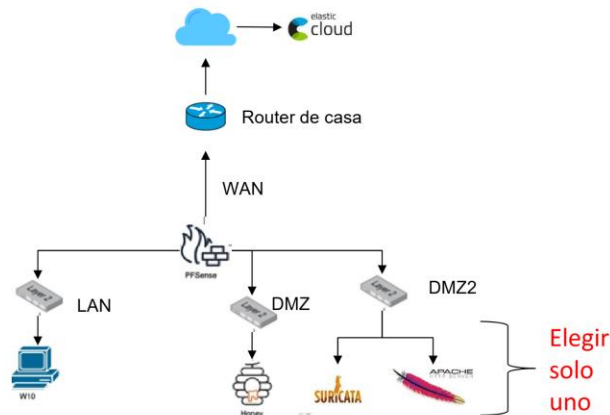


Cybersecurity Blue Team Report



Enunciado

Queremos montar la siguiente infraestructura:



Los requisitos que debe cumplir son los siguientes:

1. Debe tener un PfSense en que se interconecten las redes LAN, DMZ y DMZ2
2. En la red LAN debe haber un equipo Windows 11 que envíe logs al servidor de Elastic.
3. En la red DMZ debe haber un honeypot que envíe los logs al servidor de Elastic.
 1. Este honeypot no debe tener acceso a ninguna red interna (LAN, DMZ2...) y debe ser accesible desde el exterior (red WAN) en ambos sentidos.
4. En la red DMZ2 debe haber otra fuente diferente de logs a las dos mencionadas anteriormente. Se propone Suricata o Apache Server como posibles fuentes, pero se deja a elección del alumno.
5. El servidor de Elastic debe recibir, almacenar y poder visualizar los logs del honeypot, el Windows 11 y la fuente elegida ubicada en la DMZ2.

Criterios de evaluación de la memoria:

1. Debe contener evidencias y explicaciones que demuestren la correcta creación de la infraestructura de red en el PfSense.
2. Debe contener explicación y captura de las reglas de firewall elegidas para cada red (WAN, NAT, LAN, DMZ y DMZ2)
3. Debe contener evidencias de las políticas e integraciones asignadas a cada agente del SIEM (Elastic)
4. Debe contener evidencias que demuestren la correcta recepción de los logs, de todas las fuentes especificadas en el enunciado, en el SIEM (Elastic).

First off, I'm going to demonstrate my pfSense configurations and rules for all internal networks (LAN, DMZ, DMZ2).

1. pfSense

```

FreeBSD/amd64 (UTM.keepcoding.local) (ttyv0)

VirtualBox Virtual Machine - Netgate Device ID: 693fcc246391f50624ea

*** Welcome to pfSense 2.7.2-RELEASE (amd64) on UTM ***

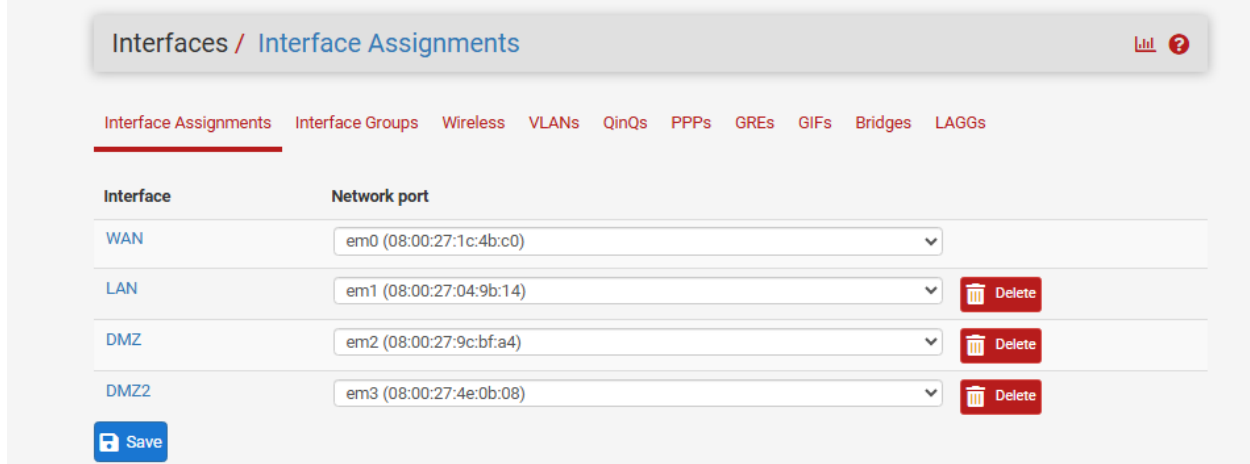
WAN (wan)      -> em0      -> v4/DHCP4: 192.168.1.222/24
                v6/DHCP6: 2a0c:5a87:8401:e500:a00:27ff:fe1c:4b
c0/64
LAN (lan)      -> em1      -> v4: 192.168.100.1/24
DMZ (opt1)     -> em2      -> v4: 192.168.200.1/24
DMZ2 (opt2)    -> em3      -> v4: 192.168.250.1/24

0) Logout (SSH only)          9) pfTop
1) Assign Interfaces          10) Filter Logs
2) Set interface(s) IP address 11) Restart webConfigurator
3) Reset webConfigurator password 12) PHP shell + pfSense tools
4) Reset to factory defaults  13) Update from console
5) Reboot system              14) Enable Secure Shell (sshd)
6) Halt system                 15) Restore recent configuration
7) Ping host                   16) Restart PHP-FPM
8) Shell

Enter an option:

```

Here's the pfSense machine displaying all the internal networks and their assigned IPs. Below this image are the individual configurations for the internal networks, as well as their rules (including port forwarding). The settings "Enable DNSSEC Support" and "Enable Python Module" must be disabled.



Network interfaces are available. This option makes the DNS resolver process recursive queries.

System Domain Transparent
Local Zone Type The local-zone type used for the pfSense system domain (System | General Setup | Domain). Transparent is the default.

DNSSEC ☐ Enable DNSSEC Support

Python Module ☐ Enable Python Module
 Enable the Python Module.

DNS Query Forwarding ☐ Enable Forwarding Mode
 If this option is set, DNS queries will be forwarded to the upstream DNS servers defined under [System > General Setup](#) or those obtained via dynamic interfaces such as DHCP, PPP, or OpenVPN (if DNS Server Override is enabled there).

☐ Use SSL/TLS for outgoing DNS Queries to Forwarding Servers
 When set in conjunction with DNS Query Forwarding, queries to all upstream forwarding DNS servers will be sent using SSL/TLS on the default port of 853. Note that ALL configured forwarding servers MUST support SSL/TLS queries on port 853.

DHCP Registration ☐ Register DHCP leases in the DNS Resolver
 If this option is set, then machines that specify their hostname when requesting an IPv4 DHCP lease will be registered in the DNS Resolver so that their name can be resolved. Note that this will cause the Resolver to reload and flush its resolution cache whenever a DHCP lease is issued. The domain in [System > General Setup](#) should also be set to the proper value.

Static DHCP ☐ Register DHCP static mappings in the DNS Resolver
 If this option is set, then DHCP static mappings will be registered in the DNS Resolver, so that their name can be resolved. The domain in [System > General Setup](#) should also be set to the proper value.

OpenVPN Clients ☐ Register connected OpenVPN clients in the DNS Resolver
 If this option is set, then the common name (CN) of connected OpenVPN clients will be registered in the DNS Resolver, so that their name can be resolved. This only works for OpenVPN servers (Remote Access SSL/TLS or User Auth with Username as Common Name option) operating in "tun" mode. The domain in [System > General Setup](#) should also be set to

Firewall / NAT / Port Forward ?

Port Forward 1:1 Outbound NPT

Rules												
			Interface	Protocol	Source Address	Source Ports	Dest. Address	Dest. Ports	NAT IP	NAT Ports	Description	Actions
<input type="checkbox"/>	<input checked="" type="checkbox"/>		WAN	TCP	*	*	WAN address	22 (SSH)	DMZ address	22 (SSH)	SSH - honeypot/cowrie	
<input type="checkbox"/>	<input checked="" type="checkbox"/>		WAN	TCP	*	*	WAN address	80 (HTTP)	192.168.1.98	80 (HTTP)	apache server	

Legend
 Pass
 Linked rule

The port "22" must be open for cowrie to be able to connect via SSH to the outside.

DMZ (Honeypot) must not be able to access LAN or DMZ2, while still having access to WAN both ways. The following screenshots are the configurations for each interface's firewall rules to make sure this works.

Firewall / Rules / WAN

Floating WAN LAN DMZ DMZ2

Rules (Drag to Change Order)

<input type="checkbox"/>	States	Protocol	Source	Port	Destination	Port	Gateway	Queue	Schedule	Description	Actions
<input type="checkbox"/>	0/0 B	IPv4 *	WAN subnets	*	DMZ subnets	*	*	none		Allow WAN to DMZ	
<input type="checkbox"/>	0/0 B	IPv4 TCP	*	*	192.168.1.98	80 (HTTP)	*	none		NAT apache server	
<input type="checkbox"/>	0/0 B	IPv4 TCP	*	*	DMZ address	22 (SSH)	*	none		NAT SSH - honeypot/cowrie	

Add Add Delete Toggle Copy Save Separator

Firewall / Rules / LAN





































Floating WAN LAN DMZ DMZ2








Rules (Drag to Change Order)

<input type="checkbox"/>	States	Protocol	Source	Port	Destination	Port	Gateway	Queue	Schedule	Description	Actions
<input checked="" type="checkbox"/>	0/3.29 MiB	*	*	*	LAN Address	443 80	*	*		Anti-Lockout Rule	
<input type="checkbox"/>	0/268 B	IPv4 *	LAN subnets	*	DMZ subnets	*	*	none		Block LAN to DMZ	
<input type="checkbox"/>	20/725.05 MiB	IPv4 *	LAN subnets	*	*	*	*	none		Default allow LAN to any rule	
<input type="checkbox"/>	0/0 B	IPv6 *	LAN subnets	*	*	*	*	none		Default allow LAN IPv6 to any rule	

























Add Add Delete Toggle Copy Save Separator








Floating WAN LAN **DMZ** DMZ2

Rules (Drag to Change Order)											
<input type="checkbox"/>	States	Protocol	Source	Port	Destination	Port	Gateway	Queue	Schedule	Description	Actions
<input type="checkbox"/>	 0/504 B	IPv4 *	DMZ subnets	*	LAN subnets	*	*	none		Block DMZ to LAN	    
<input type="checkbox"/>	 0/504 B	IPv4 *	DMZ subnets	*	DMZ2 subnets	*	*	none		Block DMZ to DMZ2	    
<input type="checkbox"/>	 0/0 B	IPv4 *	DMZ subnets	*	WAN subnets	*	*	none		Allow DMZ to WAN	    
<input type="checkbox"/>	 0/0 B	IPv4 ICMP echoreq	*	*	*	*	*	none			    
<input type="checkbox"/>	 0/234 KiB	IPv4 UDP	*	*	*	53 (DNS)	*	none		DNS traffic rule	    
<input type="checkbox"/>	 0/4.11 MiB	IPv4 TCP	*	*	*	Web	*	none		web traffic rule	    

 Add
  Add
  Delete
  Toggle
  Copy
  Save
  Separator

 Floating WAN LAN DMZ **DMZ2**

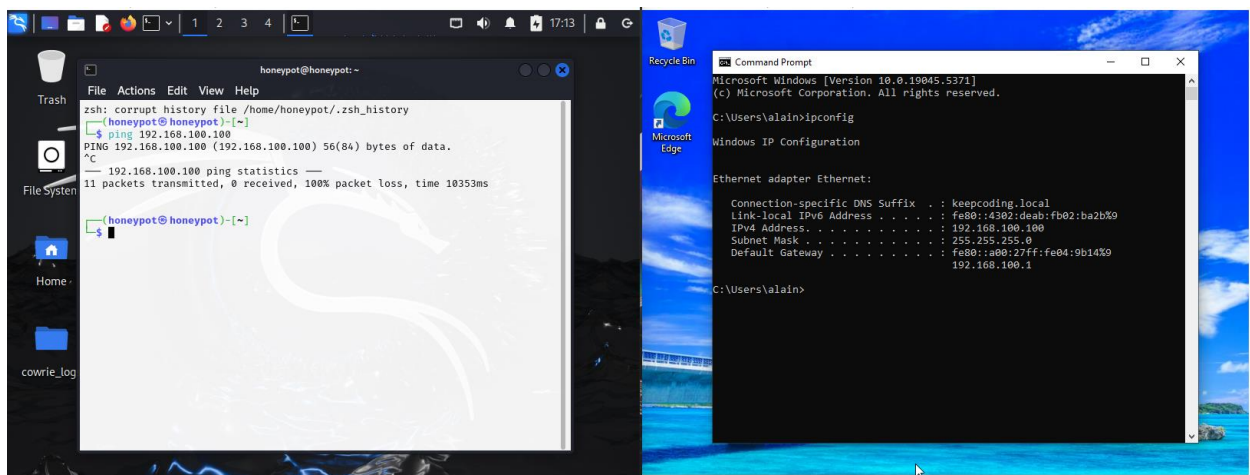
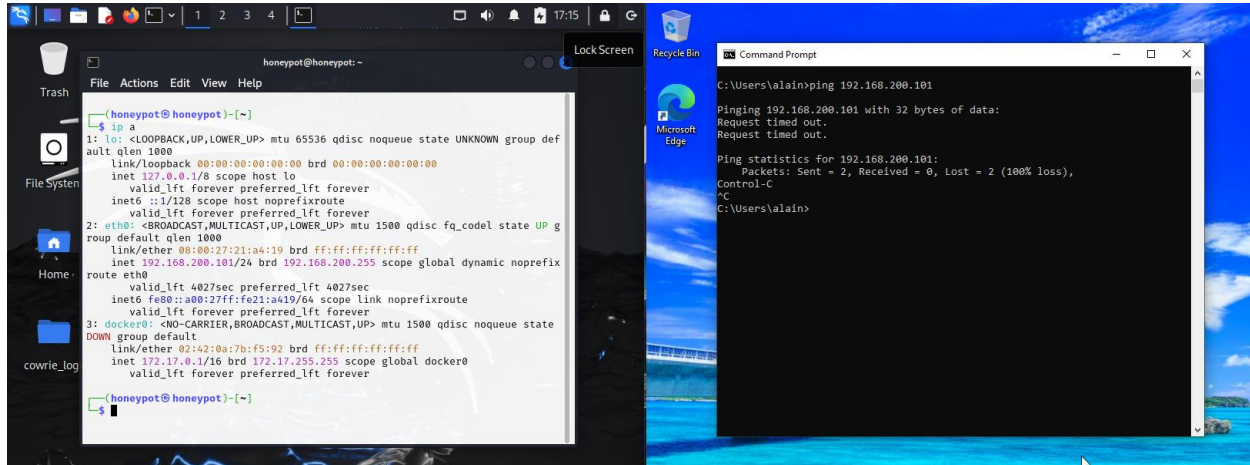
Rules (Drag to Change Order)											
<input type="checkbox"/>	States	Protocol	Source	Port	Destination	Port	Gateway	Queue	Schedule	Description	Actions
<input type="checkbox"/>	 0/252 B	IPv4 *	DMZ2 subnets	*	DMZ subnets	*	*	none		Block DMZ2 to DMZ	    
<input type="checkbox"/>	 0/0 B	IPv4 ICMP echoreq	*	*	*	*	*	none			    
<input type="checkbox"/>	 0/283 KiB	IPv4 UDP	*	*	*	53 (DNS)	*	none			    
<input type="checkbox"/>	 0/4.46 MiB	IPv4 TCP	*	*	*	Web	*	none		DNS traffic rule	    

 Add
  Add
  Delete
  Toggle
  Copy
  Save
  Separator



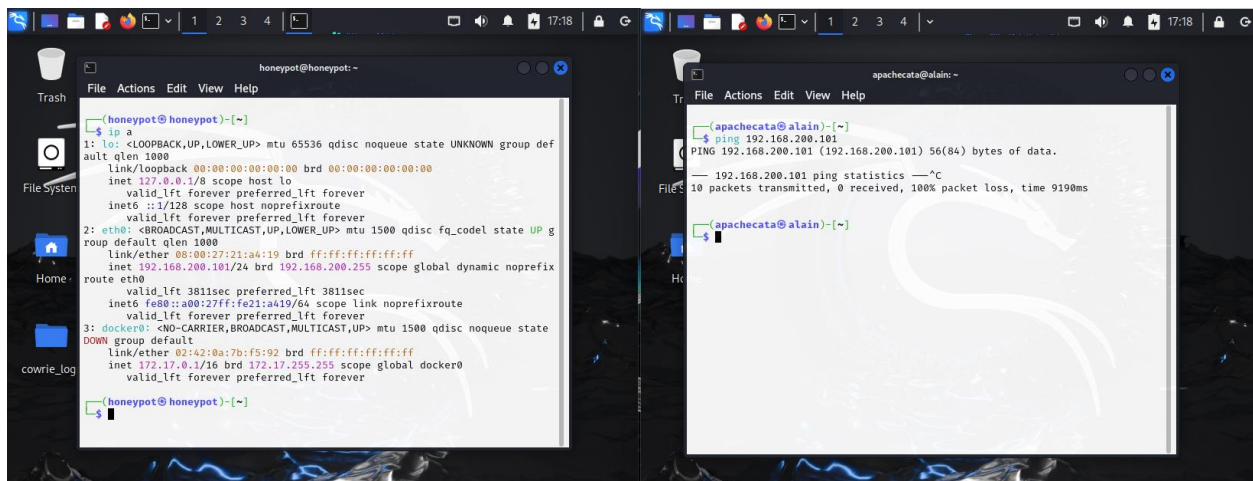
These are the firewall rules for each interface. DMZ and DMZ2 will have rules that block each other using their subnets. Same thing goes for DMZ and LAN. DMZ and WAN both share settings that allow communication to each other back and forth.

Here are screenshots displaying the rules in practice, showing that their communication between each other is indeed blocked by the firewall rules put in place. I used ping commands to present this example.



These two screenshots above are showing DMZ's inability to communicate with LAN (the Windows machine), and vice versa. When DMZ (Honeypot) attempts to ping 192.168.100.100 (the LAN machine's assigned IP), the

connection times out. The same thing occurs when the LAN machine attempts to ping the DMZ machine back.

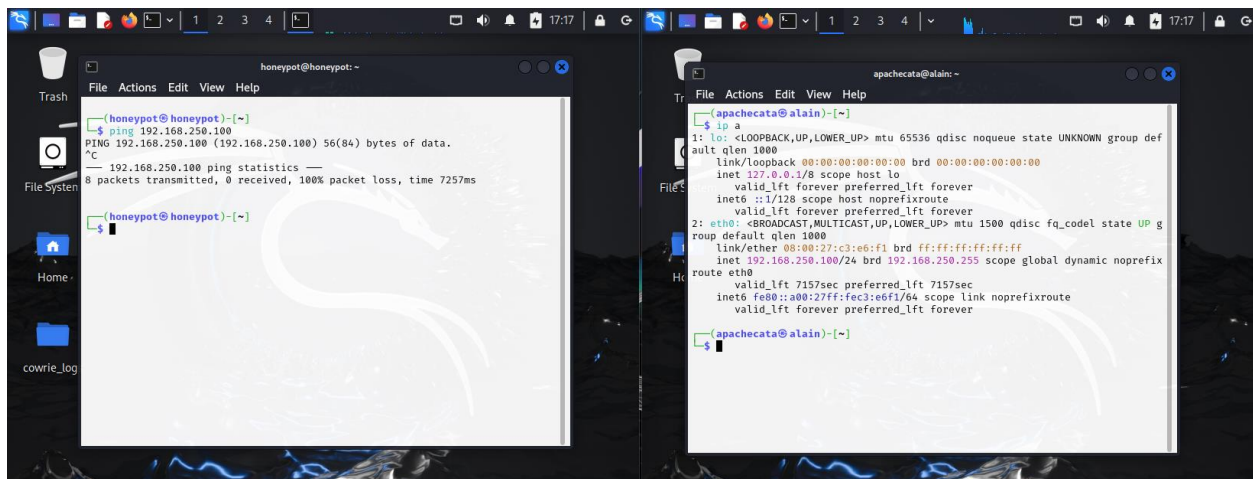


The image shows two terminal windows side-by-side. The left window is titled 'honeypot@honeypot:~' and displays the output of the 'ip a' command, showing network interfaces 'lo' and 'eth0'. The right window is titled 'apachecata@alain:~' and shows the output of a 'ping' command to 192.168.200.101, which results in a 100% packet loss and a time of 9190ms.

```
honeypot@honeypot:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group def
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP g
    link/ether 08:00:27:21:a4:19 brd ff:ff:ff:ff:ff:ff
    inet 192.168.200.101/24 brd 192.168.200.255 scope global dynamic noprefix
        valid_lft 3811sec preferred_lft 3811sec
    inet6 fe80::a00:27ff:fe21:a419/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
3: docker0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc noqueue state
    link/ether 02:42:0a:7b:f5:92 brd ff:ff:ff:ff:ff:ff
    inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0
        valid_lft forever preferred_lft forever

honeypot@honeypot:~$

apachecata@alain:~$ ping 192.168.200.101
PING 192.168.200.101 (192.168.200.101) 56(84) bytes of data.
--- 192.168.200.101 ping statistics ---
10 packets transmitted, 0 received, 100% packet loss, time 9190ms
```



The image shows two terminal windows side-by-side. The left window is titled 'honeypot@honeypot:~' and displays the output of a 'ping' command to 192.168.250.100, which results in a 100% packet loss and a time of 7257ms. The right window is titled 'apachecata@alain:~' and displays the output of the 'ip a' command, showing network interfaces 'lo' and 'eth0'.

```
honeypot@honeypot:~$ ping 192.168.250.100
PING 192.168.250.100 (192.168.250.100) 56(84) bytes of data.
--- 192.168.250.100 ping statistics ---
8 packets transmitted, 0 received, 100% packet loss, time 7257ms

honeypot@honeypot:~$

apachecata@alain:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group def
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP g
    link/ether 08:00:27:c3:e6:f1 brd ff:ff:ff:ff:ff:ff
    inet 192.168.250.100/24 brd 192.168.250.255 scope global dynamic noprefix
        valid_lft 7157sec preferred_lft 7157sec
    inet6 fe80::a00:27ff:fec3:e6f1/64 scope link noprefixroute
        valid_lft forever preferred_lft forever

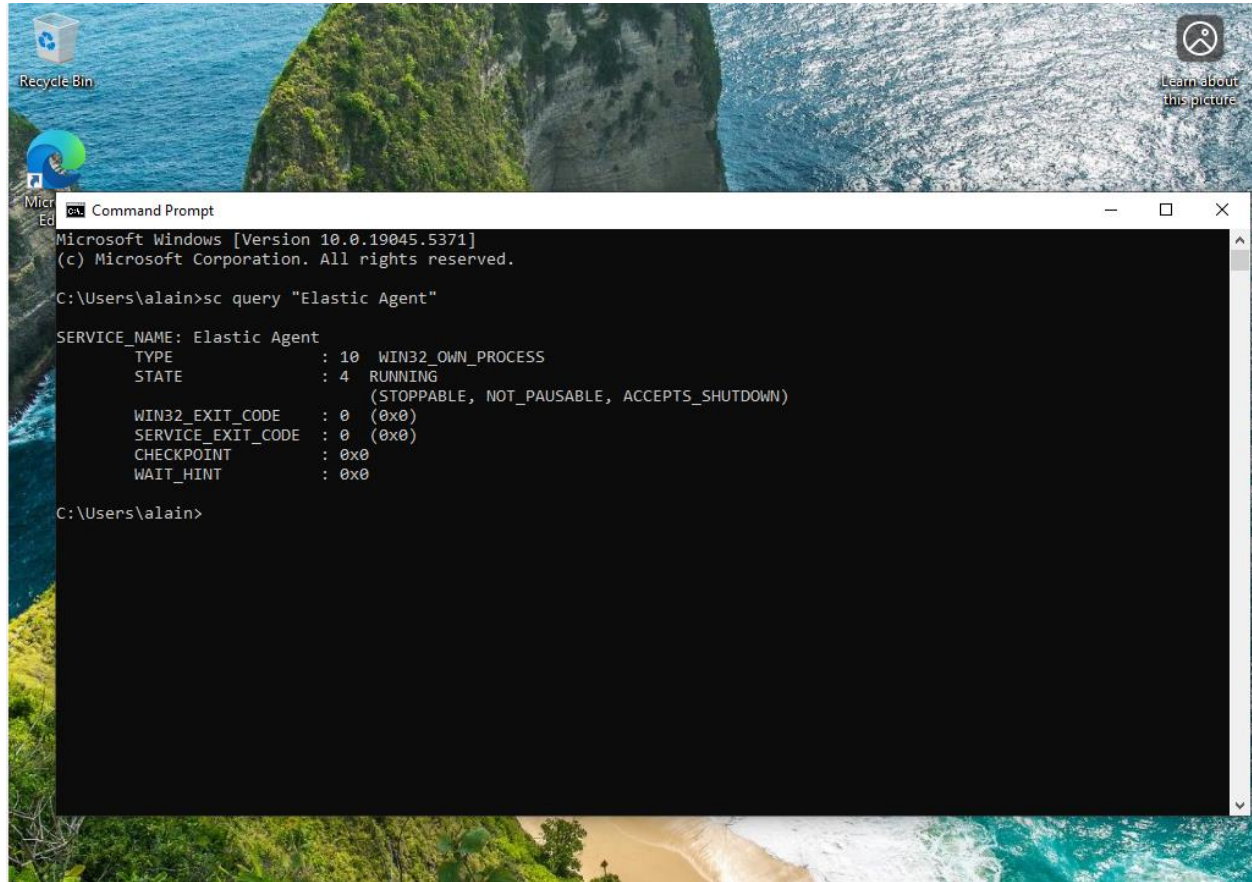
apachecata@alain:~$
```

These two machines (DMZ and DMZ2) which are both running on Kali-Linux, are running the Honeybot and Suricata. The machine on the left is the honeypot, while the one on the right is Suricata. You can see that when DMZ attempts to ping DMZ2's address (192.168.250.100) the connection times out due to the firewall rules put in place. The same thing happens in reverse.

2. LAN (Windows)

After installing Windows 10 on my virtual machine and assigning it its IP from pfSense, I created the agent policy for the machine on Elastic Cloud, added the Windows integration, and installed Elastic Agent on the Windows machine to connect it to my Elastic Cloud. This was done with the

integration's given command to run in Windows PowerShell with Administrator permissions. Below is a screenshot displaying the existence of Elastic Agent on the Windows machine, and its status as "Running".



[View all agent policies](#)

Revision

4

Integrations

2

Agents

1 agent

Last updated on

Jan 16, 2025

Actions

▼

Windows

Integrations

Settings

Q Search...

Namespace ▼

⊕ Add integration

Integration policy ↑	Integration ⇅	Namespace	Output	Actions
system-2	System v1.63.2	default	Default output ⓘ	⋮
windows-1	Windows v2.3.6	default ⓘ	Default output ⓘ	⋮

All logs and excerpts successfully ingested and transported to Elastic Cloud will be displayed at the end of the report underneath the Elastic section.

3.DMZ (Honeypot)

The DMZ internal network is on a Kali-Linux machine running the Honeypot. In order to achieve this, I installed Cowrie using GitHub and ran it with a virtual environment. This method is far more consistent for being able to access its logs, as while as being able to successfully transfer them to Elastic Cloud using Elastic Cloud’s “Custom Logs” integration. I installed Filebeat as well in order for the logs to be properly transferred to Elastic Cloud. The screenshots below are the installation process, as well as starting the Cowrie service.

(cowrie-env)honeypot@honeypot: ~/cowrie

File

Actions

Edit

View

Help

(honeypot@honeypot)-[~]

\$ git clone https://github.com/cowrie/cowrie.git

Cloning into 'cowrie'...

remote: Enumerating objects: 18681, done.

remote: Counting objects: 100% (2127/2127), done.

remote: Compressing objects: 100% (374/374), done.

remote: Total 18681 (delta 1990), reused 1753 (delta 1753), pack-reused 16554 (from 3)

Receiving objects: 100% (18681/18681), 10.43 MiB | 10.57 MiB/s, done.

Resolving deltas: 100% (13062/13062), done.

```
(honeypot@honeypot)-[~/cowrie]
$ python3 -m venv cowrie-env

(honeypot@honeypot)-[~/cowrie]
$ source cowrie-env/bin/activate
```

```
GNU nano 8.3 cowrie.cfg
# (default: not specified)
#sensor_name=myhostname

# Hostname for the honeypot. Displayed by the shell prompt of the virtual
# environment
#
# (default: svr04)
hostname = svr04

# Directory where to save log files in.
#
# (default: log)
log_path = /var/log/cowrie

# Directory where to save downloaded artifacts in.
#
# (default: downloads)
download_path = ${honeypot:state_path}/downloads

# Directory for static data files

^G Help      ^O Write Out  ^F Where Is   ^K Cut        ^T Execute
^X Exit      ^R Read File  ^\ Replace    ^U Paste      ^J Justify
```

```
(honeypot@honeypot)-[~/cowrie]
$ ./bin/cowrie start

Join the Cowrie community at: https://www.cowrie.org/slack/

Using default Python virtual environment "/home/honeypot/cowrie/cowrie-env"
Starting cowrie: [twisted --umask=0022 --pidfile=var/run/cowrie.pid --logger
cowrie.python.logfile.logger cowrie ]...
/home/honeypot/cowrie/cowrie-env/lib/python3.12/site-packages/twisted/conch/s
sh/transport.py:105: CryptographyDeprecationWarning: TripleDES has been moved
to cryptography.hazmat.decrepit.ciphers.algorithms.TripleDES and will be rem
oved from cryptography.hazmat.primitives.ciphers.algorithms in 48.0.0.
b"3des-cbc": (algorithms.TripleDES, 24, modes.CBC),
/home/honeypot/cowrie/cowrie-env/lib/python3.12/site-packages/twisted/conch/s
sh/transport.py:112: CryptographyDeprecationWarning: TripleDES has been moved
to cryptography.hazmat.decrepit.ciphers.algorithms.TripleDES and will be rem
oved from cryptography.hazmat.primitives.ciphers.algorithms in 48.0.0.
b"3des-ctr": (algorithms.TripleDES, 24, modes.CTR),
```

Here are screenshots for the integration to Elastic Cloud, as well as the installation process for Filebeat and configuration for the log path.

```
(honeypot@honeypot)-[~]
$ curl -L -O https://artifacts.elastic.co/downloads/beats/filebeat/filebeat-8.0.0-amd64.deb
% Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
           nt                                 Dload  Upload   Total   Spent    Left   Speed
  0     0    0     0    0     0      0      0  --:--:-- --:--:-- --:--:-- 
 59 34.2M   59 20.4M    0     0  10.4M      0  0:02:54 0:00:01 0:02:53 201k
100 34.2M  100 34.2M    0     0  13.9M      0  0:00:02 0:00:02 --:--:-- 14.0M

(honeypot@honeypot)-[~]
$ sudo dpkg -i filebeat-8.0.0-amd64.deb
Selecting previously unselected package filebeat.
(Reading database ... 418214 files and directories currently installed.)
Preparing to unpack filebeat-8.0.0-amd64.deb ...
Unpacking filebeat (8.0.0) ...
Setting up filebeat (8.0.0) ...
Processing triggers for kali-menu (2024.4.0) ...

# ===== Filebeat inputs =====
filebeat.inputs:
# Each - is an input. Most options can be set at the input level, so
# you can use different inputs for various configurations.
# Below are the input specific configurations.

# filestream is an input for collecting log messages from files.
- type: filestream

# Change to true to enable this input configuration.
enabled: false

# Paths that should be crawled and fetched. Glob based paths.
paths:
  - /var/log/cowrie/*.log
  #- c:\programdata\elasticsearch\logs\*
```

The log path in Filebeat's config file is set to *.log to make sure that any file ending in .log is read for Elastic Cloud. The folder is Cowrie's dedicated folder for Cowrie's logs, so despite the path being set to *.log to read any log file in the folder, it will only read Cowrie's logs due to that being the only log file in the folder. Here's a screenshot showing Elastic Cloud's integration with Honeypot.

[View all agent policies](#)

Revision 4 | Integrations 2 | Agents 1 agent | Last updated on Jan 16, 2025 | [Actions](#)

Honeypot

[Integrations](#) [Settings](#)

Namespace [Add integration](#)

Integration policy ↑	Integration ↕	Namespace	Output	Actions
log-1	Custom Logs v2.3.3	default ⓘ	Default output ⓘ	⋮
system-3	System v1.63.2	default ⓘ	Default output ⓘ	⋮

4.DMZ2 (Suricata)

DMZ2 is also a Kali-Linux machine, but running Suricata instead. The installation was very straight forward. Here’s the installation process and proof of Suricata running successfully. The integration to Elastic Cloud was also very straight forward, due to Elastic Cloud having a native integration for Suricata.

```
apachecata@alain: ~  
File Actions Edit View Help  
  
(apachecata@alain)-[~]  
$ sudo apt-get install suricata  
Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done  
suricata is already the newest version (1:7.0.8-1+b1).  
The following packages were automatically installed and are no longer required:  
  fonts-liberation2 hydra-gtk ibverbs-providers libassuan0 libavfilter9  
  libbftfio1 libboost-iostreams1.83.0 libboost-thread1.83.0 libcephfs2  
  libegl-dev libfmt9 libgail-common libgail18t64 libgeos3.12.2 libgfapi0  
  libgfrpc0 libgfxdr0 libgl1-mesa-dev libgles-dev libgles1 libglusterfs0  
  libglvnd-core-dev libglvnd-dev libgspell-1-2 libgtk2.0-0t64 libgtk2.0-bin  
  libgtk2.0-common libibverbs1 libimobiledevice6 libiniparser1  
  libjim0.82t64 libjsoncpp25 libmbcrypted07t64 libmfx1 libpaper1  
  libperl5.38t64 libplacebo338 libplist3 libpostproc57 librados2  
  librdmacm1t64 libusbmuxd6 libzip4t64 openjdk-17-jre  
  openjdk-17-jre-headless openjdk-23-jre openjdk-23-jre-headless  
  perl-modules-5.38 python3-appdirs python3-hatch-vcs python3-hatchling  
  python3-pathspect python3-pluggy python3-setuptools-scm  
  python3-trove-classifiers rwho rwho  
Use 'sudo apt autoremove' to remove them.  
0 upgraded, 0 newly installed, 0 to remove and 180 not upgraded.  
  
(apachecata@alain)-[~]  
$
```

```
(apachecata@alain)-[~]  
$ sudo systemctl status suricata  
● suricata.service - Suricata IDS/IDP daemon  
   Loaded: loaded (/usr/lib/systemd/system/suricata.service; enabled; pres  
   Active: active (running) since Fri 2025-01-17 16:14:23 CET; 2min 4s ago  
   Invocation: b3946c8fd97141d3b62be269ff5668fc  
     Docs: man:suricata(8)  
           man:suricatasc(8)  
           https://suricata.io/documentation/  
   Process: 791 ExecStart=/usr/bin/suricata -D --af-packet -c /etc/suricata  
   Main PID: 793 (Suricata-Main)  
     Tasks: 10 (limit: 4557)  
    Memory: 85.4M (peak: 87M)  
       CPU: 770ms  
    CGroup: /system.slice/suricata.service  
            └─793 /usr/bin/suricata -D --af-packet -c /etc/suricata/suricat  
  
Jan 17 16:14:23 alain systemd[1]: Starting suricata.service - Suricata IDS/I  
Jan 17 16:14:23 alain suricata[791]: i: suricata: This is Suricata version 7  
Jan 17 16:14:23 alain systemd[1]: Started suricata.service - Suricata IDS/ID  
lines 1-18/18 (END)
```

[Send feedback](#)

[View all agent policies](#)

Revision2

Integrations2

Agents1 agent

Last updated onJan 16, 2025

Actions

Suricata/Linux

IntegrationsSettings

Namespace

Add integration

Integration policy ↑	Integration ↕	Namespace	Output	Actions
suricata-1	Suricata v2.21.4	default ⓘ	Default output ⓘ	⋮
system-1	System v1.63.2	default	Default output ⓘ	⋮

5.Elastic Cloud

The process for Elastic Cloud and connecting each internal network and their services successfully to the site was very efficient and linear. Both Windows and Suricata had their own native integrations to use, and retrieving their logs was a matter of a couple clicks and 1 command each. For Honeypot, since there is no native integration available, the Custom Logs integration is the best option. It was simply giving Elastic Cloud the log file path, and installing Elastic Agent onto the host.

Fleet

Centralized management for Elastic Agents.

Agents Agent policies Enrollment tokens Uninstall tokens Data streams Settings

Ingest Overview MetricsAgent Info Metrics

Agent activityAdd agent

Filter your data using KQL syntax

Status4Tags0Agent policy3Upgrade available

Showing 3 agentsClear filters

Healthy3Unhealthy0Updating0Offline0Inactive0Unenrolled0

Status	Host	Agent policy	CPU	Memory	Last activity	Version	Actions
Healthy	honeypot	Honeypot rev. 4	0.49 %	218 MB	30 seconds ago	8.17.0	
Healthy	DESKTOP-TE6LKJ9	Windows rev. 4	N/A	N/A	14 seconds ago	8.17.0	
Healthy	alain	Suricata/Linux rev. 2	0.72 %	214 MB	26 seconds ago	8.17.0	

Rows per page: 20

This screenshot displays all the machines connected to the Elastic Cloud. Below this will be screenshots of the logs being successfully retrieved, as well as individual logs retrieved from Elastic Cloud to show more in detail.

Windows:

< View all agents

DESKTOP-TE6LKJ9Actions

Agent detailsLogsDiagnostics

Search logs...

DatasetLog levelLast dayOpen in Discover

Timestamp	event.dataset	component.id	Message	error.message
17:20:24.610	elastic_agent	windows/metrics-default	[elastic_agent][info] Unit state changed windows/metrics-default (STARTING->HEALTHY): Healthy	
17:20:24.611	elastic_agent	system/metrics-default	[elastic_agent][info] Unit state changed system/metrics-default (STARTING->HEALTHY): Healthy	
17:20:24.611	elastic_agent	winlog-default	[elastic_agent][info] Unit state changed winlog-default (STARTING->HEALTHY): Healthy	
17:20:24.621	elastic_agent	winlog-default	[elastic_agent][info] Unit state changed winlog-default-winlog-system-4603a46b-8355-4c3b-98ee-a3b103d742b3 (STARTING->HEALTHY): Healthy	
17:20:24.621	elastic_agent	winlog-default	[elastic_agent][info] Unit state changed winlog-default-winlog-windows-e80637d5-70aa-48a3-b986-1cbca80376c1 (STARTING->HEALTHY): Healthy	
17:20:24.659	elastic_agent	log-default	[elastic_agent][info] Unit state changed log-default-logfile-system-4603a46b-8355-4c3b-98ee-a3b103d742b3 (STARTING->HEALTHY): Healthy	
17:20:24.715	elastic_agent	system/metrics-default	[elastic_agent][info] Unit state changed system/metrics-default-system/metrics-system-4603a46b-8355-4c3b-98ee-a3b103d742b3 (STARTING->HEALTHY): Healthy	
17:20:24.888	elastic_agent	windows/metrics-default	[elastic_agent][info] Unit state changed windows/metrics-	

host.os.name.text

Windows 10 Home

@timestamp

Jan 18, 2025 @ 17:15:17.648

agent.ephemeral_id

fa1b7867-cdb4-4653-aacf-245b911bb471

agent.id

fd50d10d-29ce-434a-a0f9-de2f033a7d9d

agent.name

DESKTOP-TE6LKJ9

agent.type

filebeat

agent.version

8.17.0

component.binary

metricbeat

component.dataset

Honeypot:

In order to check if Elastic Cloud was receiving and ingesting Cowrie logs through Filebeat properly, I ran an SSH connection and checked if Elastic Cloud received a log entry related to Honeypot SSH, rather than only reading Filebeat's process. Below is an individual log referencing Honeypot SSH, showing that Elastic Cloud is receiving Cowrie's logs due to Filebeat successfully ingesting Cowrie's logs through the path that was put in place in the configuration file.

[View all agents](#)

honeypot

Actions

Agent details **Logs** Diagnostics

Search logs...

Dataset 1 Log level 4 Last day

Open in Discover

Timestamp	event.dataset	component.id	Message	error.message
16:55:32.005	elastic_agent	log-default	[elastic_agent][info] Spawned new component log-default: Starting: spawned pid '1026'	
16:55:32.005	elastic_agent	log-default	[elastic_agent][info] Spawned new unit log-default-logfile-logs-1c4e7679-084e-4408-95af-cccea9d1bac5: Starting: spawned pid '1026'	
16:55:32.005	elastic_agent	log-default	[elastic_agent][info] Spawned new unit log-default: Starting: spawned pid '1026'	
16:55:32.005	elastic_agent	log-default	[elastic_agent][info] Spawned new unit log-default-logfile-system-191f6184-3a9c-454e-81a3-7997e8aad75: Starting: spawned pid '1026'	
16:55:32.632	elastic_agent		[elastic_agent][info] component model updated	
16:55:32.632	elastic_agent		[elastic_agent][info] Updating running component model	
16:55:32.653	elastic_agent	system/metrics-default	[elastic_agent][info] Spawned new component system/metrics-default: Starting: spawned pid '1095'	
16:55:32.653	elastic_agent	system/metrics-default	[elastic_agent][info] Spawned new unit system/metrics-default-system/metrics-system-191f6184-3a9c-454e-81a3-7997e8aad75: Starting: spawned pid '1095'	
16:55:32.653	elastic_agent	system/metrics-default	[elastic_agent][info] Spawned new unit system/metrics-default: Starting: spawned pid '1095'	
16:55:32.653	elastic_agent	system/metrics-default	[elastic_agent][info] Spawned new unit system/metrics-default: Starting: spawned pid '1095'	

message

2025-01-18T13:21:42.926428Z [-] Ready to accept SSH connections

@timestamp

Jan 18, 2025 @ 13:21:43.147

agent.ephemeral_id

2fdc759d-e0c9-475d-b535-41870574483f

agent.id

0157a5f1-e573-4d50-a5ec-743f5f9ff3d9

agent.name

honeypot

agent.type

filebeat

agent.version

8.17

Suricata:

< View all agents

alain

Actions

Agent details

Logs

Diagnostics

Search logs...

Dataset 1Log level 4Last weekOpen in Discover

Timestamp	event.dataset	component.id	Message	error.message
16:14:28.801	elastic_agent	log-default	[elastic_agent][info] Unit state changed log-default (STARTING->HEALTHY): Healthy	
16:14:28.816	elastic_agent	log-default	[elastic_agent][info] Unit state changed log-default-logfile-system-648686ba-79e8-4a7b-b28c-85ea04c2acef (STARTING->HEALTHY): Healthy	
16:14:28.816	elastic_agent	log-default	[elastic_agent][info] Unit state changed log-default-logfile-suricata-b1598dcf-2067-49e4-8943-82cbb5581a72 (STARTING->HEALTHY): Healthy	
16:14:28.816	elastic_agent	log-default	[elastic_agent][info] Unit state changed log-default-logfile-system-648686ba-79e8-4a7b-b28c-85ea04c2acef (STARTING->HEALTHY): Healthy	
16:14:28.816	elastic_agent	log-default	[elastic_agent][info] Unit state changed log-default-logfile-suricata-b1598dcf-2067-49e4-8943-82cbb5581a72 (STARTING->HEALTHY): Healthy	
16:14:28.869	elastic_agent	system/metrics-default	[elastic_agent][info] Unit state changed system/metrics-default (STARTING->HEALTHY): Healthy	
16:14:28.869	elastic_agent	system/metrics-default	[elastic_agent][info] Unit state changed system/metrics-default (STARTING->HEALTHY): Healthy	
16:14:28.900	elastic_agent	system/metrics-default	[elastic_agent][info] Unit state changed system/metrics-default (STARTING->HEALTHY): Healthy	

@timestamp

Jan 17, 2025 @ 16:17:16.350

agent.ephemeral_id

cf184e04-68c8-4bf8-a49d-5594a8e160a0

agent.id

7bcc064f-4827-4937-9ee9-4f9af5a71b01

agent.name

alain

agent.type

filebeat

agent.version

8.17.0

data_stream.dataset

suricata.eve

data_stream.namespace

End of Report