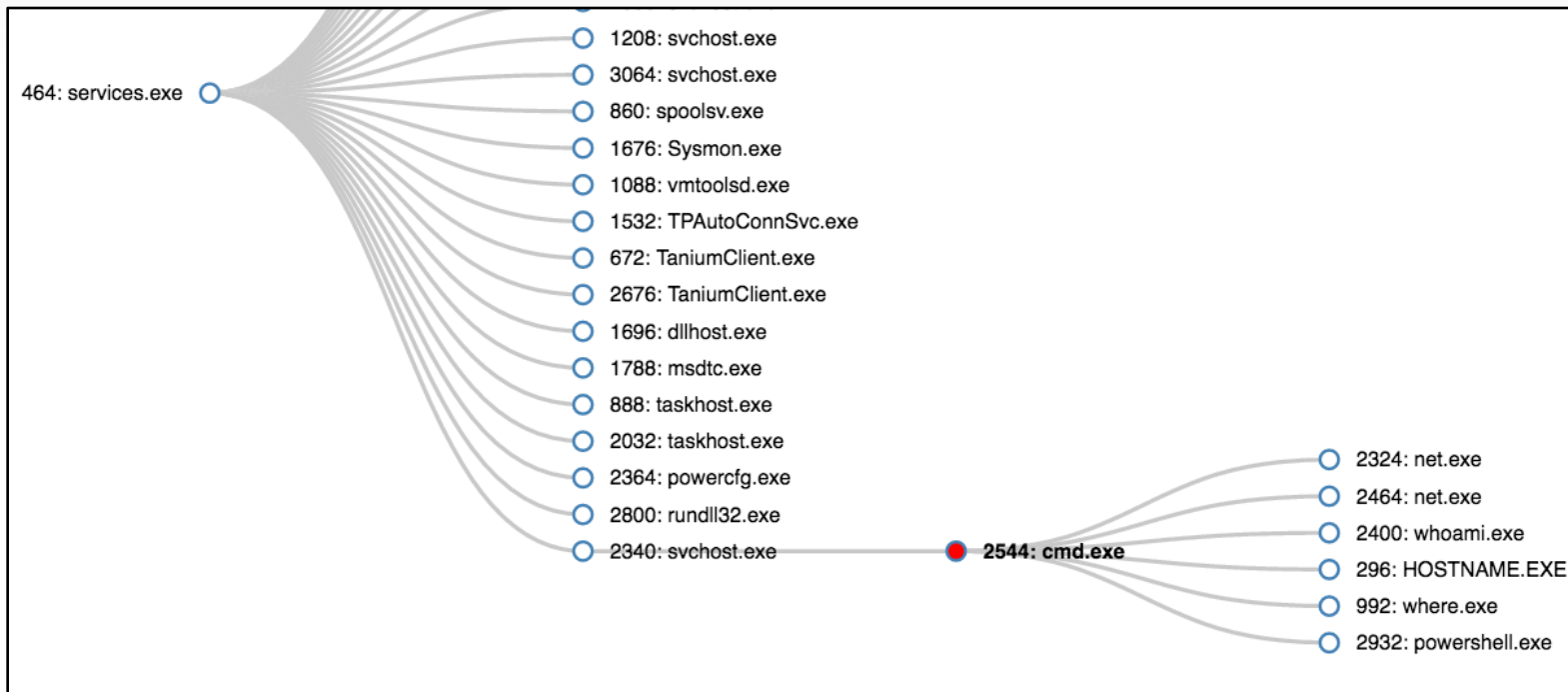
Items: **4** (501 total) Filter by Computer Group

Live Updates: **On** |  **100%**

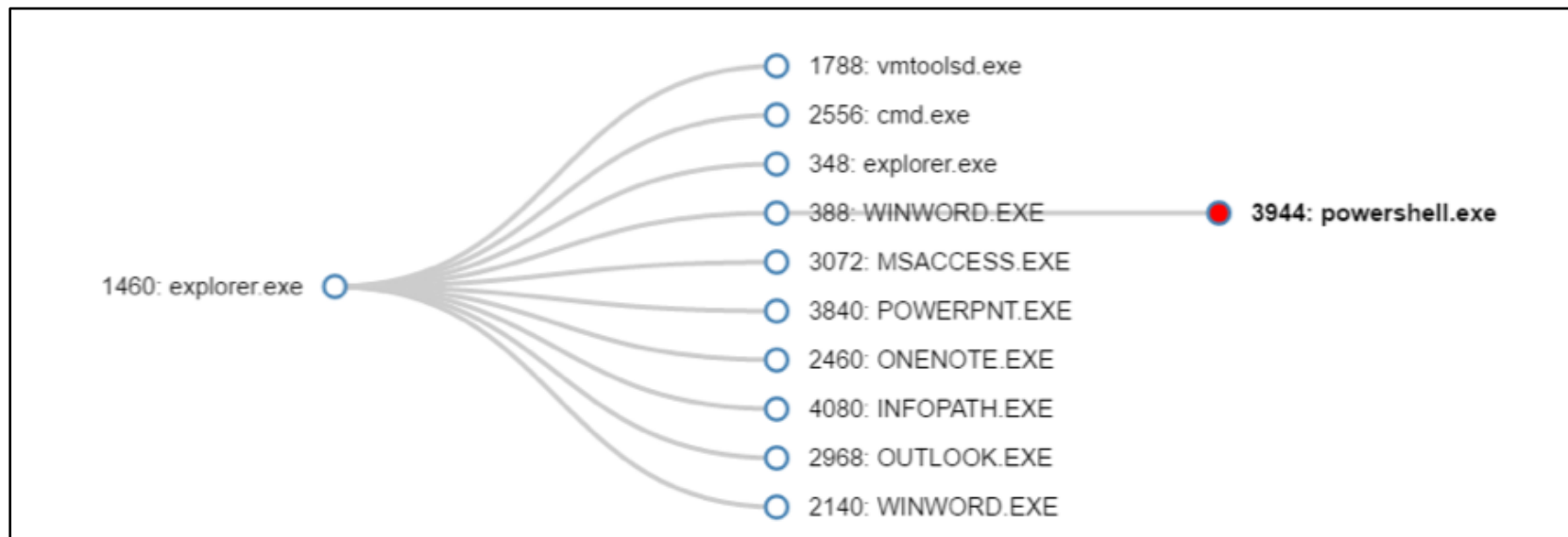
WMI Event Consumers			
	Name ↑	Class	Command Line Template
☐	SCM Event Log Consumer	NTEventLogEventConsumer	N/A
☐	BVTConsumer	CommandLineEventConsumer	cscript KernCap.vbs
☐	DoBadThings	CommandLineEventConsumer	C:\Windows\System32\eventv
☐	EvilConsumer	CommandLineEventConsumer	C:\Windows\System32\evil

Stack Analysis of Process Trees

What are process trees?

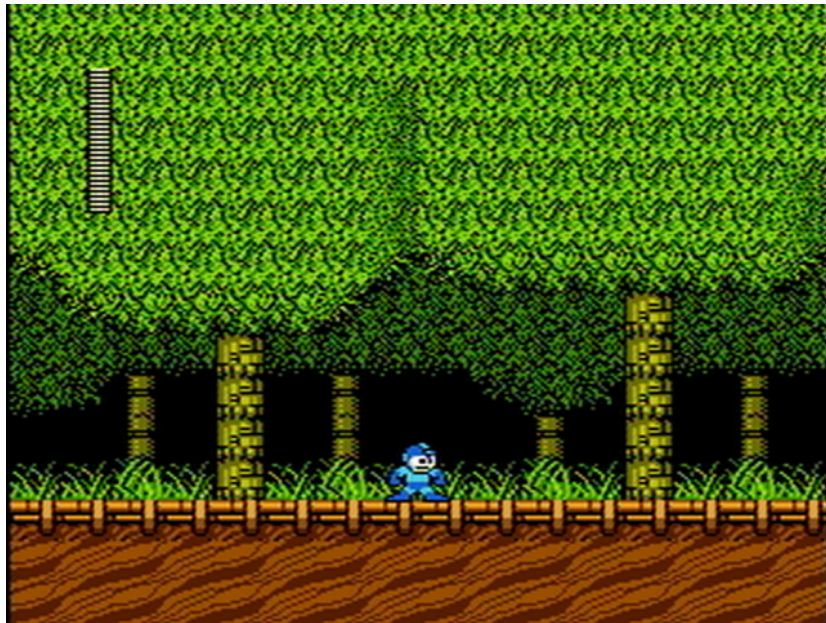


Another example



Finding interesting trees in forests

- Attackers often use native OS commands
- Attackers often exploit or leverage native OS services
- Both should result in outlier process lineages



Sources of data

- For each running process, record:
 - ImageName (or ImagePath)
 - Parent name (or path)
- Options in Windows
 - Process auditing (native, optional setting)
 - Sysmon (free Microsoft SysInternals driver)
- Post-processing to “flatten” and stack

Example: cmd.exe as child

Trace Executed Process Trees[cmd.exe, 0, As Child] ↑	Count ↓
System Idle Process -> explorer.exe -> cmd.exe	3
System Idle Process -> cmd.exe	2
System Idle Process -> cscript.exe -> cmd.exe	1
System Idle Process -> wininit.exe -> services.exe -> TaniumClient.exe -> TaniumClient.exe -> cmd.exe	1
System Idle Process -> winlogon.exe -> userinit.exe -> explorer.exe -> cmd.exe	1
System Idle Process -> winlogon.exe -> userinit.exe -> explorer.exe -> cmd.exe -> cmd.exe	1
System Idle Process -> winlogon.exe -> userinit.exe -> explorer.exe -> iexplore.exe -> iexplore.exe -> cmd.exe	1

Example: svchost.exe as parent

Trace Executed Process Trees[svchost.exe, 0, As Parent] ↑	Count ↓
svchost.exe -> dwm.exe	8
svchost.exe -> WmiPrvSE.exe	8
svchost.exe -> rdpclip.exe	3
svchost.exe -> cmd.exe -> HOSTNAME.EXE	1
svchost.exe -> cmd.exe -> net.exe	1
svchost.exe -> cmd.exe -> net.exe -> net1.exe	1
svchost.exe -> cmd.exe -> powershell.exe	1

Interesting targets

- Shells & interpreters

- cmd.exe
- powershell.exe
- cscript.exe
- wscript.exe
- mshta.exe
- rundll32.exe

- Tools for lateral movement

- net.exe
- at.exe

- schtasks.exe / taskeng.exe
- wmic.exe / wmiprvse.exe

- Commonly “hijacked” processes

- svchost.exe
- w3wp.exe
- iexplore.exe
- winword.exe
- excel.exe

Conclusion



Next steps

- Pick one of these techniques and practice!
- Learn the “noise” of your own environment
- Incorporate into red-vs-blue team exercises
- Ensure endpoint tools enable rapid search and harvesting
 - Volatile activity
 - Data “at-rest”
 - Historical telemetry

Thank you!

ryan.kazanciyan [at] tanium.com
@ryankaz42

