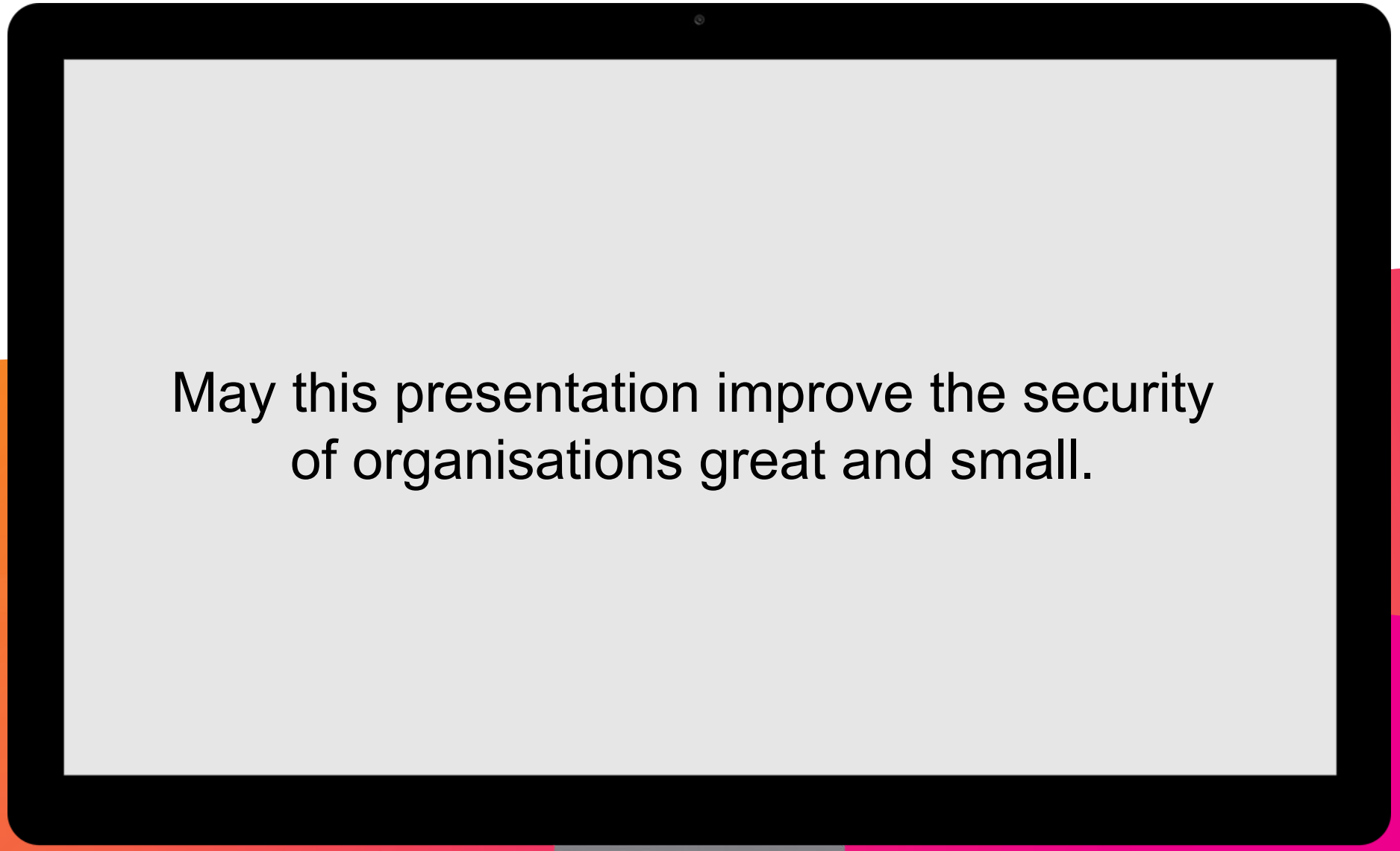
ATT&CK™ing Linux using SPL

Doug Brown
Senior Information Security Analyst | Red Hat

Forward-Looking Statements

During the course of this presentation, we may make forward-looking statements regarding future events or plans of the company. We caution you that such statements reflect our current expectations and estimates based on factors currently known to us and that actual events or results may differ materially. The forward-looking statements made in the this presentation are being made as of the time and date of its live presentation. If reviewed after its live presentation, it may not contain current or accurate information. We do not assume any obligation to update any forward-looking statements made herein.

In addition, any information about our roadmap outlines our general product direction and is subject to change at any time without notice. It is for informational purposes only, and shall not be incorporated into any contract or other commitment. Splunk undertakes no obligation either to develop the features or functionalities described or to include any such feature or functionality in a future release.



Speaker Background

Author of more than a dozen Splunkbase apps

2016 Developer Revolution Award Winner

SplunkTrustee since 2016

Masters degree - *Network Behaviour Analysis Using Formal Methods*

Contributor to ES roadmap

Previous .conf Sessions:

- 2017: *Art of Detection Using Enterprise Security*
- 2018: *Detection Technique Deep Dive*

Red Hat Operational Security

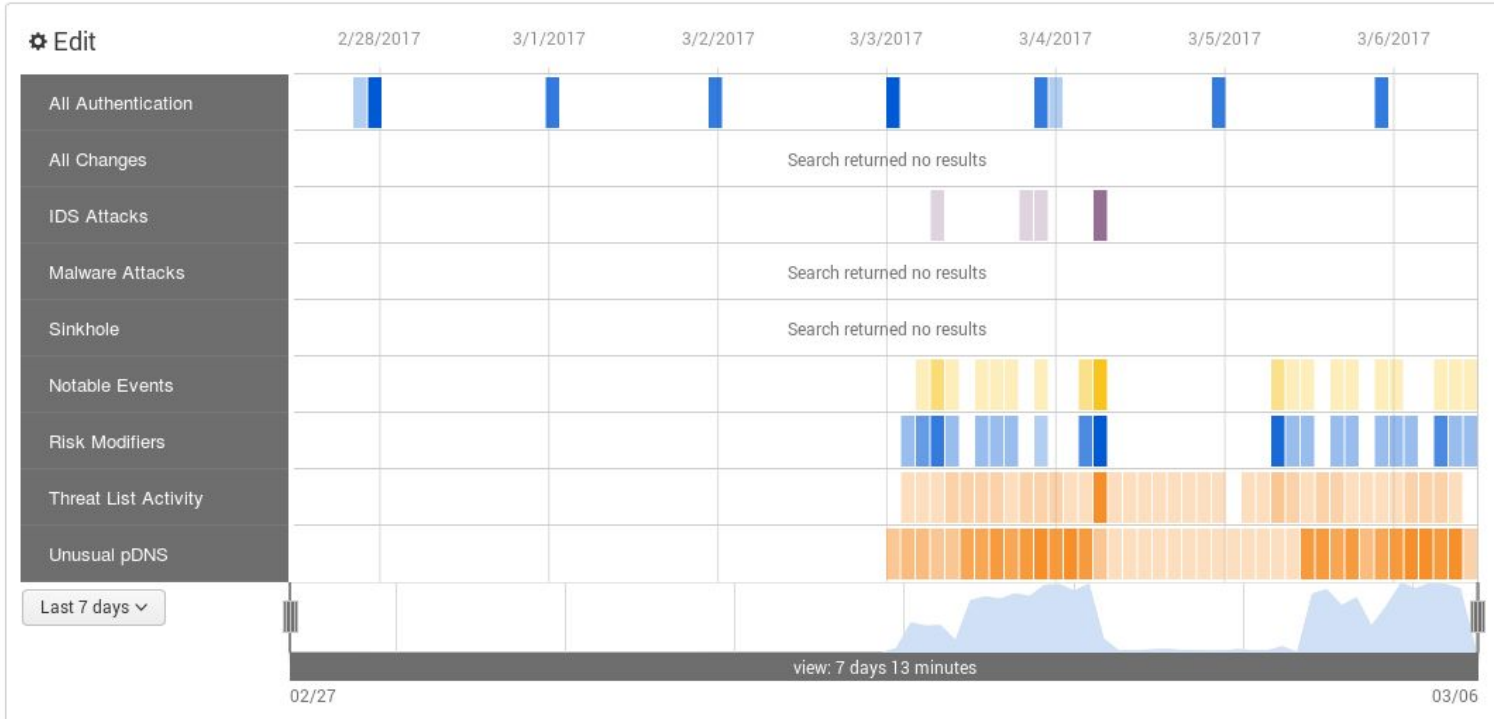
Leading Open Source Vendor

Splunk Customer Since 2012

Relatively Small Global Team

Multi TB Daily Ingestion

Splunk Enterprise Security™



The Experiment

In a parallel universe where the year is 2015…

MITRE ATT&CK™ Matrix Coverage

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery
9 items	10 items	14 items	7 items	24 items	9 items	13 items
Drive-by Compromise	Command-Line Interface	.bash_profile and .bashrc	Exploitation for Privilege Escalation	Binary Padding	Bash History	Account Discovery
Exploit Public-Facing Application	Exploitation for Client Execution	Bootkit	Process Injection	Clear Command History	Brute Force	Browser Bookmark Discovery
Hardware Additions	Graphical User Interface	Browser Extensions	Setuid and Setgid	Compile After Delivery	Credential Dumping	File and Directory Discovery
Spearphishing Attachment	Local Job Scheduling	Create Account	Sudo	Disabling Security Tools	Credentials in Files	Network Service Scanning
Spearphishing Link	Scripting	Hidden Files and Directories	Sudo Caching	Execution Guardrails	Exploitation for Credential Access	Network Sniffing
Spearphishing via Service	Source	Kernel Modules and Extensions	Valid Accounts	Exploitation for Defense Evasion	Input Capture	Password Policy Discovery
Supply Chain Compromise	Space after Filename	Local Job Scheduling	Web Shell	File Deletion	Network Sniffing	Permission Groups Discovery
Trusted Relationship	Third-party Software	Port Knocking		File Permissions Modification	Private Keys	Process Discovery
Valid Accounts	Trap	Redundant Access		Hidden Files and Directories	Two-Factor Authentication Interception	Remote System Discovery
	User Execution	Setuid and Setgid		HISTCONTROL		System Information Discovery
		Systemd Service		Indicator Removal from Tools		System Network Configuration Discovery
		Trap		Indicator Removal on Host		System Network
		Valid Accounts		Install Root Certificate		
				Masquerading		

ATT&CK™

Shellshock (CVE-2014-6271)

T1190 Exploit Public-Facing Application

```
curl -H "user-agent: () { :; }; echo; echo; /bin/bash -c 'echo \"<html><body>deface
site</body></html>\" > /var/www/html/index.html'" http://localhost/cgi-bin/shellshock
```

Dirty COW (CVE-2016-5195)

T1068 Exploitation for Privilege Escalation

Experiment Preparation

Weaponisation:

- Custom payload created that modifies /etc/sudoers
- Dirty COW exploit complied with custom payload
- Stage 2 shell script created to establish persistence
- Exploit and stage 2 encrypted with `openssl enc`
- Encrypted exploit and stage 2 uploaded to Internet

Setup target server:

- Unpatched RHEL 7.0 machine commissioned, “Basic Web Server” installed with port 80 open on firewall and the experiment’s auditd rules configured
- Simple “uptime” bash cgi script put in /var/www/cgi-bin/ with execute permissions

/etc/audit/rules.d/experiment.rules

Audit rules that provide greater visibility into pertinent system calls

-w /boot -p wa -k boot_changes
-w /etc -p wa -k etc_changes
-w /usr/bin -p wa -k usr_bin_changes
-w /usr/sbin -p wa -k usr_sbin_changes
-w /usr/include -p wa -k usr_include_changes
-w /usr/lib -p wa -k usr_lib_changes
-w /usr/lib64 -p wa -k usr_lib64_changes
-w /usr/local -p wa -k usr_local_changes
-w /var/spool/at -p wa -k at_changes
-w /var/spool/cron -p wa -k cron_changes
-a exit,always -F arch=b64 -F euid=0 -S execve -k root_exec64
-a exit,always -F arch=b32 -F euid=0 -S execve -k root_exec32
-a exit,always -F filetype=file -F obj_type=ssh_home_t -F perm=rwa -k ssh_home_access

N.B. These rules are for the purposes of this research and not a configuration recommendation from Red Hat.

Splunk

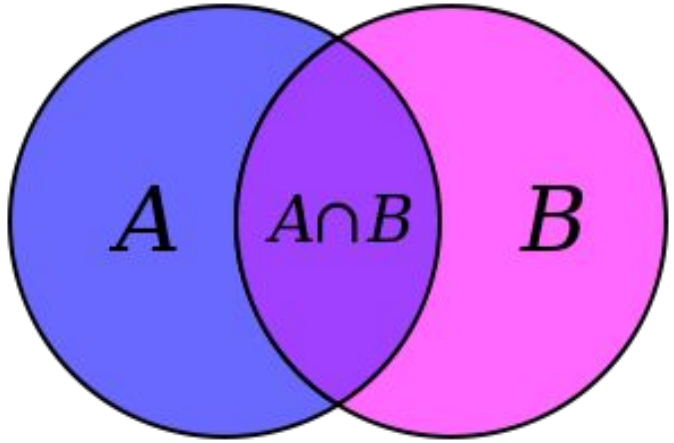
Apps used in this session

Linux Auditd v3.1+

- https://splunkbase.splunk.com/app/2642/

Set Operations Technology Add-On v1.1+

- https://splunkbase.splunk.com/app/3516/



Initial Access

splunk> .conf19

T1190 Exploit Public-Facing Application

Stage 1 produced the following events with SELinux enforcing

type=AVC msg=audit(1561636025.897:**863**): avc: denied { **execute** } for pid=31621 **comm="bash"** name="update" dev="dm-1" ino=1474358 scontext=system_u:system_r:**httpd_sys_script_t**:s0 tcontext=system_u:object_r:httpd_sys_rw_content_t:s0 tclass=file

type=CWD msg=audit(1561636025.897:**863**): **cwd="/var/www/cgi-bin"**

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery

T1190 Exploit Public-Facing Application

Stage 2 produced the following events with SELinux enforcing

type=AVC msg=audit(**1561636182**.329:905): avc: denied { **setuid** } for pid=4054 **comm="sudo"** capability=7 scontext=system_u:system_r:**httpd_sys_script_t**:s0 tcontext=system_u:system_r:httpd_sys_script_t:s0 **tclass=capability**

type=ANOM_ABEND msg=audit(**1561636182**.480:908): auid=4294967295 uid=48 gid=48 ses=4294967295 subj=system_u:system_r:**httpd_sys_script_t**:s0 pid=4050 **comm="uptime.cgi"** reason="memory violation" sig=11

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery

T1190 Exploit Public-Facing Application

Detect Crash Related To Policy Violation

```
1.  earliest=-15m eventtype=auditd_events ANOM_ABEND OR AVC
2.      [ search earliest=-15m eventtype=auditd_events ANOM_ABEND
3.          | rex field=unix_time "(?<search>^\d[9])"
4.          | table host search ]
5.  | transaction host scontext_domain maxpause=1s
6.  | where mvcount(type)>1 AND searchmatch("ANOM_ABEND")
```

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery

setenforce 0

do not try this at work

Privilege Escalation

T1166 Setuid and Setgid

Stage 2 uses Dirty COW vulnerability against a setuid binary to get root

type=PATH msg=audit(1561636398.625:**949**): item=0 name="/usr/bin/passwd" inode=33743805 dev=fd:01 **mode=0104755** ouid=0 ogid=0 rdev=00:00 obj=system_u:object_r:passwd_exec_t:s0 objtype=NORMAL

type=AVC msg=audit(1561636398.625:**949**): avc: denied { **execmem** } for pid=7377 comm="passwd" scontext=system_u:system_r:**httpd_sys_script_t**:s0 tcontext=system_u:system_r:httpd_sys_script_t:s0 tclass=process

Initial Access	Execution	Persistence	**Privilege Escalation**	Defense Evasion	Credential Access	Discovery

T1166 Setuid and Setgid

Stage 2 uses Dirty COW vulnerability against a setuid binary to get root

type=SYSCALL msg=audit(1561636398.625:949): arch=c000003e syscall=59 per=400000 success=yes exit=0 a0=19209a0 a1=191fd00 a2=191fb90 a3=7fff360b9770 items=1 ppid=7372 pid=7377 auid=4294967295 **uid=48 gid=48 euid=0** suid=0 fsuid=0 **egid=48** sgid=48 fsgid=48 tty=(none) ses=4294967295 comm="passwd" exe="/usr/bin/passwd" subj=system_u:system_r:**httpd_sys_script_t**:s0 key=(null)

type=AVC msg=audit(1561636398.626:950): avc: denied { **setuid** } for pid=7377 comm="passwd" capability=7 scontext=system_u:system_r:**httpd_sys_script_t**:s0 tcontext=system_u:system_r:httpd_sys_script_t:s0 **tclass=capability**

Initial Access	Execution	Persistence	**Privilege Escalation**	Defense Evasion	Credential Access	Discovery

T1166 Setuid and Setgid

Detect use of new capability by SELinux domain

1. | tstats summariesonly=t values(Auditd.perm) AS perm FROM

   datamodel=Auditd WHERE (nodename=Auditd.AVC Auditd.tclass=capability)

   BY _time, host, Auditd.scontext_domain span=1h]
2. | `drop_dm_object_name("Auditd")`
3. | mvexpand perm
4. | streamstats count by host, scontext_domain, perm
5. | where count==1 AND _time>relative_time(now(),"-1h")

T1169 Sudo

sudoers file modified

type=AVC msg=audit(1561636398.630:**951**): avc: denied { **dac_override** } for pid=7377 comm="sh" capability=1 scontext=system_u:system_r:**httpd_sys_script_t**:s0 tcontext=system_u:system_r:httpd_sys_script_t:s0 **tclass=capability**

type=AVC msg=audit(1561636398.630:**951**): avc: denied { **append** } for pid=7377 comm="sh" **name="sudoers"** dev="dm-1" ino=34316115 scontext=system_u:system_r:**httpd_sys_script_t**:s0 tcontext=system_u:object_r:**etc_t**:s0 **tclass=file**

type=SYSCALL msg=audit(1561636398.630:**951**): <snip> exe="/usr/bin/bash" subj=system_u:system_r:**httpd_sys_script_t:s0 key="etc_changes"**

Initial Access	Execution	Persistence	**Privilege Escalation**	Defense Evasion	Credential Access	Discovery

T1169 Sudo

Apache runs stage 3 as root using sudo

type=USER_START msg=audit(1561636398.707:963): pid=7382 uid=0 auid=4294967295 ses=4294967295 subj=system_u:system_r:**httpd_sys_script_t**:s0 msg='op=PAM:session_open **acct="root"** exe="/usr/bin/sudo" hostname=? addr=? terminal=? res=success'

type=SYSCALL msg=audit(1561636398.702:960): <snip> **uid=48 gid=48 euid=0 suid=0 fsuid=0** egid=48 sgid=48 fsgid=48 tty=(none) ses=4294967295 comm="sudo" exe="/usr/bin/sudo" subj=system_u:system_r:**httpd_sys_script_t**:s0 key=(null)

type=USER_CMD msg=audit(1561636398.702:961): pid=7382 uid=48 auid=4294967295 ses=4294967295 subj=system_u:system_r:**httpd_sys_script_t**:s0 msg='cwd="/var/www/cgi-bin" cmd="bash" terminal=? res=success'

Initial Access	Execution	Persistence	**Privilege Escalation**	Defense Evasion	Credential Access	Discovery

T1169 Sudo

Detect SELinux domains that don't normally use sudo

1. | tstats summariesonly=t values(Auditd.scontext_domain) AS scontext_domain FROM datamodel=Auditd WHERE (nodename=Auditd Auditd.type=USER_CMD) BY _time, host span=1h
2. | `drop_dm_object_name("Auditd")`
3. | mvexpand scontext_domain
4. | streamstats count by scontext_domain
5. | where count==1 AND time>relative_time(now(),"-1h")

T1168 Exploitation for Privilege Escalation

Detect unusual user/group use by SELinux domain

1. [|inputlookup auditd_indices] [|inputlookup auditd_sourcetypes] SYSCALL uid!=0
2. | where uid!=euid OR gid!=egid
3. | eval tuple=uid+":"+euid+":"+gid+":"+egid
4. | stats earliest(_time) as _time, values(host) as host by scontext_domain, tuple
5. | where _time>relative_time(now(),"-1h") AND mvcount(host)==1

Initial Access	Execution	Persistence	**Privilege Escalation**	Defense Evasion	Credential Access	Discovery

T1178 Valid Accounts

Detect SELinux domains that don't normally "login"

1. | tstats summariesonly=t values(Auditd.scontext_domain) AS scontext_domain FROM datamodel=Auditd WHERE (nodename=Auditd Auditd.type=USER_START) BY _time, host span=1h
2. | `drop_dm_object_name("Auditd")`
3. | mvexpand scontext_domain
4. | streamstats count by scontext_domain
5. | where count==1 AND _time>relative_time(now(),"-1h")

Defense Evasion

splunk> .conf19

T1054/1070 Indicator Blocking/Removal on Host

Detect New Distinct SELinux AVC Tuple

1. | tstats summariesonly=t count FROM datamodel=Auditd

   WHERE nodename=Auditd.AVC BY _time, host, Auditd.scontext_domain, Auditd.tclass,

   Auditd.perm, Auditd.tcontext_type span=1d
2. | `drop_dm_object_name("Auditd")`
3. | **distinctstream** by=scontext_domain tclass perm tcontext_type
4. | where mvcount(distinctfields)>1 AND _time>relative_time(now(), "-1d")

Discovery

T1083 File and Directory Discovery

Detect New Auditd Rules Being Triggered by an SELinux domain

1. | tstats summariesonly=t values(Auditd.key) as keys from datamodel=Auditd

   where Auditd.key=* by _time, host, Auditd.scontext_domain span=1h

2. | `drop_dm_object_name("Auditd")`

3. **| streamstats current=f values(keys) as previous_keys by host, scontext_domain**

4. **| setop op=relation keys previous_keys**

5. | where (relation=="fully disjoint" OR relation=="superset" AND _time>relative_time(now(),"-1h")

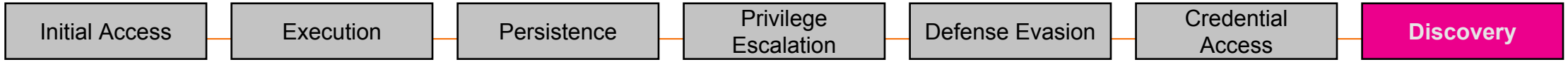
D Brown Note:
New *Technique E*

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery

T1083 File and Directory Discovery

Detect New Types Being Accessed by SELinux Domain

1. | tstats summariesonly=t values(Auditd.tcontext_type) as tcontext_types from datamodel=Auditd where (Auditd.key=* Auditd.tcontext_type=*) by _time, host, Auditd.scontext_domain span=1h
2. | `drop_dm_object_name("Auditd")`
3. | streamstats current=f values(tcontext_type) as previous_tcontext_types by host, scontext_domain
4. **| setop op=difference tcontext_types previous_tcontext_types**
5. **| where mvcount(difference)>1** AND _time>relative_time(now(),"-1h")
6. | eval risk_score=mvcount(difference)*10



Multiple Techniques

splunk> .conf19

Sequencing Small Potential Indicators

Using Auditd app's ATT&CK™ event types

1. | tstats summariesonly=t values(Auditd.mitre_attack) AS mitre_attack
   FROM datamodel=Auditd WHERE (nodename=Auditd Auditd.mitre_attack=*)
   BY _time, host span=1h
2. | streamstats current=f values(mitre_attack) as previous_mitre_attack by host
3. | setop op=difference mitre_attack previous_mitre_attack
4. | where mvcount(difference)>1 AND _time>relative_time(now(),"-4h")
5. | eval risk_score=60+mvcount(difference)*10

N.B. Patching is a known false-positive.

Key Takeaways

1. Vulnerabilities Exist – patch
2. Use Protection – setenforce 1
3. Get Insurance – auditd rules

Q&A

splunk> .conf19

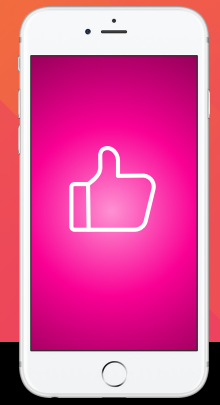
.conf19

splunk>

Thank You !

Go to the .conf19 mobile app to

RATE THIS SESSION



Other Related Sessions

.conf 2019

SEC1556

- Building Behavioral Detections: Cross-Correlating Suspicious Activity with the MITRE ATT&CK™ Framework

SEC1803

- Modernize and Mature Your SOC with Risk-Based Alerting

SEC1538

- Getting Started with Risk-Based Alerting and MITRE

SEC1908

- Tales From a Threat Team: Lessons and Strategies for Succeeding with a Risk-Based Approach

Bonus: Our Risk-Based Incident Detection

Aggregate risk, even if risk_object_type is different

1. index=risk
2. | eval risk_objects=mvdedup(mvappend(orig_host,src_ip,src_host,dest_ip,dest_host,src_user,user))
3. | eval object = risk_objects
4. | mvexpand object
5. | stats values(risk_objects) as risk_objects, dc(risk_object_type) as dc_risk_object_type,
   sum(risk_score) as sum, dc(source) as dc_correlation_search, values(source)
   as correlation_searches by object
6. | where (dc_correlation_search>1 AND sum>=80)
7. | dedup risk_objects