FIELD TECH NOTES

Azure Security Center - Manual Integration

Patrick Glynn
CE, AMER

Introduction

This guide will walk the reader through the process of adding existing Palo Alto Networks firewalls to the Azure Security Center (ASC). It is possible to deploy a firewall based upon the recommendations by ASC which already has the necessary integration configured; however, due to limitations in Azure, this can only be used for new deployments. This document can be used in cases where existing firewalls need to be added to ASC. Note that the concepts herein should be applicable to Panorama.

Azure Security Center is a security management system that facilitates securing on-premises and cloud-based resources by providing a central location for log/alert collection and processing.

Expected Outcome

This guide will walk the reader through the following configurations.

- Creating an Azure Log Analytics Workspace
- Gathering integration information for the syslog host
- Building a Linux host to forward logs to Log Analytics
- Configure a custom log format on the Firewall

Prerequisites

Before You Begin

This guide assumes prior knowledge of and access to the Azure console. The guide also assumes prior knowledge of the Palo Alto Networks VM-Series firewall.

The reader should now login into the Azure console and navigate to the desired Resource Group.

Log Analytics Workspace

Overview

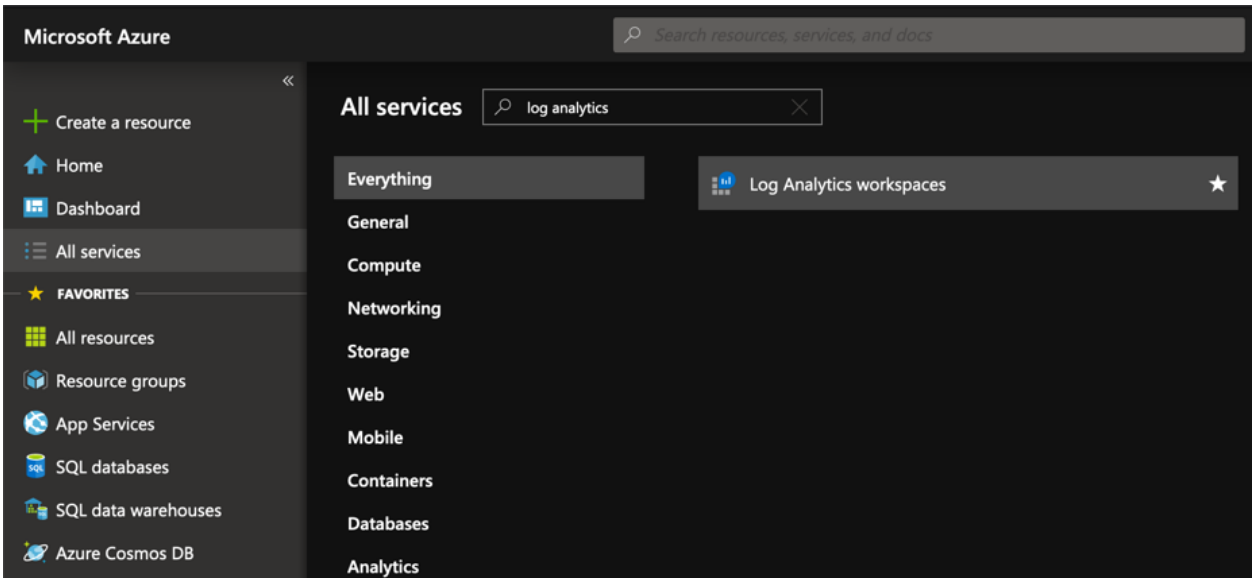
The guide will walk through the creation of a Log Analytics Workspace. The Log Analytics Workspace serves as the location to which the firewall logs are written prior to processing by ASC.

Process Flow

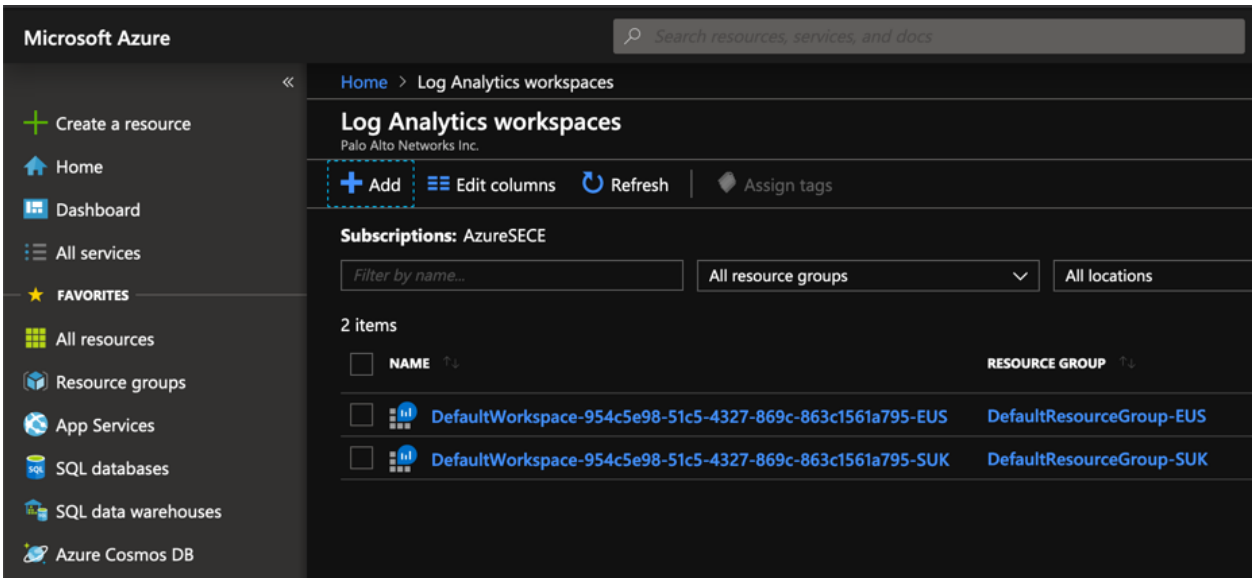
Procedure 1: **Log Analytics Workspace Creation**

Step 1 In the Azure console, open the target Resource Group.

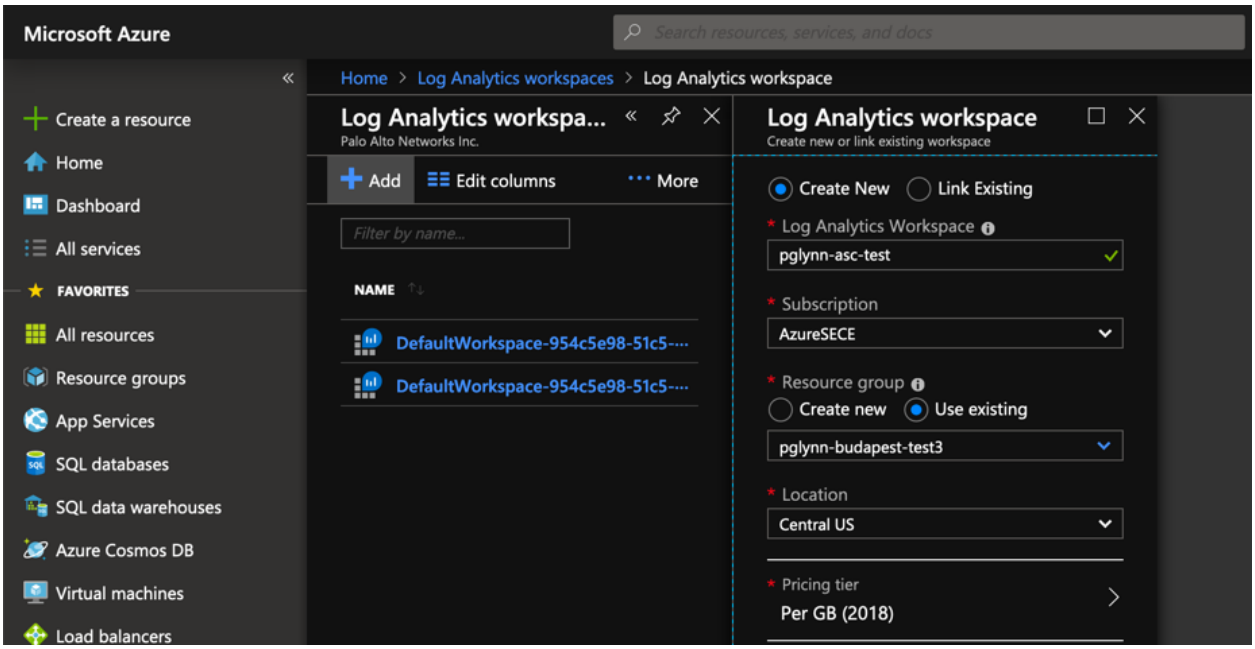
Step 2 Create a new Log Analytics Workspace



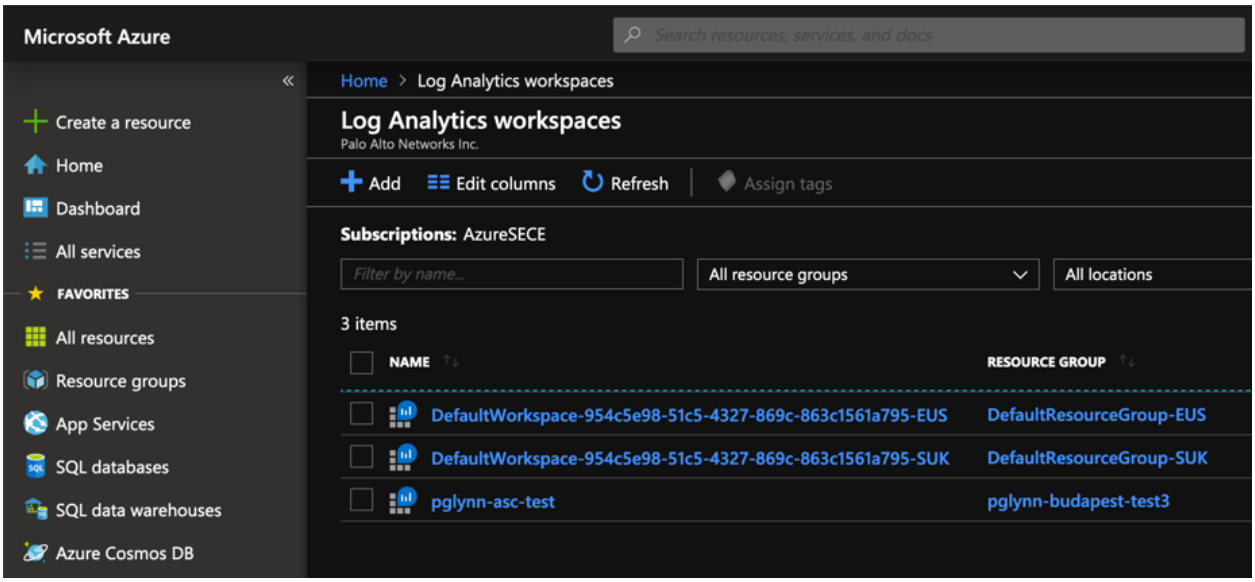
Step 3 Click “Add”



Step 4 Specify the relevant parameters and click “OK”.



Step 5 Complete.



Gather Information

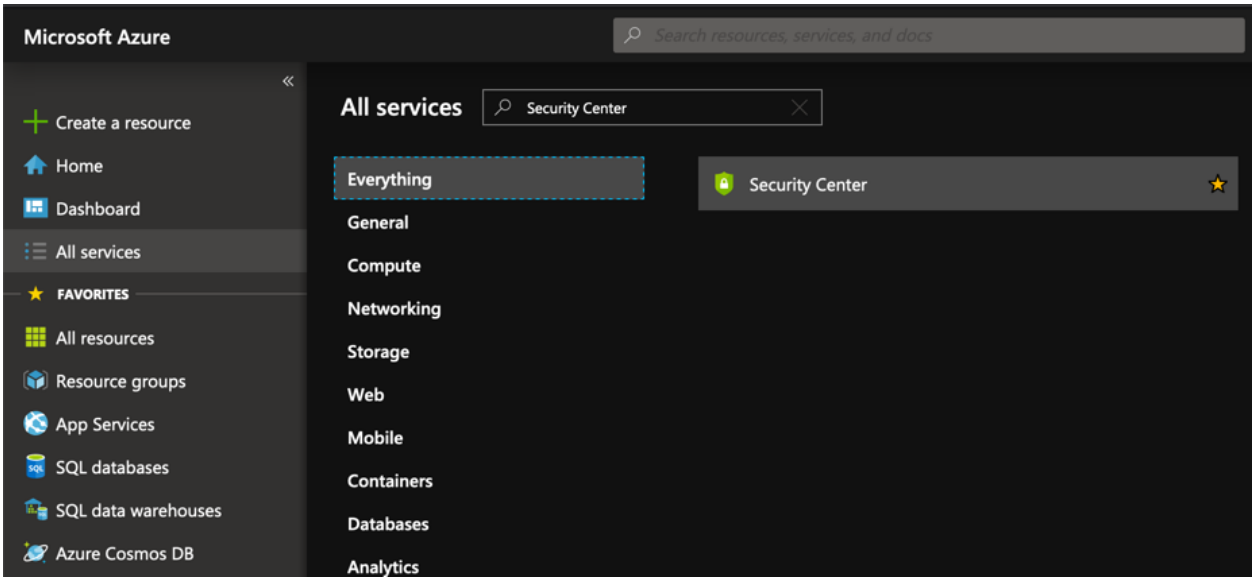
Overview

Only authenticated systems are permitted to send logs to the Log Analytics Workspace. Prior to the deployment/configuration of the syslog receiver, it is necessary to gather the authentication information from ASC.

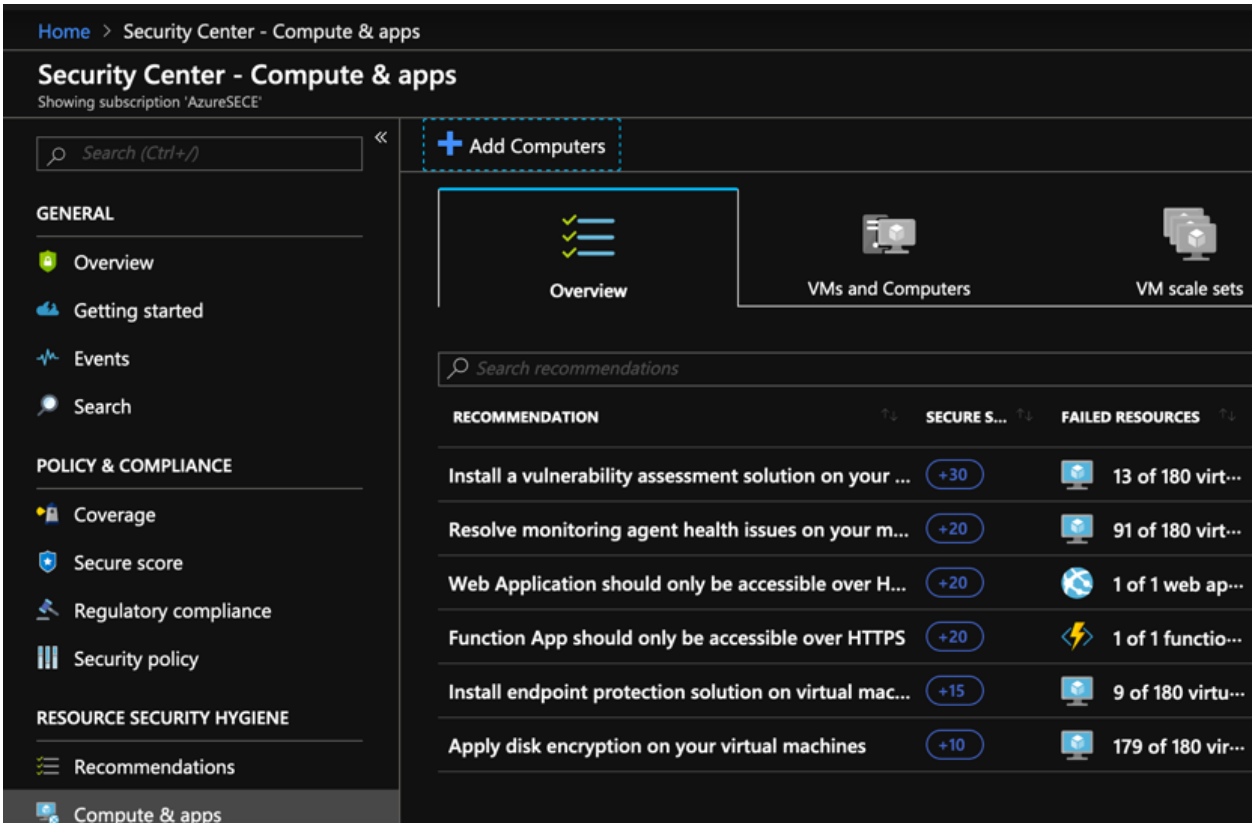
Process Flow

Procedure 2: **Information Gathering**

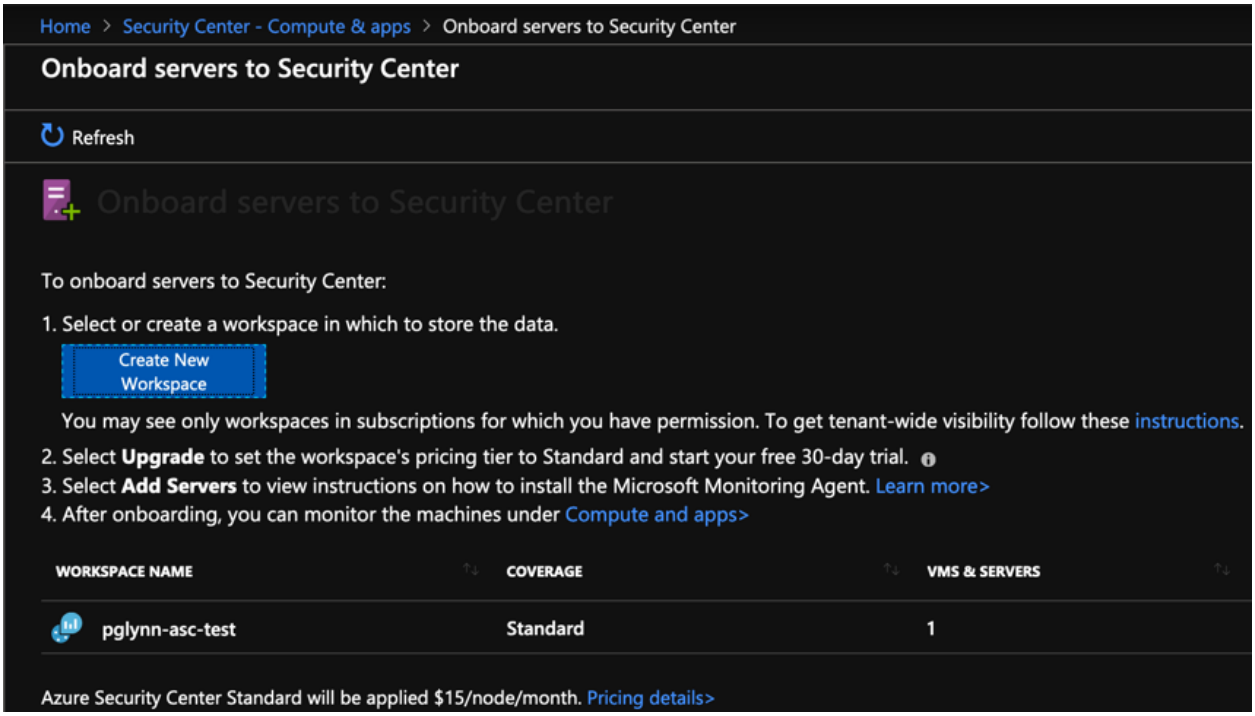
Step 1 In the Azure console, open Security Center.



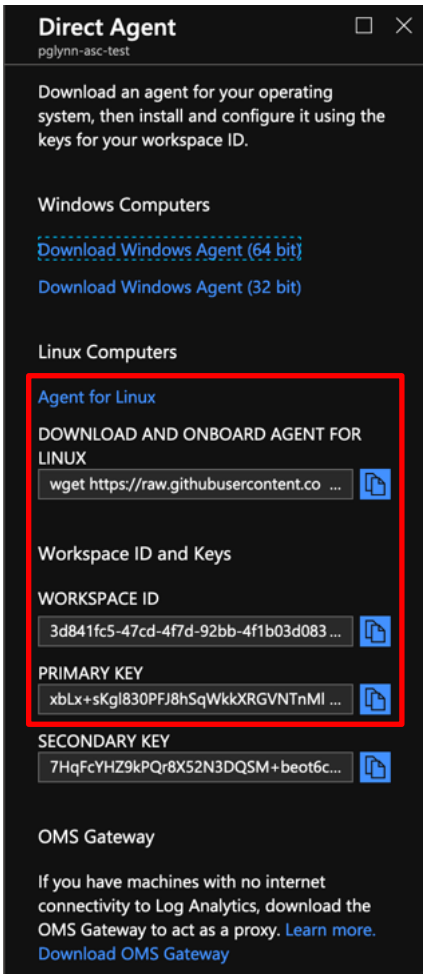
Step 2 Select “Compute & apps” and then click “+ Add Computers”.



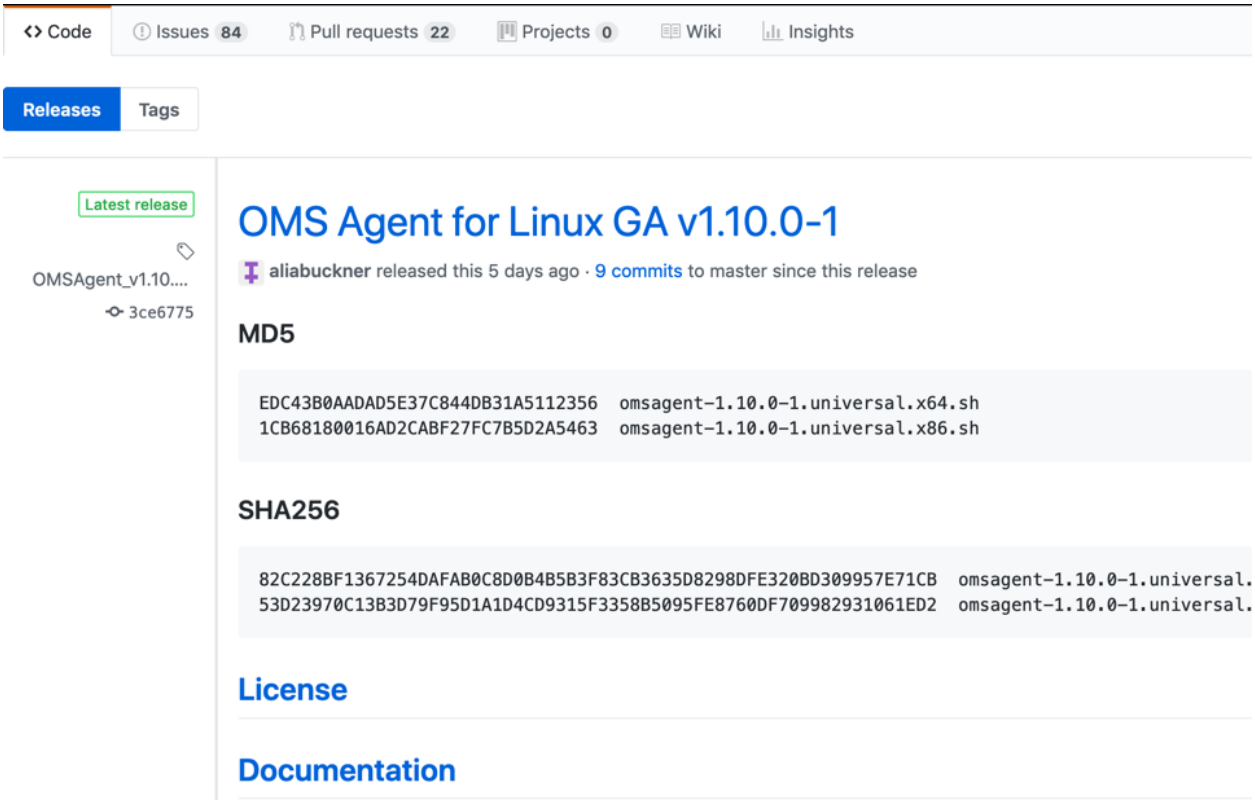
Step 3 Click the previously-created workspace name.



Step 4 Make a note of the Workspace ID and Primary Key, then download the relevant agent.



Step 5 Download the desired Linux agent version from GitHub (1.10.0-1 for 64-bit Linux variants in this case).



Build a Linux Host

Overview

For extant firewalls, traffic logs must be sent through an intermediary system to Log Analytics. In this example, a Linux host is deployed and configured to be that intermediary system. As noted earlier, multiple Linux variants or Windows systems may be used.

Process Flow

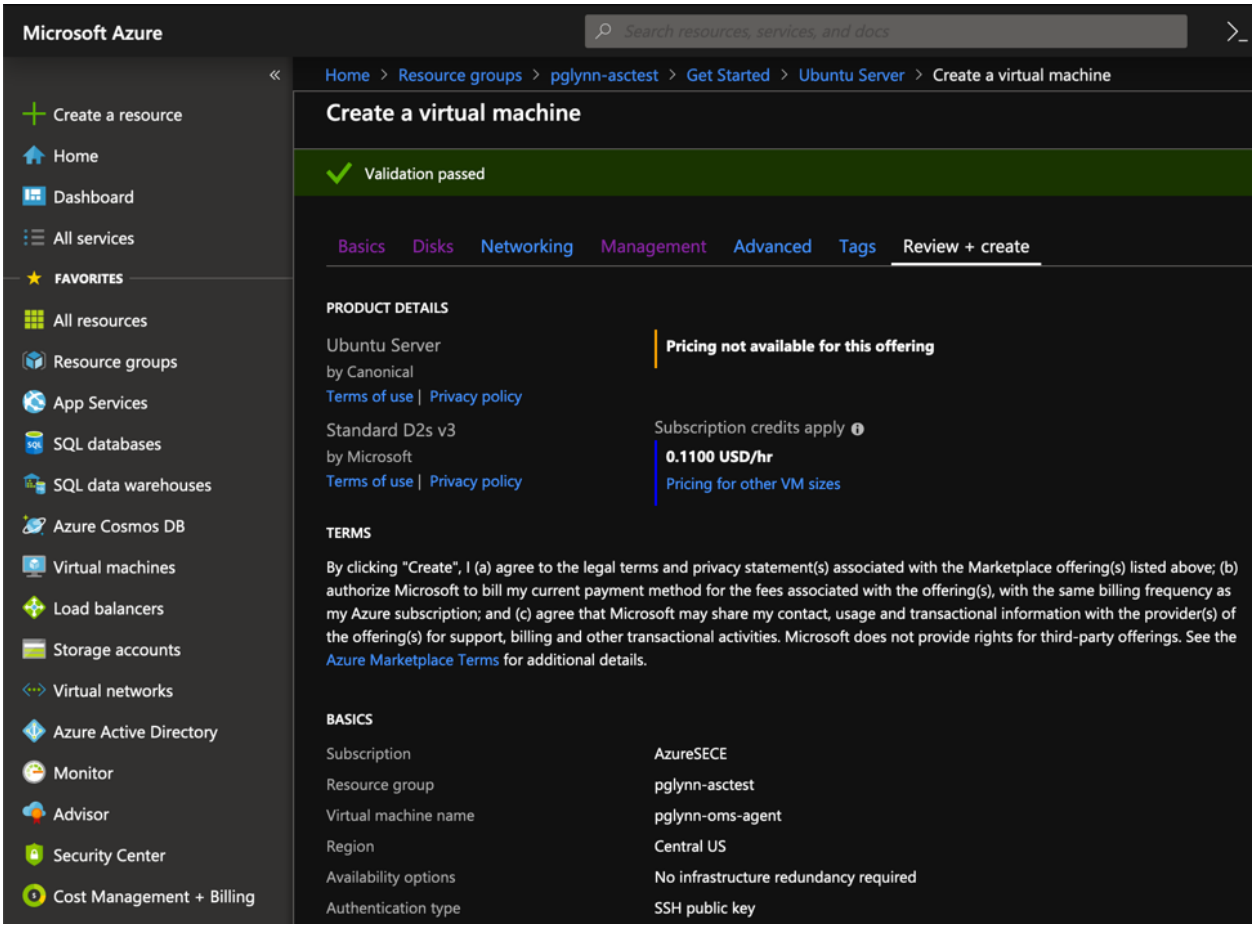
Procedure 3: **Deploy/configure a Linux Host**

The guide moves on to deploy a syslog receiver into the Resource Group. In order to forward logs from the firewall to ASC, a syslog receiver must be used to take in the firewall logs and forward them to the Log Analytics Workspace.

In this example, an Ubuntu server (16.0.4 LTS) is used but other Linux variants as well as Microsoft Windows are also supported.

The host is deployed into the subnet containing the management interface of the firewall; however, it can be deployed into a separate Resource Group/VNet/subnet provided the firewall can send syslog messages to the host.

Step 1 In the Azure console, deploy a Linux server to act as the syslog receiver/log forwarder.



Step 2 Upload the agent to the Linux host.

Step 3 Connect to the Linux host and execute the command “sudo su –“ to become the root user.

Step 4 Navigate to the directory /etc/rsyslog.d/

Step 5 Edit the file “security-config-omsagent.conf”

```
1. root@pglynn-oms-agent: /etc/rsyslog.d (ssh)
pglynn@pglynn-oms-agent:~$ sudo su -
root@pglynn-oms-agent:~# cd /etc/rsyslog.d/
root@pglynn-oms-agent:/etc/rsyslog.d# vi security-config-omsagent.conf
```

Step 6 Insert the following text into the file:

```
#OMS_facility = local4

local4.debug  @127.0.0.1:25226
```

Step 7 Save the file. This example assumes rsyslogd is in use. If syslog-ng or another syslog service is in use, refer to documentation for that service for configuration details. The desired result is for syslog messages received on the local4 syslog facility to be resent to localhost on port 25226.

Step 8 Edit the file “/etc/rsyslog.conf” and enable syslog reception on UDP/514.

```
1. root@pglynn-oms-agent: ~ (ssh)
#   /etc/rsyslog.conf      Configuration file for rsyslog.
#
#                          For more information see
#                          /usr/share/doc/rsyslog-doc/html/rsyslog_conf.html
#
#   Default logging rules can be found in /etc/rsyslog.d/50-default.conf


#################
#### MODULES ####
#################

module(load="imuxsock")  # provides support for local system logging
module(load="imklog")    # provides kernel logging support
#module(load="immark")   # provides --MARK-- message capability

# provides UDP syslog reception
module(load="imudp")
input(type="imudp" port="514")

# provides TCP syslog reception
#module(load="imtcp")
#input(type="imtcp" port="514")

"/etc/rsyslog.conf" 61L, 1371C                          19,1          Top
```

This example assumes rsyslogd is in use. If syslog-ng or another syslog service is in use, refer to documentation for that service for configuration details. The desired result is for the syslog daemon to receive logs from external hosts on UDP port 514 (or whichever port is desired).

Step 9 Install the auditd package.

```
1. root@pglynn-oms-agent: /var/tmp (ssh)
root@pglynn-oms-agent:/var/tmp# apt-get install auditd
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  libauparseo
Suggested packages:
  audispd-plugins
The following NEW packages will be installed:
  auditd libauparseo
o upgraded, 2 newly installed, o to remove and o not upgraded.
Need to get 228 kB of archives.
After this operation, 737 kB of additional disk space will be used.
Do you want to continue? [Y/n] Y
Get:1 http://azure.archive.ubuntu.com/ubuntu xenial-updates/main amd64 libaupars
eo amd64 1:2.4.5-1ubuntu2.1 [35.5 kB]
Get:2 http://azure.archive.ubuntu.com/ubuntu xenial-updates/main amd64 auditd am
d64 1:2.4.5-1ubuntu2.1 [192 kB]
Fetched 228 kB in os (1,493 kB/s)
Selecting previously unselected package libauparseo:amd64.
(Reading database ... 53440 files and directories currently installed.)
Preparing to unpack .../libauparseo_1%3a2.4.5-1ubuntu2.1_amd64.deb ...
Unpacking libauparseo:amd64 (1:2.4.5-1ubuntu2.1) ...
Selecting previously unselected package auditd.
Preparing to unpack .../auditd_1%3a2.4.5-1ubuntu2.1_amd64.deb ...
```

Step 10 Navigate to the directory where the agent software was uploaded.

```
1. root@pglynn-oms-agent: /var/tmp (ssh)
pglynn@pglynn-oms-agent:~$ sudo su -
root@pglynn-oms-agent:~# cd /etc/rsyslog.d/
root@pglynn-oms-agent:/etc/rsyslog.d# vi security-config-omsagent.conf
root@pglynn-oms-agent:/etc/rsyslog.d# cd var/tmp
-su: cd: var/tmp: No such file or directory
root@pglynn-oms-agent:/etc/rsyslog.d# cd /var/tmp
root@pglynn-oms-agent:/var/tmp# ls
cloud-init
omsagent-1.10.0-1.universal.x64.sh
systemd-private-4d627da548144e3a8261b066d97ca874-systemd-timesyncd.service-OIi3Q
3
root@pglynn-oms-agent:/var/tmp#
```

Step 11 Set the file to executable with the command “chmod +x <file_name>”.

```
1. root@pglynn-oms-agent: /var/tmp (ssh)
pglynn@pglynn-oms-agent:~$ sudo su -
root@pglynn-oms-agent:~# cd /etc/rsyslog.d/
root@pglynn-oms-agent:/etc/rsyslog.d# vi security-config-omsagent.conf
root@pglynn-oms-agent:/etc/rsyslog.d# cd var/tmp
-su: cd: var/tmp: No such file or directory
root@pglynn-oms-agent:/etc/rsyslog.d# cd /var/tmp
root@pglynn-oms-agent:/var/tmp# ls
cloud-init
omsagent-1.10.0-1.universal.x64.sh
systemd-private-4d627da548144e3a8261b066d97ca874-systemd-timesyncd.service-OIi3Q
3
root@pglynn-oms-agent:/var/tmp# chmod +x omsagent-1.10.0-1.universal.x64.sh
root@pglynn-oms-agent:/var/tmp# ls
cloud-init
omsagent-1.10.0-1.universal.x64.sh
systemd-private-4d627da548144e3a8261b066d97ca874-systemd-timesyncd.service-OIi3Q
3
root@pglynn-oms-agent:/var/tmp#
```

Step 12 Usage help can be viewed by issuing the command “omsagent-<version>.universal.x64.sh –help”. The Options of interest are “-w id” and “-s key”. These allow automatic onboarding of the agent to Security Center.

```
1. root@pglynn-oms-agent: /var/tmp (ssh)
usage: omsagent-1.10.0-1.universal.x64.sh [OPTIONS]
Options:
  --extract                     Extract contents and exit.
  --force                       Force upgrade (override version checks).
  --install                     Install the package from the system.
  --purge                       Uninstall the package and remove all related data.
  --remove                      Uninstall the package from the system.
  --restart-deps                Reconfigure and restart dependent service(s).
  --source-references           Show source code reference hashes.
  --upgrade                     Upgrade the package in the system.
  --enable-opsmgr               Enable port 1270 for usage with opsmgr.
  --version                     Version of this shell bundle.
  --version-check               Check versions already installed to see if upgradab
le.
  --debug                       use shell debug mode.

  -w id, --id id                Use workspace ID <id> for automatic onboarding.
  -s key, --shared key          Use <key> as the shared key for automatic onboardin
g.
  -d dmn, --domain dmn          Use <dmn> as the OMS domain for onboarding. Optiona
l.
                                default: opinsights.azure.com
                                ex: opinsights.azure.us (for FairFax)
  -p conf, --proxy conf         Use <conf> as the proxy configuration.
                                ex: -p [protocol://][user:password@]proxyhost[:port
```

Step 13 Run the installation using the Workspace ID and Primary Key information noted earlier:

```
omsagent-1.10.0-1.universal.x64.sh --install -w <Workspace ID> -s
<Primary Key>
```

```
1. root@pglynn-oms-agent: /var/tmp (ssh)
root@pglynn-oms-agent:/var/tmp# ./omsagent-1.10.0-1.universal.x64.sh --install -
w 3d841fc5-47cd-4f7d-92bb-4f1b03d083c2 -s xbLx+sKgl830PFJ8hSqWkkXRGVNTnMl mXD1Tsi
vGu56frwuTXf4UVlW4TK55uJcgOXbAQZok4nxLSv1a2ZoX5g==
Checking host architecture ...
Extracting...
----- Installing package: omi (omi-1.6.0-0.ulinux.x64) -----
Selecting previously unselected package omi.
(Reading database ... 53498 files and directories currently installed.)
Preparing to unpack 100/omi-1.6.0-0.ulinux.x64.deb ...
Creating omiusers group ...
sent invalidate(passwd) request, exiting
sent invalidate(group) request, exiting
sent invalidate(group) request, exiting
Creating omi group ...
sent invalidate(passwd) request, exiting
sent invalidate(group) request, exiting
sent invalidate(group) request, exiting
Creating omi service account ...
sent invalidate(passwd) request, exiting
sent invalidate(group) request, exiting
sent invalidate(passwd) request, exiting
sent invalidate(group) request, exiting
Unpacking omi (1.6.0.0) ...
Setting up omi (1.6.0.0) ...
Generating a 2048 bit RSA private key
```

Change to the directory “/etc/opt/microsoft/omsagent/<Workspace ID>/conf/omsagent.d” and edit the file “security_events.conf”. Populate it with the following text (ensure that the “format” directive is on a single line):

```
<source>
  type syslog
  port 25226
  bind 127.0.0.1
```

```
  protocol_type udp
  tag oms.security
  format /^(?<time>(?:\w+ +){2,3}(?:\d+:){2}\d+):? ?(?:(?<host>[^: ]+)
?:?)? (?<ident>[a-zA-Z0-9_%\/\.\-]*)(?:\[(?<pid>[0-9]+)\])?:
*(?<message>.*)$/
 </source>
 <filter oms.security.**>
  type filter_syslog_security
</filter>
```

N.B. – The file may also be downloaded from the MS GitHub Repository for upload to the Linux host:

https://raw.githubusercontent.com/Microsoft/OMS-Agent-for-Linux/master/installer/conf/omsagent.d/security_events.conf

Step 14 Reboot the Linux host to start/restart all services.

Configure the firewall

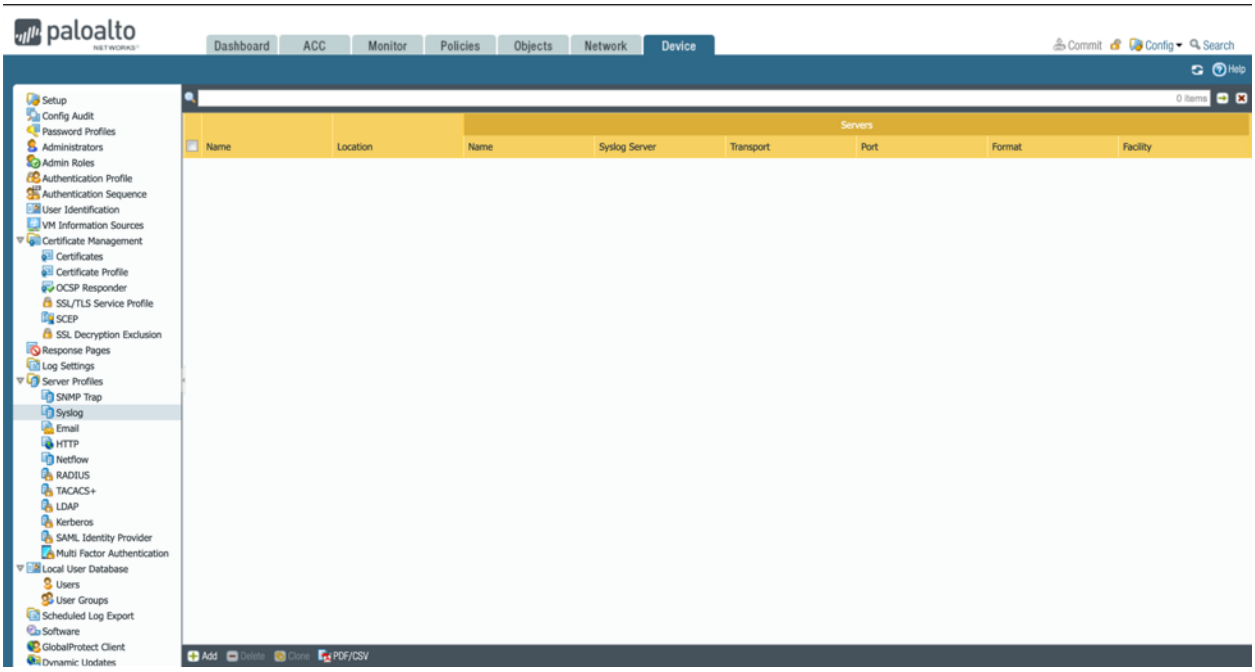
Overview

At this stage, the firewall is configured to send traffic to the syslog server and on to Log Analytics.

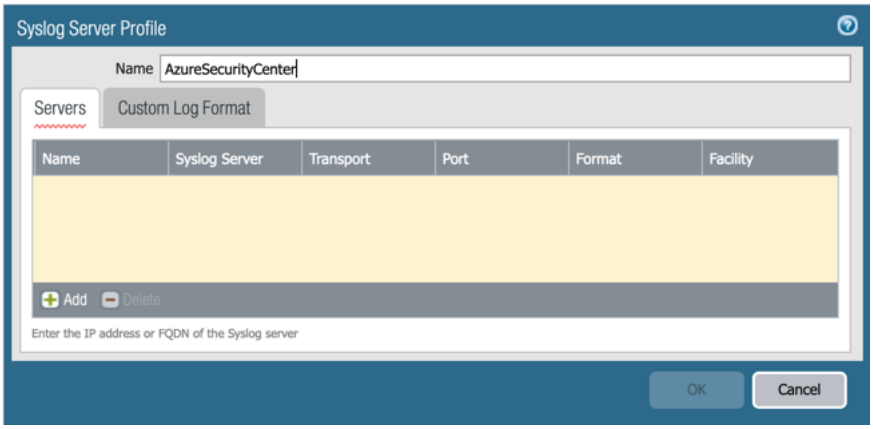
Process Flow

Procedure 4: **Configure the firewall**

Step 1 Login to the Firewall and navigate to the Device Tab > Server Profiles > Syslog.

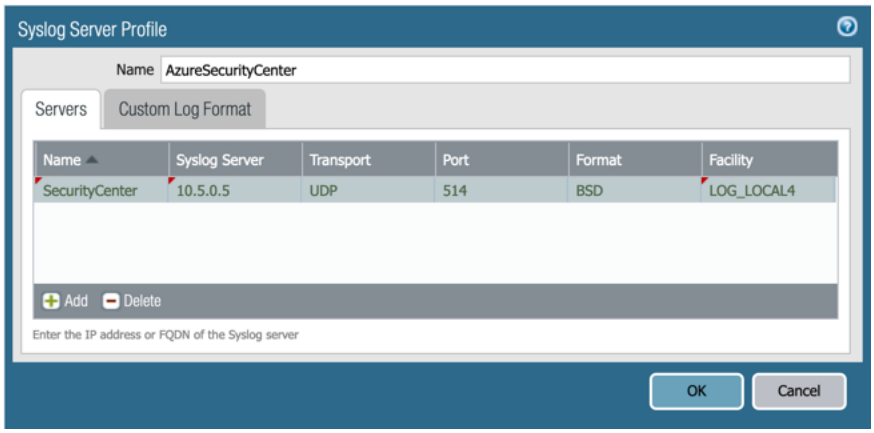


Step 2 Click “Add” and specify a Name for the profile.

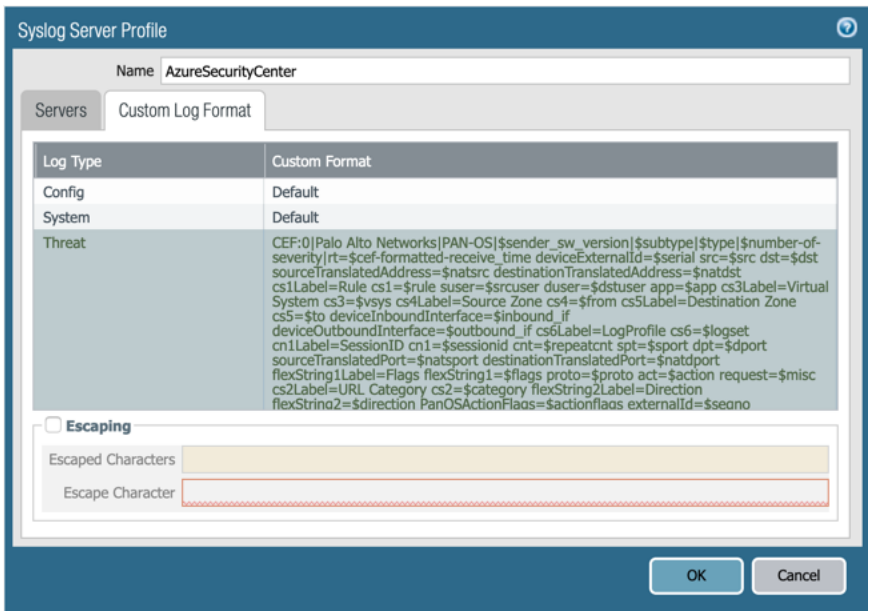


Step 3 Under Servers, click “Add”. Specify:

- Name (freeform text)
- Internal IP of the Linux host
- Transport (UDP default)
- Port (514 default)
- Facility (LOG_LOCAL4)

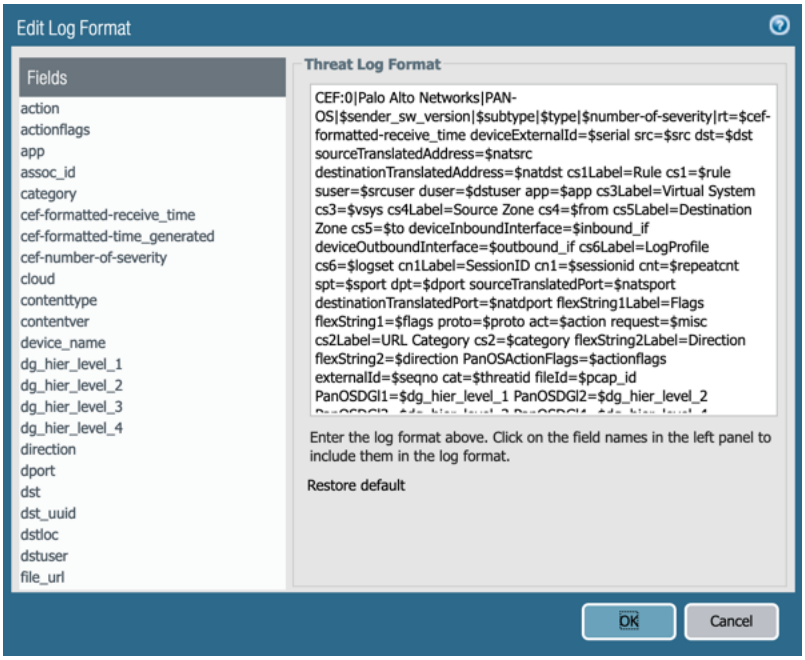


Step 4 Click on the “Custom Log Format” tab.



Step 5 Click on the log type “Threat” to create a custom format for the logs and paste in the CEF format described in the section CEF-style Log Formats in the following link:

https://docs.paloaltonetworks.com/resources/cef

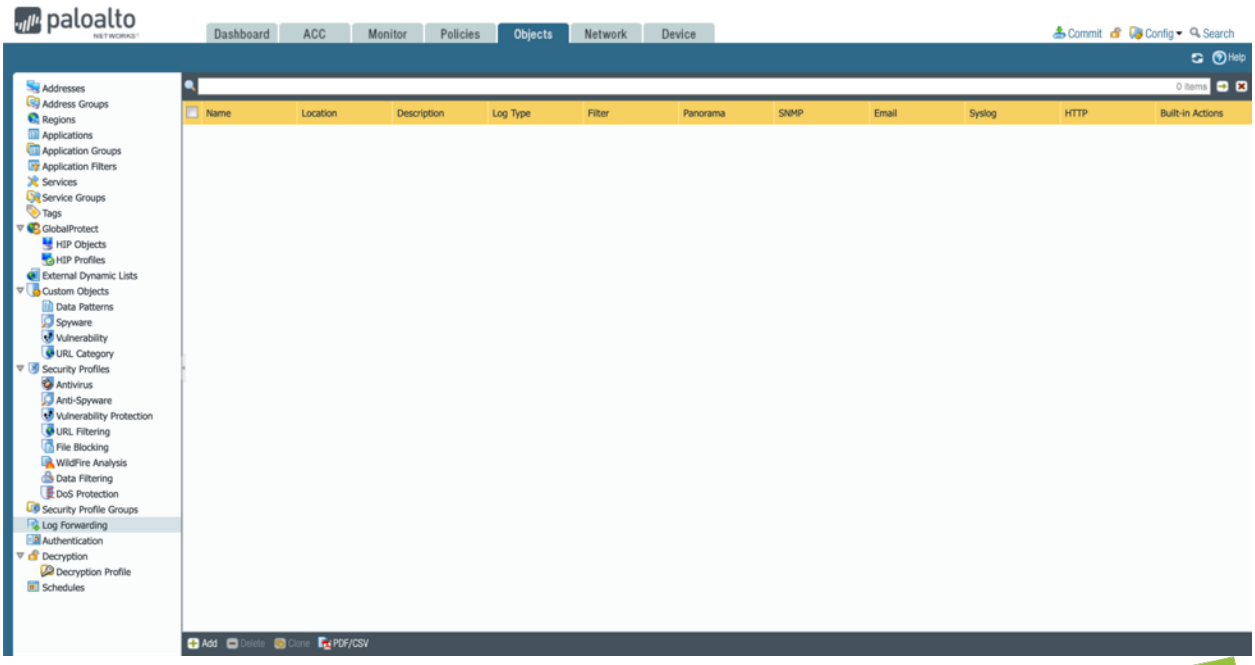


N.B. – Copy/paste actions occasionally result in spurious characters or spaces being inadvertently included. This may result in logs not showing up in the Log Analytics Database of having incorrectly-populated fields.

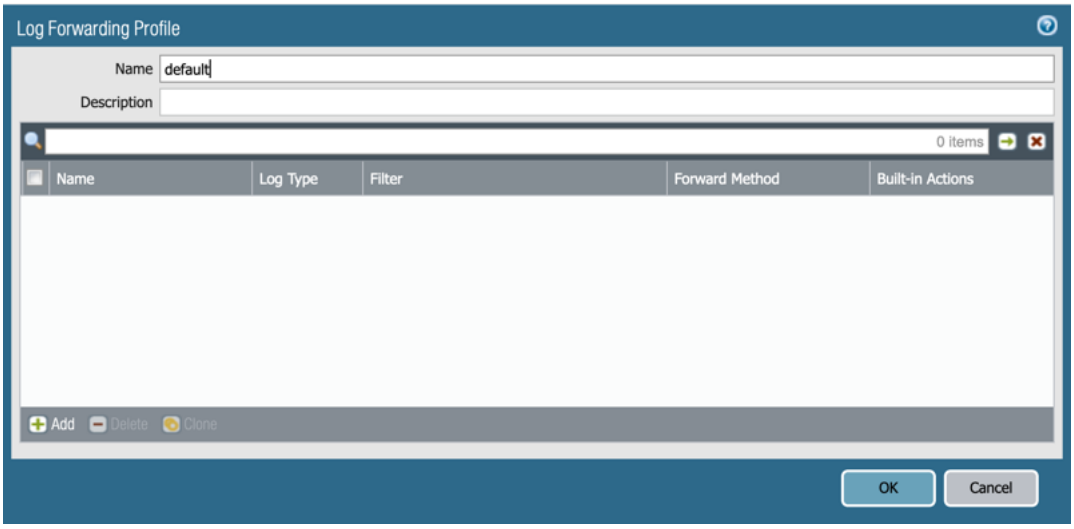
Click “OK” when done.

Step 6 Repeat the previous two steps as necessary for the other logs (e.g. Traffic, Wildfire, URL, System, Config, etc.).

Step 7 Navigate to the Objects Tab > Log Forwarding.



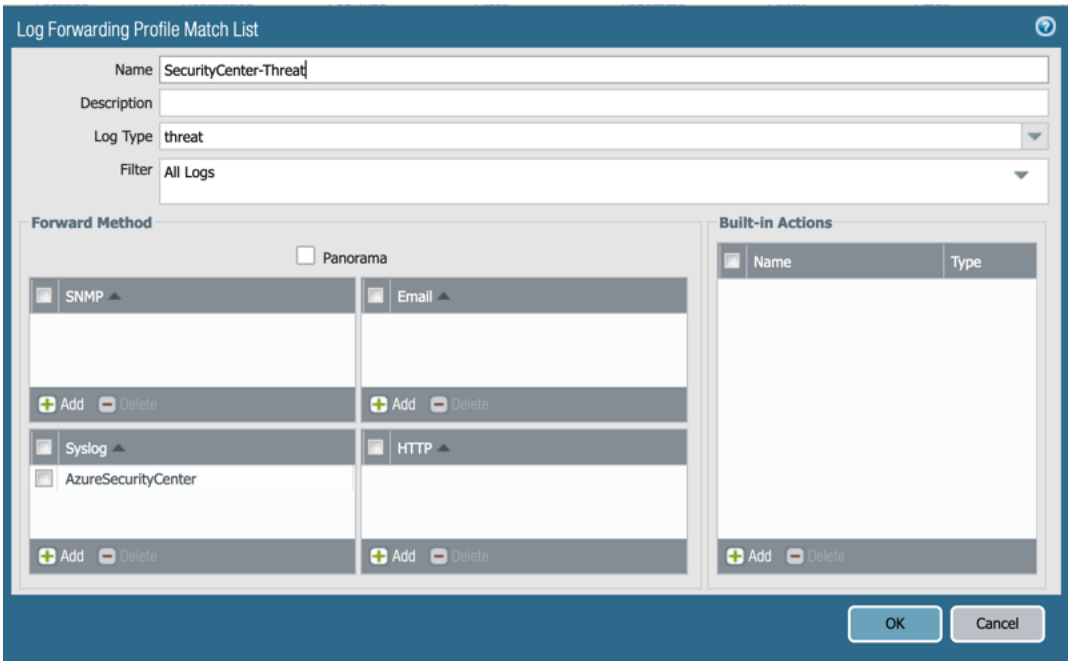
Step 8 Click “Add” and specify a name.



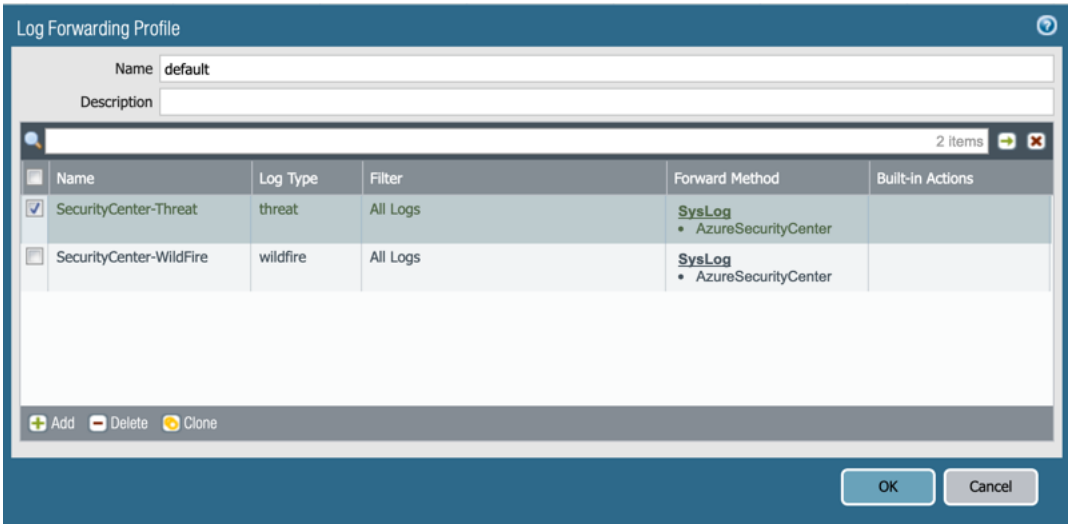
N.B. – If the profile is named “default”, it will automatically show up as the default log forwarding profile in each new rule created. It can also be created on Panorama and be pushed to the firewall(s) as well as be added to an existing log forwarding profile.

Step 9 Click “Add” and specify:

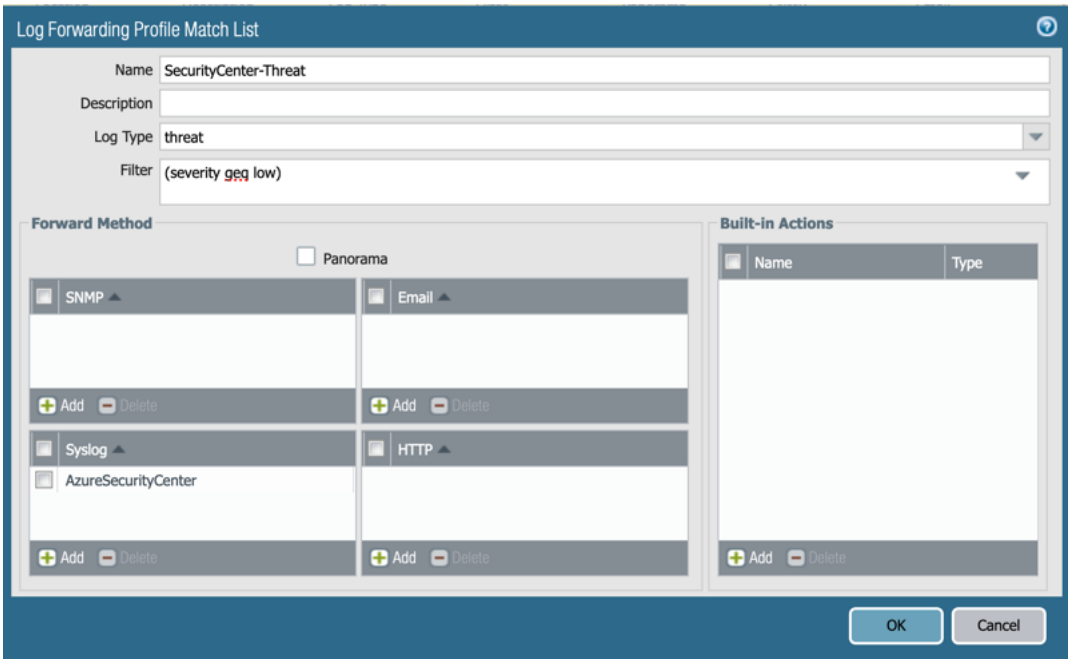
- Name (Free form text)
- Log Type
- The previously-created Syslog profile



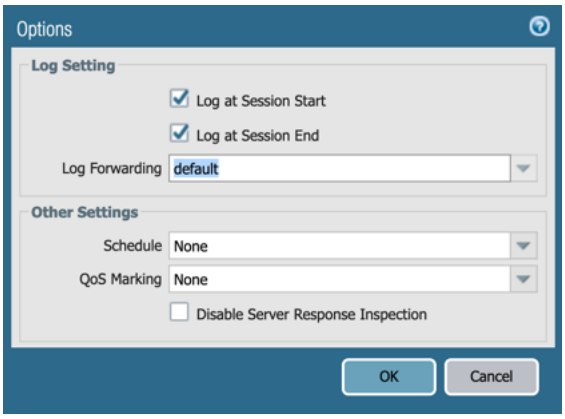
Step 10 Click “OK” when done. Repeat as required for other log types.



N.B. – By default, logs of severity “Low”, “Medium”, “High” and “Critical” are displayed in Security Center. Filters can be used to limit the logs that are sent. This can be used to send “interesting” logs and filter out low value ones (e.g. low level).



Step 11 Attach the Log Forwarding profile to the target rules and commit when done.



Verification

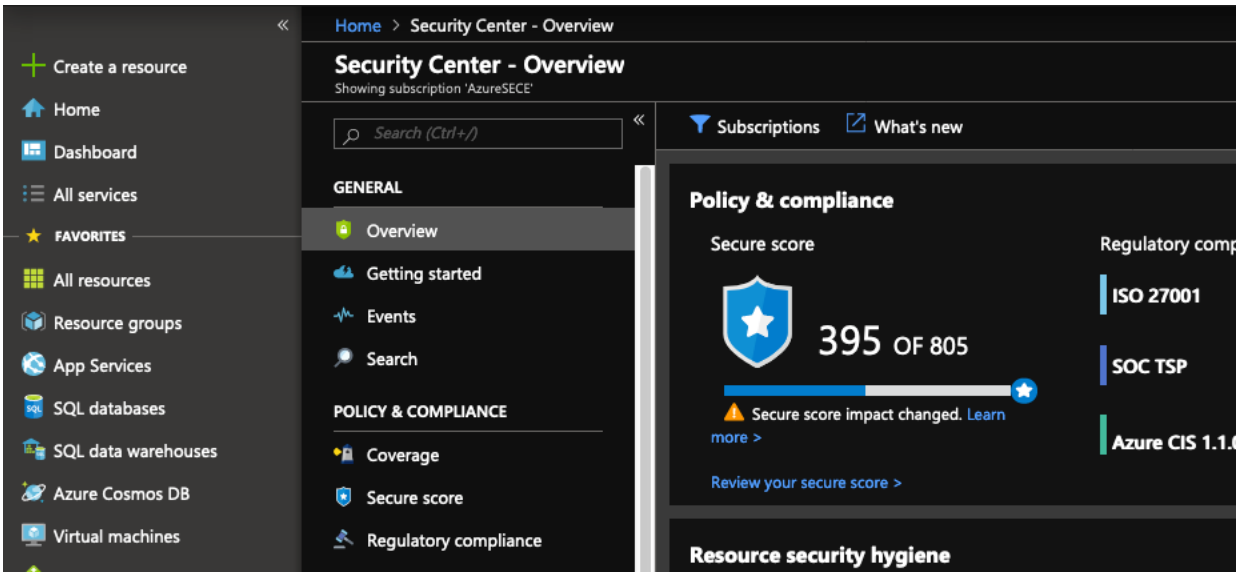
Overview

At this stage, the firewall should be sending traffic to Log Analytics although it may take some time for them to show up in the Azure console. Confirmation may be had by looking at the events in ASC.

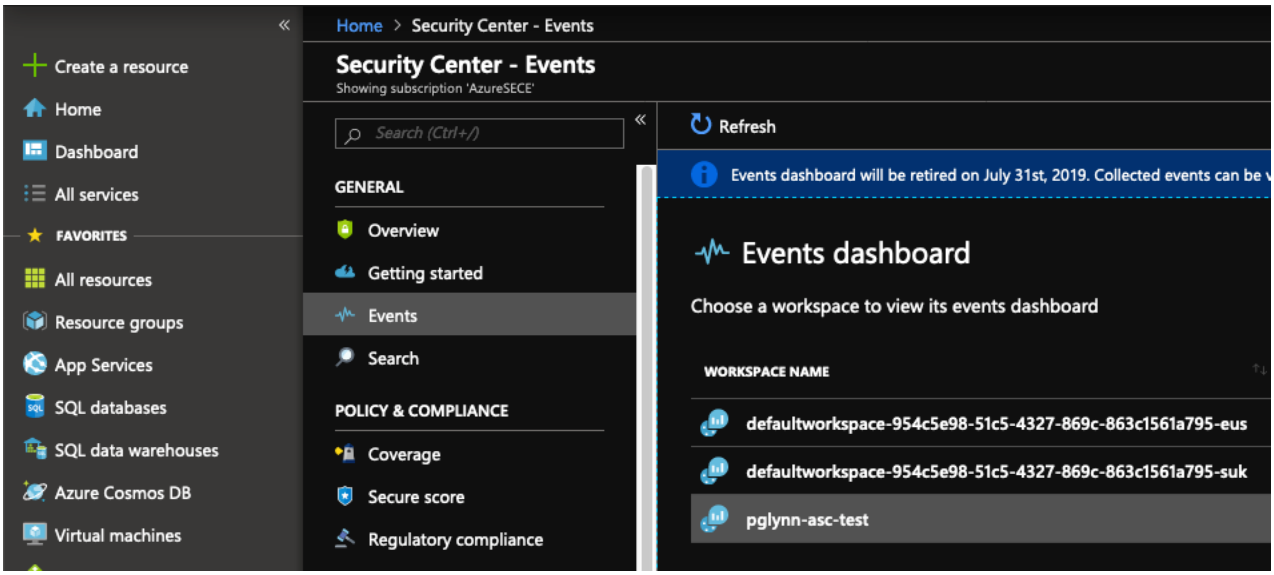
Process Flow

Procedure 5: Verify logs in ASC

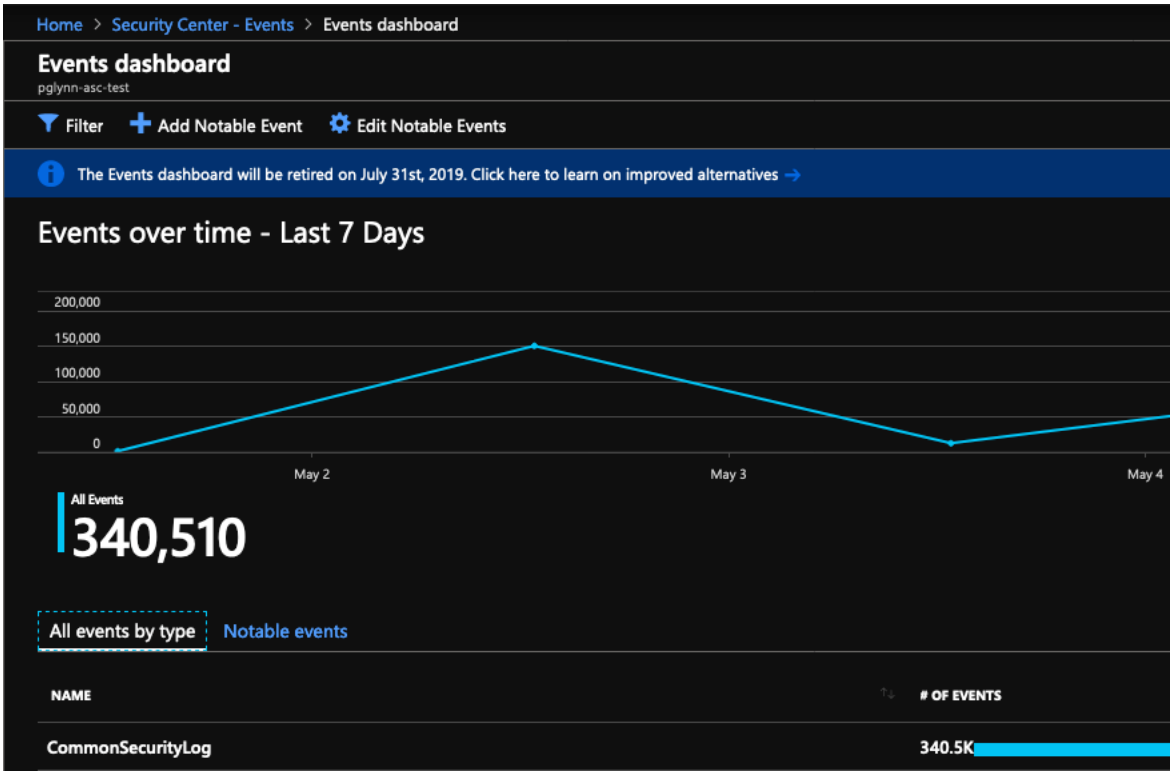
Step 1 Launch Azure Security Center



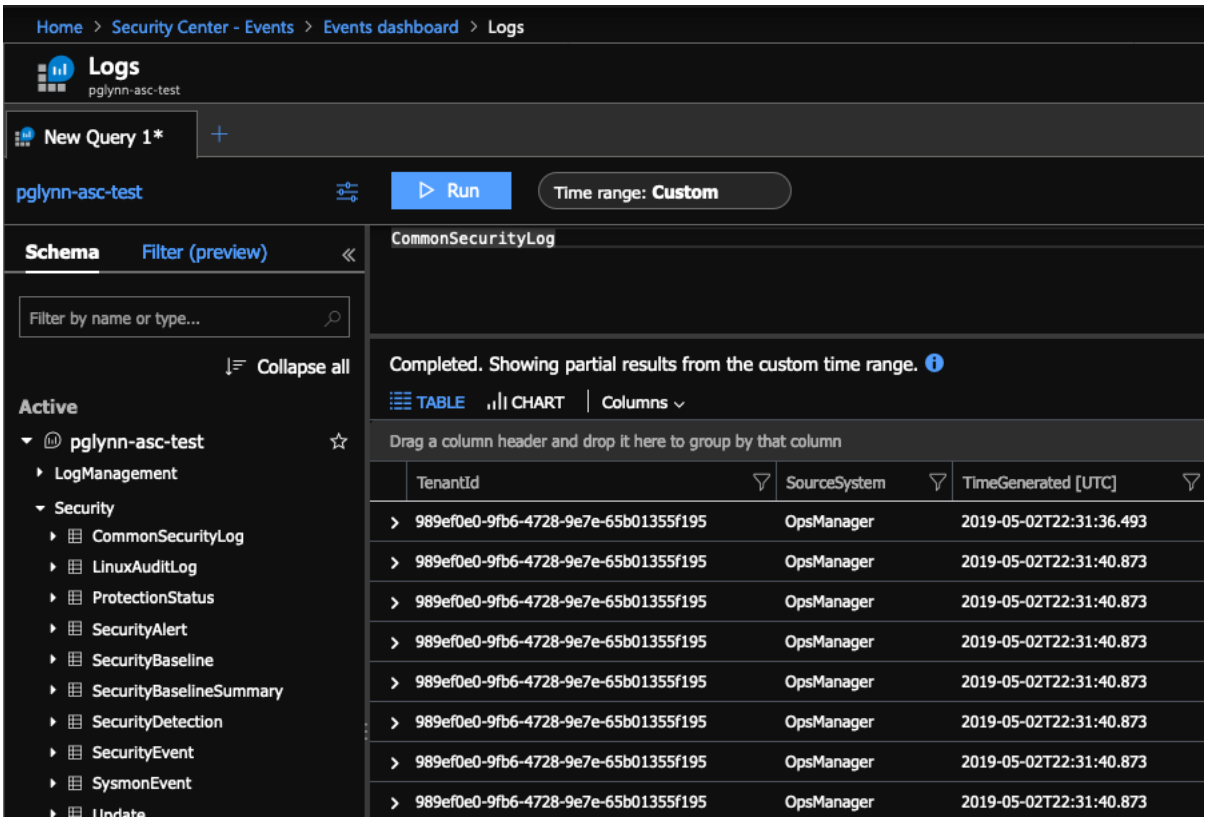
Step 2 Click on Events



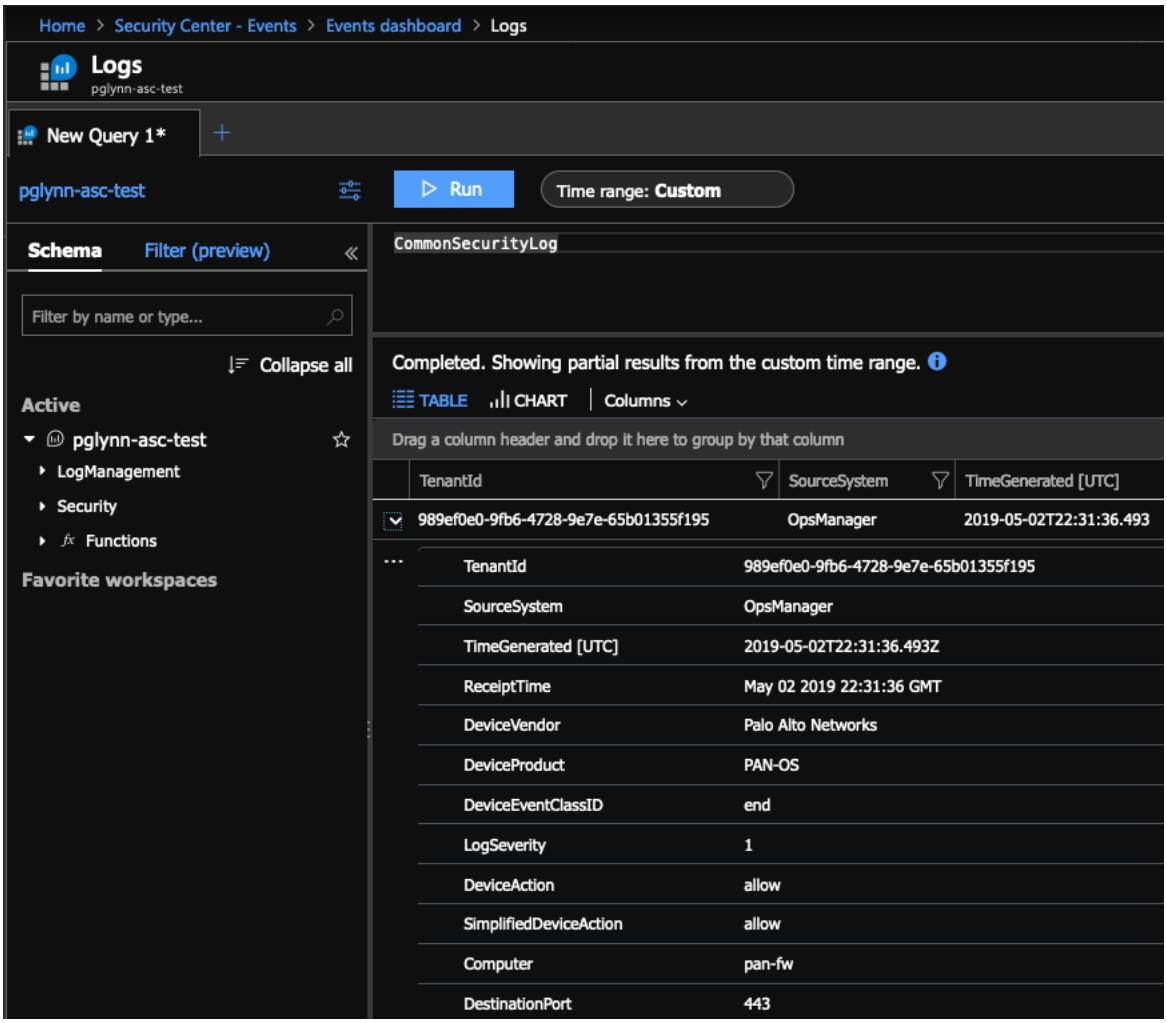
Step 3 Click on the Log Analytics Workspace name that was created earlier



Step 4 The logs will show up in the CommonSecurityLog



Step 5 Select an individual log to view the details:



Troubleshooting

Overview

Occasionally, the logs will fail to show up in ASC. When this happens, there are a number of places to start looking.

Process Flow

After verifying that the firewall is correctly configured to send generated logs to the syslog receiver, check that the messages are arriving at the receiver and being properly processed.

Procedure 6: Verify Logging

Step 1 Login to the syslog receiver and verify that syslog messages are arriving from the firewall by performing a packet capture. The source IP address should match that of the firewall.

```
Last login: Thu May  2 12:56:48 2019 from 104.219.139.2
pglynn@pglynn-asc-oms:~$ sudo su -
root@pglynn-asc-oms:~# tcpdump -nti etho 'port 514'
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on etho, link-type EN10MB (Ethernet), capture size 262144 bytes
IP 10.5.0.4.49387 > 10.5.0.5.514: SYSLOG local4.error, length: 384
IP 10.5.0.4.49387 > 10.5.0.5.514: SYSLOG local4.info, length: 441
IP 10.5.0.4.49387 > 10.5.0.5.514: SYSLOG local4.info, length: 442
IP 10.5.0.4.49387 > 10.5.0.5.514: SYSLOG local4.info, length: 401
IP 10.5.0.4.49387 > 10.5.0.5.514: SYSLOG local4.info, length: 414
IP 10.5.0.4.49387 > 10.5.0.5.514: SYSLOG local4.info, length: 426
IP 10.5.0.4.49387 > 10.5.0.5.514: SYSLOG local4.info, length: 406
IP 10.5.0.4.49387 > 10.5.0.5.514: SYSLOG local4.info, length: 384
IP 10.5.0.4.49387 > 10.5.0.5.514: SYSLOG local4.info, length: 381
IP 10.5.0.4.49387 > 10.5.0.5.514: SYSLOG local4.info, length: 387
IP 10.5.0.4.49387 > 10.5.0.5.514: SYSLOG local4.info, length: 387
IP 10.5.0.4.49387 > 10.5.0.5.514: SYSLOG local4.info, length: 411
IP 10.5.0.4.49387 > 10.5.0.5.514: SYSLOG local4.info, length: 384
IP 10.5.0.4.49387 > 10.5.0.5.514: SYSLOG local4.info, length: 378
IP 10.5.0.4.49387 > 10.5.0.5.514: SYSLOG local4.info, length: 384
IP 10.5.0.4.49387 > 10.5.0.5.514: SYSLOG local4.info, length: 431
IP 10.5.0.4.49387 > 10.5.0.5.514: SYSLOG local4.info, length: 431
^C
17 packets captured
28 packets received by filter
11 packets dropped by kernel
root@pglynn-asc-oms:~#
```

Step 2 Perform a packet capture on the loopback interface of the syslog receiver to ensure that the packets are being forwarded to the log processing service. The traffic should arrive on the port specified in the syslog configuration (25226 by default).

```
root@pglynn-asc-oms:~# tcpdump -nti lo 'port 25226'
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on lo, link-type EN10MB (Ethernet), capture size 262144 bytes
IP 127.0.0.1.47286 > 127.0.0.1.25226: UDP, length 1237
IP 127.0.0.1.47286 > 127.0.0.1.25226: UDP, length 1237
IP 127.0.0.1.47286 > 127.0.0.1.25226: UDP, length 1237
IP 127.0.0.1.47286 > 127.0.0.1.25226: UDP, length 1237
IP 127.0.0.1.47286 > 127.0.0.1.25226: UDP, length 1189
IP 127.0.0.1.47286 > 127.0.0.1.25226: UDP, length 1227
IP 127.0.0.1.47286 > 127.0.0.1.25226: UDP, length 1190
IP 127.0.0.1.47286 > 127.0.0.1.25226: UDP, length 1228
IP 127.0.0.1.47286 > 127.0.0.1.25226: UDP, length 1190
IP 127.0.0.1.47286 > 127.0.0.1.25226: UDP, length 1228
IP 127.0.0.1.47286 > 127.0.0.1.25226: UDP, length 1190
IP 127.0.0.1.47286 > 127.0.0.1.25226: UDP, length 1228
```

Step 3 Review the agent log created on the syslog receiver. It is located at /var/opt/microsoft/omsagent/< Workspace ID >/log/omsagent.log

The agent should create an entry at regular intervals for the heartbeat, when it matches (or fails to match) logs, and any other condition of interest.

```
root@pglynn-asc-oms:~# tail -100 /var/opt/microsoft/omsagent/989ef0e0-9fb
      tag oms.update_run_progress.log
      path /var/opt/microsoft/omsagent/log/urp.log
      pos_file /var/opt/microsoft/omsagent/log/urp.log.pos
      format json
   </source>
   <source>
      type exec
      tag heartbeat.output
      command echo > /dev/null
      format tsv
      keys severity,message
      run_interval 20m
   </source>
   <match oms.blob.**>
      type out_oms_blob
      log_level info
      num_threads 5
      omsadmin_conf_path /etc/opt/microsoft/omsagent/989ef0e0-9fb6-4728-9e7
      cert_path /etc/opt/microsoft/omsagent/989ef0e0-9fb6-4728-9e7e-65b0135
      key_path /etc/opt/microsoft/omsagent/989ef0e0-9fb6-4728-9e7e-65b01355
      buffer_chunk_limit 10m
      buffer_type file
      buffer_path /var/opt/microsoft/omsagent/989ef0e0-9fb6-4728-9e7e-65b01
      buffer_queue_limit 10
      buffer_queue_full_action drop_oldest_chunk
      flush_interval 60s
      retry_limit 10
      retry_wait 30s
      max_retry_wait 9m
   </match>
   <match oms.** docker.**>
      type out_oms
      log_level info
      num_threads 5
      omsadmin_conf_path /etc/opt/microsoft/omsagent/989ef0e0-9fb6-4728-9e7
      cert_path /etc/opt/microsoft/omsagent/989ef0e0-9fb6-4728-9e7e-65b0135
      key_path /etc/opt/microsoft/omsagent/989ef0e0-9fb6-4728-9e7e-65b01355
      buffer_chunk_limit 5m
      buffer_type file
```