Global Information Assurance Certification Paper

Intrusion detection through traffic analysis from the endpoint using Splunk Stream

GIAC (GCIA) Gold Certification

Author: Etrik Eddy, eddyet@hotmail.com
Advisor: Christopher Walker, CISSP
Accepted: 5/1/2017

Abstract

With technologies such as software-defined wide area networking (SD-WAN) and cloud operations, the traditional scheme of intrusion detection and packet capture at the network perimeter is quickly becoming less viable as a model for network intrusion detection. One alternative is to dynamically collect network traffic at the endpoint using the Splunk Stream and then using Splunk to analyze the traffic for indicators of compromise. This method allows for network-level detection on large, disparate networks which don't have consolidated egress points for traffic.

1. Introduction

"Behind it rose the ancient castle, its towers roofless, and its massive walls crumbling away, but telling us proudly of its own might and strength, as when, seven hundred years ago, it rang with the clash of arms, or resounded with the noise of feasting and revelry (Dickens, 1836)." Castles have for centuries been the symbol of protection and security. Their strong outer layer allowed for a peaceful, safe interior. Cyber security professionals have adopted this icon to describe the common technique of emplacing security controls near the outer edge of the network in order to protect it and detect when malicious activity was targeting them (Leuprecht, Skillicorn, & Tait, 2016, p. 1). These outer walls have been slowly eroding for a long time as technologies and social interaction methodologies change. A recent development has added another catalyst to the process which is driving a systemic shift towards borderless networking. Software-Defined Wide Area Networking (SD-WAN) is a growing topic of discussion across the network vendor space. With some vendors claiming almost 50% reduction in Total Cost of Ownership ("Cisco Meraki | SD-WAN," n.d.) there is not only technical advantages but also a business driver for change. For security professionals, this shift in the core design of the network likely means a re-engineering of the security architecture. One possible addition to the security toolset to address this change is accomplishing network-based intrusion detection at the endpoint using technologies such as Stream and Splunk.

Network-based Intrusion Detection Systems (NIDS) traditionally consists of three main components: the sensor, the ruleset, and an analysis module. The sensor ingests the network traffic. The ruleset is a dataset of indicators of malicious traffic. The analysis module looks at the network traffic ingested by the sensor, compares it to the ruleset and generates alerts based on matches. While these three parts are implemented in a myriad of ways by a variety of vendors, they all are emplaced in generally the same locations on the network: at the egress and between network segments (Stallings & Brown, 2015). In a software-defined WAN, however, there is no single, defined egress or path between

network segments. The network morphs with the data flow to best accommodate the transmission of data often utilizing a myriad of egress points ("Cisco Meraki | SD-WAN," n.d.). In a company that is utilizing a cloud-based provider for infrastructure, the network traffic for their servers may not traverse the company network at all. Instrumenting the endpoints such as these servers with Splunk and Stream helps security analysts maintain the level of visibility into network traffic that they had with traditional NIDS in the traditional network model and also allows for one central interface and alerting mechanism for network based detections. To demonstrate this, we will walk through how to set up Stream as the sensor for network traffic, a Splunk Application to ingest freely available rulesets containing network indicators of compromise, and Splunk to analyze the traffic for those indicators and generate alerts.

1.1. Lab Setup

To demonstrate one technique to accomplish network intrusion detection at the endpoint a virtual lab is set up. One server acts as the central logging and correlation point using Splunk and one endpoint simulates end user activity and is the actual sensor collecting data using Stream. Please see the Appendix A for details on system setup and configurations. It is assumed at this point that Splunk is up and running on the server and receiving the default logging from the Splunk Universal Forwarder (the formal name for the Splunk agent) on the endpoint.

2. Data Acquisition (The Sensor)

Collecting Network Traffic data from the endpoint has several benefits. It allows for actionable comparison to an indicator of compromise without having to force all traffic through a single egress. Also, in large environments that may have DHCP and multiple NAT translations, it removes the sometimes-tedious work of figuring out which endpoint is generating the traffic by the security analysts since the IP in the alert may have been changed either through its path via Network Address Translation (NAT) and/or through time with DHCP issuing the listed address to another machine. Last, using Splunk Stream to do the acquisition allows for easily deployable, highly configurable

collection using Splunk's deployment server capabilities (or any other configuration management systems) to deploy the Stream collection configuration.

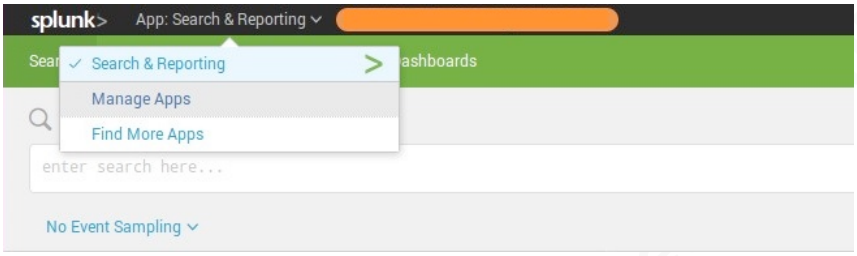
Software Defined WAN is allowing businesses to capitalize on lower cost network connectivity while still maintaining the reliability of more costly business-class circuits. For cyber security professionals, this poses a problem at requires multiple egress points to maintain reliability. By collecting network-level data at the endpoint the path and egress are no longer a concern for the scalability and coverage of network-based intrusion detection.

Traffic monitored at the edge of the network often does not contain the IP address of the true internal host generating the traffic. NAT is commonly used by network engineers. Sometimes, a path can traverse several NAT gateways prior to egressing the network. Analysts have to follow the path back through not only the NIDS alert but though NAT logs to get the actual host. If DHCP is in place, those logs need to be correlated as well to ensure the host with the IP being reviewed is the host that had the IP at the time of the alert. By capturing the traffic data at the end point both of these scenarios are moot as the data is tagged with the originating host information.

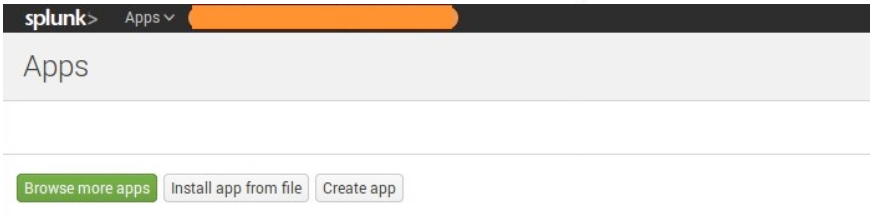
Splunk Stream can include different configurations for adjusting the level of granularity and time of data retention per host based on factors of the administrator's choosing. Servers with a higher criticality to the business, for example, could log more verbose information and retain the data for longer periods that servers or workstations that are less crucial to business operations.

2.1. Setting up Splunk Stream for Data Acquisition

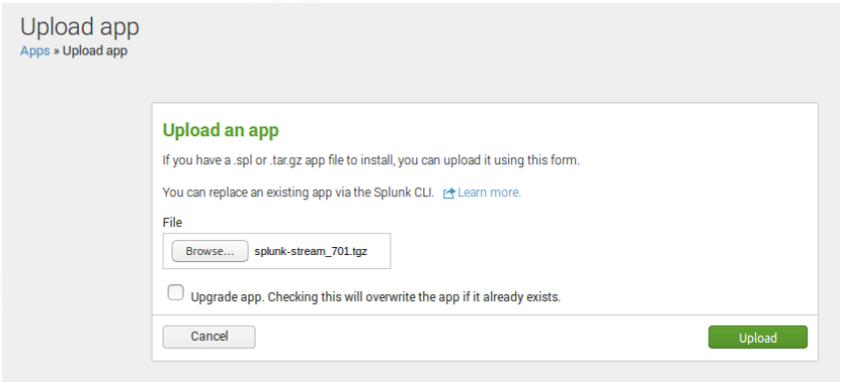
Splunk Stream components must be installed on both the server and the monitored host. Install on the server is similar to the process described previously. Download the Splunk Stream app from: https://splunkbase.splunk.com/app/1809/ On the Splunk server, log in and navigate to "Apps" – "Manage Apps". Select "Install App from file".

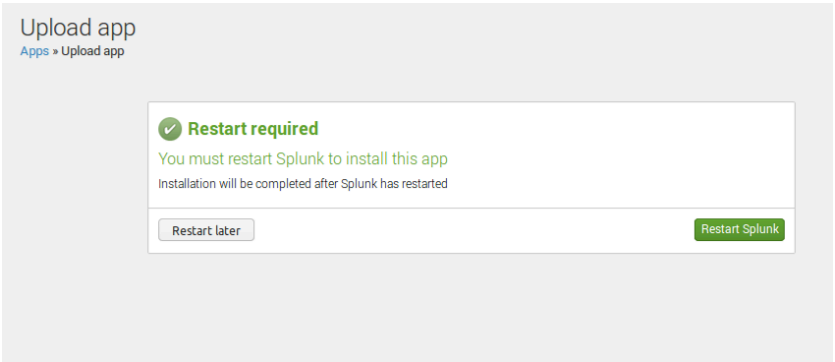


Select “Browse” and then navigate to where the file is saved.



Select “Upload”. When the upload has completed, select “Restart Spunk” and then “OK”.





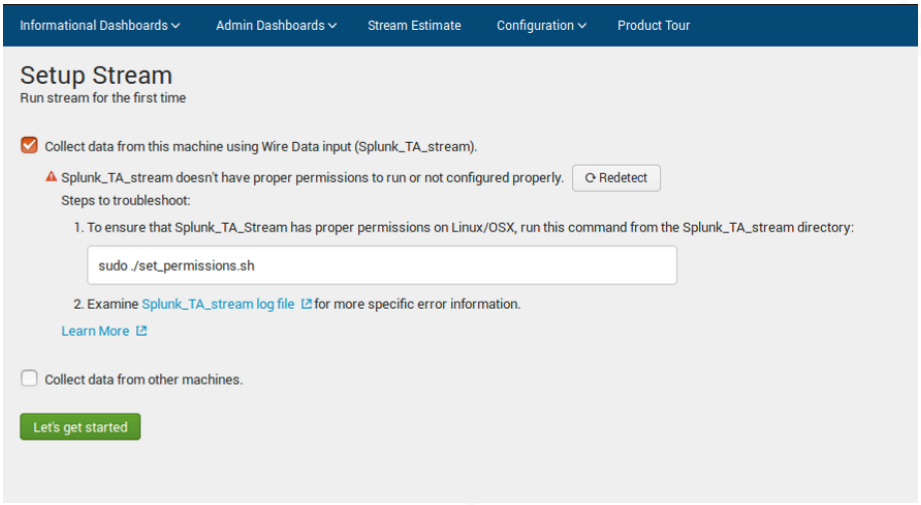
Once Splunk is back up, SSH into the server and navigate to:

/$SPLUNK_HOME/etc/apps/Splunk_TA_Stream/

Run: `$ sudo ./set_permissions.sh`

On the web page select “Redetect”. If you still get a permissions error, reboot the server.

Etrik Eddy, eddyet@hotmail.com



Copy the Splunk_TA_stream folder located at $SPLUNK_HOME/etc/deployment-apps/ to $SPLUNK_HOME/etc/apps/ on the host you wish to monitor. Edit $SPLUNK_HOME/etc/apps/ Splunk_TA_stream/local/inputs.conf and ensure that the URL listed is accurate for the lab. In a lab setup, the server name may need to be replaced with the IP address of the Splunk Server.

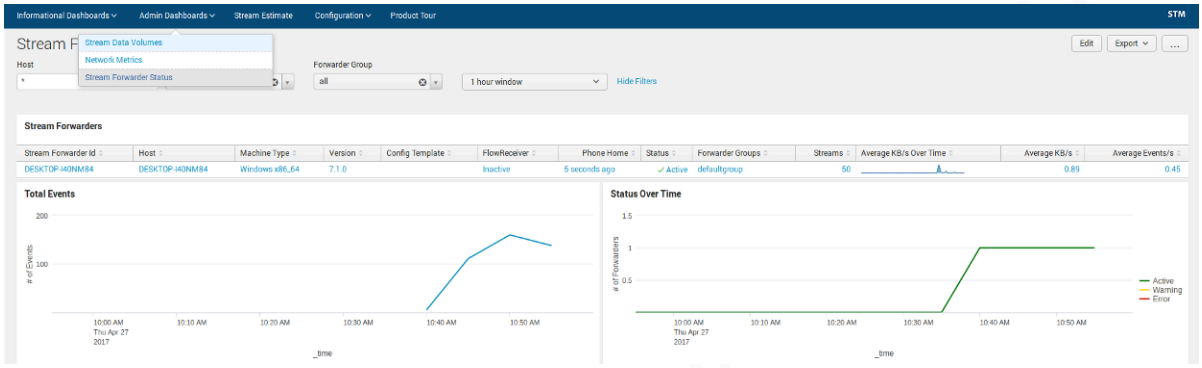
```
[streamfwd://streamfwd]
splunk_stream_app_location = http://SplunkServer:8000/en-us/custom/splunk_app_stream/
stream_forwarder_id =
disabled = 0
```

Also on the host, it is recommended that the below file be modified to allow greater throughput from Splunk Universal Forwarder:
$SPLUNK_HOME/etc/apps/SplunkUniversalForwarder/local/limits.conf
Modify the contents to reflect:

```
[thruput]
MaxKBps = 0
```

Restart Splunk on the host. Back on the server, navigate to “Admin Dashboard” and then “Stream Forwarder Status”. The hostname of the monitored host should show up and some activity should be seen in the graphs.

Etrik Eddy, eddyet@hotmail.com



The default data set is now being ingested and can be used to trigger alerts against threat intelligence indicators.

3. Threat Intelligence (The Ruleset)

Threat Intelligence, as it relates to network traffic, is the sharing of data samples that indicate malicious activity is occurring. Most of these provide samples break down into three types of network indicators: IP, Domain, and URL. Network threat indicators, by their nature, have varying levels of viability, reliability, and lifespan. IP indicators while highly viable, and if properly vetted, reliable, are only actionable for a very short period of time. IP infrastructure can change quickly without notice or impact to malicious actors. Domain indicators are somewhat viable and reliable as they may cast too wide a net containing legitimate traffic from other portions of the named domain, but their life span can be longer than IP indicators without causing additional false positives. One advantage to Domain indicators is that multiple URL indicators may be caught in a Domain, even random-generated URLs. URL indicators are highly viable since they usually are only used for the indicated malware, but the lifespan is not as long as a Domain indicators. All three type have their place and uses as indicators of compromise. To make these indicators usable three basic steps need to happen: The data needs to be downloaded; the data needs to be standardized; The data needs to be ingested into the IDS system.

To get threat intelligence, several cyber security threat intelligence lists are freely available from a variety of public websites for download. There are also several paid services that provide threat intelligence as either a downloadable feed for general

consumption or as a subscription for a specific device's use (such as when a vendor has a threat feed their product can use to enhance its detection of malicious traffic). The paid-for services are often more refined and vetted, but the free data is viable for use and better than no intelligence.

Downloaded data needs to be standardized which allows for easier comparison across data sets to make it as usable as possible. Splunk uses the Common Information Model (CIM) to standardize field names for ingested data. This allows the application of downloaded threat data across all data sets that are ingested in a CIM-compliant format (including the Splunk Stream app data that will be set up later). Once the threat intelligence data is standardized, Splunk can be set up to ingest it and compare it against the Stream data to see if there are any matches.

3.1. Setting up TA_Threat_Intel for Threat Intelligence

Based on the work of Adrian Daucourt (Daucourt, 2015), an example script was developed which ingests a sample source of each indicator type just discussed (see Appendix B). The script can easily be modified to integrate other free and/or paid data sources. Splunk then needs to be set up to run the script and ingest the contents of the text files. The manual setup process has already been completed and integrated into the Threat Intelligence Gathering application sample (TA_Threat_Intel) found at https://drive.google.com/drive/folders/0B_JoQF76rO6wR29NaHVEc1V1dkk?usp=sharing . To install the app, simply unzip it to your Splunk apps folder ($SPLUNK_HOME/etc/apps) and restart Splunk. Once installed, this app sets up an index called threat, schedules the download of the raw indicators every 12 hours, manipulates the indicators into a standardized format, de-duplicates them, and replaces the lookup files from the previous day. The resulting lookup tables include:

1. lookup_threatip
2. lookup_threatdomain
3. lookup_threaturl

They are ready to compare against source data to alert on suspicious activity.

4. Correlating the Indicators Against Traffic (The Analysis)

Analyzing volumes of network traffic data against network threat indicators allows for a cybersecurity analyst to focus their valuable time and provides a starting point for threat hunting. There are a plethora of methodologies for the follow-on actions once an alert is received (which are beyond the scope of this paper), but they are all predicated on the receipt of a trigger to initiate action. Splunk allows for the creation of alerts that can be sent through several methods to achieve this end. The most common of these triggers is an email alert. Once threat intelligence indicators and host traffic are ingesting into Splunk, the ultimate objective of this example - alerting when traffic matches the indicators - is the focus. The processes to setup and validate the alerts are similar across indicator types: get a sample indicator; generate test traffic from our monitored host; search for traffic matching indicators; save the search as an alert. The DNS indicators will be used to demonstrate this procedure. The result will be an actionable alert directly from the affected host.

** A word of caution: if there is DNS monitoring/alerting on the network being used, an alert may be triggered in its security monitoring. Always verify that permission to conduct these types of tests is obtained prior to execution. **

4.1. Setting it up Analysis of data against indicators

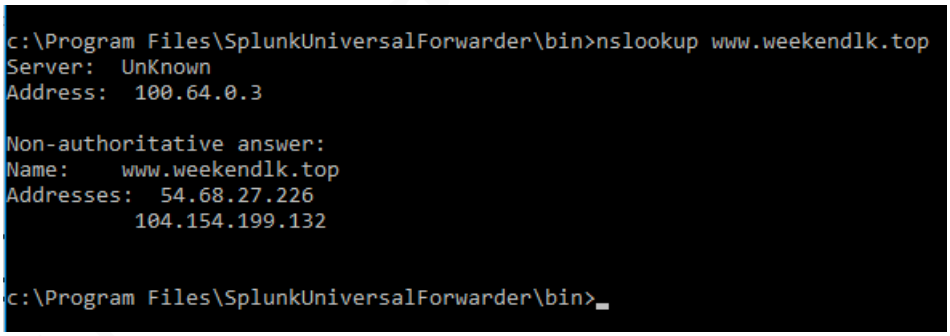
To start, list out the indicators to get a valid sample to verify the functionality of alerts with. To do this, search Splunk for:

```
|inputlookup lookup_threatdomain
```

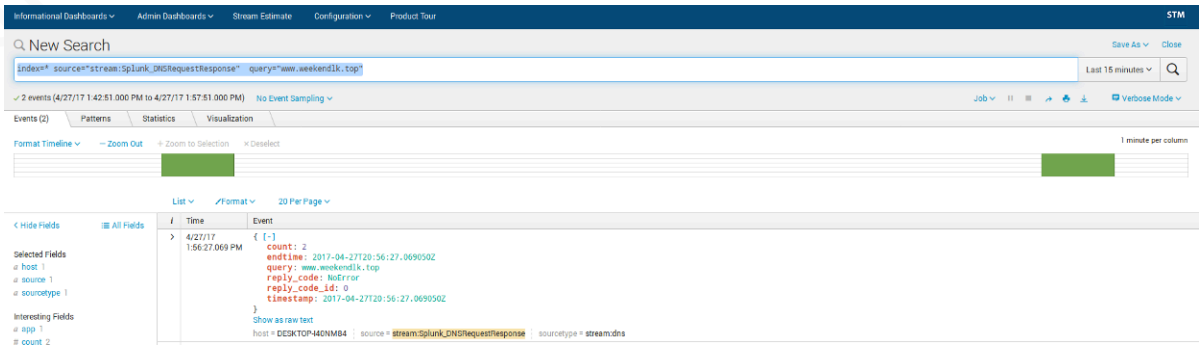
Select a few of the listed domains. Use one of the available web domain information sites check to see if they are active. One example is http://ping.eu/nslookup/ which gives not only the DNS records but also does a ping to see if the server is available.

To generate test traffic: once a domain with a DNS entry is found, on the monitored host, open a command prompt and initiate a nslookup for the domain chosen as a test by running the below command replacing www.domain.com with the chosen domain:

```
C:\>nslookup www.domain.com
```



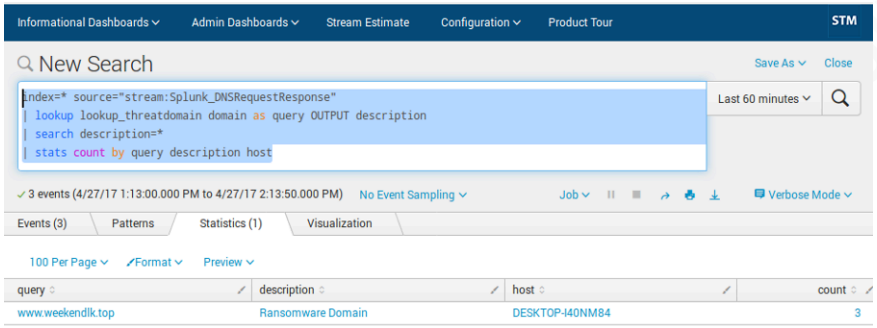
A moment later, the DNS query should show in Splunk:



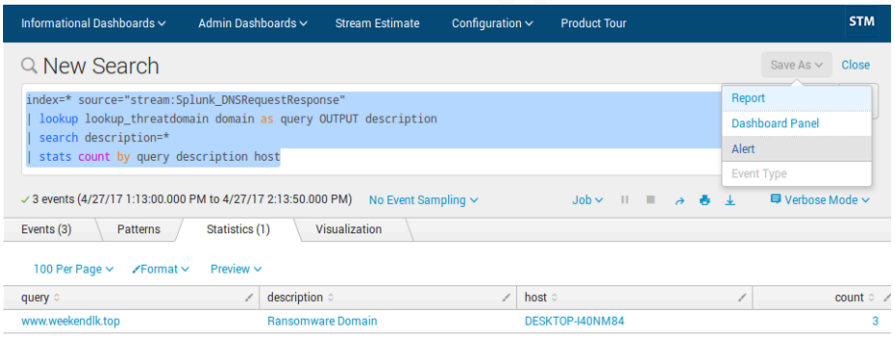
To validate the input will match the traffic, run the below query:

```
index=* source="stream:Splunk_DNSRequestResponse"
| lookup lookup_threatdomain domain as query OUTPUT description
| search description=*
| stats count by query description host
```

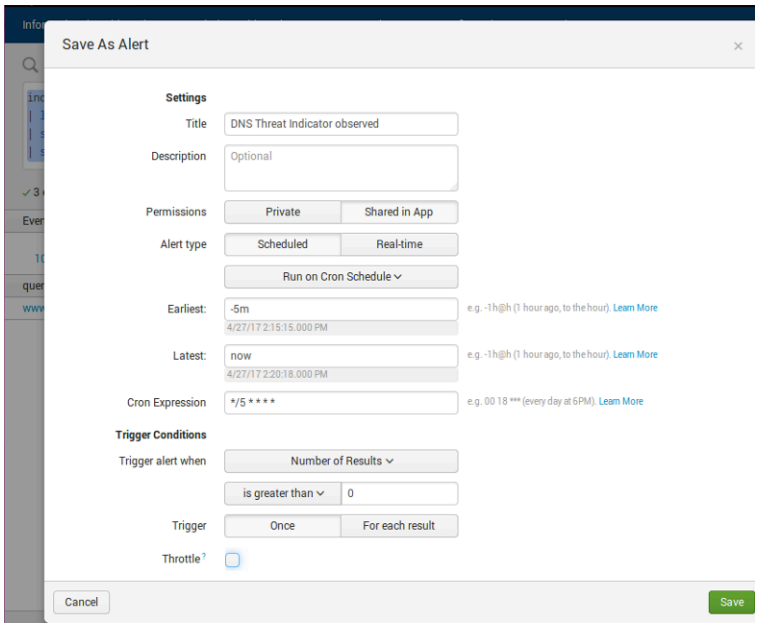
Etrik Eddy, eddyet@hotmail.com



To save this as an alert, select "Save As" – Alert

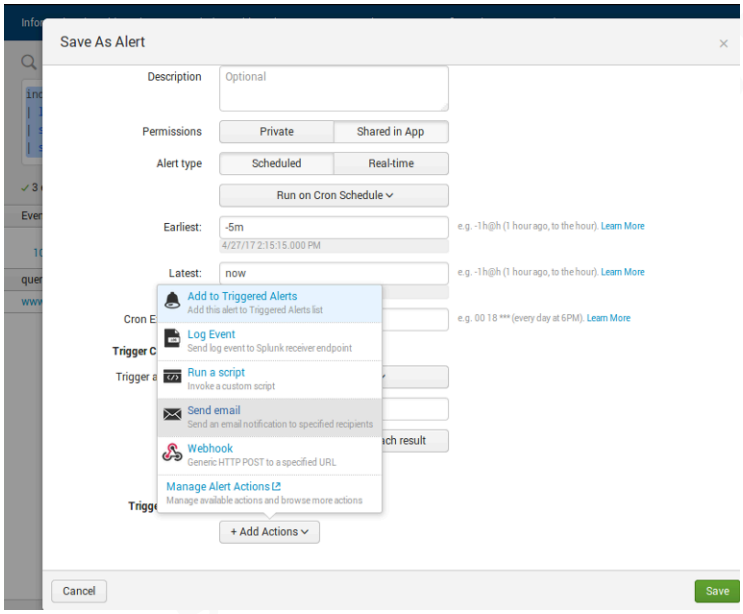


Enter a title. Select "Shared in App". Select "Run on Cron Schedule". Enter Earliest "-5m" and Latest "now". Enter Cron Expression "*/5 * * * *". Validate that Trigger alert when "Number of Results is greater than" 0 and "Trigger Once" is selected. These settings create an alert that will check every 5 minutes for any results.

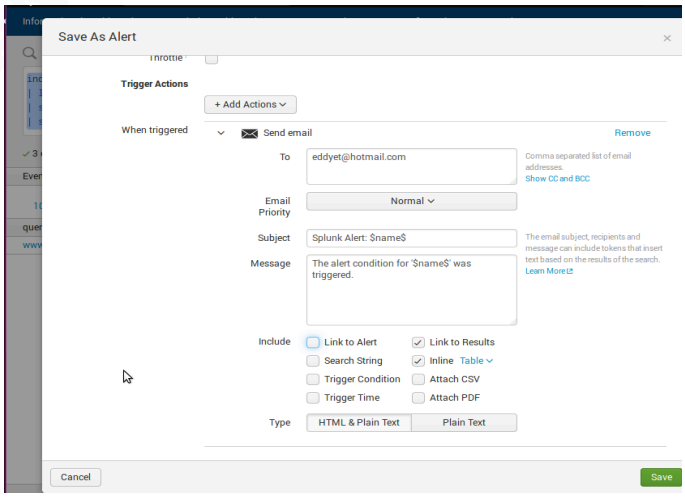


Etrik Eddy, eddyet@hotmail.com

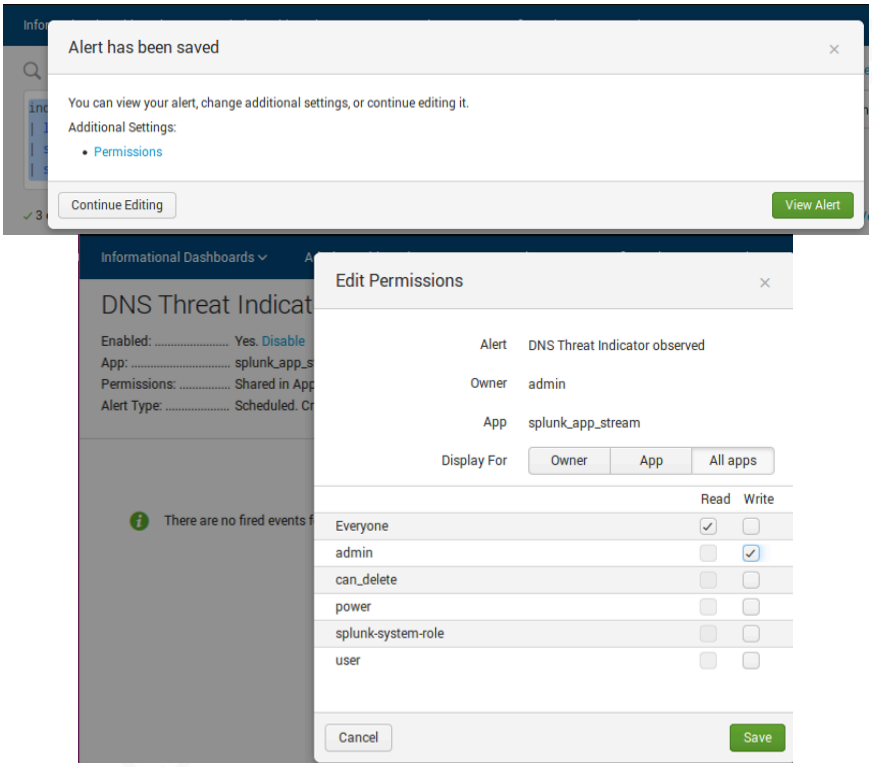
Under “Trigger Actions”, select “Add Actions”, and then select “Send email”.



Enter the email address that should be notified when the alert triggers. Uncheck “Link to Alert”. If the results should be in the email, select the “inline” checkbox. Select Save.



Select “Permissions” and set the appropriate permissions for the alert.



Back on the monitored host, again execute the nslookup for the example domain. It may be up to five minutes before the alert is triggered and a few minutes more before the email is processed.

The process can be completed for each of the sets of threat indicators. Below are the searches used to set up the other alerts:

Site indicators:

```
index=* source="stream:Splunk_HTTPURI" site!=""
| lookup lookup_threatdomain domain as site OUTPUT description
| search description=*
| stats count by site description host
```

URL indicators:

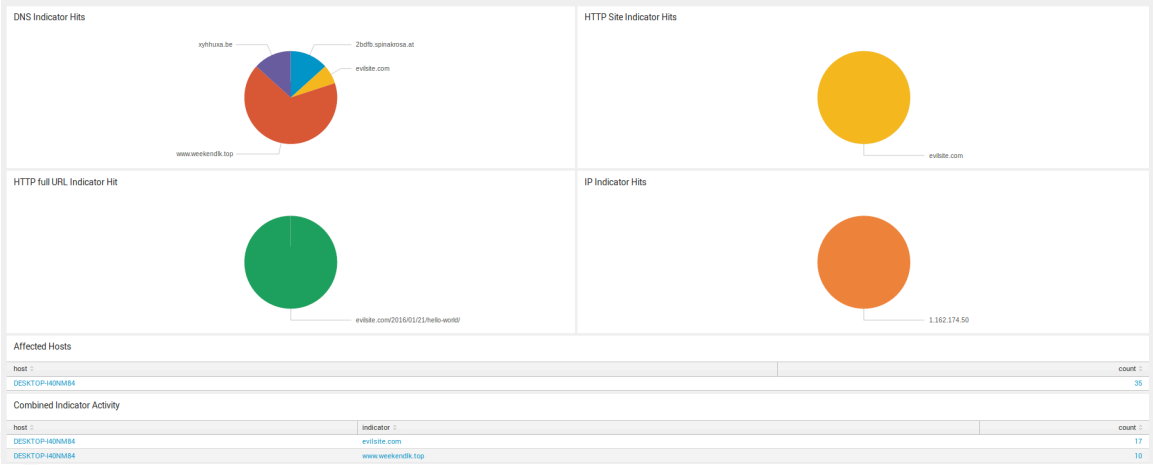
```
index=* source="stream:Splunk_HTTPURI"  uri_path!="" |eval
URL=site.uri_path
| lookup lookup_threaturl url as URL OUTPUT description
| search description=*
| stats count by URL description host
```

Etrik Eddy, eddyet@hotmail.com

IP indicators:

```
index=* source="stream:*" sourcetype="stream:ip"
| lookup lookup_threatip src_ip OUTPUT description src_ip AS
indicator
| lookup lookup_threatip src_ip as dest_ip OUTPUT description
src_ip AS indicator
| search description=*
|stats count by indicator description host
```

There is a dashboard included in the TA_Threat_Intel app (also found in Appendix C) that shows an overview of indicators that have been seen:



5. Conclusion

The landscape of networks is changing. Just as Charles Dickens's castle crumbled and succumbed to time and change, so too must the way network based intrusion detection is conducted yield to those immutable forces. To maintain visibility and ensure security in these changing network topologies, security experts will evolve their toolsets and techniques. One way traditional network intrusion detection system (NIDS) indicators of compromise can be implemented within these new worlds through gathering network data from endpoints using Stream, ingesting freely available rulesets into Splunk, and then using Splunk to analyze the traffic against the rulesets and trigger alerts for cyber security analysts. Network data directly from the endpoint provides

efficiencies in complex networks where an analyst would have to delve through layers of network address translation and DHCP logs to attempt to discover what the true origin of traffic is in the traditional NIDS alerts. They now have the authoritative host involved since it is the sensor. Network traffic could also be just the tip of the iceberg. Using the same core technology, many other data sets could be analyzed for other indicators as well (event logs, for example) from the same interface. With change comes challenge and opportunity.

References

Cisco Meraki | SD-WAN. (n.d.). Retrieved December 19, 2016, from https://meraki.cisco.com/solutions/sd-wan

Daucourt, A. (2015, February 10). splunk and free open-source threat intelligence feeds. Retrieved from http://www.deepimpact.io/blog/splunkandfreeopen-sourcethreatintelligencefeeds

Dickens, C. (1836). *The Pickwick Papers*. Retrieved from http://www.victorianlondon.org/etexts/dickens/pickwick-0005.shtml

Frincke, D., & Bishop, M. (2004). Guarding the castle keep: teaching with the fortress metaphor. *IEEE Security & Privacy Magazine*, *2*(3), 69-72. doi:10.1109/msp.2004.13

Leuprecht, C., Skillicorn, D. B., & Tait, V. E. (2016). Beyond the Castle Model of cyber-risk and cyber-security. *Government Information Quarterly*, *33*(2), 250-257. doi:10.1016/j.giq.2016.01.012

Stallings, W., & Brown, L. (2015). Introduction to Network-Based Intrusion Detection Systems. In *Computer security: Principles and practice*.

Appendix - A
Lab Setup

The lab for this research was comprised of two virtual machines built in Oracle's VirtualBox. They both had the network set to "NAT-Network" allowing them to talk to each other and the internet.

1. The Server:

Specs:

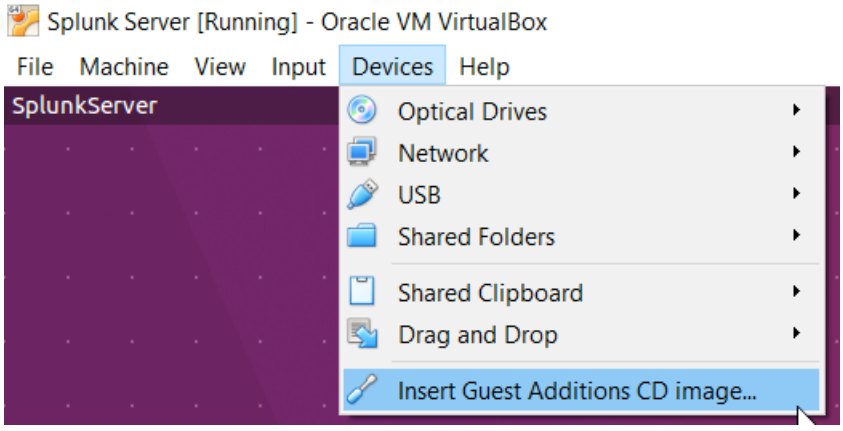
-8Gb ram

-50Gb hard drive

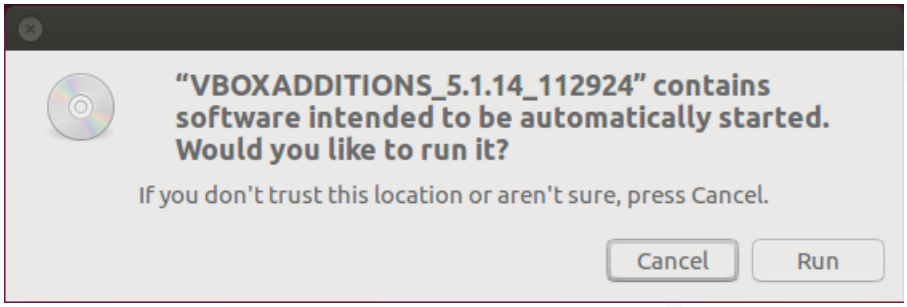
The operating system is Ubuntu 16.04. The basic steps for setup were:

1. Install OS
2. Update OS and software by running:

   ```
   $ sudo apt-get update
   ```
3. Install Virtual Box Guest Additions by selecting Devices- Insert Guest Additions CD image…



Then select "Run" from the prompt that appears.



4. Add User for Splunk to run as by running:

```
$ sudo useradd splunk
$ sudo groupadd splunk
```

5. Install Splunk

After installing the OS installing the Virtual Box Guest Additions and running updates, the below commands were run to prepare and install Splunk Enterprise latest version (6.5.2 as of this writing):

```
$ wget -O splunk-6.5.2-67571ef4b87d-Linux-x86_64.tgz
'https://www.splunk.com/bin/splunk/DownloadActivityServlet?archit
ecture=x86_64&platform=linux&version=6.5.2&product=splunk&filenam
e=splunk-6.5.2-67571ef4b87d-Linux-x86_64.tgz&wget=true'
$ sudo mv ~/splunk-6.5.2-67571ef4b87d-Linux-x86_64.tgz /opt/
$ cd /opt
$ sudo tar xzvf splunk-6.5.2-67571ef4b87d-Linux-x86_64.tgz
$ sudo chown -R splunk:splunk /opt/splunk
$ sudo su splunk
$ cd /opt/splunk/bin
$ sudo ./splunk start
```

Press the spacebar to scroll through EULA then type "y"

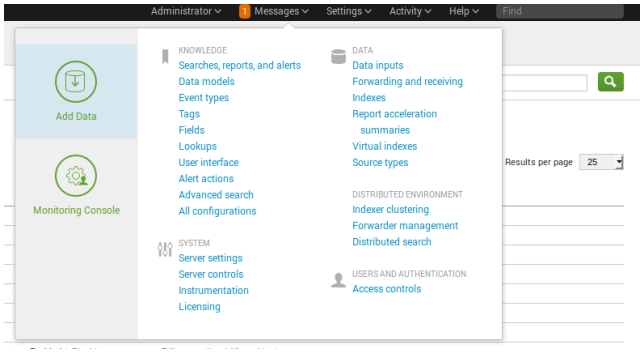
Open a browser and browse to: http://localhost:8000

Log in using the default credentials admin:changeme

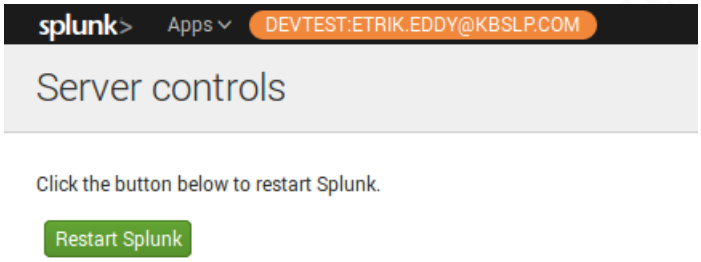
CHANGE THE PASSWORD

Optional step: Install a license if you have one:

Restart Splunk by going to Settings- Server controls:



Select Restart Splunk



2. The Host:

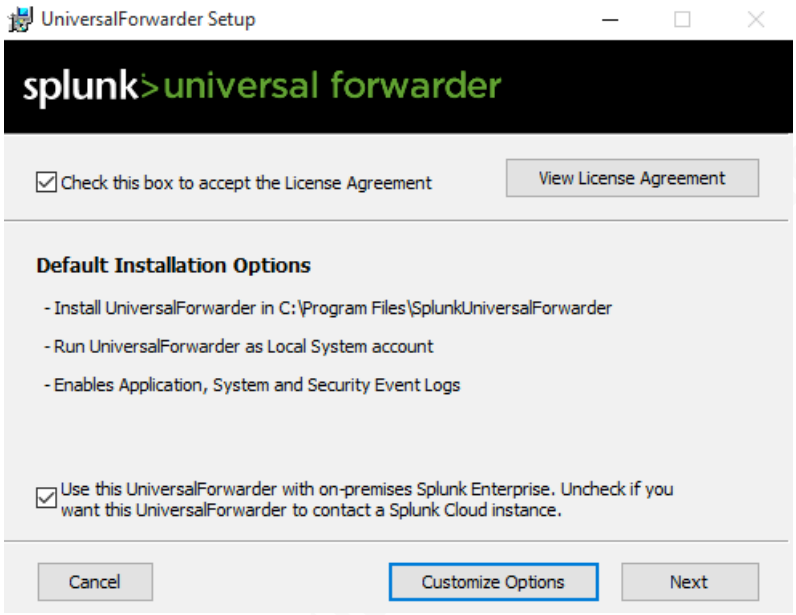
Specs:

-2Gb ram

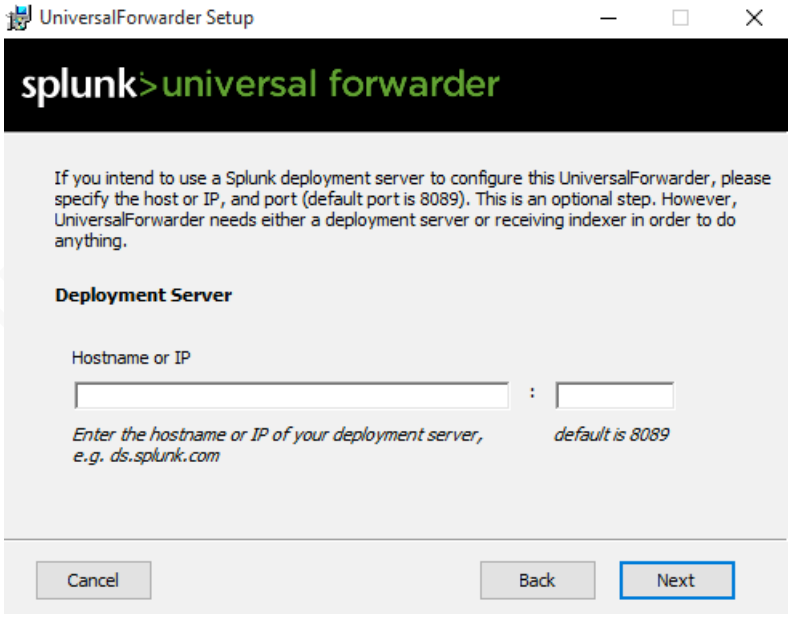
-35Gb hard drive

The operating system is Windows 10. The basic steps for setup were:

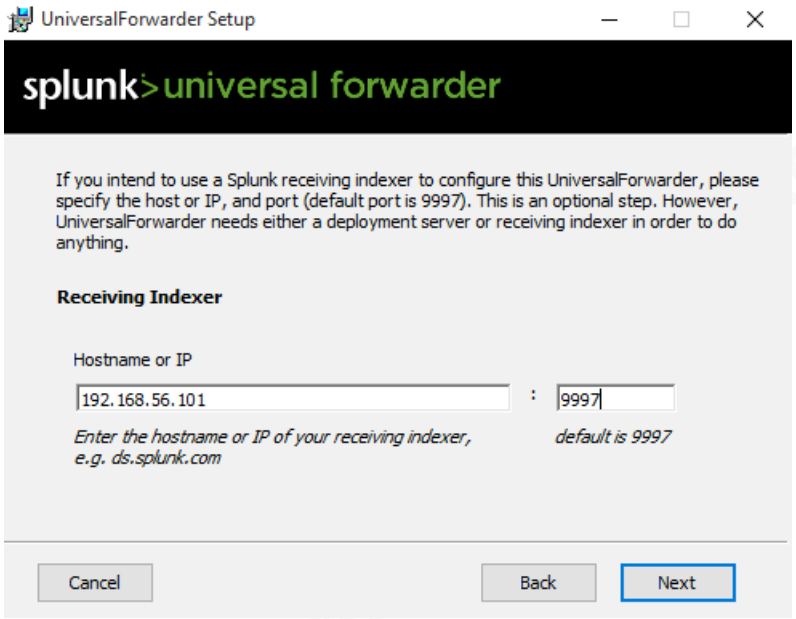
1. Install the OS
2. Update the OS by running windows update
3. Download the Splunk Univeral Forwarder (6.5.2 as of this writing) by running the below command:
   `wget -O splunkforwarder-6.5.2-67571ef4b87d-x64-release.msi 'https://www.splunk.com/bin/splunk/DownloadActivityServlet?architecture=x86_64&platform=windows&version=6.5.2&product=universalforwarder&filename=splunkforwarder-6.5.2-67571ef4b87d-x64-release.msi&wget=true'`
4. Execute the MSI. Click the checkbox to accept the EULA and select next.



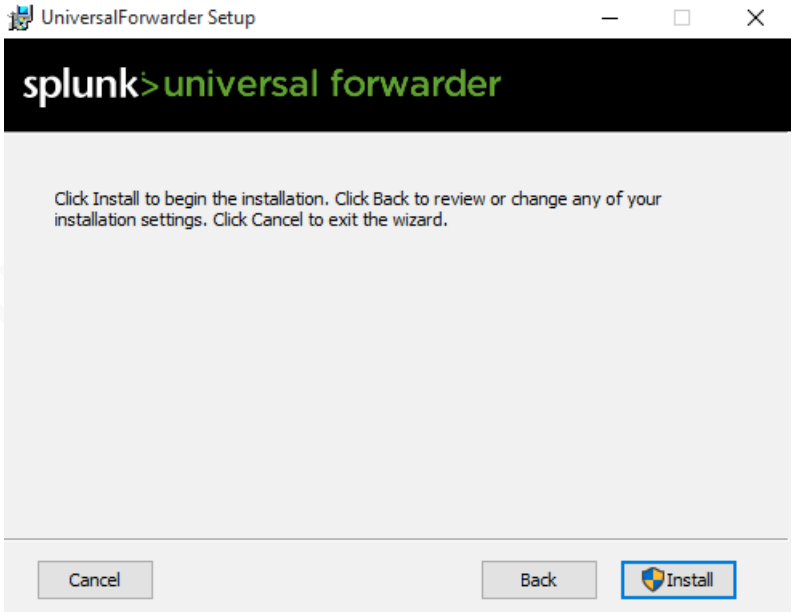
5. Select Next:



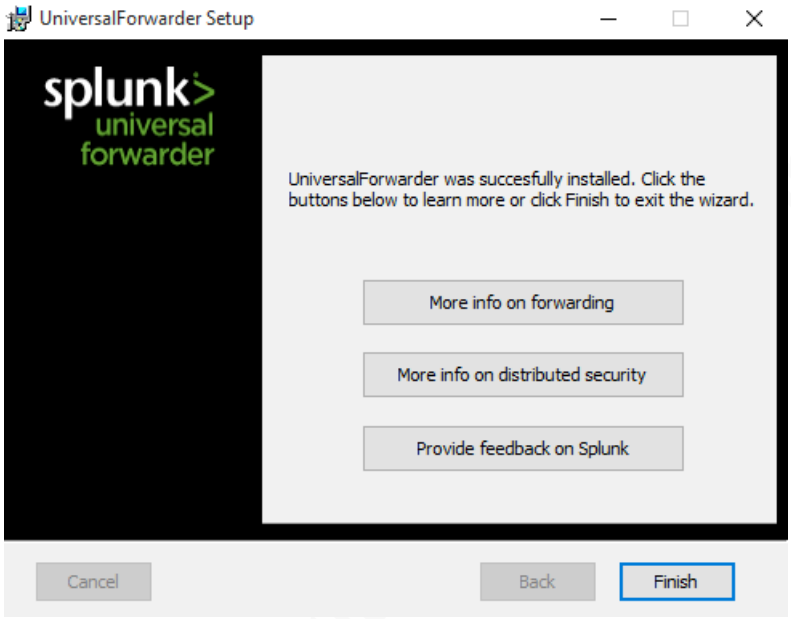
6. Enter the IP address of the Splunk Server and port 9997 then click next:



7. Select "Install":



8. Select "Finish":



9. Modify the below file (or create it if it doesn't exist) so it reflects the IP of your server (this is where you can change this setting if your server's IP changes for any reason also):

C:\Program Files\SplunkUniversalForwarder\etc\system\local\outputs.conf

```
[tcpout]
defaultGroup = default-autolb-group

[tcpout:default-autolb-group]
server = 192.168.56.101:9997

[tcpout-server://192.168.56.101:9997]
```

Appendix - B
Threat Intelligence Ingestion Script

```
#!/bin/bash
#Developed By Etrik Eddy based on example from
#  http://www.deepimpact.io/blog/splunkandfreeopen-
sourcethreatintelligencefeeds

unset LD_LIBRARY_PATH
DOWNLOAD_DIR="/tmp/threat_intel_download"

mkdir -p $DOWNLOAD_DIR


#================================================================
===============
#Malc0de - Malc0de Blacklist
#================================================================
===============

wget http://malc0de.com/bl/IP_Blacklist.txt -O /tmp/IP_Blacklist.txt --no-check-
certificate -N
echo "# Generated: `date`" > $DOWNLOAD_DIR/ip_malc0de_black_list.txt
cat /tmp/IP_Blacklist.txt | sed -n '/^[0-9]/p' | sed 's/$/ Malc0de IP/' >>
$DOWNLOAD_DIR/ip_malc0de_black_list.txt
rm /tmp/IP_Blacklist.txt

#================================================================
===============
#Binary Defense Systems Artillery Threat Intelligence Feed and Banlist Feed
#================================================================
===============

wget http://www.binarydefense.com/banlist.txt -O /tmp/binary_defense_ips.txt --
no-check-certificate -N
echo "# Generated: `date`" > $DOWNLOAD_DIR/ip_binary_defense_ban_list.txt
cat /tmp/binary_defense_ips.txt | sed -n '/^[0-9]/p' | sed 's/$/ Binary Defense IP/'
>> $DOWNLOAD_DIR/ip_binary_defense_ban_list.txt
rm /tmp/binary_defense_ips.txt

#================================================================
===============
#AlienVault - IP Reputation Database
```

Etrik Eddy, eddyet@hotmail.com

```
#================================================================
===============

wget https://reputation.alienvault.com/reputation.snort.gz -P /tmp --no-check-
certificate -N
gzip -d /tmp/reputation.snort.gz
echo "# Generated: `date`" > $DOWNLOAD_DIR/ip_av_rep_list.txt
cat /tmp/reputation.snort | sed -n '/^[0-9]/p' | sed "s/# //">>
$DOWNLOAD_DIR/ip_av_rep_list.txt
rm /tmp/reputation.snort

#================================================================
===============
#Ransomware Tracker - IP Block List
#================================================================
===============

wget https://ransomwaretracker.abuse.ch/downloads/RW_IPBL.txt -O
/tmp/ransomwaretracker.txt --no-check-certificate -N
echo "# Generated: `date`" > $DOWNLOAD_DIR/ip_ransomware_block_list.txt
cat /tmp/ransomwaretracker.txt | sed -n '/^[0-9]/p' | sed 's/$/ Ransomware IP/' >>
$DOWNLOAD_DIR/ip_ransomware_block_list.txt
rm /tmp/ransomwaretracker.txt

#================================================================
===============
#Ransomware Tracker - domain Block List
#================================================================
===============

wget https://ransomwaretracker.abuse.ch/downloads/RW_DOMBL.txt -O
/tmp/ransomwareDomainTracker.txt --no-check-certificate -N
echo "# Generated: `date`" > $DOWNLOAD_DIR/domain_ransomware_block_list.txt
cat /tmp/ransomwareDomainTracker.txt | sed  '/^#/d'| sed 's/$/ Ransomware
Domain/' >> $DOWNLOAD_DIR/domain_ransomware_block_list.txt
rm /tmp/ransomwareDomainTracker.txt

#================================================================
===============
#Ransomware Tracker - url Block List
#================================================================
===============
```

Etrik Eddy, eddyet@hotmail.com

```
wget https://ransomwaretracker.abuse.ch/downloads/RW_URLBL.txt -O
/tmp/ransomwareURLTracker.txt --no-check-certificate -N
echo "# Generated: `date`" > $DOWNLOAD_DIR/url_ransomware_block_list.txt
cat /tmp/ransomwareURLTracker.txt | sed  '/^#/d'| sed 's/$/ Ransomware URL/'
>> $DOWNLOAD_DIR/url_ransomware_block_list.txt
rm /tmp/ransomwareURLTracker.txt

#============================================================
===============
#VirusShare Hash list
#============================================================
===============

wget https://virusshare.com/hashes/VirusShare_00000.md5 -O
/tmp/virussharehash.txt --no-check-certificate -N
echo "# Generated: `date`" > $DOWNLOAD_DIR/hash_virusshare_list.txt
cat /tmp/virussharehash.txt | sed  '/^#/d'| sed 's/$/ Malware Hash/' >>
$DOWNLOAD_DIR/hash_virusshare_list.txt
rm /tmp/virussharehash.txt


#============================================================
===============
# Alexa 1 M
#============================================================
===============

wget http://s3.amazonaws.com/alexa-static/top-1m.csv.zip -O /tmp/top-1m.csv.zip
--no-check-certificate -N
unzip -o /tmp/top-1m.csv.zip -d /tmp/
mv /tmp/top-1m.csv $DOWNLOAD_DIR/
```

Appendix - C
Threat Intelligence Dashboard

Import the below XML to recreate the Threat Intelligence Dashboard:

```
<form>
  <label>Threat Intelligence Hits</label>
  <fieldset submitButton="false">
    <input type="time" token="field1">
      <label></label>
      <default>
        <earliest>-24h@h</earliest>
        <latest>now</latest>
      </default>
    </input>
  </fieldset>
  <row>
    <panel>
      <title>DNS Indicator Hits</title>
      <chart>
        <search>
          <query>index=*
source="stream:Splunk_DNSRequestResponse"
| lookup lookup_threatdomain domain as query OUTPUT description
| search description=*
| stats count by query</query>
          <earliest>$field1.earliest$</earliest>
          <latest>$field1.latest$</latest>
          <sampleRatio>1</sampleRatio>
        </search>
        <option
name="charting.axisLabelsX.majorLabelStyle.overflowMode">ellipsis
None</option>
        <option
name="charting.axisLabelsX.majorLabelStyle.rotation">0</option>
        <option
name="charting.axisTitleX.visibility">visible</option>
        <option
name="charting.axisTitleY.visibility">visible</option>
        <option
name="charting.axisTitleY2.visibility">visible</option>
        <option name="charting.axisX.scale">linear</option>
        <option name="charting.axisY.scale">linear</option>
        <option name="charting.axisY2.enabled">0</option>
        <option name="charting.axisY2.scale">inherit</option>
        <option name="charting.chart">pie</option>
        <option
name="charting.chart.bubbleMaximumSize">50</option>
        <option
name="charting.chart.bubbleMinimumSize">10</option>
        <option name="charting.chart.bubbleSizeBy">area</option>
```

Etrik Eddy, eddyet@hotmail.com

```
        <option name="charting.chart.nullValueMode">gaps</option>
        <option
name="charting.chart.showDataLabels">none</option>
        <option
name="charting.chart.sliceCollapsingThreshold">0.01</option>
        <option name="charting.chart.stackMode">default</option>
        <option name="charting.chart.style">shiny</option>
        <option name="charting.drilldown">all</option>
        <option name="charting.layout.splitSeries">0</option>
        <option
name="charting.layout.splitSeries.allowIndependentYRanges">0</opt
ion>
        <option
name="charting.legend.labelStyle.overflowMode">ellipsisMiddle</op
tion>
        <option name="charting.legend.placement">right</option>
      </chart>
    </panel>
    <panel>
      <title>HTTP Site Indicator Hits</title>
      <chart>
        <search>
          <query>index=* source="stream:Splunk_HTTPURI" site!=""
| lookup lookup_threatdomain domain as site OUTPUT description
| search description=*
| stats count by site</query>
          <earliest>$field1.earliest$</earliest>
          <latest>$field1.latest$</latest>
          <sampleRatio>1</sampleRatio>
        </search>
        <option
name="charting.axisLabelsX.majorLabelStyle.overflowMode">ellipsis
None</option>
        <option
name="charting.axisLabelsX.majorLabelStyle.rotation">0</option>
        <option
name="charting.axisTitleX.visibility">visible</option>
        <option
name="charting.axisTitleY.visibility">visible</option>
        <option
name="charting.axisTitleY2.visibility">visible</option>
        <option name="charting.axisX.scale">linear</option>
        <option name="charting.axisY.scale">linear</option>
        <option name="charting.axisY2.enabled">0</option>
        <option name="charting.axisY2.scale">inherit</option>
        <option name="charting.chart">pie</option>
        <option
name="charting.chart.bubbleMaximumSize">50</option>
        <option
name="charting.chart.bubbleMinimumSize">10</option>
        <option name="charting.chart.bubbleSizeBy">area</option>
        <option name="charting.chart.nullValueMode">gaps</option>
```

```
        <option
name="charting.chart.showDataLabels">none</option>
        <option
name="charting.chart.sliceCollapsingThreshold">0.01</option>
        <option name="charting.chart.stackMode">default</option>
        <option name="charting.chart.style">shiny</option>
        <option name="charting.drilldown">all</option>
        <option name="charting.layout.splitSeries">0</option>
        <option
name="charting.layout.splitSeries.allowIndependentYRanges">0</opt
ion>
        <option
name="charting.legend.labelStyle.overflowMode">ellipsisMiddle</op
tion>
        <option name="charting.legend.placement">right</option>
      </chart>
    </panel>
  </row>
  <row>
    <panel>
      <title>HTTP full URL Indicator Hit</title>
      <chart>
        <search>
          <query>index=* source="stream:Splunk_HTTPURI"
uri_path!="" |eval URL=site.uri_path
| lookup lookup_threaturl url as URL OUTPUT description
| search description=*
| stats count by URL</query>
          <earliest>$field1.earliest$</earliest>
          <latest>$field1.latest$</latest>
          <sampleRatio>1</sampleRatio>
        </search>
        <option name="charting.chart">pie</option>
      </chart>
    </panel>
    <panel>
      <title>IP Indicator Hits</title>
      <chart>
        <search>
          <query>index=* source="stream:*" sourcetype="stream:ip"
| lookup lookup_threatip src_ip OUTPUT description src_ip AS
indicator
| lookup lookup_threatip src_ip as dest_ip OUTPUT description
src_ip AS indicator
| search description=* | stats count by indicator</query>
          <earliest>$field1.earliest$</earliest>
          <latest>$field1.latest$</latest>
          <sampleRatio>1</sampleRatio>
        </search>
        <option
name="charting.axisLabelsX.majorLabelStyle.overflowMode">ellipsis
None</option>
```

```
        <option
name="charting.axisLabelsX.majorLabelStyle.rotation">0</option>
        <option
name="charting.axisTitleX.visibility">visible</option>
        <option
name="charting.axisTitleY.visibility">visible</option>
        <option
name="charting.axisTitleY2.visibility">visible</option>
        <option name="charting.axisX.scale">linear</option>
        <option name="charting.axisY.scale">linear</option>
        <option name="charting.axisY2.enabled">0</option>
        <option name="charting.axisY2.scale">inherit</option>
        <option name="charting.chart">pie</option>
        <option
name="charting.chart.bubbleMaximumSize">50</option>
        <option
name="charting.chart.bubbleMinimumSize">10</option>
        <option name="charting.chart.bubbleSizeBy">area</option>
        <option name="charting.chart.nullValueMode">gaps</option>
        <option
name="charting.chart.showDataLabels">none</option>
        <option
name="charting.chart.sliceCollapsingThreshold">0.01</option>
        <option name="charting.chart.stackMode">default</option>
        <option name="charting.chart.style">shiny</option>
        <option name="charting.drilldown">all</option>
        <option name="charting.layout.splitSeries">0</option>
        <option
name="charting.layout.splitSeries.allowIndependentYRanges">0</opt
ion>
        <option
name="charting.legend.labelStyle.overflowMode">ellipsisMiddle</op
tion>
        <option name="charting.legend.placement">right</option>
      </chart>
    </panel>
  </row>
  <row>
    <panel>
      <title>Affected Hosts</title>
      <table>
        <search>
          <query>index=*
source="stream:Splunk_DNSRequestResponse"
| lookup lookup_threatdomain domain as query OUTPUT description
| search description=*
| rename query as indicator
| APPEND
    [ search index=* source="stream:Splunk_HTTPURI" site!=""
    | lookup lookup_threatdomain domain as site OUTPUT
description
    | search description=*
```

```
    | rename site as indicator]
| APPEND
    [ search index=* source="stream:Splunk_HTTPURI" uri_path!=""
    | eval URL=site.uri_path
    | lookup lookup_threaturl url as URL OUTPUT description
    | search description=*
    | rename URL as indicator ]
| APPEND
    [ search index=* sourcetype="stream:ip"
    | lookup lookup_threatip src_ip OUTPUT description src_ip AS
indicator
    | lookup lookup_threatip src_ip as dest_ip OUTPUT description
src_ip AS indicator
    | search description=*]
| stats count by host</query>
          <earliest>$field1.earliest$</earliest>
          <latest>$field1.latest$</latest>
          <sampleRatio>1</sampleRatio>
        </search>
      </table>
    </panel>
  </row>
  <row>
    <panel>
      <title>Combined Indicator Activity</title>
      <table>
        <search>
          <query>index=*
source="stream:Splunk_DNSRequestResponse"
| lookup lookup_threatdomain domain as query OUTPUT description
| search description=* | rename query as indicator
| APPEND
    [ search index=* source="stream:Splunk_HTTPURI" site!=""
    | lookup lookup_threatdomain domain as site OUTPUT
description
    | search description=* | rename site as indicator]
| APPEND
    [ search index=* source="stream:Splunk_HTTPURI" uri_path!=""
    | eval URL=site.uri_path
    | lookup lookup_threaturl url as URL OUTPUT description
    | search description=* | rename URL as indicator ]
| APPEND
    [ search index=* sourcetype="stream:ip"
    | lookup lookup_threatip src_ip OUTPUT description src_ip AS
indicator
    | lookup lookup_threatip src_ip as dest_ip OUTPUT description
src_ip AS indicator
    | search description=*]
| stats count by host indicator | sort - count</query>
          <earliest>$field1.earliest$</earliest>
          <latest>$field1.latest$</latest>
          <sampleRatio>1</sampleRatio>
```

```
        </search>
        <option name="count">100</option>
        <option name="dataOverlayMode">none</option>
        <option name="drilldown">cell</option>
        <option name="percentagesRow">false</option>
        <option name="rowNumbers">false</option>
        <option name="totalsRow">false</option>
        <option name="wrap">true</option>
      </table>
    </panel>
  </row>
</form>
```