

Authors: Olajuwon Olawale Faleke and  
Philip Jeremiah Kadiri

Preforming network intrusion detection  
using Microsoft azure and open-source  
tools

25/10/2022

# Table of contents

- 1. List of Figures
- 2. Introduction
- 3. Technical Report
  - Setting up an Azure Virtual Machine
  - Downloading PCAP Files for Network Analysis
  - Suricata
    - Installing Suricata
    - Configuring Suricata
    - Testing Suricata
    - Checking QakBot Traffic with Suricata
  - Setting up the Elastic Stack
    - Elasticsearch
      - Installing Elasticsearch
      - Configuring Elasticsearch
      - Testing Elasticsearch
    - Logstash
      - Installing Logstash
      - Configuring Logstash
      - Testing Logstash
    - Kibana
      - Installing Kibana
      - Testing Kibana
      - Creating the Kibana Dashboard
      - Visualizing Alerts in Kibana
      - Analyzing Alerts in Kibana
- 4. Recommendations
- 5. Personal Reflection
- 6. Conclusion
- 7. References
- 8. Appendices
  - Appendix A: Setting up an Azure Virtual Machine
    - Registration of Free Microsoft Azure Account
    - Creating the Virtual Machine

# List of Figures

- **Figure 1:** Setting Permissions for SSH Keys
- **Figure 2:** Successful Login to Azure Security Audit Virtual Machine
- **Figure 3:** Successful Update
- **Figure 4** Shows successful installation of Nginx and access to Azure Server
- **Figure 5:** PCAP files
- **Figure 6:** Successful Installation of Suricata
- **Figure 7:** Enabling and verifying status of Suricata
- **Figure 8:** Specifying subnet in Suricata's configuration file
- **Figure 9:** Updating network interface in Suricata's config
- **Figure 10:** Enabling community-id in Suricata's config
- **Figure 11:** Default rule files in Suricata
- **Figure 12:** Updating Suricata
- **Figure 13:** Disabling the MQTT app-layer protocol
- **Figure 14:** Enabling the RDP app-layer protocol
- **Figure 15:** Disabling the SIP app-layer protocol
- **Figure 16:** Successful update of Suricata without app-layer errors
- **Figure 17:** Adding two new source, updating, and testing Suricata configurations
- **Figure 18:** Suricata configuration loaded successfully
- **Figure 19:** Running status for Suricata
- **Figure 20:** Suricata log files
- **Figure 21:** Testing NID detection
- **Figure 22:** Suricata detects information leak attempt on the network 1
- **Figure 23:** Suricata logs outbound destination IP
- **Figure 24:** *QakBot Infection Stages Overview*
- **Figure 25:** Parsing QakBot traffic into Suricata
- **Figure 26:** QakBot malicious `.zip` file downloads (one)
- **Figure 27:** QakBot malicious `.zip` file downloads (two)
- **Figure 28:** Elastic Stack support matrix
- **Figure 29:** Java not found on system
- **Figure 30:** Java version and home path
- **Figure 31:** Elasticsearch installation
- **Figure 32:** Elasticsearch running status
- **Figure 33:** Modifying Elasticsearch configuration
- **Figure 34:** Successful installation of Logstash
- **Figure 35:** Logstash pipeline configuration 1
- **Figure 36:** Logstash pipeline configuration 2
- **Figure 37:** Logstash pipeline configuration 3

- **Figure 38:** Logstash pipeline configuration 4
- **Figure 39:** Logstash running successfully
- **Figure 40:** Kibana running successfully
- **Figure 41:** Setting up inbound port rules for Kibana on Azure
- **Figure 42:** Kibana management screen
- **Figure 43:** ELK/B Stack functionality
- **Figure 44:** Access to create index-pattern after Suricata PCAP ingestion
- **Figure 45:** Created `logstash-*` pattern
- **Figure 46:** Sample Suricata Alert Dashbort import
- **Figure 47:** Alert summary search import
- **Figure 48:** Visualization imports
- **Figure 49:** Bar chart showing the year and date of QakBot malware attack
- **Figure 50:** Alert by Geo-IP
- **Figure 51:** Top 10 Alerts
- **Figure 52:** Number of Alerts
- **Figure 53:** Alerts by `src-dest-IP` and `src-dest-port`
- **Figure 54:** Alert Summary
- **Figure 55:** Timeline graph
- **Figure 56:** Potential Second Stage Download and DLL Windows file download HTTP
- **Figure 57:** HTTP Headers for Downloaded File
- **Figure 58:** Alerts informing usage of a Malware Anti-Degubbung EXE
- **Figure 59:** Malicious DLL FIle and Potential **QakBot** malware
- **Figure 60:** Potential Execution of CnC Server (QakBot DLL)
- **Figure 61:** QakBot (Banking Trojan) Signature in Kibana
- **Figure 62:** Geolocation Data of Host the Executed Malware
- **Figure 63:** Complete Registration of Free Microsoft Azure Account.
- **Figure 64:** Azure Machine - Project Details
- **Figure 65:** Azure VM Account settings and virtual machine memory size
- **Figure 66:** Virtual Machine Disk Size Allocation
- **Figure 67:** Basic Settings
- **Figure 68:** Disks and Networking Settings
- **Figure 69:** Management, Monitoring, and Advanced Settings
- **Figure 70:** Generating the SSH Key Pairs
- **Figure 71:** Complete Deployment of Azure VM
- **Figure 72:** Virtual Machine Properties
- **Figure 73:** Virtual Machine Capabilities
- **Figure 74:** VM Essentials

# **Introduction**

This experiment will practically demonstrate how to setup and configure: a logging and data visualization stack called the Elastic Stack v5.2.0, and how to setup, configure, and parse an intrusion detection system's (Suricata) JSON log output into that stack. The experiment will also review the QakBot malware PCAP traffic, analyze the infection process using the aforementioned open source tools, suggest recommendations, and document the reflection on the procedure.

# Technical Report

This technical report contains the configuration, setup, and use of the Elastic Stack and Suricata. Additionally, it aims to analyze the QakBot malware infection process and any other malicious traffic.

## Setting up an Azure Virtual Machine

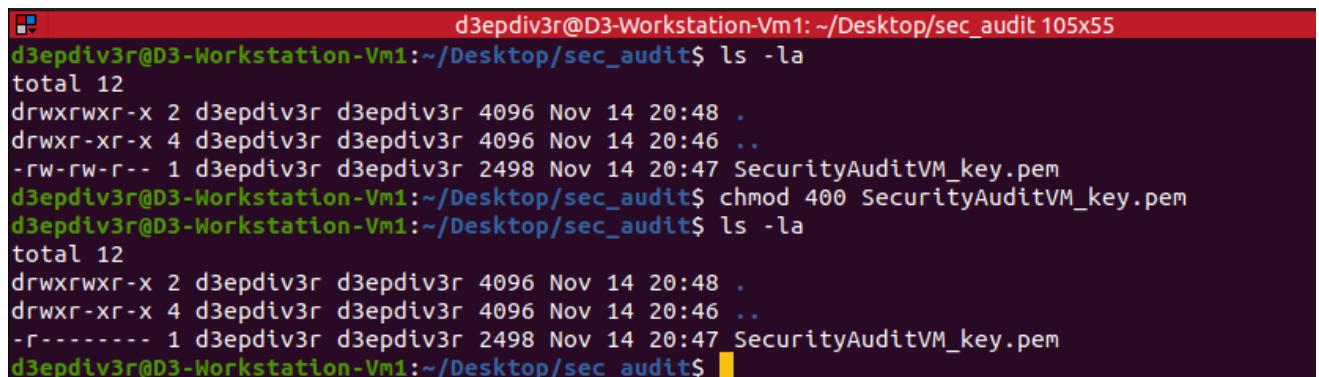
A free account has to be created in order to utilize Azure's services, as exhibited in *Appendix A*, Figures 63 to 74. The account created came with \$200 worth of credits and free access to Azure's service for 12 months which assisted in the developer acquiring a virtual machine (vm) of choice.

The Azure validated creation process and account settings can be found in *Appendix A*, Figures 63 to 74.

When selecting the size of the VM, there are a number of limitations as only certain regions were allowed to use specific sizes. We settled with the East US region, 64GiB Hard Disk space, 16Gib RAM memory, 2 CPUs with 2 Cores each was selected, which is believed to be sufficient for this project.

Next the SSH keys are initialised by setting permissions to read-only.

```
# Setting Permissions for SSH key to Read Only
chmod 400 SecurityAuditVM_key.pem
```



```
d3epdiv3r@d3-Workstation-Vm1: ~/Desktop/sec_audit 105x55
d3epdiv3r@d3-Workstation-Vm1:~/Desktop/sec_audit$ ls -la
total 12
drwxrwxr-x 2 d3epdiv3r d3epdiv3r 4096 Nov 14 20:48 .
drwxr-xr-x 4 d3epdiv3r d3epdiv3r 4096 Nov 14 20:46 ..
-rw-rw-r-- 1 d3epdiv3r d3epdiv3r 2498 Nov 14 20:47 SecurityAuditVM_key.pem
d3epdiv3r@d3-Workstation-Vm1:~/Desktop/sec_audit$ chmod 400 SecurityAuditVM_key.pem
d3epdiv3r@d3-Workstation-Vm1:~/Desktop/sec_audit$ ls -la
total 12
drwxrwxr-x 2 d3epdiv3r d3epdiv3r 4096 Nov 14 20:48 .
drwxr-xr-x 4 d3epdiv3r d3epdiv3r 4096 Nov 14 20:46 ..
-r----- 1 d3epdiv3r d3epdiv3r 2498 Nov 14 20:47 SecurityAuditVM_key.pem
d3epdiv3r@d3-Workstation-Vm1:~/Desktop/sec_audit$ █
```

Figure 1: Setting Permissions for SSH Keys

Afterwards, the login was attempted, which was successful on the first attempt, indicating that Azure has been successfully setup.

```
# Login to Azure VM via SSH
```

```

azureuser@SecurityAuditVM:~ 105x55
d3epdiv3r@D3-Workstation-Vm1:~/Desktop/sec_audit$ ssh -i SecurityAuditVM_key.pem azureuser@20.106.233.71
The authenticity of host '20.106.233.71 (20.106.233.71)' can't be established.
ED25519 key fingerprint is SHA256:KK7hWHSEHrWpxkzaN6u+5PELerHXdX0aXAbFvx9EpPY.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '20.106.233.71' (ED25519) to the list of known hosts.
Welcome to Ubuntu 20.04.5 LTS (GNU/Linux 5.15.0-1022-azure x86_64)

 * Documentation: https://help.ubuntu.com
 * Management: https://landscape.canonical.com
 * Support: https://ubuntu.com/advantage

 System information as of Mon Nov 14 20:55:59 UTC 2022

 System load: 0.0          Processes:           126
 Usage of /: 5.0% of 28.89GB Users logged in:      0
 Memory usage: 2%          IPv4 address for eth0: 10.0.0.4
 Swap usage: 0%

0 updates can be applied immediately.

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

azureuser@SecurityAuditVM:~$ whoami
azureuser

```

Figure 2: Successful Login to Azure Security Audit Virtual Machine

To complete the setup, the Ubuntu's package manager was updated, and installed Nginx's web server just incase we needed it. However, an update was not carried out, as the latest version of Ubuntu 20.04 was already in use. This was also done to avoid breaking any working packages.

```

# Updating package manager
sudo apt update -y

sudo apt -y install nginx

sudo apt update -y

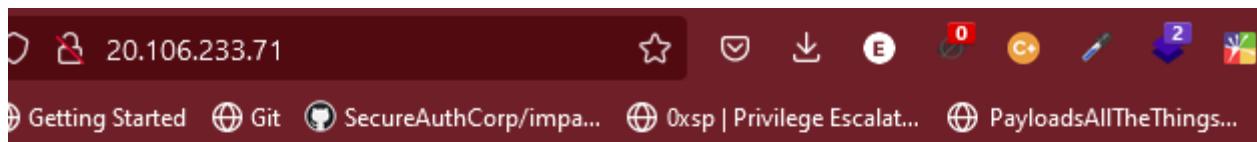
```

```

Processing triggers for libc-bin (2.31-0ubuntu9.1) ...
azureuser@SecurityAuditVM:~$ sudo apt update -y
Hit:1 http://azure.archive.ubuntu.com/ubuntu focal InRelease
Hit:2 http://azure.archive.ubuntu.com/ubuntu focal-updates InRelease
Hit:3 http://azure.archive.ubuntu.com/ubuntu focal-backports InRelease
Hit:4 http://azure.archive.ubuntu.com/ubuntu focal-security InRelease
Reading package lists... Done
Building dependency tree
Reading state information... Done
24 packages can be upgraded. Run 'apt list --upgradable' to see them.
azureuser@SecurityAuditVM:~$ █

```

Figure 3: Successful Update



## Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to [nginx.org](https://nginx.org). Commercial support is available at [nginx.com](https://nginx.com).

*Thank you for using nginx.*

Figure 4 Shows successful installation of Nginx and access to Azure Server

## Downloading PCAP Files for Network Analysis

Before proceeding, the PCAP files had to be downloaded so they would be available for the analysis. A variety of PCAP files have been downloaded from the following sources: Palo Alto Networks (Unit 42, 2020) and Cyber Defenders (CyberDefenders, 2020) to be used by Suricata and the other open source tools that need it.

```
azureuser@SecurityAuditVM:~/PCAP_Files$ ls -la
total 135248
drwxrwxr-x 2 azureuser azureuser    4096 Nov 14 23:19 .
drwxr-xr-x 7 azureuser azureuser    4096 Nov 15 00:06 ..
-rw-rw-r-- 1 azureuser azureuser  3805423 Nov 14 23:15 1-Emotet-infection.pcap
-rw-rw-r-- 1 azureuser azureuser  7825363 Nov 14 23:15 2-Emotet-with-spambot-traffic-part-1.pcap
-rw-rw-r-- 1 azureuser azureuser 21621569 Nov 14 23:15 3-Emotet-with-spambot-traffic-part-2.pcap
-rw-rw-r-- 1 azureuser azureuser 14915074 Nov 14 23:15 4-Emotet-infection-with-Trickbot.pcap
-rw-rw-r-- 1 azureuser azureuser 15657225 Nov 14 23:15 5-Emotet-infection-with-Qakbot.pcap
-rw-rw-r-- 1 azureuser azureuser 53173810 Nov 14 23:15 Qbot-infection-traffic.pcap
-rw-rw-r-- 1 azureuser azureuser 21474604 Nov 14 23:15 hp_challenge.pcap
azureuser@SecurityAuditVM:~/PCAP_Files$
```

Figure 5: PCAP files

## Suricata

Suricata has been installed according to the guide provided by Microsoft (Microsoft, 2022) which required that new PPA repositories be added to the package manager.

### Installing Suricata

```
sudo add-apt-repository ppa:oisf/suricata-stable
sudo apt-get update
```

```
Setting up libevent-pthreads-2.1-7:amd64 (2.1.11-stable-1) ...
Setting up libhiredis0.14:amd64 (0.14.0-6) ...
Setting up libluajit-5.1-2:amd64 (2.1.0~beta3+dfsg-5.1build1) ...
Setting up suricata (6.0.8-0ubuntu4) ...
Processing triggers for systemd (245.4-4ubuntu3.18) ...
Processing triggers for libc-bin (2.31-0ubuntu9.9) ...
azureuser@SecurityAuditVM:~$
```

Figure 6: Successful Installation of Suricata

After setting up Suricata, the service was enabled and checked to see if it was running successfully.

```
# Enabling Suricata

# Checking Status of Suricata
sudo systemctl status suricata
```

```
azureuser@SecurityAuditVM:~$ sudo systemctl enable suricata.service
suricata.service is not a native service, redirecting to systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable suricata
azureuser@SecurityAuditVM:~$ sudo systemctl status suricata
● suricata.service - LSB: Next Generation IDS/IPS
  Loaded: loaded (/etc/init.d/suricata; generated)
  Active: active (running) since Mon 2022-11-14 23:33:24 UTC; 20h ago
    Docs: man:systemd-sysv-generator(8)
    Tasks: 8 (limit: 19193)
   Memory: 216.5M
   CGroup: /system.slice/suricata.service
           └─16118 /usr/bin/suricata -c /etc/suricata/suricata.yaml --pidfile /var/run/suricata.pid --af-packet -D -vvv

Nov 14 23:33:24 SecurityAuditVM systemd[1]: Starting LSB: Next Generation IDS/IPS...
Nov 14 23:33:24 SecurityAuditVM suricata[16087]: Starting suricata in IDS (af-packet) mode... done.
Nov 14 23:33:24 SecurityAuditVM systemd[1]: Started LSB: Next Generation IDS/IPS.
azureuser@SecurityAuditVM:~$
```

Figure 7: Enabling and verifying status of Suricata

## Configuring Suricata

After validating that Suricata runs as expected, the service was stopped using `sudo systemctl stop suricata` in order to make changes to the configuration file.

Suricata's configuration file exists in `/etc/suricata/suricata.yaml`. Being a YAML file, it made modifications easy and Suricata did a great job in clearly identifying what each configuration does which also made changes easy.

The home subnet needed to be specified in order to increase the accuracy and performance of the alerts.

```
# Checking the IP Address of the VM for use in configuration file
azureuser@SecurityAuditVM:~$ ip -br -c a

eth0              UP      10.0.0.4/24 fe80::20d:3aff:fe53:f5b5/64
enP59470s1        UP
```

```

## Step 1: Inform Suricata about your network
##

vars:
  # more specific is better for alert accuracy and performance
  address-groups:
    HOME_NET: "[10.0.0.4/24]" ←
    #HOME_NET: "[192.168.0.0/16]"
    #HOME_NET: "[10.0.0.0/8]"
    #HOME_NET: "[172.16.0.0/12]"
    #HOME_NET: "any"

  EXTERNAL_NET: "!$HOME_NET"
  #EXTERNAL_NET: "any"

```

Figure 8: Specifying subnet in Suricata's configuration file

Next the interface had to be changed.

```

## Step 3: Configure common capture settings
##
## See "Advanced Capture Options" below for more options, including Netmap
## and PF_RING.
##

# Linux high speed capture support
af-packet:
  - interface: eth0
    # Number of receive threads. "auto" uses the number of cores
    #threads: auto
    # Default clusterid. AF_PACKET will load balance packets based on flow.
    cluster-id: 99
    # Default AF_PACKET cluster type. AF_PACKET can load balance per flow or per hash.
    #cluster-type: 1

```

Figure 9: Updating network interface in Suricata's config

Next, the community flow ID was enabled, which is useful for event correlation and helpful when using tools like Zeek or when trying to import logs into Suricata in JSON format.

```

# Community Flow ID
# Adds a 'community_id' field to EVE records. These are meant to give
# records a predictable flow ID that can be used to match records to
# output of other tools such as Zeek (Bro).
#
# Takes a 'seed' that needs to be same across sensors and tools
# to make the id less predictable.

# enable/disable the community id feature.
community-id: true ←
# Seed value for the ID output. Valid values are 0-65535.
community-id-seed: 0

```

Figure 10: Enabling community-id in Suricata's config

Afterwards, it was verified that Suricata's rules were put in place.

```
##  
## Configure Suricata to load Suricata-Update managed rules.  
##  
  
default-rule-path: /var/lib/suricata/rules  
  
rule-files:  
  - suricata.rules  
##  
## Auxiliary configuration files.  
##
```

Figure 11: Default rule files in Suricata

Since Suricata operates using rules similar to other IDS systems, an update was run to load all rules and initialize the resources needed.

```
# Update Suricata  
sudo suricata-update
```

```

azureuser@SecurityAuditVM:~$ sudo suricata-update
15/11/2022 -- 20:49:35 - <Info> -- Using data-directory /var/lib/suricata.
15/11/2022 -- 20:49:35 - <Info> -- Using Suricata configuration /etc/suricata/suricata.yaml
15/11/2022 -- 20:49:35 - <Info> -- Using /etc/suricata/rules for Suricata provided rules.
15/11/2022 -- 20:49:35 - <Info> -- Found Suricata version 6.0.8 at /usr/bin/suricata.
15/11/2022 -- 20:49:35 - <Info> -- Loading /etc/suricata/suricata.yaml
15/11/2022 -- 20:49:35 - <Info> -- Disabling rules for protocol http2
15/11/2022 -- 20:49:35 - <Info> -- Disabling rules for protocol modbus
15/11/2022 -- 20:49:35 - <Info> -- Disabling rules for protocol dnp3
15/11/2022 -- 20:49:35 - <Info> -- Disabling rules for protocol enip
15/11/2022 -- 20:49:35 - <Info> -- No sources configured, will use Emerging Threats Open
15/11/2022 -- 20:49:35 - <Info> -- Fetching https://rules.emergingthreats.net/open/suricata-6.0.8/emerging.rules.tar.gz.
100% - 3532015/3532015
15/11/2022 -- 20:49:35 - <Info> -- Done.
15/11/2022 -- 20:49:35 - <Info> -- Loading distribution rule file /etc/suricata/rules/app-layer-events.rules
15/11/2022 -- 20:49:35 - <Info> -- Loading distribution rule file /etc/suricata/rules/decoder-events.rules
15/11/2022 -- 20:49:35 - <Info> -- Loading distribution rule file /etc/suricata/rules/dhcp-events.rules
15/11/2022 -- 20:49:35 - <Info> -- Loading distribution rule file /etc/suricata/rules/dnp3-events.rules
15/11/2022 -- 20:49:35 - <Info> -- Loading distribution rule file /etc/suricata/rules/dns-events.rules
15/11/2022 -- 20:49:35 - <Info> -- Loading distribution rule file /etc/suricata/rules/files.rules
15/11/2022 -- 20:49:35 - <Info> -- Loading distribution rule file /etc/suricata/rules/http-events.rules
15/11/2022 -- 20:49:35 - <Info> -- Loading distribution rule file /etc/suricata/rules/ipsec-events.rules
15/11/2022 -- 20:49:35 - <Info> -- Loading distribution rule file /etc/suricata/rules/kerberos-events.rules
15/11/2022 -- 20:49:35 - <Info> -- Loading distribution rule file /etc/suricata/rules/modbus-events.rules
15/11/2022 -- 20:49:35 - <Info> -- Loading distribution rule file /etc/suricata/rules/nfs-events.rules
15/11/2022 -- 20:49:35 - <Info> -- Loading distribution rule file /etc/suricata/rules/ntp-events.rules
15/11/2022 -- 20:49:35 - <Info> -- Loading distribution rule file /etc/suricata/rules/smb-events.rules
15/11/2022 -- 20:49:35 - <Info> -- Loading distribution rule file /etc/suricata/rules/smtp-events.rules
15/11/2022 -- 20:49:35 - <Info> -- Loading distribution rule file /etc/suricata/rules/stream-events.rules
15/11/2022 -- 20:49:35 - <Info> -- Loading distribution rule file /etc/suricata/rules/tls-events.rules
15/11/2022 -- 20:49:35 - <Info> -- Ignoring file rules/emerging-deleted.rules
15/11/2022 -- 20:49:37 - <Info> -- Loaded 36497 rules.
15/11/2022 -- 20:49:37 - <Info> -- Disabled 14 rules.
15/11/2022 -- 20:49:37 - <Info> -- Enabled 0 rules.
15/11/2022 -- 20:49:37 - <Info> -- Modified 0 rules.
15/11/2022 -- 20:49:37 - <Info> -- Dropped 0 rules.
15/11/2022 -- 20:49:37 - <Info> -- Enabled 131 rules for flowbit dependencies.
15/11/2022 -- 20:49:37 - <Info> -- Creating directory /var/lib/suricata/rules.
15/11/2022 -- 20:49:37 - <Info> -- Backing up current rules.
15/11/2022 -- 20:49:37 - <Info> -- Writing rules to /var/lib/suricata/rules/suricata.rules: total: 36497; enabled: 28921; added: 36497; removed 0; modified: 0
15/11/2022 -- 20:49:37 - <Info> -- Writing /var/lib/suricata/rules/classification.config
15/11/2022 -- 20:49:37 - <Info> -- Testing with suricata -T.
15/11/2022 -- 20:49:37 - <Warning> -- [ERRCODE: SC_ERR_CONF_YAML_ERROR(242)] - App-Layer protocol sip enable status not set, so enabling by default. This behavior will change in Suricata 7, so please update your config. See ticket #4744 for more details.
15/11/2022 -- 20:49:38 - <Warning> -- [ERRCODE: SC_ERR_CONF_YAML_ERROR(242)] - App-Layer protocol mqtt enable status not set, so enabling by default. This behavior will change in Suricata 7, so please update your config. See ticket #4744 for more details.
15/11/2022 -- 20:49:38 - <Warning> -- [ERRCODE: SC_ERR_CONF_YAML_ERROR(242)] - App-Layer protocol rdp enable status not set, so enabling by default. This behavior will change in Suricata 7, so please update your config. See ticket #4744 for more details.
15/11/2022 -- 20:49:55 - <Info> -- Done.
azureuser@SecurityAuditVM:~$
```

Figure 12: Updating Suricata

On running the update, it is evident that it automatically fetched the rules from Emerging Threats, but more importantly, issues were experienced (with the app-layer protocol) that seemed to be corrected in Suricata 7 but has not been changed in Suricata 6.x.x. The configurations have to be changed to fix these issues.

```

dp: 5900, 5901, 5902, 5903, 5904, 5905, 5906, 5907,
# MQTT, disabled by default.
mqtt:
    enabled: no
    # max-msg-length: 1mb
    # subscribe-topic-match-limit: 100
    # unsubscribe-topic-match-limit: 100
    # Maximum number of live MQTT transactions per flow
    # max-tx: 4096
krb5:
    enabled: yes
```

Figure 13: Disabling the MQTT app-layer protocol

```
# memcap: 64mb
rdp:
  enabled: yes
ssh:
  enabled: yes
#hassh: yes
```

Figure 14: Enabling the RDP app-layer protocol

```
enabled: yes
sip:
  enabled: no
```

Figure 15: Disabling the SIP app-layer protocol

Then we updated Suricata and all worked fine.

```
azureuser@SecurityAuditVM:~$ sudo suricata-update
15/11/2022 -- 21:04:12 - <Info> -- Using data-directory /var/lib/suricata.
15/11/2022 -- 21:04:12 - <Info> -- Using Suricata configuration /etc/suricata/suricata.yaml
15/11/2022 -- 21:04:12 - <Info> -- Using /etc/suricata/rules for Suricata provided rules.
15/11/2022 -- 21:04:12 - <Info> -- Found Suricata version 6.0.8 at /usr/bin/suricata.
15/11/2022 -- 21:04:12 - <Info> -- Loading /etc/suricata/suricata.yaml
15/11/2022 -- 21:04:12 - <Info> -- Disabling rules for protocol mqtt
15/11/2022 -- 21:04:12 - <Info> -- Disabling rules for protocol http2
15/11/2022 -- 21:04:12 - <Info> -- Disabling rules for protocol modbus
15/11/2022 -- 21:04:12 - <Info> -- Disabling rules for protocol dnp3
15/11/2022 -- 21:04:12 - <Info> -- Disabling rules for protocol enip
15/11/2022 -- 21:04:12 - <Info> -- Disabling rules for protocol sip
15/11/2022 -- 21:04:12 - <Info> -- No sources configured, will use Emerging Threats Open
15/11/2022 -- 21:04:12 - <Info> -- Last download less than 15 minutes ago. Not downloading https://rules.emergingthreats.net/open/suricata-6.0.8/emerging.rules.tar.gz.
15/11/2022 -- 21:04:13 - <Info> -- Loading distribution rule file /etc/suricata/rules/app-layer-events.rules
15/11/2022 -- 21:04:13 - <Info> -- Loading distribution rule file /etc/suricata/rules/decoder-events.rules
15/11/2022 -- 21:04:13 - <Info> -- Loading distribution rule file /etc/suricata/rules/dhcp-events.rules
15/11/2022 -- 21:04:13 - <Info> -- Loading distribution rule file /etc/suricata/rules/dnp3-events.rules
15/11/2022 -- 21:04:13 - <Info> -- Loading distribution rule file /etc/suricata/rules/dns-events.rules
15/11/2022 -- 21:04:13 - <Info> -- Loading distribution rule file /etc/suricata/rules/files.rules
15/11/2022 -- 21:04:13 - <Info> -- Loading distribution rule file /etc/suricata/rules/http-events.rules
15/11/2022 -- 21:04:13 - <Info> -- Loading distribution rule file /etc/suricata/rules/ipsec-events.rules
15/11/2022 -- 21:04:13 - <Info> -- Loading distribution rule file /etc/suricata/rules/kerberos-events.rules
15/11/2022 -- 21:04:13 - <Info> -- Loading distribution rule file /etc/suricata/rules/modbus-events.rules
15/11/2022 -- 21:04:13 - <Info> -- Loading distribution rule file /etc/suricata/rules/nfs-events.rules
15/11/2022 -- 21:04:13 - <Info> -- Loading distribution rule file /etc/suricata/rules/ntp-events.rules
15/11/2022 -- 21:04:13 - <Info> -- Loading distribution rule file /etc/suricata/rules/smb-events.rules
15/11/2022 -- 21:04:13 - <Info> -- Loading distribution rule file /etc/suricata/rules/smtp-events.rules
15/11/2022 -- 21:04:13 - <Info> -- Loading distribution rule file /etc/suricata/rules/stream-events.rules
15/11/2022 -- 21:04:13 - <Info> -- Loading distribution rule file /etc/suricata/rules/tls-events.rules
15/11/2022 -- 21:04:13 - <Info> -- Ignoring file rules/emerging-deleted.rules
15/11/2022 -- 21:04:14 - <Info> -- Loaded 36497 rules.
15/11/2022 -- 21:04:15 - <Info> -- Disabled 14 rules.
15/11/2022 -- 21:04:15 - <Info> -- Enabled 0 rules.
15/11/2022 -- 21:04:15 - <Info> -- Modified 0 rules.
15/11/2022 -- 21:04:15 - <Info> -- Dropped 0 rules.
15/11/2022 -- 21:04:15 - <Info> -- Enabled 131 rules for flowbit dependencies.
15/11/2022 -- 21:04:15 - <Info> -- Backing up current rules.
15/11/2022 -- 21:04:17 - <Info> -- Writing rules to /var/lib/suricata/rules/suricata.rules: total: 36497; enabled: 28921; added: 0; removed 0; modified: 0
15/11/2022 -- 21:04:17 - <Info> -- Writing /var/lib/suricata/rules/classification.config
15/11/2022 -- 21:04:17 - <Info> -- No changes detected, exiting.
azureuser@SecurityAuditVM:~$ sudo suricata-update -T
```

Figure 16: Successful update of Suricata without app-layer errors

Understanding what resources exist is essential, since Suricata was pulling rule lists from Emerging Threats. Here `sudo suricata-update list-sources` was run to get the information as seen below:

```
azureuser@SecurityAuditVM:~$ sudo suricata-update list-sources
[...]
Vendor: Proofpoint
License: MIT
[...]
Vendor: malsilo
License: MIT
[...]
```

It is recognisable that some of the sources are commercial and some MIT (Open source or Non-commercial) which meant that we could add additional rules can be added for free. The [malsilo/win-malware](#) is then added to aid Suricata conduct a more in-depth scan.

```
# Add two sources and update Suricata
```

As displayed below, everything is working correctly and the Suricata tests run successfully.

```

azuser@SecurityAuditVM:~$ sudo suricata-update enable-source malsilo/win-malware && sudo suricata-update enable-source etnetera/aggressive && sudo suricata-update
15/11/2022 -- 21:42:10 - <Info> -- Using data-directory /var/lib/suricata.
15/11/2022 -- 21:42:10 - <Info> -- Using Suricata configuration /etc/suricata/suricata.yaml
15/11/2022 -- 21:42:10 - <Info> -- Using /etc/suricata/rules for Suricata provided rules.
15/11/2022 -- 21:42:10 - <Info> -- Found Suricata version 6.0.8 at /usr/bin/suricata.
15/11/2022 -- 21:42:10 - <Info> -- Creating directory /var/lib/suricata/update/sources
15/11/2022 -- 21:42:10 - <Info> -- Enabling default source et/open
15/11/2022 -- 21:42:10 - <Info> -- Source malsilo/win-malware enabled
15/11/2022 -- 21:42:10 - <Info> -- Using data-directory /var/lib/suricata.
15/11/2022 -- 21:42:10 - <Info> -- Using Suricata configuration /etc/suricata/suricata.yaml
15/11/2022 -- 21:42:10 - <Info> -- Using /etc/suricata/rules for Suricata provided rules.
15/11/2022 -- 21:42:10 - <Info> -- Found Suricata version 6.0.8 at /usr/bin/suricata.
15/11/2022 -- 21:42:10 - <Info> -- Source etnetera/aggressive enabled
15/11/2022 -- 21:42:10 - <Info> -- Using data-directory /var/lib/suricata.
15/11/2022 -- 21:42:10 - <Info> -- Using Suricata configuration /etc/suricata/suricata.yaml
15/11/2022 -- 21:42:10 - <Info> -- Using /etc/suricata/rules for Suricata provided rules.
15/11/2022 -- 21:42:10 - <Info> -- Found Suricata version 6.0.8 at /usr/bin/suricata.
15/11/2022 -- 21:42:10 - <Info> -- Loading /etc/suricata/suricata.yaml
15/11/2022 -- 21:42:10 - <Info> -- Disabling rules for protocol mqtt
15/11/2022 -- 21:42:10 - <Info> -- Disabling rules for protocol http2
15/11/2022 -- 21:42:10 - <Info> -- Disabling rules for protocol modbus
15/11/2022 -- 21:42:10 - <Info> -- Disabling rules for protocol dnp3
15/11/2022 -- 21:42:10 - <Info> -- Disabling rules for protocol enip
15/11/2022 -- 21:42:10 - <Info> -- Disabling rules for protocol sip
15/11/2022 -- 21:42:10 - <Info> -- Fetching https://security.etnetera.cz/feeds/etn_aggressive.rules.
100% - 204438/204438
15/11/2022 -- 21:42:11 - <Info> -- Done.
15/11/2022 -- 21:42:11 - <Info> -- Fetching https://malsilo.gitlab.io/feeds/dumps/malsilo.rules.tar.gz.
100% - 2371/2371
15/11/2022 -- 21:42:11 - <Info> -- Done.
15/11/2022 -- 21:42:11 - <Info> -- Checking https://rules.emergingthreats.net/open/suricata-6.0.8/emerging.rules.tar.gz.md5
.
15/11/2022 -- 21:42:11 - <Info> -- Remote checksum has not changed. Not fetching.
15/11/2022 -- 21:42:11 - <Info> -- Loading distribution rule file /etc/suricata/rules/app-layer-events.rules
15/11/2022 -- 21:42:11 - <Info> -- Loading distribution rule file /etc/suricata/rules/decoder-events.rules
15/11/2022 -- 21:42:11 - <Info> -- Loading distribution rule file /etc/suricata/rules/dhcp-events.rules
15/11/2022 -- 21:42:11 - <Info> -- Loading distribution rule file /etc/suricata/rules/dnp3-events.rules
15/11/2022 -- 21:42:11 - <Info> -- Loading distribution rule file /etc/suricata/rules/dns-events.rules
15/11/2022 -- 21:42:11 - <Info> -- Loading distribution rule file /etc/suricata/rules/files.rules
15/11/2022 -- 21:42:11 - <Info> -- Loading distribution rule file /etc/suricata/rules/http-events.rules
15/11/2022 -- 21:42:11 - <Info> -- Loading distribution rule file /etc/suricata/rules/ipsec-events.rules
15/11/2022 -- 21:42:11 - <Info> -- Loading distribution rule file /etc/suricata/rules/kerberos-events.rules
15/11/2022 -- 21:42:11 - <Info> -- Loading distribution rule file /etc/suricata/rules/modbus-events.rules
15/11/2022 -- 21:42:11 - <Info> -- Loading distribution rule file /etc/suricata/rules/nfs-events.rules
15/11/2022 -- 21:42:11 - <Info> -- Loading distribution rule file /etc/suricata/rules/ntp-events.rules
15/11/2022 -- 21:42:11 - <Info> -- Loading distribution rule file /etc/suricata/rules/smb-events.rules
15/11/2022 -- 21:42:11 - <Info> -- Loading distribution rule file /etc/suricata/rules/smtp-events.rules
15/11/2022 -- 21:42:11 - <Info> -- Loading distribution rule file /etc/suricata/rules/stream-events.rules
15/11/2022 -- 21:42:11 - <Info> -- Loading distribution rule file /etc/suricata/rules/tls-events.rules
15/11/2022 -- 21:42:12 - <Info> -- Ignoring file rules/emerging-deleted.rules
15/11/2022 -- 21:42:13 - <Info> -- Loaded 36868 rules.
15/11/2022 -- 21:42:13 - <Info> -- Disabled 14 rules.
15/11/2022 -- 21:42:13 - <Info> -- Enabled 0 rules.
15/11/2022 -- 21:42:13 - <Info> -- Modified 0 rules.
15/11/2022 -- 21:42:13 - <Info> -- Dropped 0 rules.
15/11/2022 -- 21:42:14 - <Info> -- Enabled 131 rules for flowbit dependencies.
15/11/2022 -- 21:42:14 - <Info> -- Backing up current rules.
15/11/2022 -- 21:42:16 - <Info> -- Writing rules to /var/lib/suricata/rules/suricata.rules: total: 36800; enabled: 29224; added: 303; removed 0; modified: 0
15/11/2022 -- 21:42:16 - <Info> -- Writing /var/lib/suricata/rules/classification.config
15/11/2022 -- 21:42:16 - <Info> -- Testing with suricata -T.
15/11/2022 -- 21:42:34 - <Info> -- Done.
azuser@SecurityAuditVM:~$ █

```

Figure 17: Adding two new source, updating, and testing Suricata configurations

The next step was to manually test Suricata's configuration to ensure everything was setup properly.

```

# Testing Suricata's Configuration File and Rules
sudo suricata -T -c /etc/suricata/suricata.yaml -v

```

```

azureuser@SecurityAuditVM:~$ sudo suricata -T -c /etc/suricata/suricata.yaml -v
16/11/2022 -- 02:06:56 - <Info> - Running suricata under test mode
16/11/2022 -- 02:06:56 - <Notice> - This is Suricata version 6.0.8 RELEASE running in SYSTEM mode
16/11/2022 -- 02:06:56 - <Info> - CPUs/cores online: 2
16/11/2022 -- 02:06:56 - <Info> - fast output device (regular) initialized: fast.log
16/11/2022 -- 02:06:56 - <Info> - eve-log output device (regular) initialized: eve.json
16/11/2022 -- 02:06:56 - <Notice> - JsonSIPLog logger not enabled: protocol sip is disabled
16/11/2022 -- 02:06:56 - <Notice> - JsonMQTTLog logger not enabled: protocol mqtt is disabled
16/11/2022 -- 02:06:56 - <Info> - stats output device (regular) initialized: stats.log
16/11/2022 -- 02:07:03 - <Info> - 1 rule files processed. 29240 rules successfully loaded, 0 rules failed
16/11/2022 -- 02:07:03 - <Info> - Threshold config parsed: 0 rule(s) found
16/11/2022 -- 02:07:03 - <Info> - 29243 signatures processed. 1483 are IP-only rules, 5171 are inspecting packet payload, 22386 inspect application layer, 108 are decoder event only
16/11/2022 -- 02:07:14 - <Notice> - Configuration provided was successfully loaded. Exiting.
16/11/2022 -- 02:07:14 - <Info> - cleaning up signature grouping structure... complete
azureuser@SecurityAuditVM:~$ █

```

Figure 18: Suricata configuration loaded successfully

The configuration was loaded successfully: one rule file (`suricata.rules`) was processed and 29,240 rules loaded successfully. Zero rules failed.

Afterwards, Suricata is run to ensure everything works as expected.

```

azureuser@SecurityAuditVM:~$ sudo systemctl start suricata.service
azureuser@SecurityAuditVM:~$ sudo systemctl status suricata.service
● suricata.service - LSB: Next Generation IDS/IPS
    Loaded: loaded (/etc/init.d/suricata; generated)
    Active: active (running) since Wed 2022-11-16 02:28:29 UTC; 13s ago
      Docs: man:systemd-sysv-generator(8)
   Process: 31093 ExecStart=/etc/init.d/suricata start (code=exited, status=0/SUCCESS)
     Tasks: 1 (limit: 19193)
    Memory: 424.7M
       CGroup: /system.slice/suricata.service
               └─31099 /usr/bin/suricata -c /etc/suricata/suricata.yaml --pidfile /var/run/suricata.pid --af-pa

Nov 16 02:28:29 SecurityAuditVM systemd[1]: Starting LSB: Next Generation IDS/IPS...
Nov 16 02:28:29 SecurityAuditVM suricata[31093]: Starting suricata in IDS (af-packet) mode... done.
Nov 16 02:28:29 SecurityAuditVM systemd[1]: Started LSB: Next Generation IDS/IPS.
azureuser@SecurityAuditVM:~$ █

```

Figure 19: Running status for Suricata

## Testing Suricata

When testing Suricata, the logs were checked to see if they have been created correctly for verification purposes, as logs can be written to whichever directory the user specifies.

```

azureuser@SecurityAuditVM:~$ ls -al /var/log/suricata/
total 1228
drwxr-xr-x  5 root root      4096 Nov 16 00:47 .
drwxrwxr-x 12 root syslog    4096 Nov 16 00:47 ..
drwxr-xr-x  2 root root      4096 Sep 28 05:05 certs
drwxr-xr-x  2 root root      4096 Sep 28 05:05 core
-rw-r--r--  1 root root    954990 Nov 16 02:32 eve.json
-rw-r--r--  1 root root     1961 Nov 16 02:29 fast.log
drwxr-xr-x  2 root root      4096 Sep 28 05:05 files
-rw-r--r--  1 root root   208742 Nov 16 02:32 stats.log
-rw-r--r--  1 root root    1542 Nov 16 02:28 suricata-start.log
-rw-r--r--  1 root root   57203 Nov 16 02:28 suricata.log
azureuser@SecurityAuditVM:~$ █

```

Figure 20: Suricata log files

As seen in *Figure 20* above, `eve.json` (contains the intrusion logs stored in `JSON` format) and `fast.log` (contains the same log stored in standard format) the two essential files to the scope of this project are created.

Next, Suricata was tested to see if it was monitoring the network by making a request to

```
curl http://testmynids.org/uid/index.html
```

. A website and framework for testing NIDS detection (3CORESec, 2021).

```
azureuser@SecurityAuditVM:~$ curl http://testmynids.org/uid/index.html
uid=0(root) gid=0(root) groups=0(root)
```

Figure 21: Testing NID detection

Viewing `sudo cat /var/log/suricata/fast.log` proves that Suricata produces attack alerts coming from `testmynids.com` in *Figure 22* and *Figure 23*

```
[**] [1:2013028:7] ET POLICY curl User-Agent Outbound [**] [Classification: Attempted Information Leak] [Priority: 2]
[**] [1:2100498:7] GPL ATTACK_RESPONSE id check returned root [**] [Classification: Potentially Bad Traffic]
[**] [1:2013028:7] ET POLICY curl User-Agent Outbound [**] [Classification: Attempted Information Leak] [Priority: 2]
```

Figure 22: Suricata detects information leak attempt on the network 1

```
leak] [Priority: 2] {TCP} 10.0.0.4:36520 -> 18.165.83.29:80
Traffic] [Priority: 2] {TCP} 18.165.83.29:80 -> 10.0.0.4:36520
leak] [Priority: 2] {TCP} 10.0.0.4:37624 -> 18.165.83.127:80
```

Figure 23: Suricata logs outbound destination IP

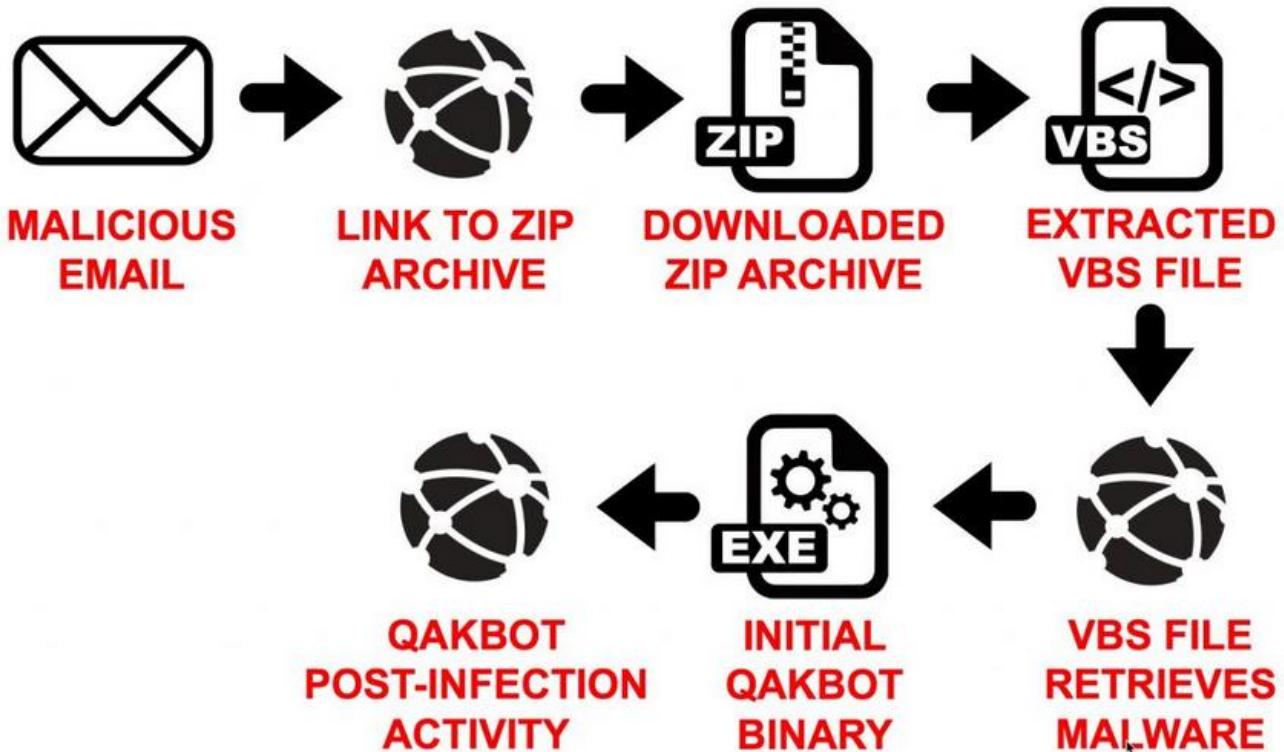
This shows that Suricata is set up correctly and running in the background!

## Checking QakBot Traffic with Suricata

Seeing as the amount of PCAP files we downloaded earlier (as seen in *Figure 5*) analysing of all may be beyond the scope, it is wiser to focus specifically on the `Qbot-infection-traffic.pcap` file for analysis.

Figure 24

*QakBot Infection Stages Overview*



Directly copied from Duncan, B. (2020).

Knowing QakBot's infection process is helpful because it assist in detecting intrusion on the network.

Next, *Figure 25* below displays an attempt to parse the PCAP file into Suricata.

```

# Parsing PCAP to Suricata

/var/log/suricata
# OR
  
```

```

azureuser@SecurityAuditVM:~/PCAP_Files$ ls
1-Emotet-infection.pcap          3-Emotet-with-spambot-traffic-part-2.pcap  5-Emotet-infection-with-Qakbot.pcap
2-Emotet-with-spambot-traffic-part-1.pcap  4-Emotet-infection-with-Trickbot.pcap  Qbot-infection-traffic.pcap
azureuser@SecurityAuditVM:~/PCAP_Files$ sudo suricata -c /etc/suricata/suricata.yaml -r Qbot-infection-traffic.pcap
16/11/2022 -- 04:26:24 - <Notice> - This is Suricata version 6.0.8 RELEASE running in USER mode
16/11/2022 -- 04:26:24 - <Notice> - JsonSIPLog logger not enabled: protocol sip is disabled
16/11/2022 -- 04:26:24 - <Notice> - JsonMQTTLog logger not enabled: protocol mqtt is disabled
16/11/2022 -- 04:26:42 - <Notice> - all 3 packet processing threads, 4 management threads initialized, engine started.
16/11/2022 -- 04:26:43 - <Notice> - Signal_Received_Signaling
16/11/2022 -- 04:26:43 - <Notice> - Pcap-file module read 1 files, 74404 packets, 51983322 bytes
azureuser@SecurityAuditVM:~/PCAP_Files$ 
  
```

**Figure 25:** Parsing QakBot traffic into Suricata

It is evident in Figure above that Suricata ingested the PCAP file successfully, as 74,404 packets were read.

Suricata's digested output is stored in `eve.json` (JSON Format) and `fast.log` (standard format). The output in the `eve.json` file was chosen to be utilised, meaning `jq` had to be installed to make the JSON easier to read.

```
# Install jq
```

Since we know the infection process of **QakBot** is made aware, developers understood what to look for. The `eve.json` file was analysed and GET requests were found to download malicious `9312.zip` files from `bhatner.com` what appears to be a WordPress CMS (indicated by `wp-`) `L3006EHZ.zip` being sent from `"daikei-  
"toiwase@monotaro.com`. These events happen over and over again and this sort of behaviour is common with QakBot malware. This shows that the PCAP file was being ingested properly.

```
# Searching for patterns that match ".zip" in Suricata's log file (eve.json)
jq -C ." /var/log/suricata/eve.json | grep ".zip"
```

```
[{"timestamp":"2020-01-29T15:41:27.337110+0000","flow_id":1730951941733493,"pcap_cnt":455,"event_type":"fileinfo","src_ip":"103.91.92.1","src_port":80,"dest_ip":"bhatner.com","url":"/wp-content/uploads/2020/01/ahead/9312.zip","http_user_agent":"Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like  
content_type : application/zip","http_method":"GET","protocol":"HTTP/1.1","status":200,"length":112420}, "app_proto":"http","fileinfo":{"filename":"9312.zip","s  
400,"tx_id":0}]}
[{"timestamp":"2020-01-29T15:41:33.449319+0000","flow_id":1730951941733493,"pcap_cnt":1352,"event_type":"http","src_ip":"10.1.29.101","src_port":49679,"dest_ip":  
strname:"bhatner.com","url":"/wp-content/uploads/2020/01/ahead/9312.zip","http_user_agent":"Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML  
,"http_content_type":application/zip,"http_method":GET,"protocol":HTTP/1.1,"status":200,"length":1046724}}
[{"timestamp":"2021-01-05T20:16:06.195035+0000","flow_id":13846051927218,"pcap_cnt":360,"event_type":"smtp","src_ip":"10.1.4.205","src_port":50458,"dest_ip":  
"lo": "[10.0.0.8]","mail_from": "<skammy.tang@suburfarm.com>","rcpt_to": ["<toiwase@monotaro.com>"],"email":{"status":"PARSE_DONE","from": "\\\\"daikei-koumuten@gaia.  
yPSLy56S+TW9ub3RhUk8=?= <toiwase@monotaro.com>"], "attachment": ["L3006EHZ.zip"]}, "url": ["k2k.sagawa-exp.co.jp/p/sagawa/web/okurijoinput.jsp"]}]
[{"timestamp":"2021-01-05T20:16:06.290970+0000","flow_id":13846051927218,"pcap_cnt":366,"event_type":"fileinfo","src_ip": "10.1.4.205","src_port":50458,"dest_ip":  
"10.0.0.8}","mail_from": "<skammy.tang@suburfarm.com>","rcpt_to": ["<toiwase@monotaro.com>"],"email":{"status":"PARSE_DONE","from": "\\\\"daikei-koumuten@gaia.eonet.  
56S+TW9ub3RhUk8=?= <toiwase@monotaro.com>"], "attachment": ["L3006EHZ.zip"]}, "url": ["k2k.sagawa-exp.co.jp/p/sagawa/web/okurijoinput.jsp"]}, "app_proto": "smtp","fi  
"CLOSED","stored":false,"size":65929,"tx_id":0}]
[{"timestamp":"2020-01-29T15:41:27.337110+0000","flow_id":1179147428445301,"pcap_cnt":455,"event_type":"fileinfo","src_ip": "103.91.92.1","src_port":80,"dest_ip":  
"bhatner.com","url":"/wp-content/uploads/2020/01/ahead/9312.zip","http_user_agent":"Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like  
content_type : application/zip","http_method":"GET","protocol":"HTTP/1.1","status":200,"length":112420}, "app_proto": "http","fileinfo":{"filename":"9312.zip","s  
400,"tx_id":0}]}
[{"timestamp":"2020-01-29T15:41:33.449319+0000","flow_id":1179147428445301,"pcap_cnt":1352,"event_type":"http","src_ip": "10.1.29.101","src_port":49679,"dest_ip":  
strname:"bhatner.com","url":"/wp-content/uploads/2020/01/ahead/9312.zip","http_user_agent":"Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like  
,"http_content_type":application/zip,"http_method":GET,"protocol":HTTP/1.1,"status":200,"length":1046724}}
[{"timestamp":"2020-01-29T15:41:27.337110+0000","flow_id":1053055778569333,"pcap_cnt":455,"event_type":"fileinfo","src_ip": "103.91.92.1","src_port":80,"dest_ip":  
"bhatner.com","url":"/wp-content/uploads/2020/01/ahead/9312.zip","http_user_agent":"Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like  
content_type : application/zip,"http_method":GET,"protocol":HTTP/1.1,"status":200,"length":112420}, "app_proto": "http","fileinfo":{"filename":"9312.zip","s  
400,"tx_id":0}]}
[{"timestamp":"2020-01-29T15:41:33.449319+0000","flow_id":1053055778569333,"pcap_cnt":1352,"event_type":"http","src_ip": "10.1.29.101","src_port":49679,"dest_ip":  
["flowInts": {"tcp.retransmission.count":2}], "community_id": "1:jFq2NfY7zn/g8xhfsUDPDtgzMc=", "http": [{"hostname": "bhatner.com", "url": "/wp-content/uploads/2020/01  
Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/79.0.3945.130 Safari/537.36 Edg/79.0.309.71", "http_content_type": "application/zip", "http_method": "GE
```

Figure 26: QakBot malicious `.zip` file downloads (one)

```

    "L3006EHZ.zip",
    "L3006EHZ.zip",
    "filename": "L3006EHZ.zip",
    "url": "/wp-content/uploads/2020/01/ahead/9312.zip",
    "http_content_type": "application/zip",
    "filename": "9312.zip",
    "url": "/wp-content/uploads/2020/01/ahead/9312.zip",
    "http_content_type": "application/zip",
    "L3006EHZ.zip"
    "L3006EHZ.zip"
    "filename": "L3006EHZ.zip",
    "url": "/wp-content/uploads/2020/01/ahead/9312.zip",
    "http_content_type": "application/zip",
    "filename": "9312.zip",
    "url": "/wp-content/uploads/2020/01/ahead/9312.zip",
    "http_content_type": "application/zip",
    "L3006EHZ.zip"
    "L3006EHZ.zip"
    "filename": "L3006EHZ.zip",
    "url": "/wp-content/uploads/2020/01/ahead/9312.zip",
    "http_content_type": "application/zip",
    "filename": "9312.zip",
    "url": "/wp-content/uploads/2020/01/ahead/9312.zip",
    "http_content_type": "application/zip",
    "url": "/wp-content/uploads/2020/01/ahead/9312.zip",
    "http_content_type": "application/zip",
    "filename": "9312.zip",
    "url": "/wp-content/uploads/2020/01/ahead/9312.zip",
    "http_content_type": "application/zip",
    "azureuser@SecurityAuditVM:~/PCAP_Files$ █

```

Figure 27: QakBot malicious `.zip` file downloads (two)

## Setting up the Elastic Stack

As displayed *Figure 26* and *Figure 27* the logs produced by Suricata are not very presentable and they are a little difficult to read and understand. "By connecting Suricata with the Elastic Stack (a SaaS which combine Elasticsearch, Logstash and Kibana), we can create a Kibana dashboard which allows us to search, graph, analyze, and derive insights from our logs" Microsoft. (2022).

**Note** that the Microsoft documentation provided is out-of-date as it does not state that the versions of Elastic Stack (5.2.\*.) written about, can only work on Ubuntu v16.04: 32bit specifically (and other specific Linux versions) as seen in *Figure 28* below.

Azure only offers Ubuntu v18.04 and greater for free. In order to use v16.04 on Azure, the user would have to pay! As a result, from this point on, another server was provisioned with Ubuntu v16.04 to carry out the task. The previous steps were repeated on that server.

	RHEL/CentOS 6‡	RHEL/CentOS 7	RHEL 8	RHEL 9	CentOS 8	Oracle Linux 6*‡	Oracle Linux 7*	Oracle Linux 8	Ubuntu 14.04	Ubuntu 16.04	Ubuntu 18.04	Ubuntu 20.04	Ubuntu 22.04
Elasticsearch 5.0.x	✓	✓	✗	✗	✗	✓	✓	✗	✓	✓	✗	✗	✗
Elasticsearch 5.1.x	✓	✓	✗	✗	✗	✓	✓	✗	✓	✓	✗	✗	✗
Elasticsearch 5.2.x	✓	✓	✗	✗	✗	✓	✓	✗	✓	✓	✗	✗	✗
Elasticsearch 5.3.x	✓	✓	✗	✗	✗	✓	✓	✗	✓	✓	✗	✗	✗
Elasticsearch 5.4.x	✓	✓	✗	✗	✗	✓	✓	✗	✓	✓	✗	✗	✗
Elasticsearch 5.5.x	✓	✓	✗	✗	✗	✓	✓	✗	✓	✓	✗	✗	✗

**Figure 28:** Elastic Stack support matrix

The next step here was to install all the required dependencies to use the Elastic Stack.

## Elasticsearch

Elasticsearch 5.0 and above requires Java 8 to run, since Java did not exist on the system, it had to be installed.

### Installing Elasticsearch

```
azureuser@SecurityAuditVM:~/PCAP_Files$ java -version
```

```
Command 'java' not found, but can be installed with:
```

```
sudo apt install default-jre          # version 2:1.11-72, or
sudo apt install openjdk-11-jre-headless # version 11.0.17+8-1ubuntu2~20.04
sudo apt install openjdk-13-jre-headless # version 13.0.7+5-0ubuntu1~20.04
sudo apt install openjdk-16-jre-headless # version 16.0.1+9-1~20.04
sudo apt install openjdk-17-jre-headless # version 17.0.5+8-2ubuntu1~20.04
sudo apt install openjdk-8-jre-headless  # version 8u352-ga-1~20.04
```

```
azureuser@SecurityAuditVM:~/PCAP_Files$ uname -a
```

**Figure 29:** Java not found on system

```
# We updated the system first
sudo apt update

sudo apt install apt-transport-https
# Then we Installed Java-8

# After downloading Java 8, we added the path to our environment
sudo nano /etc/environment

source /etc/environment
java -version
```

```

azureuser@SecurityAudittVM:~$ java -version
openjdk version "1.8.0_352"
OpenJDK Runtime Environment (build 1.8.0_352-8u352-ga-1~20.04-b08)
OpenJDK 64-Bit Server VM (build 25.352-b08, mixed mode)
azureuser@SecurityAudittVM:~$ echo $JAVA_HOME
/usr/lib/jvm/java-8-openjdk-amd64
azureuser@SecurityAudittVM:~$ █

```

Figure 30: Java version and home path

The next stage of the experiment was to install Elasticsearch.

```

# Update system
sudo apt-get update

curl -L -O https://artifacts.elastic.co/downloads/elasticsearch/elasticsearch-
5.2.0.deb

sudo dpkg -i elasticsearch-5.2.0.deb
# Update system

```

```

azureuser@SecurityAudittVM:~$ curl -L -O https://artifacts.elastic.co/downloads/elasticsearch/elasticsearch-5.2.0.deb
  % Total    % Received % Xferd  Average Speed   Time     Time      Current
          Dload  Upload Total Spent   Left Speed
100 31.8M 100 31.8M    0     0  12.5M      0:00:02  0:00:02  ---:--- 12.5M
azureuser@SecurityAudittVM:~$ sudo dpkg -i elasticsearch-5.2.0.deb
Selecting previously unselected package elasticsearch.
(Reading database ... 73794 files and directories currently installed.)
Preparing to unpack elasticsearch-5.2.0.deb ...
Creating elasticsearch group... OK
Creating elasticsearch user... OK
Unpacking elasticsearch (5.2.0) ...
Setting up elasticsearch (5.2.0) ...
Not setting net/ipv4/conf/all/promote_secondaries (explicit setting exists).
Not setting net/ipv4/conf/default/promote_secondaries (explicit setting exists).
Processing triggers for systemd (245.4-4ubuntu3.18) ...
azureuser@SecurityAudittVM:~$ sudo apt-get update
Hit:1 http://azure.archive.ubuntu.com/ubuntu focal InRelease
Hit:2 http://azure.archive.ubuntu.com/ubuntu focal-updates InRelease
Hit:3 http://azure.archive.ubuntu.com/ubuntu focal-backports InRelease
Hit:4 http://azure.archive.ubuntu.com/ubuntu focal-security InRelease
Hit:5 http://ppa.launchpad.net/oisf/suricata-stable/ubuntu focal InRelease
Reading package lists... Done
azureuser@SecurityAudittVM:~$ █

```

Figure 31: Elasticsearch installation

Next, the daemon was restarted, enabled and started Elasticsearch. This ran successfully, denoted by the active status as seen in *Figure 32* below.

```

# Reload systemctl daemon
sudo systemctl daemon-reload

sudo /etc/init.d/elasticsearch start
# Enable the Elasticsearch service

# Start Elasticsearch

# Check Elasticsearch running status

```

```

azureuser@SecurityAudittVM:~$ sudo systemctl status elasticsearch.service
● elasticsearch.service - Elasticsearch
   Loaded: loaded (/lib/systemd/system/elasticsearch.service; enabled; vendor preset: enabled)
     Active: active (running) since Thu 2022-11-17 18:55:00 UTC; 2s ago
       Docs: http://www.elastic.co
   Process: 8894 ExecStartPre=/usr/share/elasticsearch/bin/elasticsearch-systemd-pre-exec (code=exi
 Main PID: 8895 (java)
    Tasks: 16 (limit: 19193)
      Memory: 2.1G
     CGroup: /system.slice/elasticsearch.service
             └─8895 /bin/java -Xms2g -Xmx2g -XX:+UseConcMarkSweepGC -XX:CMSInitiatingOccupancyFraction=70

Nov 17 18:55:00 SecurityAudittVM systemd[1]: Starting Elasticsearch...
Nov 17 18:55:00 SecurityAudittVM systemd[1]: Started Elasticsearch.

```

Figure 32: Elasticsearch running status

## Configuring Elasticsearch

Some of the configurations within Elasticsearch's config file needed to be modified.

```

# Stop Elasticsearch before configuration
sudo systemctl stop elasticsearch.service
# Editing Elasticsearch configurations

# Afterward we started the service
sudo systemctl start elasticsearch

```

`network.host` was uncommented and the IP to `0.0.0.0` was changed. The `http.port: 9200` from within the config file was also uncommented.

```

# Elasticsearch performs poorly when the system is swapping the memory.
#
# ----- Network -----
#
# Set the bind address to a specific IP (IPv4 or IPv6):
#
network.host: 0.0.0.0
#
# Set a custom port for HTTP:
#
http.port: 9200
#
# For more information, consult the network module documentation.
#
# ----- Discovery -----

```

Figure 33: Modifying Elasticsearch configuration

## Testing Elasticsearch

After the setup, a test on Elasticsearch was conducted.

```

// JSON response from Elasticsearch API endpoint
azureuser@SecurityAudittVM:~/PCAP_Files$ curl -X GET "localhost:9200"
{
  "cluster_name" : "elasticsearch",

```

```

"cluster_uuid" : "zIJMKN3RkWMmgsfiAlMvg",
"number" : "5.2.0",
"build_date" : "2017-02-24T17:26:45.835Z",
"lucene_version" : "6.4.0"
>tagline" : "You Know, for Search"
}

```

The JSON response shows that Elasticsearch has been configured correctly!

## Logstash

In this section of the report, Logstash is installed. This is "a tool that collects data from different sources. The data it collects is parsed by Kibana and stored in Elasticsearch" (Mamidwar, S, 2022).

### Installing Logstash

```

# Downloading logstash

sudo dpkg -i logstash-5.2.0.deb
# Starting Logstash

# Enabling Logstash

# Checking Logstash status
sudo systemctl status logstash

```

```

Setting up logstash (1.8.3-1-1) ...
azureuser@SecurityAuditVM:~/PCAP_Files$ sudo systemctl start logstash
azureuser@SecurityAuditVM:~/PCAP_Files$ sudo systemctl enable logstash
Created symlink /etc/systemd/system/multi-user.target.wants/logstash.service → /lib/systemd/system/logstash.service.
azureuser@SecurityAuditVM:~/PCAP_Files$ sudo systemctl status logstash
● logstash.service - logstash
   Loaded: loaded (/lib/systemd/system/logstash.service; enabled; vendor preset: enabled)
   Active: active (running) since Wed 2022-11-16 17:34:56 UTC; 21s ago
     Main PID: 48434 (java)
        Tasks: 21 (limit: 19193)
       Memory: 583.1M
      CGroup: /system.slice/logstash.service
              └─48434 /usr/share/logstash/jdk/bin/java -Xms1g -Xmx1g -Djava.awt.headless=true -Dfile.encoding=UTF-8 -Dlogstash.log.level=info

```

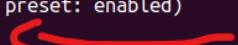


Figure 34: Successful installation of Logstash

### Configuring Logstash

Logstash was stopped before proceeding to configure it to read from the output of the `eve.json` file created by Suricata.

```

# Stop Logstash

# We changed permissions of eve.json and logstash file so it can digest data
# both were initially 777 then switched permissions back to 776

sudo chmod -R 776 /usr/share/logstash
# Creating configuration file

# Adding configurations

```

We added the data below into `/etc/logstash/conf.d/logstash.conf`.

```

input {
file {

    codec => "json"

}

}

filter {

date {

}

code => ""

    event.set('[fileinfo][type]', event.get('[fileinfo]
[magic]').to_s.split(',')[0])

    ""

}

ruby{

    if event.get('[event_type]') == 'alert'

        if (sp.length == 2) and /\A\d+\z/.match(sp[1])

            end

            ""

    }

    if [src_ip]  {

```

```

geoip {
    target => "geoip"
    add_field => [ "[geoip][coordinates]", "%{[geoip][longitude]}" ]
}

convert => [ "[geoip][coordinates]", "float" ]

if ![geoip.ip] {
    geoip {
        target => "geoip"
        add_field => [ "[geoip][coordinates]", "%{[geoip][longitude]}" ]
    }

    convert => [ "[geoip][coordinates]", "float" ]
}

}

output {
    elasticsearch {
    }
}

```

Next the following changes are made, as seen in (*Figures: 35, 36, 37, 38*) below to the

`/etc/logstash/logstash.yml` file.

```
#  
# Settings can be specified either in hierarchical f  
#  
#     pipeline:  
#         batch:  
#             size: 1000  
#             delay: 20  
#  
# Or as flat keys:  
#  
#     pipeline.batch.size: 125  
#     pipeline.batch.delay: 5  
#
```

Figure 35: Logstash pipeline configuration 1

```
"# ----- Data path -----  
#  
# Which directory should be used by logstash  
# for any persistent needs. Defaults to LOGS  
#  
path.data: /var/lib/logstash  
#  
# ----- Pipeline Settings -----
```

Figure 36: Logstash pipeline configuration 2

```
# ----- Pipeline Configuration Settings -----  
#  
# Where to fetch the pipeline configuration for the main p  
#  
path.config: /etc/logstash/conf.d  
#  
# Pipeline configuration string for the main pipeline  
#  
# config.string:  
#
```

Figure 37: Logstash pipeline configuration 3

```
# ----- Metrics Settings -----
#
# Bind address for the metrics REST endpoint
#
http.host: "0.0.0.0" L
#
# Bind port for the metrics REST endpoint, this option also accept a range
# (9600-9700) and logstash will pick up the first available ports.
#
http.port: 9600 L
#
# ----- Debugging Settings -----
#
# Options for log.level:
#   * fatal
#   * error
#   * warn
#   * info (default)
#   * debug
#   * trace
#           I
# log.level: info I
path.logs: /var/log/logstash
#
# ----- Other Settings -----
```

**Figure 38:** Logstash pipeline configuration 4

# Testing Logstash

Next, Logstash was manually started as `systemctl` was causing some issues.

**Figure 39:** Logstash running successfully

## Kibana

Afterwards, Kibana was installed. "It is a graphical user interface for parsing and interpreting collected log files" (Mamidwar, S, 2022).

# Installing Kibana

```
# Download Kibana  
  
# Verify and extract kibana  
  
tar -xzf kibana-5.2.0-linux-x86.tar.gz  
cd kibana-5.2.0-linux-x86/
```

```
# Start Kibana
./bin/kibana
```

```
log [23:08:05.770] [info][status][plugin:kibana@5.2.0] Status changed from uninitialized to green - Ready
log [23:08:05.831] [info][status][plugin:elasticsearch@5.2.0] Status changed from uninitialized to yellow - Waiting for El
asticsearch
log [23:08:05.869] [info][status][plugin:console@5.2.0] Status changed from uninitialized to green - Ready
log [23:08:05.891] [info][status][plugin:elasticsearch@5.2.0] Status changed from yellow to green - Kibana index ready
log [23:08:06.047] [info][status][plugin:timelion@5.2.0] Status changed from uninitialized to green - Ready
log [23:08:06.052] [info][listening] Server running at http://localhost:5601
log [23:08:06.053] [info][status][ui settings] Status changed from uninitialized to green - Ready
```

**Figure 40:** Kibana running successfully

There was no need to make any change in the Kibana configuration file.

After setting up the configurations above, the Azure Portal needed to be logged into, then navigate to the **Virtual Machine, Settings**, then selected **Networking**, within this section an **inbound port rule** was created to allow public access to the server port at **5601** for the purpose of this project.

The screenshot shows the Azure portal's interface for managing a virtual machine's network settings. At the top, it displays the VM name: securityauditvm732\_z1. Below this, under 'IP configuration', it shows 'ipconfig1 (Primary)'. In the 'Network Interface' section, there is a note about a network security group: 'Network security group SecurityAuditVM-nsg (attached to network interface: securityauditvm732\_z1) Impacts 0 subnets, 1 network interfaces'. Under 'Inbound port rules', a new rule is being added, highlighted with a red border. The rule details are as follows:

Priority	Name	Port	Protocol	Source	Destination	Action	More Options
300	SSH	22	TCP	Any	Any	Allow	...
320	HTTP	80	TCP	Any	Any	Allow	...
330	AllowAnyCustom5601Inbound	5601	Any	Any	Any	Allow	...
65000	AllowVnetInBound	Any	Any	VirtualNetwork	VirtualNetwork	Allow	...
65001	AllowAzureLoadBalancerInBound	Any	Any	AzureLoadBalancer	Any	Allow	...
65500	DenyAllInBound	Any	Any	Any	Any	Deny	...

**Figure 41:** Setting up inbound port rules for Kibana on Azure

After the changes, Kibana's service were restarted and it ran successfully.

## Testing Kibana

To access Kibana, a web browser was opened and browsed to our Azure's public IP address and port

```
# Kibana interface
http://20.106.233.71:5601
```

As shown in *Figure 42* below, Kibana has been installed, configured, and loaded correctly!

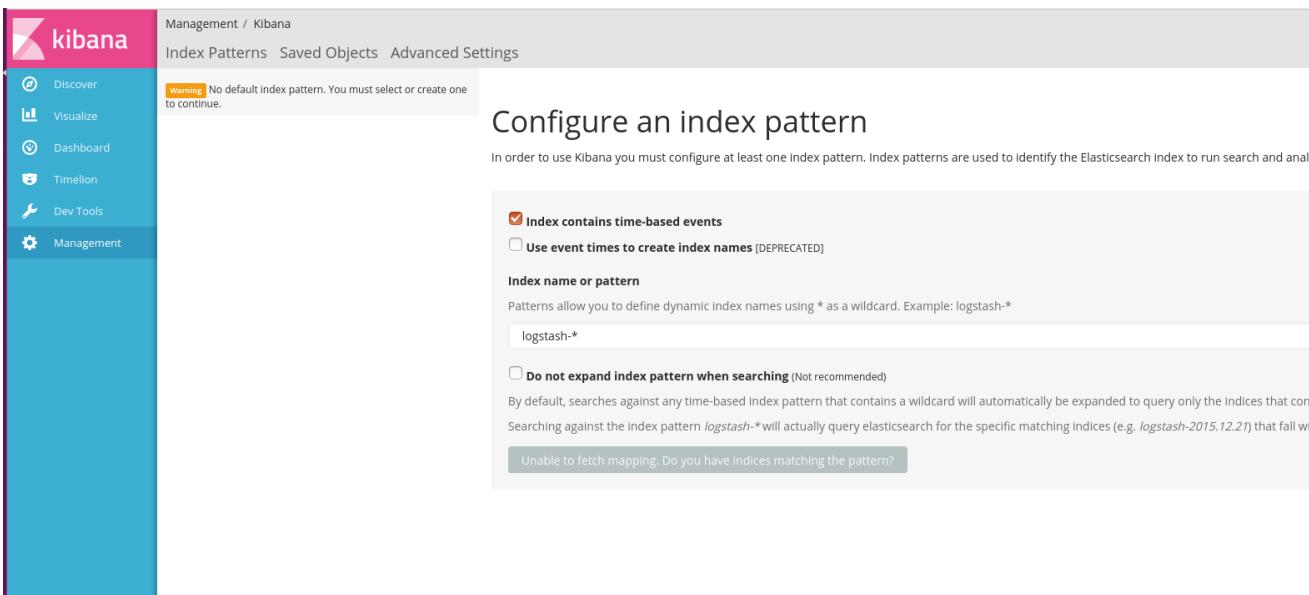


Figure 42: Kibana management screen

## Creating the Kibana Dashboard

Developers needed to create a default index pattern and download the dashboard, visualization, and saved search file.

At this point in the experiment all services had to be restarted, so that they can be synchronized to properly load all index patterns. As seen in *Figure 42* above, no index pattern for `logstash-*` was found which prevented the experiment from moving forward.

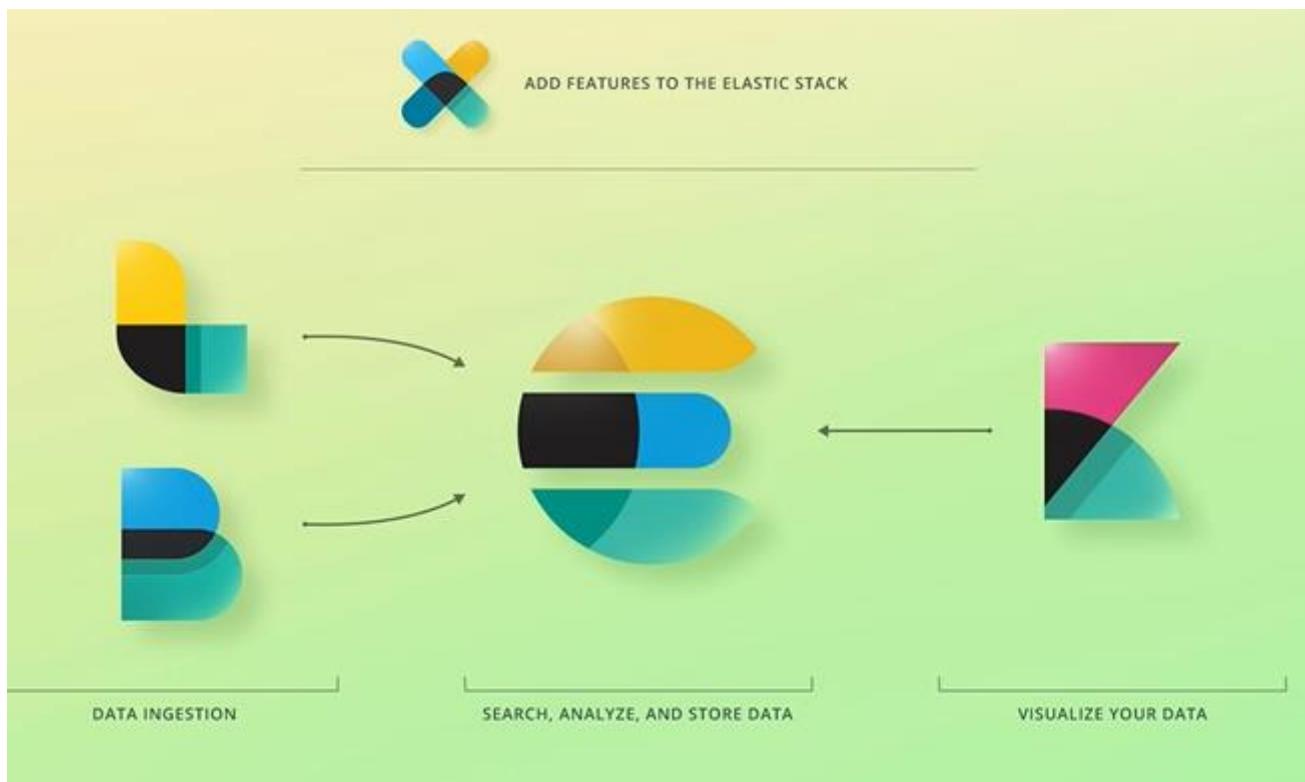


Figure 43: ELK/B Stack functionality

Next, after all services were running in the background, Suricata's PCAP ingestion process had to be re-ran, so that as events generated by Suricata reflect in the `eve.json` file,

Logstash will pick it up and parse it to Elasticsearch, so that it can be displayed by Kibana.

## Configure an index pattern

In order to use Kibana you must configure at least one index pattern. Index patterns are used to identify the Elasticsearch index to run searches against.

Index contains time-based events

Use event times to create index names [DEPRECATED]

**Index name or pattern**

Patterns allow you to define dynamic index names using \* as a wildcard. Example: logstash-\*

logstash-\*

Do not expand index pattern when searching (Not recommended)

By default, searches against any time-based index pattern that contains a wildcard will automatically be expanded to query only the indices matching the pattern. Searching against the index pattern `logstash-*` will actually query Elasticsearch for the specific matching indices (e.g. `logstash-2015.12..`)

**Time-field name** refresh fields

@timestamp

Create

Figure 44: Access to create index-pattern after Suricata PCAP ingestion

Index Patterns Saved Objects Advanced Settings

+ Add New

★ logstash-\*

Filter

fields (502) scripted fields (0) source filters (0)

name	type	format
files.tx_id	number	
geolp.region_name.keyword	string	
dns.authorities.rrname	string	
files.stored	boolean	
path	string	
flow.reason	string	
geolp.region_code.keyword	string	
tcp.tcp_flags_tc.keyword	string	
dns.version	number	
tls.version	string	
dns.answers.rttms	string	

Figure 45: Created `logstash-*` pattern

Next, the dashboards were imported.

III THE DEFAULT LIST.

The screenshot shows a user interface for managing dashboards. At the top, there are three tabs: 'Dashboards (1)', 'Searches (1)', and 'Visualizatio'. The 'Dashboards (1)' tab is highlighted with a blue border. Below the tabs is a search bar with the placeholder text 'Search...'. Underneath the search bar is a list item with a checkbox next to the title 'Sample Suricata Alert Dashboard'. The entire list item is enclosed in a thin horizontal line.

Figure 46: Sample Suricata Alert Dashbort import

in the default list.

The screenshot shows a user interface for managing searches. At the top, there are three tabs: 'Dashboards (1)', 'Searches (1)', and 'Visualizatio'. The 'Searches (1)' tab is highlighted with a blue border. Below the tabs is a search bar with the placeholder text 'Search...'. Underneath the search bar is a list item with a checkbox next to the title 'Alert Summary'. The entire list item is enclosed in a thin horizontal line.

Figure 47: Alert summary search import

Dashboards (1)    Searches (1)    Visualizations (7)

Title

Alert by GeIP

Number of Alerts

Top 10 Alerts

Top 20 DestIP - Alerts

Top 20 DestPort - Alerts

Top 20 Script - Alerts

Top 20 SrcPort - Alerts

**Figure 48:** Visualization imports

## Visualizing Alerts in Kibana

As shown in *Figure 49* below, the output of the **Discover** tab in Kibana shows a spike-bar in the years and dates for when the **QakBot** malware hit the network. It also, shows that it ingests the logs from Suricata `eve.json` properly.



**Figure 49:** Bar chart showing the year and date of QakBot malware attack



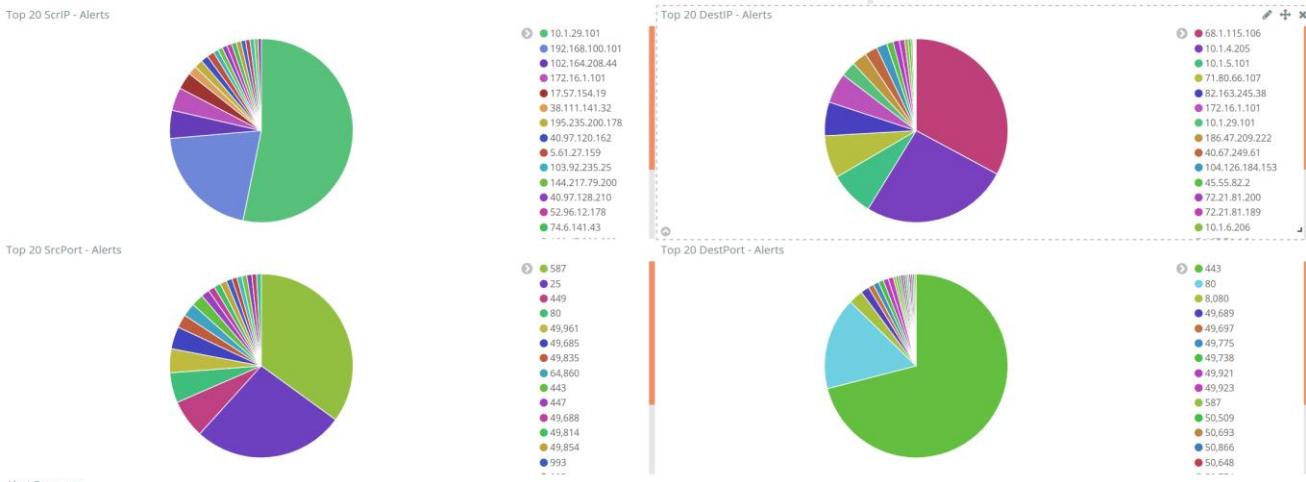
Figure 50: Alert by Geo-IP

Top 10 Alerts	
alert.signature.keyword: Descending	Count
ET JA3 Hash - [Abuse.ch] Possible Quakbot	1,548
SURICATA Applayer Detect protocol only one direction	1,188
ET POLICY OpenSSL Demo CA - Internet Widgits Pty (O)	168
ET MALWARE Win32/Emotet CnC Activity (POST) M9	142
ET USER_AGENTS Microsoft Device Metadata Retrieval Client User-Agent	140
ET MALWARE Win32/Emotet CnC Activity (POST) M8	126
ET JA3 Hash - [Abuse.ch] Possible Gozi	84
SURICATA SMTP invalid reply	78
SURICATA Applayer Wrong direction first Data	50
ET POLICY PE EXE or DLL Windows file download HTTP	42

Figure 51: Top 10 Alerts



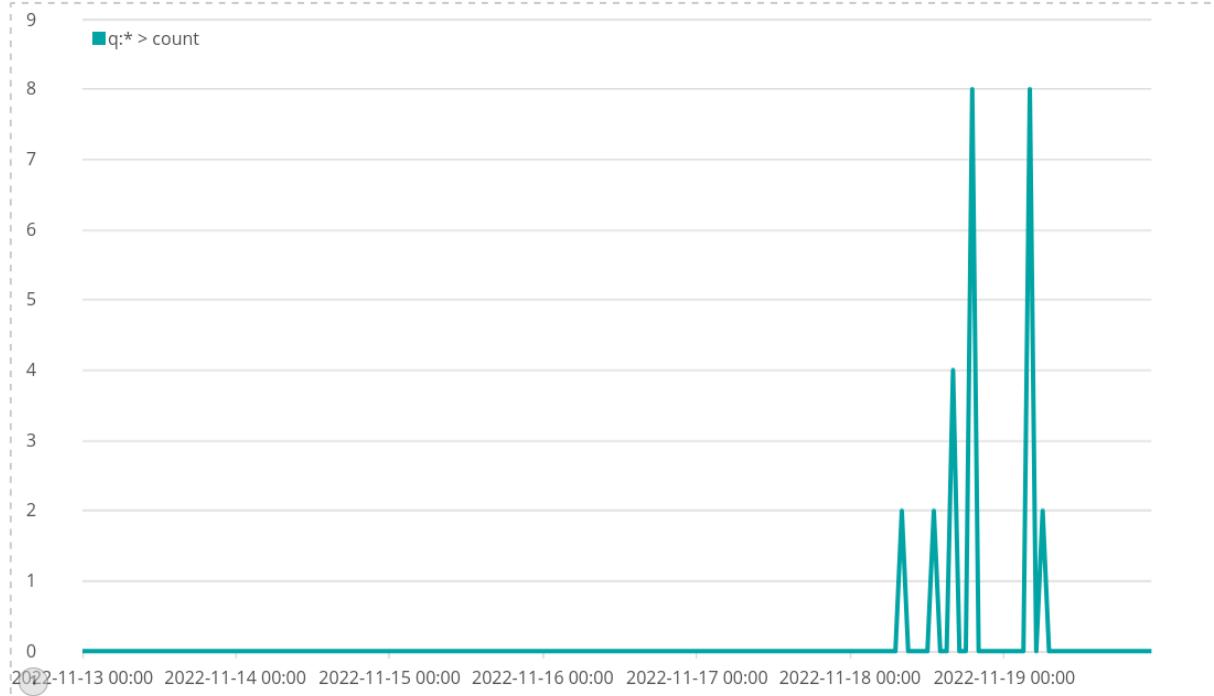
Figure 52: Number of Alerts



**Figure 53:** Alerts by `src-dest-IP` and `src-dest-port`

Alert Summary					1	2	3	4	5	...10	»	
Time	alert.signature	src_ip	src_port	dest_ip	dest_port							
▶ January 6th 2021, 10:17:31.562	ET CNC Feodo Tracker Reported CnC Server	10.1.6.206	49,845	167.71.4.0	8,080							
▶ January 6th 2021, 10:17:31.562	ET CNC Feodo Tracker Reported CnC Server	10.1.6.206	49,845	167.71.4.0	8,080							
▶ January 6th 2021, 10:17:31.562	ET CNC Feodo Tracker Reported CnC Server	10.1.6.206	49,845	167.71.4.0	8,080							
▶ January 6th 2021, 10:17:31.562	ET CNC Feodo Tracker Reported CnC Server	10.1.6.206	49,845	167.71.4.0	8,080							
▶ January 6th 2021, 10:17:31.562	ET CNC Feodo Tracker Reported CnC Server	10.1.6.206	49,845	167.71.4.0	8,080							
▶ January 6th 2021, 10:17:31.562	ET CNC Feodo Tracker Reported CnC Server	10.1.6.206	49,845	167.71.4.0	8,080							
▶ January 6th 2021, 10:17:31.562	ET CNC Feodo Tracker Reported CnC Server	10.1.6.206	49,845	167.71.4.0	8,080							
▶ January 6th 2021, 08:43:00.227	ET CNC Feodo Tracker Reported CnC Server	10.1.6.206	49,794	167.71.4.0	8,080							
▶ January 6th 2021, 08:43:00.227	ET CNC Feodo Tracker Reported CnC Server	10.1.6.206	49,794	167.71.4.0	8,080							

**Figure 54:** Alert Summary



**Figure 55:** Timeline graph

## Analyzing Alerts in Kibana

As seen in the *Visualizing Alerts in Kibana* section the benefits of the Elastic Stack is beyond question, as the logs generated by Suricata have been displayed in a visually and graphically grouped manner that makes analyzing, assimilation, and understanding easy. Suricata does a great job in detecting intrusions (for an open source software), but by parsing the log output to the Elastic Stack more sense can be made out of the data. As seen above, Kibana displays: the **number of Alerts** (3,850), the **attack by geolocation**, the **date of the attack** (2019-12-31/2020-12-31), the **top ten alerts** (in which it immediately picks out QakBot and its infection process in a glance), and many more information!

Here, the aim is to briefly analyze the data generated by Kibana.

In addition to the discovered **9312.zip** file by Suricata (see *Figure 27*) which contains the malicious VBS file that retrieves the QakBot malware (see *Figure 24*), as shown in *Figure 56* and *Figure 57*, it is clear that Kibana displays alerts of a potential second stage download (presumably the malware retrieved by the malicious VBS file) in a more human readable and noticeable way.

▶ January 6th 2021, 08:42:26.512 ET INFO EXE - Served Attached HTTP	103.92.235.25	80	10.1.6.206	49,775
▶ January 6th 2021, 08:42:26.512 ET POLICY PE EXE or DLL Windows file download HTTP	103.92.235.25	80	10.1.6.206	49,775
▶ January 6th 2021, 08:42:26.512 ET INFO EXE - Served Attached HTTP	103.92.235.25	80	10.1.6.206	49,775
▶ January 6th 2021, 08:42:26.512 ET INFO Executable Retrieved With Minimal HTTP Headers - Potential Second Stage Download	103.92.235.25	80	10.1.6.206	49,775
▶ January 6th 2021, 08:42:26.512 ET INFO Executable Retrieved With Minimal HTTP Headers - Potential Second Stage Download	103.92.235.25	80	10.1.6.206	49,775
▶ January 6th 2021, 08:42:26.512 ET POLICY PE EXE or DLL Windows file download HTTP	103.92.235.25	80	10.1.6.206	49,775
▶ January 6th 2021, 08:42:26.512 ET INFO Executable Retrieved With Minimal HTTP Headers - Potential Second Stage Download	103.92.235.25	80	10.1.6.206	49,775
▶ January 6th 2021, 08:42:26.512 ET INFO Executable Retrieved With Minimal HTTP Headers - Potential Second Stage Download	103.92.235.25	80	10.1.6.206	49,775

**Figure 56:** Potential Second Stage Download and DLL Windows file download HTTP

t http.hostname	seo.udaipurkart.com
t http.http_content_type	application/octet-stream
t http.http_method	GET
# http.length	45,044
t http.protocol	HTTP/1.1
# http.status	200
t http.url	/rx-5700-6hn7/Sgms/
t metadata.flowbits	min.gethttp, ET.http.binary
t path	/var/log/suricata/eve.json
# pcap_cnt	131,231
t proto	TCP
t src_ip	103.92.235.25
# src_port	80

**Figure 57:** HTTP Headers for Downloaded File

The attack chain occurs continually in a loop.

▶ January 6th 2021, 08:42:27.328	ET INFO EXE IsDebuggerPresent (Used in Malware Anti-Debugging)	103.92.235.25	80	10.1.6.206	49,775
▶ January 6th 2021, 08:42:27.328	ET INFO EXE IsDebuggerPresent (Used in Malware Anti-Debugging)	103.92.235.25	80	10.1.6.206	49,775
▶ January 6th 2021, 08:42:27.328	ET INFO EXE IsDebuggerPresent (Used in Malware Anti-Debugging)	103.92.235.25	80	10.1.6.206	49,775
▶ January 6th 2021, 08:42:27.328	ET INFO EXE IsDebuggerPresent (Used in Malware Anti-Debugging)	103.92.235.25	80	10.1.6.206	49,775
▶ January 6th 2021, 08:42:27.328	ET INFO EXE IsDebuggerPresent (Used in Malware Anti-Debugging)	103.92.235.25	80	10.1.6.206	49,775

**Figure 58: Alerts informing usage of a Malware Anti-Degubbung EXE**

```
# alert.severity          * 3
t alert.signature        * ET INFO EXE IsDebuggerPresent (Used in Malware Anti-Debugging)
# alert.signature_id      * 2,015,744
t app_proto               * http
t community_id            * 1:to7YeekESWCT2IQYVYRc/E+Qdu8=
t dest_ip                 * 10.1.6.206
# dest_port                * 49,775
t event_type              * alert
? files                   {
    "filename": "nDURg8uFD5hl.dll",
    "size": 76792,
    "stored": false,
    "state": "UNKNOWN",
    "tx_id": 0,
    "gaps": false,
    "sid": []
}
▲ ↴
```

**Figure 59: Malicious DLL File and Potential QakBot malware**

In *Figure 58* and *Figure 59*, there are continuous requests to download the malicious DLL file which is believed to be the QakBot malware.

Time	alert.signature	src_ip	src_port	dest_ip	dest_port
▶ January 6th 2021, 10:17:31.562	ET CNC Feodo Tracker Reported CnC Server	10.1.6.206	49,845	167.71.4.0	8,080
▶ January 6th 2021, 10:17:31.562	ET CNC Feodo Tracker Reported CnC Server	10.1.6.206	49,845	167.71.4.0	8,080
▶ January 6th 2021, 10:17:31.562	ET CNC Feodo Tracker Reported CnC Server	10.1.6.206	49,845	167.71.4.0	8,080
▶ January 6th 2021, 10:17:31.562	ET CNC Feodo Tracker Reported CnC Server	10.1.6.206	49,845	167.71.4.0	8,080
▶ January 6th 2021, 10:17:31.562	ET CNC Feodo Tracker Reported CnC Server	10.1.6.206	49,845	167.71.4.0	8,080
▶ January 6th 2021, 10:17:31.562	ET CNC Feodo Tracker Reported CnC Server	10.1.6.206	49,845	167.71.4.0	8,080
▶ January 6th 2021, 08:43:00.227	ET CNC Feodo Tracker Reported CnC Server	10.1.6.206	49,794	167.71.4.0	8,080
▶ January 6th 2021, 08:43:00.227	ET CNC Feodo Tracker Reported CnC Server	10.1.6.206	49,794	167.71.4.0	8,080

**Figure 60: Potential Execution of CnC Server (QakBot DLL)**

```

t alert.metadata.signature_severity   Q Q □ * Major
t alert.metadata.tag                 Q Q □ * Banking_Trojan
t alert.metadata.updated_at         Q Q □ * 2022_11_17
# alert.rev                         Q Q □ * 6,649
# alert.severity                   Q Q □ * 1
t alert.signature                  Q Q □ * ET CNC Feodo Tracker Reported CnC Server
# alert.signature_id               Q Q □ * 2,404,306
t community_id                     Q Q □ * 1:lhcDElazEZDXC3/Krwn/cAffIBg=
t dest_ip                          Q Q □ * 167.71.4.0
# dest_port                        Q Q □ * 8,080
t event_type                       Q Q □ * alert

```

Figure 61: QakBot (Banking Trojan) Signature in Kibana

```

t geoip.region_code                Q Q □ * NY
t geoip.region_name               Q Q □ * New York
t geoip.timezone                  Q Q □ * America/New_York
t host                            Q Q □ * ubuntu
t metadata.flowbits              Q Q □ * ET.Evil, ET.BotccIP
t path                            Q Q □ * /var/log/suricata/eve.json
# pcap_cnt                         Q Q □ * 135,322
t proto                           Q Q □ * TCP
t src_ip                          Q Q □ * 10.1.6.206
# src_port                         Q Q □ * 49,845
t tags                            Q Q □ * _geoip_lookup_failure
@ timestamp                       Q Q □ * January 6th 2021, 10:17:31.562
t type                            Q Q □ * SuricataIDPS

```

Figure 62: Geolocation Data of Host the Executed Malware

Obviously, *Figures 60, 61, and 62* shows that the QakBot malware had begun its post infection activities at this stage, which shows that it had successfully exploited an Ubuntu node on a network in New York, USA!

This is how multiple open-source tools can be combined to successfully detect and analyze intrusions on a network.

# **Recommendations**

International Organization for Standardization. (2013). Section 12.2.1 states that we need to: "scan electronic mail attachments and downloads for malware before use; this scan should be carried out at different places, e.g. at electronic mail servers, desk top computers and when entering the network of the organization;"

Therefore, this attack would have been prevented if the user(s) followed the ISO controls set against malware.

# Personal Reflection

This experiment taught the developers a lot about security monitoring and intrusion detection. The process of using an IDS to ingest and log data (to detect attacks on the network) was learned, and how that data can be graphically represented by combining multiple open source tools (Azure, Logstash, Elasticsearch, and Kibana).

During this experiment, issues were encountered, some of which are:

- The Microsoft tutorial did not show how to properly setup and configure Suricata, and the Elastic Stack. This information had to be learnt from a different resource. Although challenging, this gave the developers more understanding of the inner workings of Suricata.
- As seen in **Figure 28**, only Ubuntu v16.04 perfectly works with Elasticsearch v5.2.\*. A failure always occurs when the Elasticsearch service is ran on versions > v16.04. A way had to be figured out to get the machine, as every Ubuntu v16.04 on Azure requires payment to use (only versions higher than that: v18.04, v20.04, and v22.04 are free). The Microsoft documentation does not state this.
- Running different releases of Kibana and Elasticsearch (e.g. Kibana 5.x and Elasticsearch 6.x) is not supported, nor is running a version of Kibana that is newer than the version of Elasticsearch (e.g. Kibana 5.1 and Elasticsearch 5.0).

Regardless of all the errors, the general experience and practical knowledge gained, of setting up, configuring, and deploying an IDS and using a data visualization stack to view logs produced by the IDS was very intriguing, helpful as much as necessary for network monitoring.

# Conclusion

The benefits of the Elastic Stack in network monitoring is beyond question, as it makes the job easier. The experiment shows basic configuration of the stack and Suricata. The utilization of the open source tools helped to practically analyze malicious network traffic containing events that correlate to the operations of the QakBot malware. Not only did the stack and IDS help analyze, they also identified and successfully detected the malware execution process and attack.

The Elastic Stack and Suricata picked up on these malicious activities and graphically displayed the alerts in a grouped manner making it easy to understand.

In all, it was a great experience. Learning how to setup and configure an IDS, data visualization software, read attack logs, and more.

# References

- Duncan, B. (2020). *Wireshark Tutorial: Examining Qakbot Infections*. Unit 42.  
<https://unit42.paloaltonetworks.com/tutorial-qakbot-infection/>
- Duncan, B. (2021). *Wireshark Tutorial: Examining Emotet Infection Traffic*. Unit 42.  
<https://unit42.paloaltonetworks.com/wireshark-tutorial-emotet-infection/>
- The Honeynet Project. (2020). *EscapeRoom*. CyberDefenders. <https://cyberdefenders.org/>
- Microsoft. (2022). *Perform network intrusion detection with Network Watcher and open source tools*. <https://learn.microsoft.com/en-us/azure/network-watcher/network-watcher-intrusion-detection-open-source-tools>
- 3CORESec. (2021). *3CORESec/testmynids.org: A website and framework for testing NIDS detection*. GitHub. <https://github.com/3CORESec/testmynids.org>
- Mamidwar, S. (2022). *How to Install Elastic Stack 8 on Ubuntu 20.04 LTS* FOSSTechNix.  
<https://www.fosstechnix.com/how-to-install-elastic-stack-8-on-ubuntu-20-04/>
- Elastic. (2022). *Support Matrix*. <https://www.elastic.co/support/matrix>
- International Organization for Standardization. (2013). *Information technology Security techniques Code of practice for information security controls* ( ISO Standard No. 27002:2013(E) ). <https://www.iso.org/obp/ui#iso:std:iso-iec:27002:dis:ed-3:v1:en>

# Appendices

## Appendix A

### Setting up an Azure Virtual Machine

This appendix documents the necessary steps that were taken to successfully create an Azure virtual machine.

### Registration of Free Microsoft Azure Account

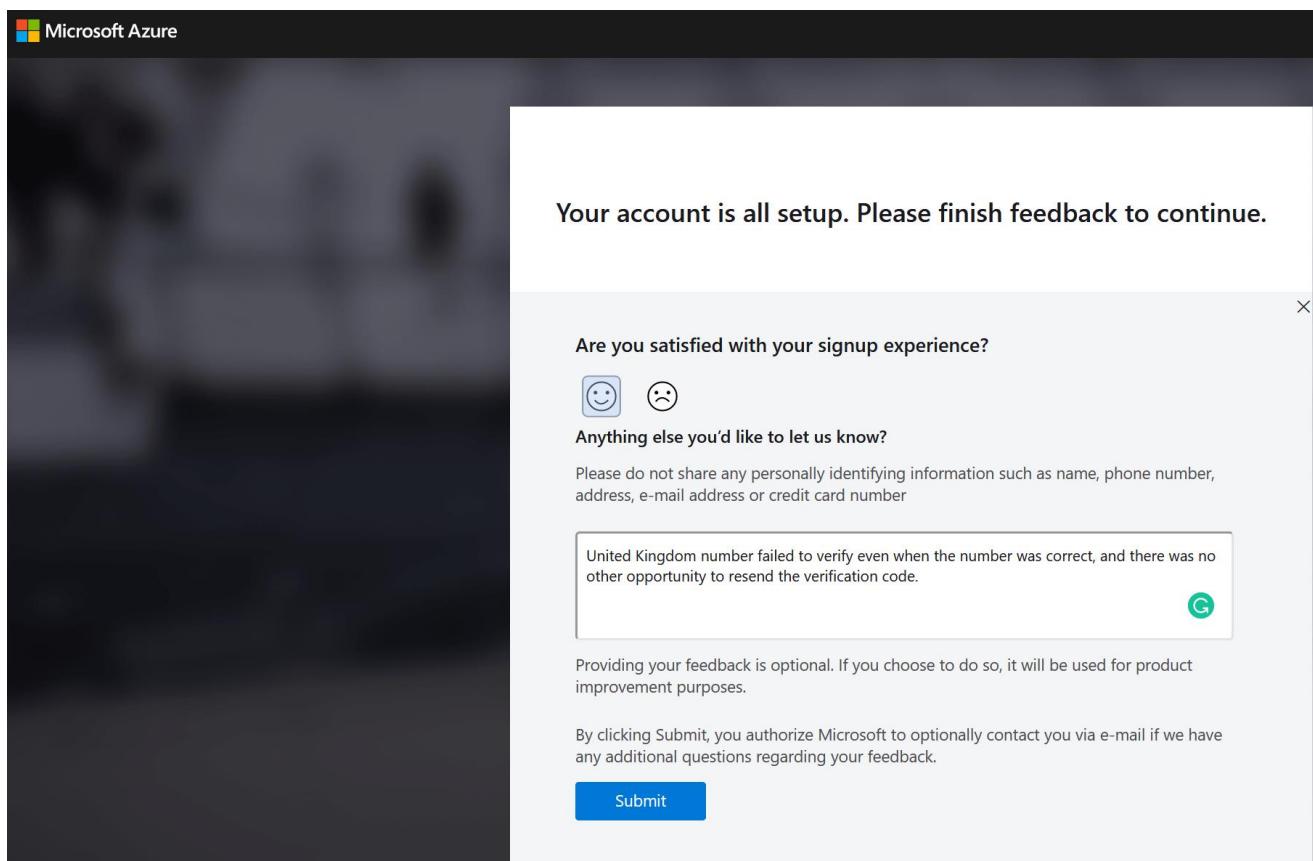


Figure 63: Complete Registration of Free Microsoft Azure Account.

### Creating the Virtual Machine

**i** This subscription may not be eligible to deploy VMs of certain sizes in certain regions.

### Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription *	Azure subscription 1
Resource group *	(New) SecurityAuditResourceGroup
	<a href="#">Create new</a>

### Instance details

Virtual machine name *	SecurityAuditVM
Region *	(US) East US
Availability options	Availability zone
Availability zone *	Zones 1
	<p> You can now select multiple zones. Selecting multiple zones will create one VM per zone. <a href="#">Learn more</a></p>
Security type	Standard
Image *	Ubuntu Server 20.04 LTS - Gen2 (free services eligible)
	<a href="#">See all images</a>   <a href="#">Configure VM generation</a>
VM architecture	<input type="radio"/> Arm64 <input checked="" type="radio"/> x64

**Figure 64:** Azure Machine - Project Details

**i** You are in the free trial period. Costs associated with this VM can be covered by any remaining credits on your subscription.  
[Learn more](#)

Size \* ⓘ

Standard\_E2s\_v3 - 2 vcpus, 16 GiB memory (US\$91.98/month)



[See all sizes](#)

#### Administrator account

Authentication type ⓘ

SSH public key

Password

**i** Azure now automatically generates an SSH key pair for you and allows you to store it for future use. It is a fast, simple, and secure way to connect to your virtual machine.

Username \* ⓘ

azureuser



SSH public key source

Generate new key pair



Key pair name \*

SecurityAuditVM\_key



#### Inbound port rules

Select which virtual machine network ports are accessible from the public internet. You can specify more limited or granular network access on the Networking tab.

Public inbound ports \* ⓘ

None

Allow selected ports

Select inbound ports \*

HTTP (80), SSH (22)



**⚠** This will allow all IP addresses to access your virtual machine. This is only recommended for testing. Use the Advanced controls in the Networking tab to create rules to limit inbound traffic to known IP addresses.

**Figure 65:** Azure VM Account settings and virtual machine memory size



#### OS disk

OS disk type * ⓘ	Premium SSD (locally-redundant storage) <span style="float: right;">▼</span>
Delete with VM ⓘ	<input checked="" type="checkbox"/>
Key management ⓘ	Platform-managed key <span style="float: right;">▼</span>
Enable Ultra Disk compatibility ⓘ	<input type="checkbox"/>

#### Data disks for SecurityAuditVM

You can add and configure additional data disks for your virtual machine or attach existing disks. This VM also comes with a temporary disk.

LUN	Name	Size (GiB)	Disk type	Host cache...	Delete with VM ⓘ
1	SecurityAuditVM_Data...	64	Premium SSD LRS	Read-only <span style="float: right;">▼</span>	<input type="checkbox"/>

[Create and attach a new disk](#)    [Attach an existing disk](#)

**Figure 66:** Virtual Machine Disk Size Allocation



#### Basics

Subscription	Azure subscription 1
Resource group	(new) SecurityAuditResourceGroup
Virtual machine name	SecurityAuditVM
Region	East US
Availability options	Availability zone
Availability zone	1
Security type	Standard
Image	Ubuntu Server 20.04 LTS - Gen2
VM architecture	x64
Size	Standard E2s v3 (2 vcpus, 16 GiB memory)
Authentication type	SSH public key
Username	azureuser
Key pair name	SecurityAuditVM_key
Public inbound ports	SSH, HTTP
Azure Spot	No

**Figure 67:** Basic Settings

## Disks

OS disk type	Premium SSD LRS
Use managed disks	Yes
Delete OS disk with VM	Enabled
Data disks	1
Delete data disk with VM	0 disks enabled
Ephemeral OS disk	No

## Networking

Virtual network	(new) SecurityAuditResourceGroup-vnet
Subnet	(new) default (10.0.0.0/24)
Public IP	(new) SecurityAuditVM-ip
Accelerated networking	On
Place this virtual machine behind an existing load balancing solution?	No
Delete public IP and NIC when VM is deleted	Disabled

**Figure 68:** Disks and Networking Settings

## Management

Microsoft Defender for Cloud	Basic (free)
System assigned managed identity	Off
Login with Azure AD	Off
Auto-shutdown	Off
Backup	Disabled
Enable hotpatch	Off
Patch orchestration options	Image Default

## Monitoring

Alerts	Off
Boot diagnostics	On
Enable OS guest diagnostics	Off

## Advanced

Extensions	None
VM applications	None
Cloud init	No
User data	No
Disk controller type	SCSI
Proximity placement group	None
Capacity reservation group	None

Figure 69: Management, Monitoring, and Advanced Settings

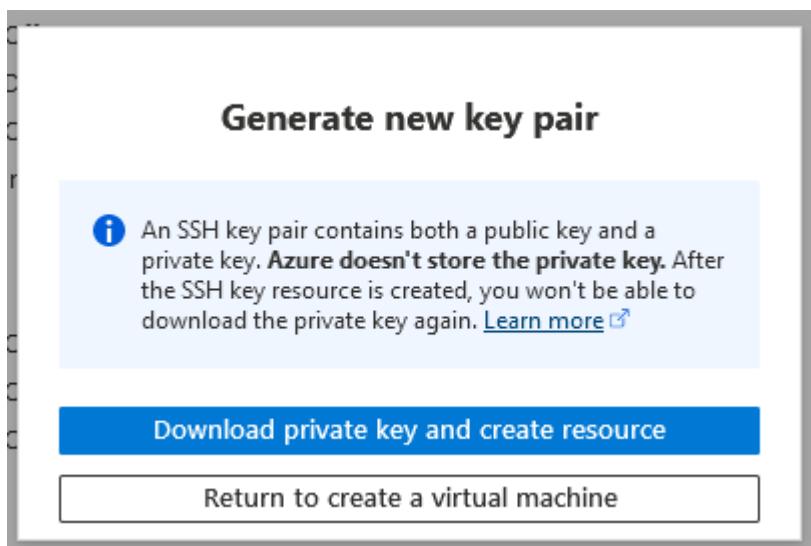
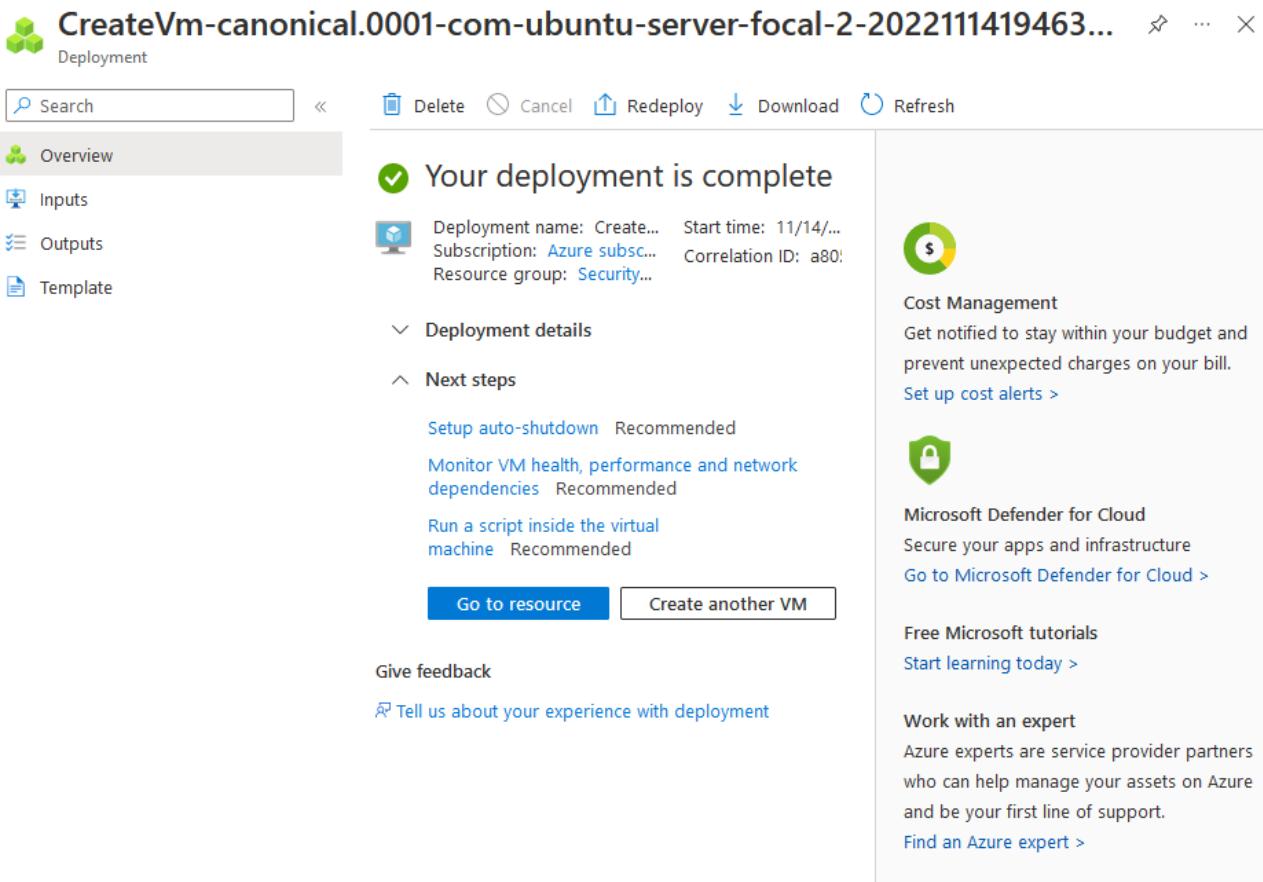


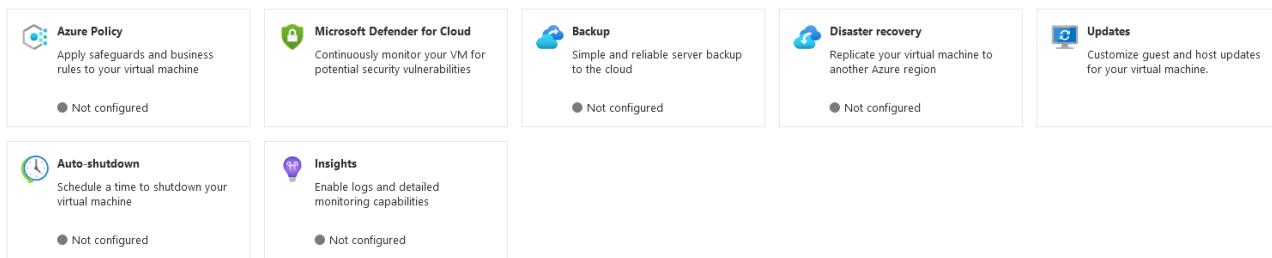
Figure 70: Generating the SSH Key Pairs



**Figure 71:** Complete Deployment of Azure VM

Properties		Monitoring	Capabilities (7)	Recommendations	Tutorials																																								
<p><b>Virtual machine</b></p> <table> <tbody> <tr><td>Computer name</td><td>SecurityAuditVM</td></tr> <tr><td>Health state</td><td>-</td></tr> <tr><td>Operating system</td><td>Linux (ubuntu 20.04)</td></tr> <tr><td>Publisher</td><td>canonical</td></tr> <tr><td>Offer</td><td>0001-com-ubuntu-server-focal</td></tr> <tr><td>Plan</td><td>20_04-lts-gen2</td></tr> <tr><td>VM generation</td><td>V2</td></tr> <tr><td>VM architecture</td><td>x64</td></tr> <tr><td>Agent status</td><td>Ready</td></tr> <tr><td>Agent version</td><td>2.8.0.11</td></tr> <tr><td>Host group</td><td>None</td></tr> <tr><td>Host</td><td>-</td></tr> <tr><td>Proximity placement group</td><td>-</td></tr> <tr><td>Colocation status</td><td>N/A</td></tr> <tr><td>Capacity reservation group</td><td>-</td></tr> </tbody> </table> <p><b>Availability + scaling</b></p> <table> <tbody> <tr><td>Availability zone</td><td>1</td></tr> <tr><td>Availability set</td><td>-</td></tr> <tr><td>Scale Set</td><td>-</td></tr> </tbody> </table> <p><b>Security type</b></p> <table> <tbody> <tr><td>Security type</td><td>Standard</td></tr> </tbody> </table> <p><b>Extensions + applications</b></p> <table> <tbody> <tr><td>Extensions</td><td>-</td></tr> </tbody> </table>						Computer name	SecurityAuditVM	Health state	-	Operating system	Linux (ubuntu 20.04)	Publisher	canonical	Offer	0001-com-ubuntu-server-focal	Plan	20_04-lts-gen2	VM generation	V2	VM architecture	x64	Agent status	Ready	Agent version	2.8.0.11	Host group	None	Host	-	Proximity placement group	-	Colocation status	N/A	Capacity reservation group	-	Availability zone	1	Availability set	-	Scale Set	-	Security type	Standard	Extensions	-
Computer name	SecurityAuditVM																																												
Health state	-																																												
Operating system	Linux (ubuntu 20.04)																																												
Publisher	canonical																																												
Offer	0001-com-ubuntu-server-focal																																												
Plan	20_04-lts-gen2																																												
VM generation	V2																																												
VM architecture	x64																																												
Agent status	Ready																																												
Agent version	2.8.0.11																																												
Host group	None																																												
Host	-																																												
Proximity placement group	-																																												
Colocation status	N/A																																												
Capacity reservation group	-																																												
Availability zone	1																																												
Availability set	-																																												
Scale Set	-																																												
Security type	Standard																																												
Extensions	-																																												
<p><b>Networking</b></p> <table> <tbody> <tr><td>Public IP address</td><td>20.106.233.71</td></tr> <tr><td>Public IP address (IPv6)</td><td>-</td></tr> <tr><td>Private IP address</td><td>10.0.0.4</td></tr> <tr><td>Private IP address (IPv6)</td><td>-</td></tr> <tr><td>Virtual network/subnet</td><td>SecurityAuditResourceGroup-vnet/default</td></tr> <tr><td>DNS name</td><td>Configure</td></tr> </tbody> </table> <p><b>Size</b></p> <table> <tbody> <tr><td>Size</td><td>Standard E2s v3</td></tr> <tr><td>vCPUs</td><td>2</td></tr> <tr><td>RAM</td><td>16 GiB</td></tr> </tbody> </table> <p><b>Disk</b></p> <table> <tbody> <tr><td>OS disk</td><td>SecurityAuditVM_OsDisk_1_879394ca3de547448b2962dd7c4debd7</td></tr> <tr><td>Encryption at host</td><td>Disabled</td></tr> <tr><td>Azure disk encryption</td><td>Not enabled</td></tr> <tr><td>Ephemeral OS disk</td><td>N/A</td></tr> <tr><td>Data disks</td><td>1</td></tr> </tbody> </table> <p><b>Auto-shutdown</b></p> <table> <tbody> <tr><td>Auto-shutdown</td><td>Not enabled</td></tr> <tr><td>Scheduled shutdown</td><td>-</td></tr> </tbody> </table> <p><b>Azure Spot</b></p> <table> <tbody> <tr><td>Azure Spot</td><td>-</td></tr> <tr><td>Azure Spot eviction policy</td><td>-</td></tr> </tbody> </table>						Public IP address	20.106.233.71	Public IP address (IPv6)	-	Private IP address	10.0.0.4	Private IP address (IPv6)	-	Virtual network/subnet	SecurityAuditResourceGroup-vnet/default	DNS name	Configure	Size	Standard E2s v3	vCPUs	2	RAM	16 GiB	OS disk	SecurityAuditVM_OsDisk_1_879394ca3de547448b2962dd7c4debd7	Encryption at host	Disabled	Azure disk encryption	Not enabled	Ephemeral OS disk	N/A	Data disks	1	Auto-shutdown	Not enabled	Scheduled shutdown	-	Azure Spot	-	Azure Spot eviction policy	-				
Public IP address	20.106.233.71																																												
Public IP address (IPv6)	-																																												
Private IP address	10.0.0.4																																												
Private IP address (IPv6)	-																																												
Virtual network/subnet	SecurityAuditResourceGroup-vnet/default																																												
DNS name	Configure																																												
Size	Standard E2s v3																																												
vCPUs	2																																												
RAM	16 GiB																																												
OS disk	SecurityAuditVM_OsDisk_1_879394ca3de547448b2962dd7c4debd7																																												
Encryption at host	Disabled																																												
Azure disk encryption	Not enabled																																												
Ephemeral OS disk	N/A																																												
Data disks	1																																												
Auto-shutdown	Not enabled																																												
Scheduled shutdown	-																																												
Azure Spot	-																																												
Azure Spot eviction policy	-																																												

**Figure 72:** Virtual Machine Properties

**Figure 73: Virtual Machine Capabilities****^ Essentials**Resource group ([move](#)) : [SecurityAuditResourceGroup](#)

Status : Running

Location : East US (Zone 1)

Subscription ([move](#)) : [Azure subscription 1](#)

Subscription ID : 7e5efc75-3d3f-43c5-bf24-6337fa995500

Availability zone : 1

Tags ([edit](#)) : [Click here to add tags](#)

Operating system : Linux (ubuntu 20.04)

Size : Standard E2s v3 (2 vcpus, 16 GiB memory)

Public IP address : [20.106.233.71](#)Virtual network/subnet : [SecurityAuditResourceGroup-vnet/default](#)DNS name : [Not configured](#)**Figure 74: VM Essentials**