Splunk for Healthcare

Jake McAleer
Senior Manager, IT Security

Mike McGinnis
Security Engineer

athenahealth.com
Our Focus Today Will Be Security

Think about how you can use these general concepts to query whatever data you care about
Some Healthcare Definitions
Who are the players?

- **Single-payer healthcare** is a healthcare system financed by taxes that covers the costs of essential healthcare for all residents, with costs covered by **single public system** (hence 'single-payer'). Alternatively, a **multi-payer system** is one in which **private individuals or their employers** buy health insurance or healthcare services from private or public providers.


- **Patients** - individuals who **receive** medical care from providers

- **Providers** - Institutions that **provide** care to patients, charge payers for that care, and buy products from vendors

- **Payers** - Institutions that **pay** providers for healthcare services, which includes insurance carriers, private employers, the government, and also individuals.

How Healthcare Can Benefit From Splunk
Besides easy searching...

▶ Providers
  • Meaningful use analysis
  • Overall patient health
  • Metrics by doctor
  • Success rates by procedure
  • Metrics on collections

▶ Payers
  • Common errors made by submitting providers
  • Trends on claims data
    • Breakdowns by submitter, quarter of the year, time of day, etc.
  • Metrics on payouts
Founded in 1997, provides cloud based services such as network-enabled EHR, practice management and population health services

Connecting more than 72,000 providers and health systems nationwide

5,000+ employees

We were voted Forbes “Most Innovative Growth Company” and a Deloitte “Fast 500 Company” in 2014 and have earned numerous employer awards

Three InfoSec Towers
  Risk, SIRT (Security Incident Response Team), and ITSec

Sad fact of the day: We process over 2 million faxes a day

We are not a payer or provider, we’re a weird mix of everything
The goal of Splunk is to take raw data and turn it into actionable context

Easily consume data from various sources (syslog, text files, threat feeds, etc.)

Splunk Enterprise Security (ES) for the SIRT

Crafted alerts and reporting to look for high value targets

If we see a bad pattern within our network, we can quickly alert and take action

We can tweak and tailor alerts and reports over time

Well supported: Very few issues and when we call, they answer

Official Splunk and 3rd party apps:
Value to athenahealth

Why we like Splunk

- Immediate visibility (near real-time data)
- Virtually any data, even mainframe and other legacy infrastructure
- Less “alert fatigue” via very detailed and deep control
- Ability to dig in and investigate, correlate (it’s not a proprietary black hole)
- Better team efficiency - Reduce confusion and wasted time over where to look for information
- Granular permissioning
- Intuitive, easy-to-use, and responsive UI
- Designed to scale, runs on both Windows and Linux servers
- Easy win on audits: Regulators and Auditors love Splunk
Some examples of how we use Splunk within our custom SaaS application:

- Help prove/disprove suspected compromised accounts
- Help prove/disprove account abuse by malicious practice employees
- Accessing of particularly sensitive information
- Controlled prescription abuse
- Employee activity within the application
- Accessing sensitive records (such as celebrities)
- Database activity
What Splunk is not...
Magic, Silver Bullet, One solution, Set it and forget it

► Splunk needs to understand the data you’re throwing at it
  • Vendors change log formats constantly
  • Proprietary in-house apps and logging follow no format

► Splunk has little to no pre-canned alerts by default
  • You need to pick and chose what you want
  • There are additional apps and licenses you can buy, but they’re not magical either

► Splunk needs TLC
  • Just like all infrastructure, it needs attention and curation
  • This includes the hosted Splunk offering

► Splunk doesn’t magically get the logs, you send them to Splunk
  • Build processes need to include the syslog/Universal forwarder configuration steps
Splunk attempts to classify logs as they come in:

- Many common formats are recognized: Router logs, Windows logs, Linux logs, etc.
- If it's not something Splunk recognizes, it makes a best guess effort.
- You can "teach" Splunk by giving it input formatting information (great for proprietary logs).
- Some vendors (BlueCoat) change their log formats often; Splunk tries to keep up.

Why do we care? Consistent formatting allows for correlation.

http://docs.splunk.com/Documentation/CIM/4.8.0/User/UsetheCIMtonormalizeOSSECdata
Correlations
The real reason we love logs

▶ Without logs that can be correlated, it’s nearly impossible to relate events across the different platforms

▶ In order to correlate, it must be in CIM format!

▶ Example: Infection reported via anti-malware agent to Splunk
  • Alert notifies SIRT
    • End User Windows Logs: Who was on the computer? What files are on there? What was touched?
    • Web Proxy Logs: Did the machine reach out to known C&C servers?
    • File Share Logs: Did the machine read/exfiltrate or alter (ransomware) files on network shares?
How does it work under the hood?

Trust is important: How do we ensure we don’t lose them?
Over ten “power users”, many regular IT staff users

- Anti-malware, anti-virus, system data, system logs, VPN/firewall/router logs, O365, various other unstructured data
- >500GB/day license
- Example: 964,201,274 events/day
- Goal: Retain up to two years of searchable data
- Retention varies by the type and value of the data
- Windows logs are the most verbose
- There are ways to ignore verbose data you don’t need
General Workflow of Splunk

How data gets into Splunk

- Network Device
- Universal Forwarder Agent
- SyslogNG Server
- Linux/Unix/Mainframe Server
- Universal Forwarder Agent
- Windows Server
- Indexers
- Search Heads
- Security Staff
General Workflow of Splunk
Universal Forwarder: How it really works

- Security Staff
- Indexers
- Search Heads
- Windows Server
- Universal Forwarder Agent

Primary path
Alternate path
General Workflow of Splunk

Universal Forwarder: Worst Case

Agent will queue

Security Staff

Indexers

Search Heads

Windows Server
General Workflow of Splunk

syslog: Worst Case

Network Device

Linux/Unix/Mainframe Server

SyslogNG Server

SyslogNG via Universal Forwarder will queue

Security Staff

Indexers

Search Heads

Universal Forwarder Agent

Agent
Hardware
No need for fancy SAN, NAS, etc. Keep it simple!

Commodity Linux servers that our systems team runs for us

▶ Indexers
  • The most important thing for the indexers is IOPS (fast hard drives)
  • Server(s) with SSDs in RAID5 configuration
  • You can do spinning disks in RAID10, but it’s much slower
  • We have a mix of SSD and rusting disk: New data is written to SSDs and after a few days it’s moved to slower and cheaper HDDs since most people are searching only recent events

▶ Search Heads
  • Server(s) with minimal hard drive requirements and lots of CPU and RAM

Splunk offers a hosted solution in AWS
The obvious...bad things
• Malware, IDS alerts, Data exfiltration

Splunk licensing issues
• More likely to catch “real time” as people are ingesting data
• Daily isn’t enough and runs at midnight; we monitor 4 times a day

Servers not reporting into Splunk
• Network issues and reporting services dying (Carbon Black bug)
• Server maintenance and deprecation

General IT problems
• Active Directory Account lockouts and RSA token lockouts (goes to Support)
• Service Account Lockouts

Alerts must be clear and actionable or they’re a waste!
https://www.pagerduty.com/blog/lets-talk-about-alert-fatigue/

Action by severity:
• E-mail
• Slack (webhooks)
• Page via PagerDuty, OpsGenie, etc.
• Notify our NOC, who calls us day or night
Example Of An Actual Alert

Improve your awareness and visibility

Splunk Alert: Save - No Carbon Black logs for 30min

The alert condition for 'Save - No Carbon Black logs for 30min' was triggered.

Alert:   Save - No Carbon Black logs for 30min

View results in Splunk

<table>
<thead>
<tr>
<th>host</th>
<th>lastTime</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>'17 04:44:57</td>
</tr>
</tbody>
</table>

If you believe you've received this email in error, please see your Splunk administrator.

splunk > the engine for machine data
Deployment and Configuration Management
Keeping the environment consistently configured

- Audit requirement to put client on all new server builds
  - This is a good thing, it means buy in from the business
- Universal forwarder install is automated using Puppet and PowerShell
- Install uses deployment server to pull down configuration settings
- Server classes are broken down by OS
  - We don’t get much fancier than that, keep it simple
- Distributed Management Console (DMC) is used to monitor system health monitoring
  - Replaces other apps like Splunk on Splunk (SOS)
In the AWS VPC, we have multiple forwarders

• Typical universal forwarder install on each EC2 instance
• We chose to have the individual server universal forwarders report to a central set of heavy forwarders for compression and transform reasons

3rd party apps:

• API: Install Splunk app that brings in logs like Amazon Cloud Trail, O365, CASB, etc. into Splunk
• Syslog: 3rd party syslog servers (like Cylance) send data to a publicly facing forwarder in our DMZ with special ACLs, which then populates the indexers

Other services like Azure are on our roadmap and are consumed in a very similar manner
Lookups
More information is better

A lookup is a CSV file used to populate more information based upon a value you look up

- Example: AD has a field with a site code, which a lookup table could add a value that tells you the office location

<table>
<thead>
<tr>
<th>Name</th>
<th>Site Code</th>
<th>Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jake McAleer</td>
<td>39</td>
<td>Watertown, MA</td>
</tr>
</tbody>
</table>

A great way to simplify searching and adding more context for users

Search populate lookups which populate second searches

- Example: Find service account list -> Report of service accounts locked out
- The lookup tables are automated so it’s always up to date

Example of how we use it:

- Service account lookup to pull in description and who owns the account
DNS Logs via Stream
Network tap to suck up DNS data out of band

- We have over 1TB/day of DNS logs on very busy servers, so traditional universal forwarders were out of the question
  - Too much data
  - Too much load on the servers
  - We don’t want to be even possibly associated with jeopardizing production

- An out of band network tap sending data to a heavy forwarder running the stream app, which acts as a tcpdump type collector

- From there, we suck down the DNS logs we want into Splunk with the ability to filter out logs we don’t need and we do it all without impacting production

- “Estimate mode” helps you determine how much license it will use

- DNS logs are awesome, they help with all sorts of incident investigation
Get people hooked!

Make it searchable by your users for diagnostics and they’ll love it!

- Linux server logs
- Windows server and domain controller logs (including account lockouts)
- Virtual Server Infrastructure (ESXi, OpenStack, etc.)
- DHCP and DNS logs
- SSO logs (PingFed, Okta, Azure, etc.)
- In-house developed application logs, SFTP server logs
- VPN, firewall, and router logs
- Two-factor, web proxy, and MDM logs
- Endpoint logs (anti-virus, anti-malware, Bit9, Carbon Black, etc.)
- AWS, Azure, and other IaaS/SaaS providers
General Tips and Tricks For Splunk

 Lessons we learned the hard way

- Permissioning in cloud management platforms like AWS is very granular and took some back and forth to get just right so we could scrape the data we needed

- CI/CD pipeline overwriting our changes
  - Puppet, etc. was accidentally overwriting the work we were doing to test out changes in AWS

- Stream helps with out-of-band collection (we use it for DNS)

- Test before upgrading....many software updates break CIM format
  - Make sure your fields are populating correctly post upgrade

- Clean up old apps you don’t use; they suck up resources

- Use the main index (udp/514) as a catch-all to find misconfigured apps
  - Our main index should always be empty
  - Same concept in syslog-ng so we always capture it, but we’re aware where in the ingestion process it’s misconfigured and needs rework
Interesting Work To Check Out
Shout outs to others

▶ BSidesCharm 2017 T201 Weaponizing Splunk Using Blue Teams for Evil by Ryan Hays
  • [https://www.youtube.com/watch?v=QmpoWwG0IPs](https://www.youtube.com/watch?v=QmpoWwG0IPs)
  • [https://github.com/TBGSecurity/weaponize_splunk](https://github.com/TBGSecurity/weaponize_splunk)

▶ JA3 TLS Client Fingerprinting
  • [https://engineering.salesforce.com/open-sourcing-ja3-92c9e53c3c41](https://engineering.salesforce.com/open-sourcing-ja3-92c9e53c3c41)
  • Referenced yesterday at “Hunting the Known Unknowns: Finding Evil With SSL Traffic”

▶ Setting up Splunk to use SSO
  • [https://www.splunk.com/blog/2013/03/28/splunkweb-sso-samlv2.html](https://www.splunk.com/blog/2013/03/28/splunkweb-sso-samlv2.html)