RSA® Charge 2016

RSA Mass Triage: Hunting Polar Bears in a Blizzard

Brian Baskin - RSA Incident Response Practice - @bbaskin

Steve Brzozowski - RSA Incident Response Practice - @stevebuho

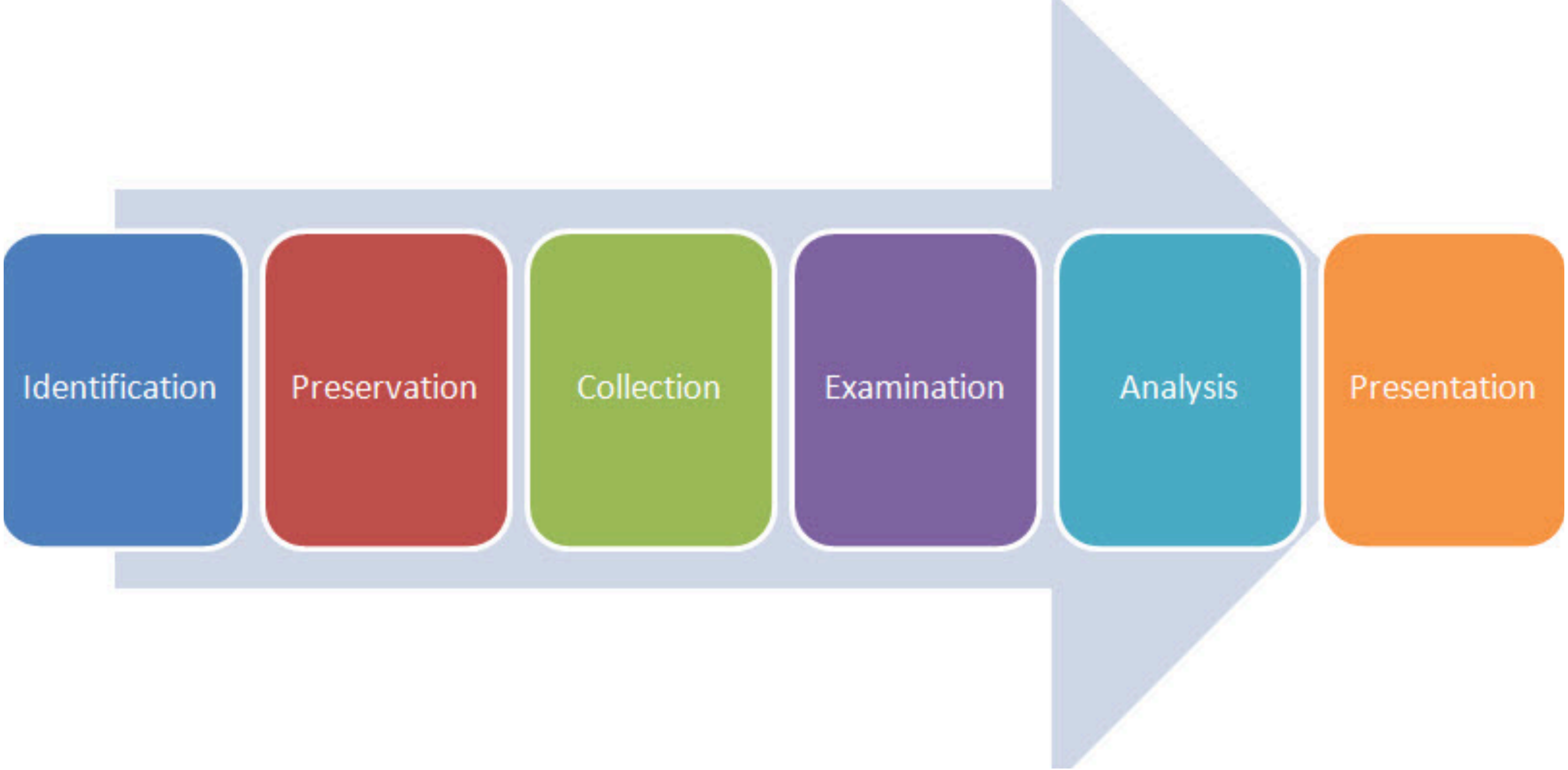
#RSACharge

RSA Incident Response Practice

- Global Practice across North America, Europe, & Asia
- RSA NetWitness Packets, Logs & Endpoint as well as other industry, open source, and custom tools for:
  - Network intrusions
  - Host-based forensics
  - Malware analysis
  - Reverse engineering

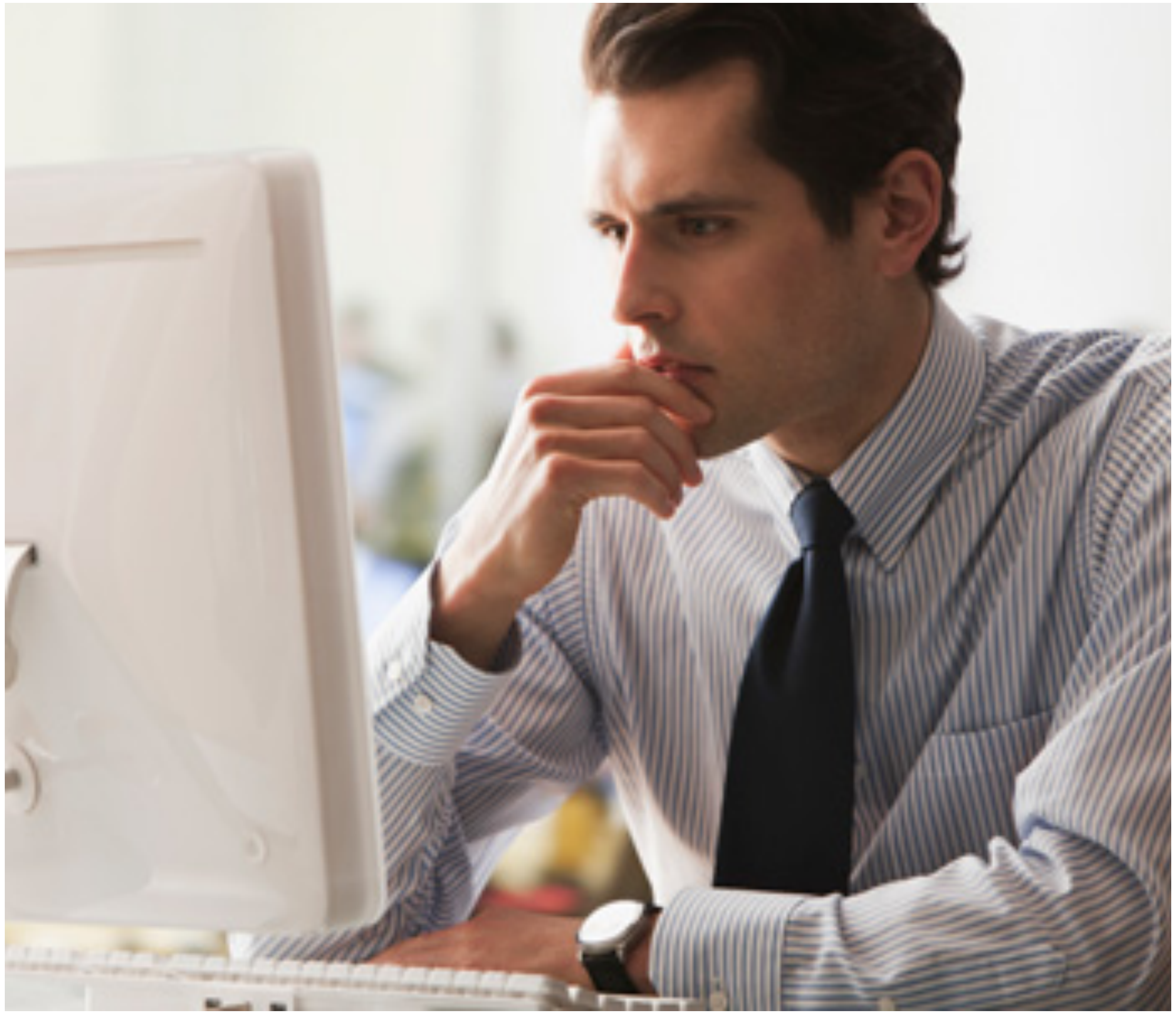
The Forensic Process

Digital Forensics Process



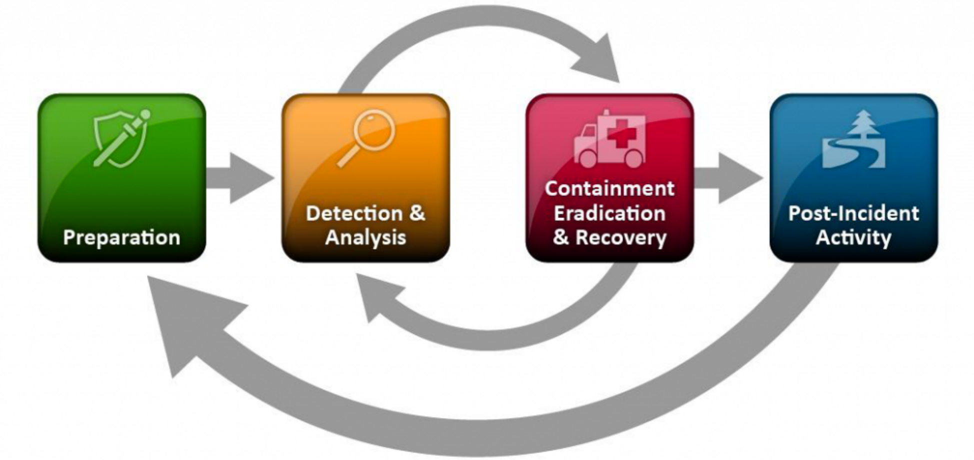
Traditional Incident Response

- **Review Systems One-at-a-Time**
  Single-step analysis and scoping
- **Collect Thousand of Artifacts to find Single Indicators**
  Manually pull data from various sources, files, folders
  Use collection of specialized commercial and free tools to analyze
- **Can Take Weeks and Months to Investigate**
  "Dead box" – Turn off, image, analyze.
  Slow process for large compromises



Traditional IR is not working

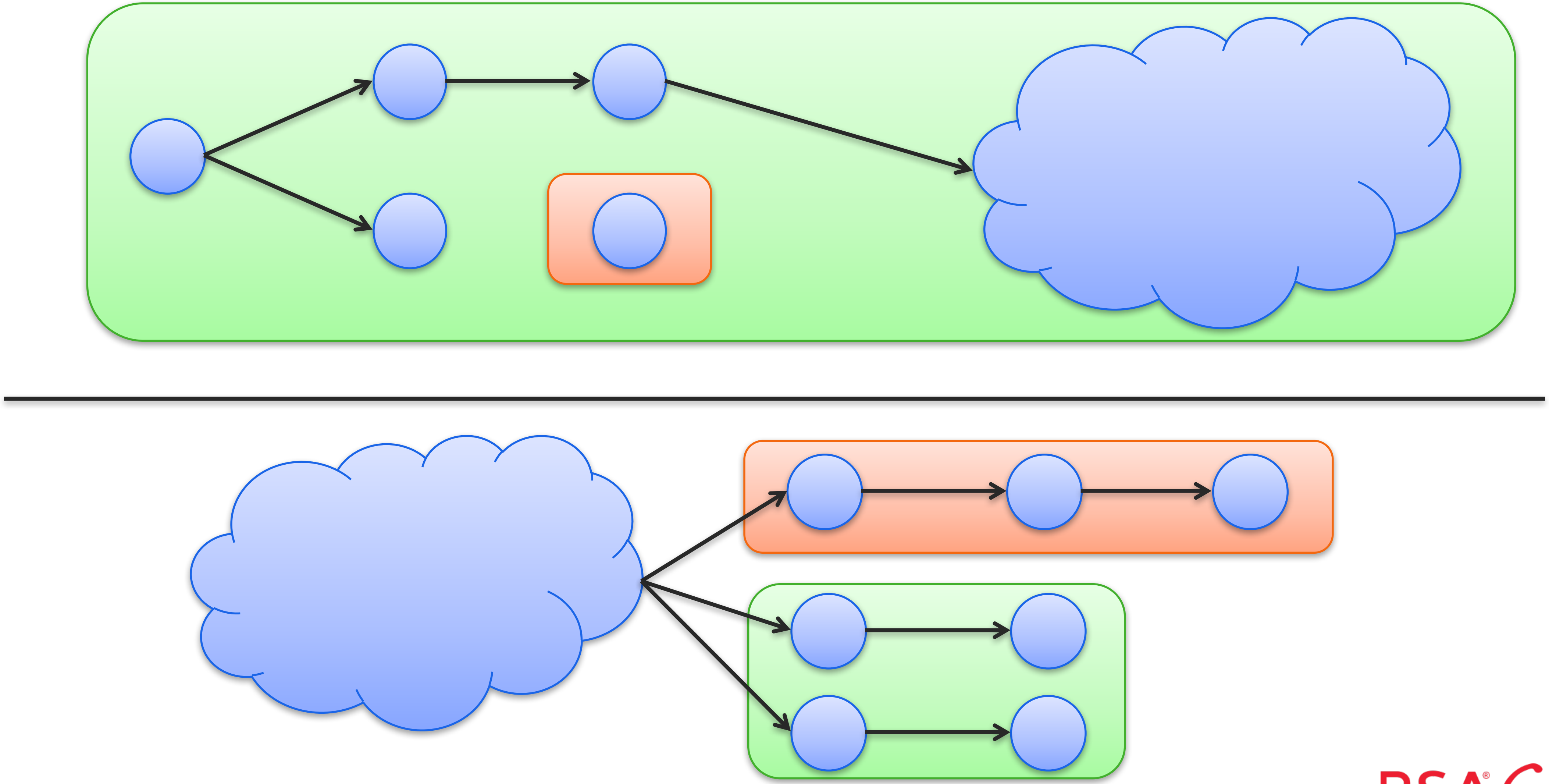
- **Preparation** can't account for everything
- **Detection** can fail
    - Often companies are notified by third parties of a breach
- **Analysis** can take a long time and have a narrow focus
- **Containment, Eradication & Recovery** can be premature if the whole story isn't known
    - Response efforts can be limited to a single incident and miss the larger picture.



Forensics at Scale

- **Response Scope Must Equal Incident Scope**
  One-off host-based analysis is ineffective and wasteful for an enterprise-wide compromise
- **Evidence Will Not Wait**
  Ability to scope, triage, and re-scope at a moment's notice
- **Fluidity in Analysis Techniques**
  Shift rapidly between network logs, packet captures, system logs, and file system artifacts
- **Resources At The Ready**
  Employees, Tools, Policies prepared and ready to react

Standard Triage vs. Mass Triage



Hunting polar bears in a blizzard

- Visibility blinded by vast amount of snow
- Can’t tell good from bad
- Risk of missing your hunt completely
- Can’t focus on the bear, focus on the bear’s effect

Two polar bears fighting in a snow storm

- Notice their fighting stance and graceful movements!

RSA Mass Triage

Methodology and Process

RSA Charge 2016

Forensic Methodologies

Traditional Incident Response

- Get alerted to activity from third-party alert
- Physically retrieve system, create forensic image
- Analyze system for malicious indicator
- Look for activity that may reference other systems
- Expand scope system by system

RSA Mass Triage

- Collect Mass Set of System Profiles
- Analyze for outliers and alerted indicators
- Perform remote forensics on flagged systems
- Analyze Mass Set for New Indicators
- Expand scope network by network

Mass Triage in a Nutshell

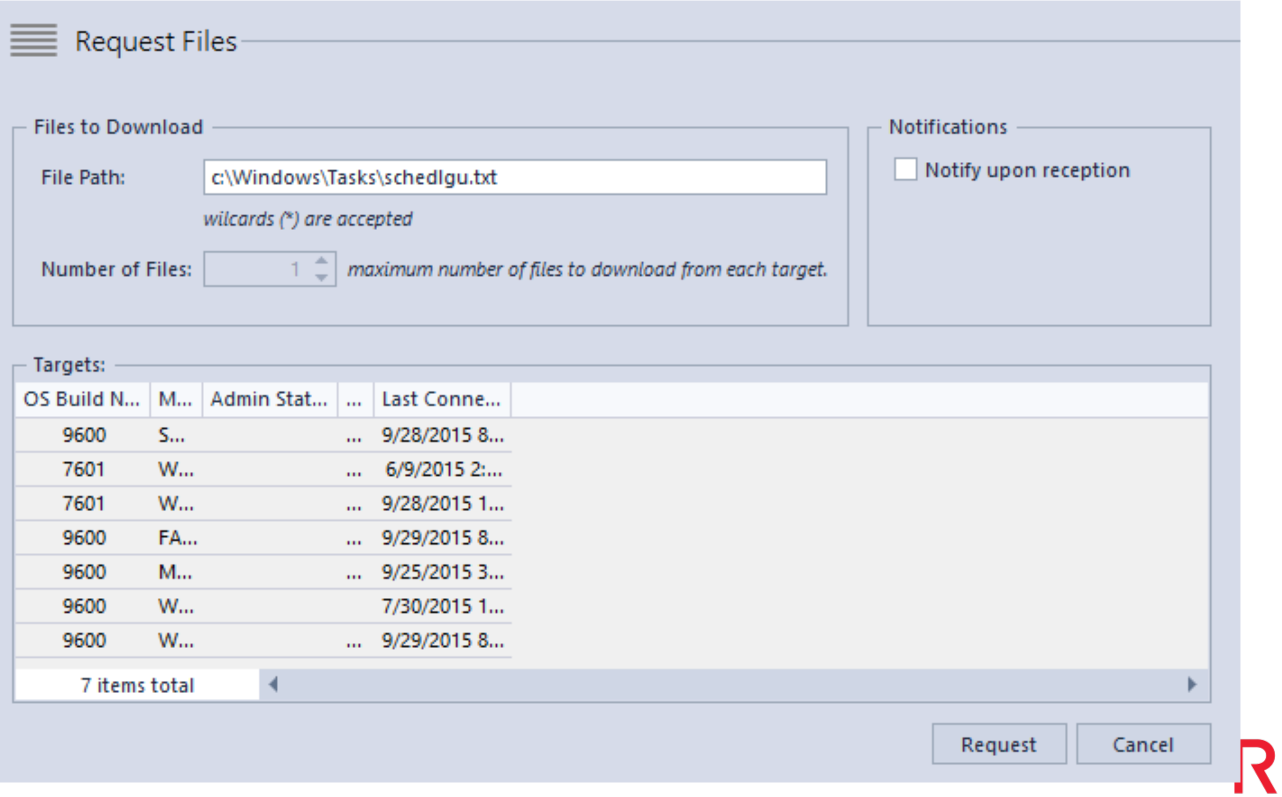
- Selectively download Files using NetWitness Endpoint (NWE)
  - From single or multiple systems
- Tag downloaded files with hostname from NWE database
- Processing Data Ensues
- Interpret the Results

Windows Triage

Windows Triage - Requesting Files

NetWitness Endpoint can request files from systems

- One of the key features to Mass Triage
- Request files that are forensically significant



Request Files

Files to Download

File Path: c:\Windows\Tasks\schedlgu.txt

wilcards (*) are accepted

Number of Files: 1 *maximum number of files to download from each target.*

Notifications

□ Notify upon reception

Targets:

OS Build N...	M...	Admin Stat...	...	Last Conne...
9600	S...		...	9/28/2015 8...
7601	W...		...	6/9/2015 2:...
7601	W...		...	9/28/2015 1...
9600	FA...		...	9/29/2015 8...
9600	M...		...	9/25/2015 3...
9600	W...			7/30/2015 1...
9600	W...		...	9/29/2015 8...

7 items total

Process Execution Tracking

- What files, where from, at what time
- Multiple Windows-based sources
  - Application Compatibility Cache (AppCompatCache / ShimCache)
  - RecentFileCache (Win7 and below)
  - Amcache (Win8 and above)
  - Prefetch
  - Scheduled Tasks (At Jobs)

NetWitness Endpoint Downloaded Files

Files downloaded by NetWitness Endpoint will be placed in the Server\Files directory

- Hints for searching in Windows for downloaded files
  - **Starts with**
    - System.Filename:~<system_
    - System.Filename:~<amcache_
    - System.Filename:~<recentfilecache_
    - System.Filename:~<at_
    - System.Filename:~<schedlgu_
  - **Contains**
    - System.Filename:~=
  - **Ends with**
    - System.Filename:~>

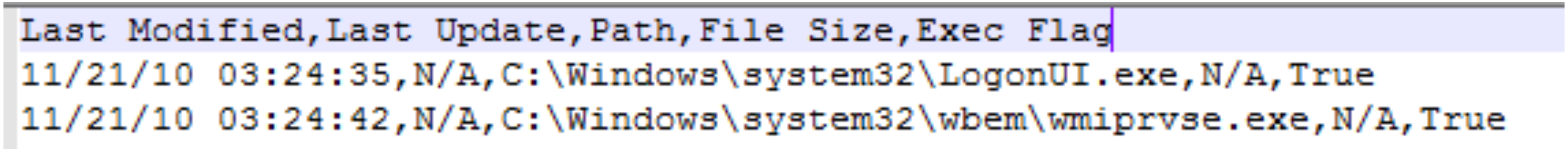
Shimcache

- Shimcache or AppCompatCache
  - Tracks compatibility issues
  - https://dl.mandiant.com/EE/library/Whitepaper_ShimCacheParser.pdf
- File execution logged if file executed via CreateProcess().
  - HKLM\SYSTEM\CurrentControlSet\Control\Session Manager\AppCompactCache\AppCompatCache
- Records file path, size, **last modified**, last exec time (if supported by OS)
- Stored within Windows SYSTEM registry hive

Shim Cache / SYSTEM Hive

- Shimcache Parser
  - https://github.com/mandiant/ShimCacheParser
  - Developed by Mandiant and continually updated

```
c:\tools> python ShimCacheParser.py –v -i SYSTEM -o system.csv
[+] Reading registry hive: SYSTEM_...
[+] Found 64bit Windows 7/2k8-R2 Shim Cache data...
[+] Found 64bit Windows 7/2k8-R2 Shim Cache data...
[+] Writing output to system.csv...
```



```
Last Modified,Last Update,Path,File Size,Exec Flag
11/21/10 03:24:35,N/A,C:\Windows\system32\LogonUI.exe,N/A,True
11/21/10 03:24:42,N/A,C:\Windows\system32\wbem\wmiprvse.exe,N/A,True
```

Shim Cache / SYSTEM Hive – Warnings

- **Hives live in memory**
- Hives written to disk after reboot
- **Requesting Hives from disk may not contain most recent information**
- Many analysts and investigators miss critical information by relying on hives from disk
- Risk rebooting a critical server for updated hive?

Shim Cache / SYSTEM Hive Memory Options

1. **Reboot system then request Registry Hive**

2. **Memory + Volatility**
   - Dump System memory
   - Use volatility to parse memory shimcache
   - https://github.com/volatilityfoundation/volatility/wiki/Command%20Reference#shimcache

```
$ python vol.py -f win7.vmem --profile=Win7SP1x86 shimcache
Volatility Foundation Volatility Framework 2.4
Last Modified                  Path
------------------------------ ----
2009-07-14 01:14:22 UTC+0000   \??\C:\Windows\system32\LogonUI.exe
2009-07-14 01:14:18 UTC+0000   \??\C:\Windows\system32\DllHost.exe
2009-07-14 01:16:03 UTC+0000   \??\C:\Windows\System32\networkexplorer.dll
2009-07-14 01:14:31 UTC+0000   \??\C:\WINDOWS\SYSTEM32\RUNDLL32.EXE
2011-03-22 18:18:16 UTC+0000   \??\C:\Program Files\VMware\VMware Tools\TPAutoConnect.exe
2009-07-14 01:14:25 UTC+0000   \??\C:\Windows\System32\msdtc.exe
2009-07-14 01:14:27 UTC+0000   \??\C:\Windows\system32\net1.exe
2009-07-14 01:14:27 UTC+0000   \??\C:\Windows\System32\net.exe
[snip]
```

Recent File Cache

- ProgramDataUpdater (Application Experience Service) stores data during process creation

- Contains simple path and filename of files executed since ProgramDataUpdater has been run

- C:\Windows\AppCompat\Programs\RecentFilecache.bcf (Win7)

```
11
c:\program files (x86)\mozilla firefox\uninstall\helper.exe
c:\program files (x86)\mozilla firefox\updater.exe
c:\program files (x86)\mozilla maintenance service\maintenanceservice.exe
c:\program files (x86)\mozilla maintenance service\update\updater.exe
c:\windows\psexesvc.exe
c:\windows\system32\malware.exe
c:\programdata\backupsql\malwaremelt.bat
c:\windows\system32\tasklist.exe
c:\windows\NWE_agent.exe
c:\windows\system32\NWEservice.exe
```

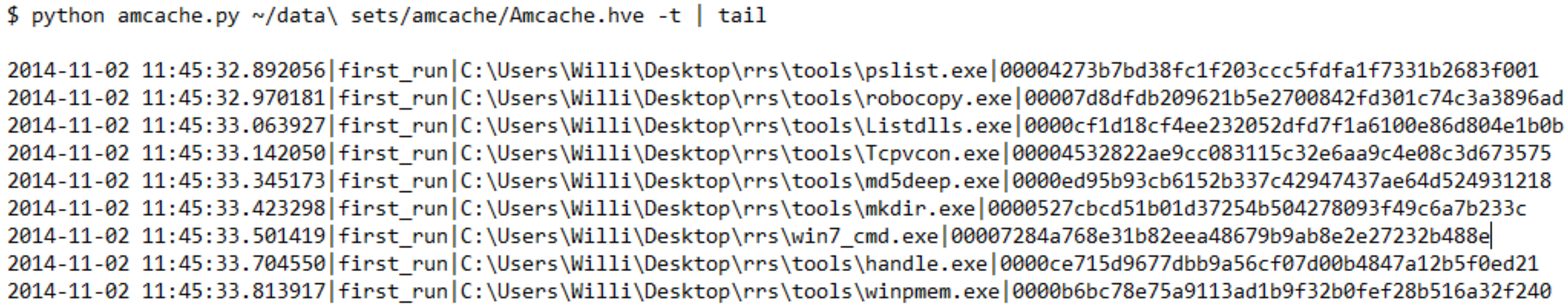
Amcache.hve

Replaced Recent File Cache

- Now in a registry hive format
- C:\Windows\AppCompat\Programs\Amcache.hve
  - (Windows 8+)
- Amcache.hve\Root\File\{Volume GUID}\#######
  - Entry for every executable run, full path information, File's
  - Last Modification Time and Disk volume the executable was run from
  - First Run Time = Last Modification Time of Key
  - SHA1 hash of executable also contained in the key (sometimes)

Amcache.hve

Path	SHA1	Created Time
C:\Windows\PSEXESVC.exe	f1e36e0e34276a5015040780e14b58efd1112b76	9/6/16 03:49:19
C:\Windows\NWE_agent.exe	c277d569265db6062d379eb745577863445946 50	9/6/16 03:49:20
C:\Windows\system32\NWE Service.exe	acc2f9beed1077901b5fbf13b215665b672779a2	9/6/16 09:15:33



```
$ python amcache.py ~/data\ sets/amcache/Amcache.hve -t | tail

2014-11-02 11:45:32.892056|first_run|C:\Users\Willi\Desktop\rrs\tools\pslist.exe|00004273b7bd38fc1f203ccc5fdfa1f7331b2683f001
2014-11-02 11:45:32.970181|first_run|C:\Users\Willi\Desktop\rrs\tools\robocopy.exe|00007d8dfdb209621b5e2700842fd301c74c3a3896ad
2014-11-02 11:45:33.063927|first_run|C:\Users\Willi\Desktop\rrs\tools\Listdlls.exe|0000cf1d18cf4ee232052dfd7f1a6100e86d804e1b0b
2014-11-02 11:45:33.142050|first_run|C:\Users\Willi\Desktop\rrs\tools\Tcpvcon.exe|00004532822ae9cc083115c32e6aa9c4e08c3d673575
2014-11-02 11:45:33.345173|first_run|C:\Users\Willi\Desktop\rrs\tools\md5deep.exe|0000ed95b93cb6152b337c42947437ae64d524931218
2014-11-02 11:45:33.423298|first_run|C:\Users\Willi\Desktop\rrs\tools\mkdir.exe|0000527cbcd51b01d37254b504278093f49c6a7b233c
2014-11-02 11:45:33.501419|first_run|C:\Users\Willi\Desktop\rrs\win7_cmd.exe|00007284a768e31b82eea48679b9ab8e2e27232b488e
2014-11-02 11:45:33.704550|first_run|C:\Users\Willi\Desktop\rrs\tools\handle.exe|0000ce715d9677dbb9a56cf07d00b4847a12b5f0ed21
2014-11-02 11:45:33.813917|first_run|C:\Users\Willi\Desktop\rrs\tools\winpmem.exe|0000b6bc78e75a9113ad1b9f32b0fef28b516a32f240
```

What Next?

Translating Artifacts Into Wins

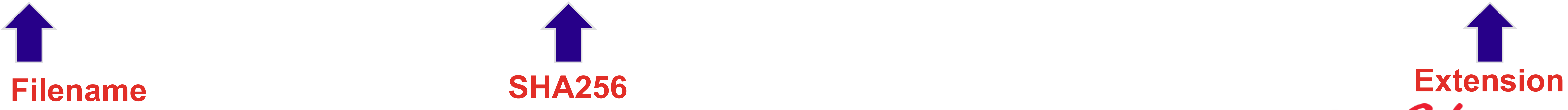
Normalize Data Set

Associate data found with a machine in NWE

- `"\At1","/c c:\temp\a.bat”`
- `"$~$Sys0$.job" (rundll32.exe)`
- `c:\perflogs\svc.exe|ModTime: Wed Mar 12 16:30:57 2014 Z|Executed|LN14`
- `c:\windows\debug\svc.exe|ModTime: Tue Mar 11 09:07:45 2014 Z|Executed|LN14`
- `c:\temp\a.bat`

For each artifact, determine which is the corresponding data file
e.g. `"\At1","/c c:\temp\a.bat”` is from

```
at1_eb41aa5b1bba1b1f42e1e8ba6e454f1a81bb6919a8217b5ce5db4c02e26b0a42_42423nm.job_
```



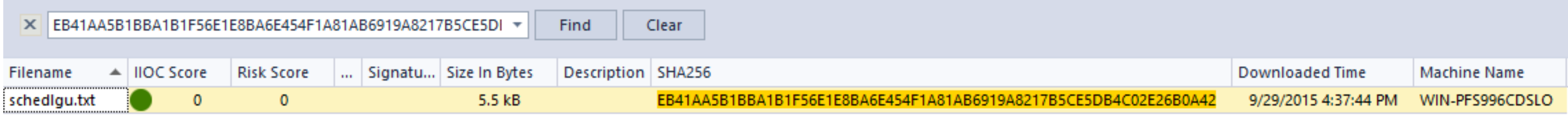
Normalize Data Set

Old Method

```
at1_eb41aa5b1bba1b1f42e1e8ba6e454f1a81bb6919a8217b5ce5db4c02e26b0a42_42423nm.job_
```

On the NWE Downloads Tab

- Add the column File.Download -> Downloaded Time
- Add the column Machine.OperatingSystem -> Machine Name
- Control-F to bring up the find feature
- Copy and paste SHA256 hash from the filenames to get machine



EB41AA5B1BBA1B1F56E1E8BA6E454F1A81AB6919A8217B5CE5DI

Filename	IIOC Score	Risk Score	...	Signatu...	Size In Bytes	Description	SHA256	Downloaded Time	Machine Name
schedlgu.txt	● 0	0			5.5 kB		EB41AA5B1BBA1B1F56E1E8BA6E454F1A81AB6919A8217B5CE5DB4C02E26B0A42	9/29/2015 4:37:44 PM	WIN-PFS996CDSLO

Normalize Data Set

Another Method

- Use NetWitness Endpoint database to determine Machine Name
- Lookup machine name based on downloaded filename:

```
SELECT DISTINCT mn.MachineName FROM
    [dbo].[MachineDownloaded] AS [md] WITH(NOLOCK)
    INNER JOIN [dbo].[FileNames] AS [fn] WITH(NOLOCK) ON ([fn].[PK_FileNames] =
[md].[FK_FileNames__RelativeFileName])
    INNER JOIN [dbo].[machines] AS [mn] WITH(NOLOCK) ON ([mn].[PK_Machines] =
[md].[FK_Machines])
    WHERE fn.filename = “X”
```

Normalize Data Set

Best Method

- Automate the querying of data from NetWitness Endpoint database
- Automatically rename files in a directory to include the Machine Name from which it was downloaded

```
$ python ECAT_Download_File_Renamer.py -h
usage: ECAT_Download_File_Renamer.py [-h] -d <directory> [-u <user>]
                                     [-p <password>] [-s <hostname or IP>]
                                     [-db <database>] [--dsn <dsn>]

optional arguments:
  -d <directory>, --dir <directory>
                        Directory where files are stored
  -u <user>, --user <user>
                        Username for SQL Database. Default: Windows Credentials
  -p <password>, --pass <password>
                        Password for SQL Database. Default: Windows Credentials
  -s <hostname or IP>, --server <hostname or IP>
                        Hostname or IP for SQL Server. Default: localhost
  -db <database>, --database <database>
                        ECAT database
  --dsn <dsn>           SQL DSN
```

Process Data Set

- Convert all gathered data files into a massive set of events
- Parse binary data structures to extract metadata
- Place all metadata into a single CSV
- Typically deal with millions of events at one time

```
E:\RMT>wc -l RMT_Oct2016_results.csv
18152987
```

Data Set Structure

- Normalized based on Mandiant ShimCacheParser output
- Add fields for Hostname, Data Source

```
CNF315,2016-05-04 22:44:52,N/A,C:\Program Files (x86)\Citrix\GoToAssist Remote Support
Customer\888\g2ax_user_customer.exe,N/A,N/A,,shimcache
BG12,2016-05-02 21:31:44,N/A,C:\Windows\TEMP\CR_1DDB0.tmp\setup.exe,N/A,True,,shimcache
BG19,2016-05-03 09:15:18,N/A,C:\Windows\TEMP\CR_0D6B8.tmp\setup.exe,N/A,True,,shimcache
DC245,2016-05-02 20:45:08,N/A,C:\windows\TEMP\CR_65C30.tmp\setup.exe,N/A,True,,shimcache
DC283,2016-05-02 21:52:37,N/A,C:\WINDOWS\TEMP\CR_D2561.tmp\setup.exe,N/A,N/A,,shimcache
DC300,2016-05-03
15:31:12,N/A,\\fs8\packages\FireAMP\WINDOWS_DESKTOPS_US_GROUP_FireAMPSetup.exe,N/A,N/A,,shimcache
DC300,2016-05-03 20:53:23,N/A,C:\Program
Files\WindowsApps\Microsoft.WindowsStore_11602.1.26.0_x64__8wekyb3d8bbwe\Application,N/A,N/A,,shimcache
DC314,2016-05-02 22:02:24,N/A,C:\Program Files
(x86)\Google\Chrome\Application\50.0.2661.94\Installer\setup.exe,N/A,N/A,,shimcache
DC314,2016-05-02 22:02:24,N/A,C:\Windows\TEMP\CR_D314E.tmp\setup.exe,N/A,N/A,,shimcache
DC314,2016-05-02 22:04:34,N/A,c:\users\benchea\appdata\local\temp\skypesetup.exe,0,N/A,,amcache
DC314,2016-05-02 22:04:34,N/A,c:\users\benchea\appdata\local\temp\skypesetup.exe,47405184,N/A,,amcache
SA2,2016-05-02 20:53:10,N/A,C:\Windows\TEMP\CR_76839.tmp\setup.exe,N/A,True,,shimcache
SA2,2016-05-03 06:29:05,N/A,C:\Windows\Temp\SecurityScan_Release.exe,N/A,True,,shimcache
```

Searching Data

- Review and filter millions of events down to a manageable few:
  - Relevant timestamps ($SI Modified time)
  - Suspicious or known-bad filenames
  - Unusual file paths for executables (%temp%, $Recycle.bin, appdata, programdata)
  - Look for atypical file extentions (.txt, .gif, .jpg, .log)

Searching Data

Things to look for

- Reserved names
- Windows folder
- System32 folder
- TEMP / TMP folder
- One-two char filenames
- Filenames with suspicious extensions
- Filenames with .tmp extension
- Files one directory deep
- Self-extracting folders
- Batch filenames
- Keywords related to the incident

Sample keywords

“\\temp\\temp"
scvhost.exe
psexec.exe
"\\pwd.exe"
"\\port.exe"
bulk-ps
output.bat
mkatz.bat
"\\tar.exe"
wce.exe
whoami.exe

Filtering Data

- Regular Expressions to hunt for unusual indicators

- Files run from web server folders:
  - `‘(tomcat|inetpub|wwwroot|webapps|clientaccess)’`
- Files run directly from Windows folder:
  - `‘(:\\windows\\.{1,15},)’`
- Files of small size (batch or PowerShell scripts):
  - `‘\,([0-9]{2})\,N\/A’, ‘\,([0-9]{3})\,N\/A’`
- Files with unusual extensions:
  - `‘(\.bin,|\.dat,|\.log,|\.gif,|\.txt,|\.jpg,|\.rar,|\.sql,)’`
- Files running one-folder deep from volume root:
  - `‘(:\\[a-zA-Z0-9]{1,12}\\[a-zA-Z0-9]*\....,)’`

Filtering Data

- One Character File Names:

```
   6 c:\tdm-gcc-64_4.9.2\work\a.exe
   1 c:\accbk\army\g.bat
   1 c:\accbk\agusta\y.bat
   1 e:\move_qual\x.exe
   1 c:\users\jsmith\appdata\local\microsoft\windows\temporary
internet files\content.ie5\4unu162n\..exe
   1 c:\_inbox\boxer text editor\b.exe
   1 sysvol\users\k2service\downloads\..exe
   1 c:\g77\a.exe
   1 c:\acc pc\agusta\g.bat
   1 c:\qmerge\release_8.214n\live\x.exe
   1 sysvol\program files (x86)\k2 for sharepoint 2013\z.bat
```

Filtering Data

- One Deep Folders:

```
1 \??\e:\agent\procexp.exe
1 c:\apps\run.bat
1 c:\batch\upload.bat
1 c:\dangerous\splashappis.exe
1 c:\downloads\mtben1721su.exe
1 c:\g77\a.exe
1 c:\tools\dbgview.exe
1 c:\xxxxxx\usbmake.exe
1 g:\av\combofix.exe
1 g:\av\keyfinderinstaller.exe
1 z:\50320t00\flash.exe
```

RSA Mass Triage (RMT)

Automating the Drudgery of Triage

RSA Mass Triage

Custom Scripts to automate much of these tasks

- Rename NWE files to provide context
- Parse Amcache, RFC, and ShimCache for indicators
- Perform Frequency Analysis of results
- Provide results in easy to format, CSV

Demonstration of Use

During the live conference, this slide will contain video links and updated examples of analysis through RSA Mass Triage

NWE Mass Triage Wins

Leveraging Endpoints for Hunting and Forensics

Large Scale MFT Scanning

- Conduct a Full Scan of the suspect machine(s)
- Download the $MFT
  - Look for other tools and exfil
  - Unleash Timetology

Longboltsecurity @Longboltsec · Mar 22

Timetology. It's a thing.

- Write Yara signatures for the malware found
- Rinse and Repeat to find additional compromises

Windows Event Logs

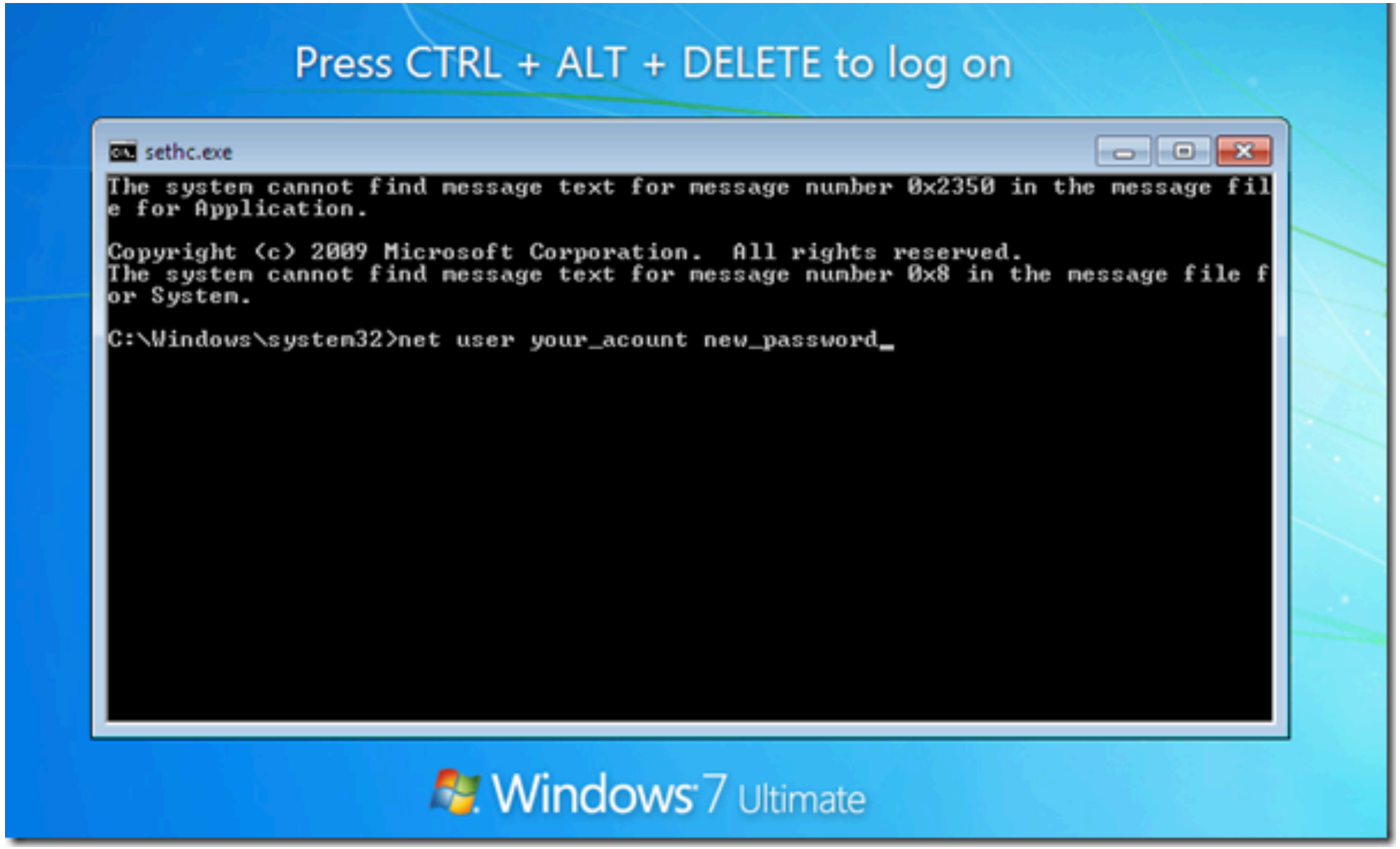
- Request for C:\Windows\System32\winevt\Logs\*.evtx
- Use File Renamer
- Load results into Plaso / Log2Timeline
- Perform bulk analysis on:
  - All Security Events
  - All RDP events
  - etc

Scheduled Jobs

- Download job forensic artifacts from systems
  - *.job
  - Schedlgu.txt
- GREP for the file extensions of executable files (.exe, .dll, .cmd, .ps1, .vbs, .vbe, .bat, etc.)
- Reviews results for interesting attributes
  - Filename
  - File Path
  - Suspicious administrative commands
  - UNC paths or Network access (potential lateral movement)

Sticky Keys Exploit

- Mass download of C:\Windows\System32\sethc.exe
- Perform quick analysis of all results for any unusual versions



```
The system cannot find message text for message number 0x2350 in the message fil
e for Application.

Copyright (c) 2009 Microsoft Corporation.  All rights reserved.
The system cannot find message text for message number 0x8  in the message file f
or System.

C:\Windows\system32>net user your_acount new_password_
```

Thank You

Brian Baskin – brian.baskin@rsa.com

Steve Brzozowski – steve.brzozowski@rsa.com

Please Complete Session Evaluation

#RSACharge

RSA Charge 2016

RSA® Charge 2016

#RSACharge