

# Threat Intelligence for SOC

Is your organisation prepared to handle emerging threats like new malware IOCs or zero days? And in any case, can you determine unknown adversaries or apply known indicators from reliable sources in your Security Operations pipeline?

Such questions arise when you think of the ever-going cat-and-mouse game of threat actors and security analysts on a typical day. As a group working to secure an organisation, the security team is expected to be prepared to handle the never-ending evolution of threats and anticipate unknown possibilities of potential compromises. Doing all these might be challenging, but you are not alone in this battle.

## Learning Objectives:

In this room, we will highlight the impact of Threat Intelligence in the Security Operations pipeline; how important information shared across different groups can be utilised by your organisation. In addition, we will tackle topics such as the following throughout the room:

- Threat Intelligence Consumers and Producers
- Types of Threat Intelligence
- Utilising Threat Intelligence to Prevent and Detect malicious activities

## Room Prerequisites:

It is highly suggested to clear the following rooms first before proceeding with this room:

- Threat Intelligence Tools
- Introduction to Detection Engineering
- Tactical Detection
- Investigating with ELK 101

Let's put our intelligence to work and stay ahead of potential threats with Threat Intelligence.

## Threat Intelligence Feeds

### Threat Intelligence Recap

For a quick review, let's reiterate the definition of Threat Intelligence discussed in the Intro to Cyber Threat Intel room.

Threat Intelligence is the analysis of data and information using tools and techniques to generate meaningful patterns to mitigate against potential risks associated with existing or emerging threats targeting organisations, industries, sectors or governments.

There are different classifications of Threat Intelligence, and the primary types of it are:

- Strategic Intel: High-level intel that looks into the organisation's threat landscape and maps out the risk areas based on trends, patterns and emerging threats that may impact business decisions.
- Technical Intel: Examines evidence and artefacts of attacks an adversary uses. Incident Response teams can use this intel to create a baseline attack surface to analyse and develop defence mechanisms.
- Tactical Intel: Assesses adversaries' tactics, techniques, and procedures (TTPs). This intel can strengthen security controls and address vulnerabilities through real-time investigations.
- Operational Intel: Assesses an adversary's specific motives and intent to perform an attack. Security teams may use this intel to understand the critical assets available in the organisation (people, processes, and technologies) that threat actors may target.

These classifications may give you an idea of how you operate with the data. But our only focus in this room is on Technical Intel, utilising artefacts generated by adversaries to improve the Security Operations pipeline. This type is the most common of the four classes and is mainly known as IOC-based Threat Intelligence.

Before applying Threat Intelligence to Security Operations, let's first deal with your understanding of how organisations differ in roles regarding Threat Intelligence.

### **Consumers and Producers**

Do you build the knowledge base, or do you consume the knowledge of others?

The common notion of Threat Intelligence is the dataset of known bad IOCs collated by different entities. It may be malicious URLs hosting malware or IP addresses of suspicious connections. But would you know how this information is gathered for the disposal of security analysts? Let's first differentiate the concept of Producers and Consumers of Threat Intelligence.

### **Producers**

Threat Intelligence Producers gather, analyse and disseminate threat intelligence data for others and themselves. These Producers create reports, advisories, and resources that are shared within the broader cybersecurity community. This group includes cybersecurity vendors, research labs and organisations specialising in collecting and interpreting data on emerging cyber threats.

Now, the Producers typically collect data using various methods and techniques. Standard methods include network monitoring, which involves monitoring an

organisation's network traffic to identify abnormal behaviour from the inside or a honeypot server exposed externally.

Another example could be a collection of IOCs based on internal incidents handled by an organisation. These organisations expect a more significant number of incidents compared to small organisations with fewer assets to be compromised or user activity to be monitored. The results of these collections are then further analysed, attributed to potential threat actors, and published eventually to help other organisations.

These examples summarise that not every organisation can be a Producer. It requires a vast set of collected data, the capacity to determine expected normal behaviour, and the capability to analyse and pinpoint unknown potential threats.

## **Consumers**

On the other hand, Threat Intelligence Consumers are organisations or individuals who consume Threat Intelligence created by Producers. The information gathered from different sources is utilised to improve the organisation's security posture.

Example of an analyst being a threat intelligence consumer. How does this group typically leverage the intelligence data shared with them?

- Identifying vulnerabilities - Consumers can use published vulnerabilities discovered due to zero days launched by threat actors to identify vulnerabilities in an organisation's infrastructure. Advisories such as CVE publications are commonly utilized to determine if an organisation is impacted by it and apply mitigations if needed.
- Prevention and Detection - Consumers can use IOCs to prevent intrusions by blocking these artefacts or detect them by applying them to threat detection rules.
- Incident Response - Consumers can use intelligence data to respond more effectively to incidents as the data may confirm the likelihood of the attack and the potentially attributed tactics and techniques.
- Collaborating with others - Information sharing is not only for Producers but also for Consumers. Data analysis of IOCs may still require human assessment, so it is helpful to share information that is validated to be beneficial for Security Operations.

## **Are you a Consumer or a Producer?**

Assessing whether your organisation is a Threat Intelligence Consumer or Producer depends on the roles and responsibilities of your security team and the overall cybersecurity strategy of your organisation.

<b>Producer</b>	<b>Consumer</b>
Collect and analyse internal and external data to produce actionable threat intelligence that helps identify and prevent cyber threats.	Monitor the organisation's network and systems for potential security threats and vulnerabilities, and leverage external intelligence to supplement their analysis and understanding of those threats.
Create and distribute threat intelligence reports to other organisations, such as industry peers, regulators, or law enforcement agencies.	Use threat intelligence feeds and reports from third-party providers to identify potential security threats and vulnerabilities and integrate that information into your organisation's security posture.

Once you have defined your role, you may also consider assessing your current practices based on the following:

<b>Classification</b>	<b>Producer</b>	<b>Consumer</b>
<b>Understanding</b>	Evaluate the quality of the intelligence produced by your organisation, including the information's relevance, accuracy, and timeliness.	Assess your organisation's understanding of threat intelligence and whether it is effectively being used to enhance your organisation's security posture.
<b>Collection</b>	Evaluate your organisation's ability to collect and analyse data from various sources, including network logs, endpoint data, and other sources.	Evaluate your organisation's ability to collect and consume threat intelligence from various sources.
<b>Analytics</b>	Assess your security team's technical and analytical skills, including their ability to detect and analyse threats and communicate their findings to other groups.	Evaluate your organisation's ability to analyse and process the threat intelligence that is being collected.
<b>Application</b>	Evaluate your organisation's ability to respond to threats based on the threat intelligence produced.	Evaluate your organisation's ability to respond to threats identified through threat

		intelligence.
--	--	---------------

## Consuming Threat Intelligence

o conclude this task, we will hunt and consume IOCs provided by Threat Intelligence Producers.

Click on the Start Machine button at the top right corner of this task. The machine provides access to the following:

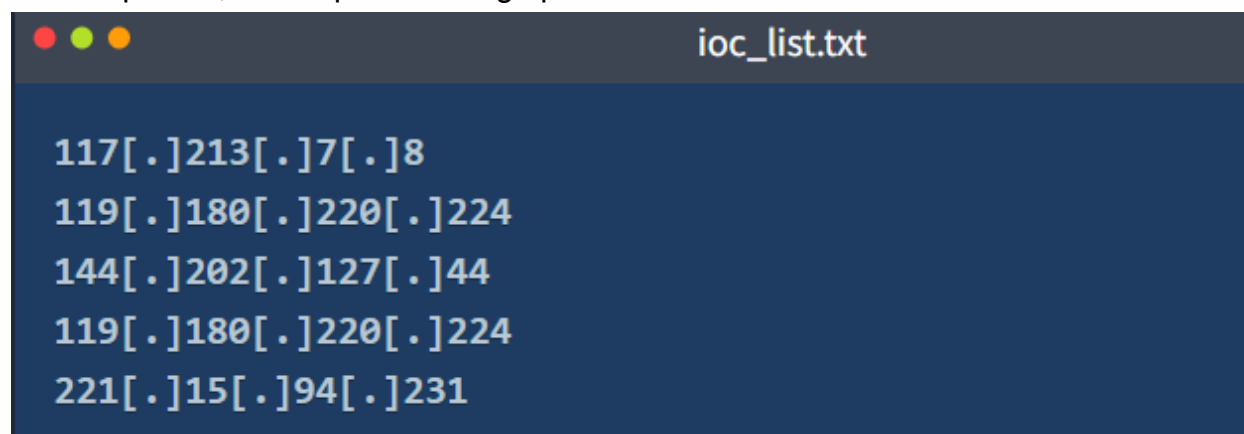
- Kibana Instance via `http://MACHINE_IP` with the following credentials:  
elastic:elastic

## Uncoder.io

Uncoder.io is an online tool that transforms Sigma rules, IOC lists, and other platform query syntaxes into custom hunting queries prepared for execution in SIEM and XDR. It is an easy-to-use tool that could assist us in hunting the following IOCs up for investigation. For IOCs, the tool accepts six different types of IOCs, namely:

- IPs
- Domains
- URLs
- Hashes
- Emails
- Files

Let's use the following set of IPs and feed them into Uncoder. Do take note that with recent updates, this requires setting up a free account on the [uncoder.io](https://uncoder.io) website.



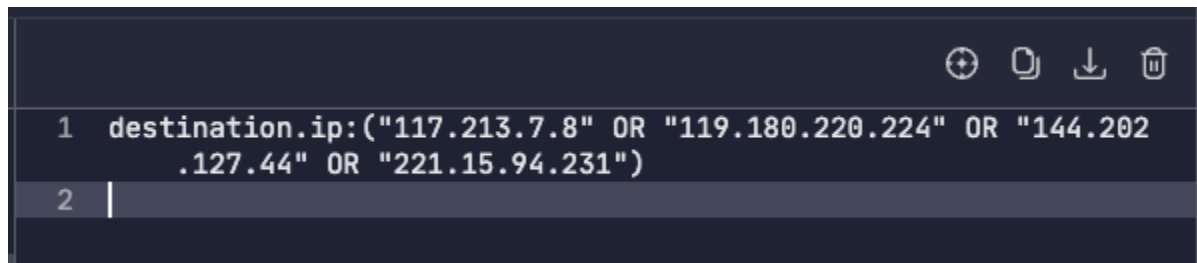
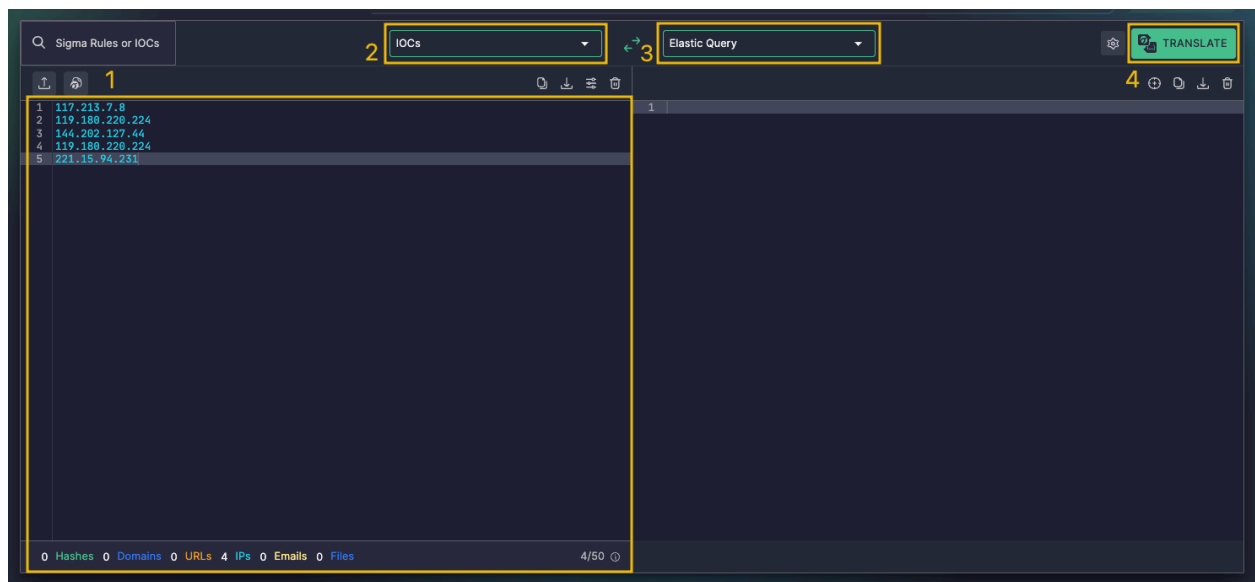
```
ioc_list.txt

117[.]213[.]7[.]8
119[.]180[.]220[.]224
144[.]202[.]127[.]44
119[.]180[.]220[.]224
221[.]15[.]94[.]231
```

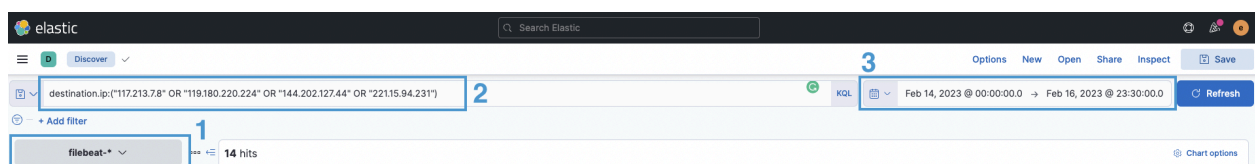
Follow along with the steps and the image below:

- Paste the list of IOCs. Note that defanged IPs will be cleaned, and redundancies will be removed, so only four IPs are detected, as shown below.
- Configure the source platform and set it to IOCs.
- Set the target platform to Elastic Query since the provided SIEM is built on that.

- Click translate and view the produced query syntax.



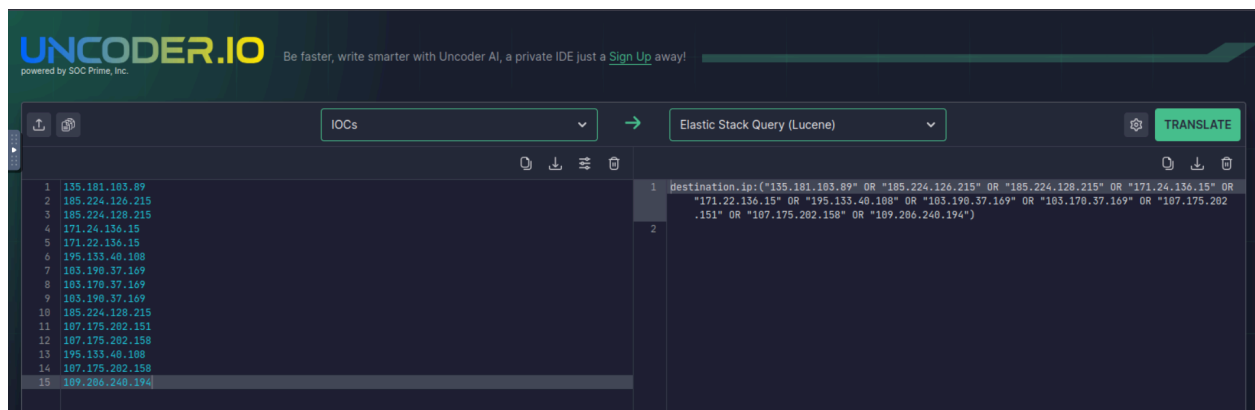
The result of using the tool can be utilised in our Kibana Instance via the Discover feature. Ensure that the query is under the filebeat-\* index and searches between 02/14/2023 and 02/17/2023.



To complete the task, answer the following questions using the set of IOCs below. You must use the same index and timeframe mentioned above.

```
ioc_list.txt

135[.]181[.]103[.]89
185[.]224[.]126[.]215
185[.]224[.]128[.]215
171[.]24[.]136[.]15
171[.]22[.]136[.]15
195[.]133[.]40[.]108
103[.]190[.]37[.]169
103[.]170[.]37[.]169
103[.]190[.]37[.]169
185[.]224[.]128[.]215
107[.]175[.]202[.]151
107[.]175[.]202[.]158
195[.]133[.]40[.]108
107[.]175[.]202[.]158
109[.]206[.]240[.]194
```



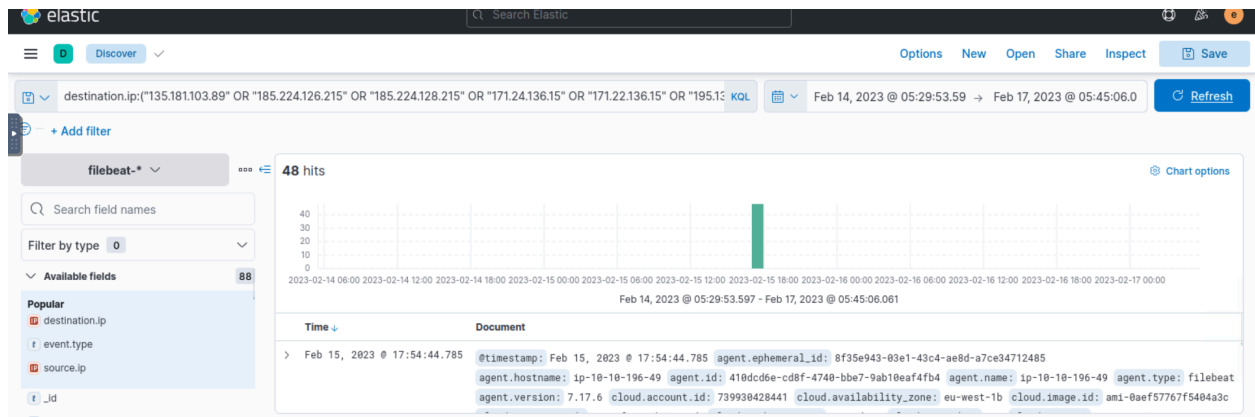
\*\*\*\*\*

Answer the questions below:

How many unique IP addresses were provided in the IOC list?

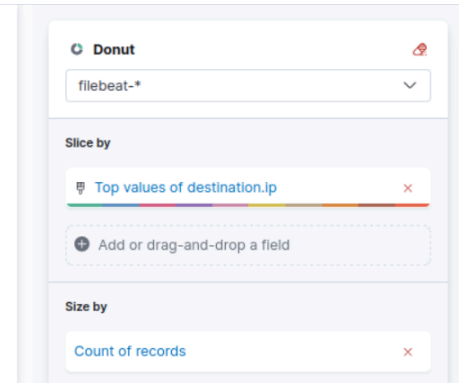
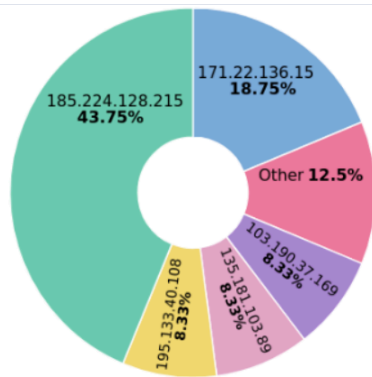
Answer: 11

Based on the set of IOCs, how many IOC hits were discovered in the logs?



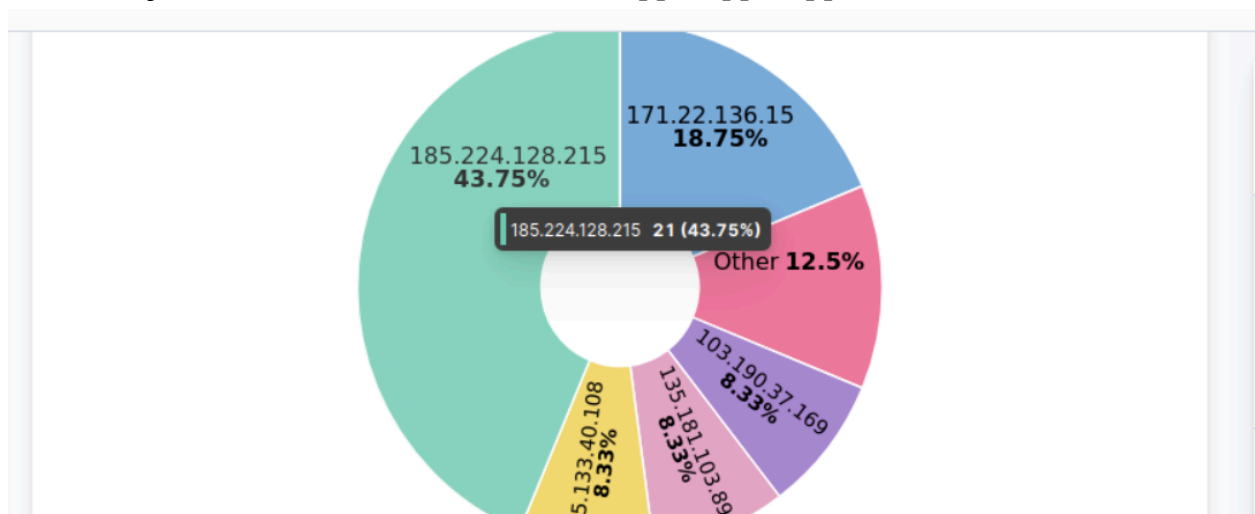
Answer: 48

Out of the total number of IOCs, how many unique IP addresses were discovered in the logs?



Answer: 7

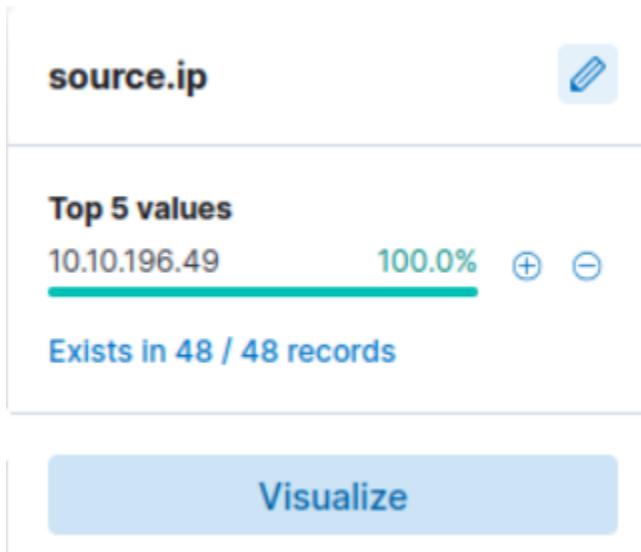
How many connections were made to 185.[.]224.[.]128.[.]215?



Answer: 21

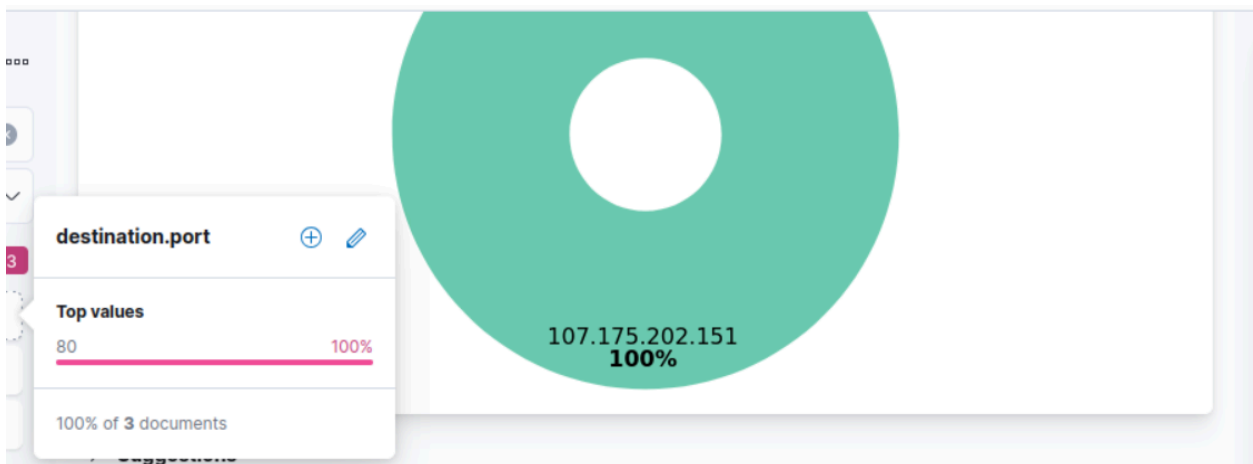


What is the IP address of the compromised host?



Answer: 10.10.196.49

What is the destination port of connections made to 107[.]175[.]202[.]151?



Answer: 80

## Intelligence-Driven Prevention

Your organisation has determined that you are a consumer of Threat Intelligence from reliable sources; your task is to apply the concepts of being a consumer by deploying controls to prevent threats in your infrastructure.

Using our current knowledge of Threat Intelligence, we will utilize the IOCs from reliable sources to deploy security controls that will prevent malicious activity in our infrastructure.

To start with, we can first simplify the types of IOCs that are commonly distinguished in Threat Intelligence feeds:

- Domains - Typically attributed to URLs used to host malicious files, C2 callbacks or email domains used for spam.
- IP Addresses - Commonly attributed to addresses known to execute attacks seen from external assets or outbound callbacks from malware.

### **IP Blocking via Firewall**

IP blocking is a well-known security measure that involves blocking ingress or egressing network traffic based on the device's IP address attempting to initiate a network connection. It is typically done using a Firewall, a security system that controls the traffic based on predetermined rules.

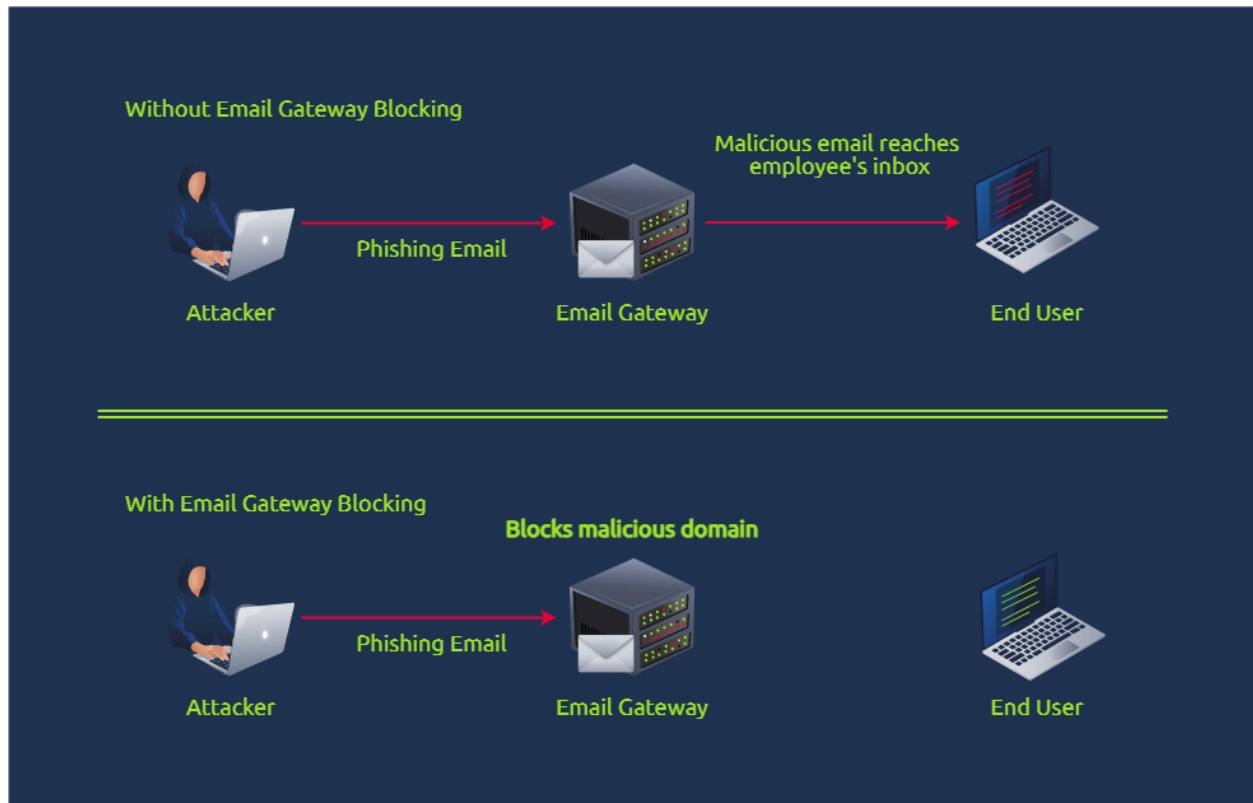
Configuring firewall rules could be overwhelming, but having a direction to deny connections from a known malicious IP address is a good start in preventing malicious connections, such as:

- Prevent intrusive connections against external applications that may affect service uptime or compromise via a known vulnerability.
- Prevent connection attempts to the threat actor's infrastructure after successful malware execution.



### Domain Blocking through Email Gateways

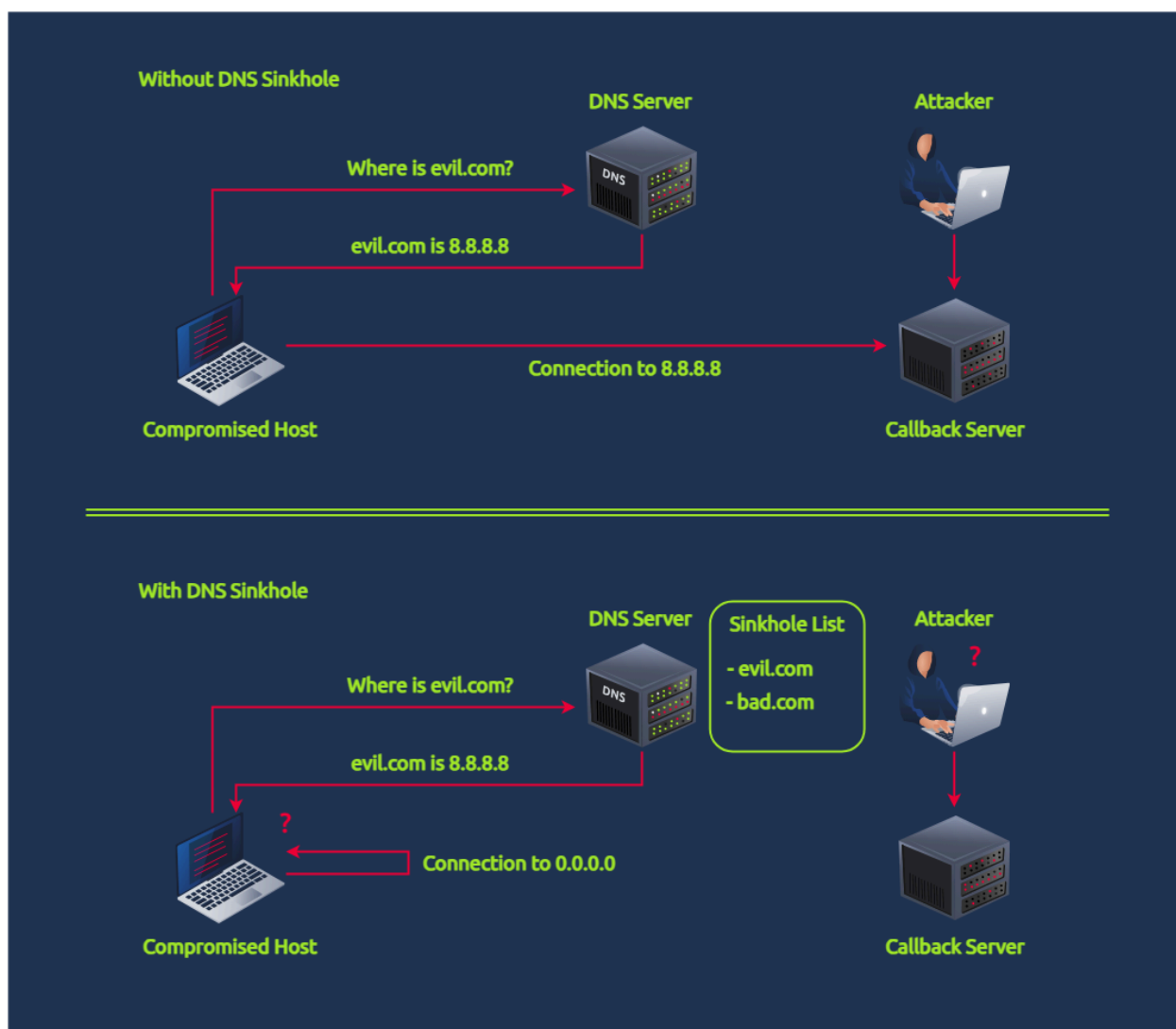
Similar to IP blocking, we can configure Email Gateways to prevent known malicious domains from forwarding incoming email messages based on the sender's domain. Email Gateways also depend on a ruleset, which should contain the block list of domains known to send spam or phishing emails. Once the block list is populated, the Email Gateway prevents threat actors from reaching the inbox of the target users in the organization.



Preventing spam emails from reaching employees' inboxes reduces the potential attack surfaces of a threat actor in compromising the organisation. Most of the time, an organisation takeover starts with the execution of a malicious attachment or submitting credentials to a phishing website. An additional prevention layer slightly reduces the burden on the users' phishing awareness capabilities.

### **Domain Blocking through DNS Sinkhole**

DNS Sinkhole is a security measure that mitigates connections to a malicious domain. This is typically done by redirecting all DNS requests from a known malicious domain to a sinkhole, preventing the resolution to their counterpart IP addresses.



## Hunting Sinkholed Domains

Using the logs generated by the SIEM, you are tasked to hunt all domains identified as malicious by your DNS Sinkhole. The Threat Intelligence team has provided a single domain to start with your activity:

- agrosaaxe[.]info

In addition, you may also need the following KQL query templates to complete the investigation:

- dns.question.name: "replace with domain"
- dns.answers.data: "replace with sinkhole IP"

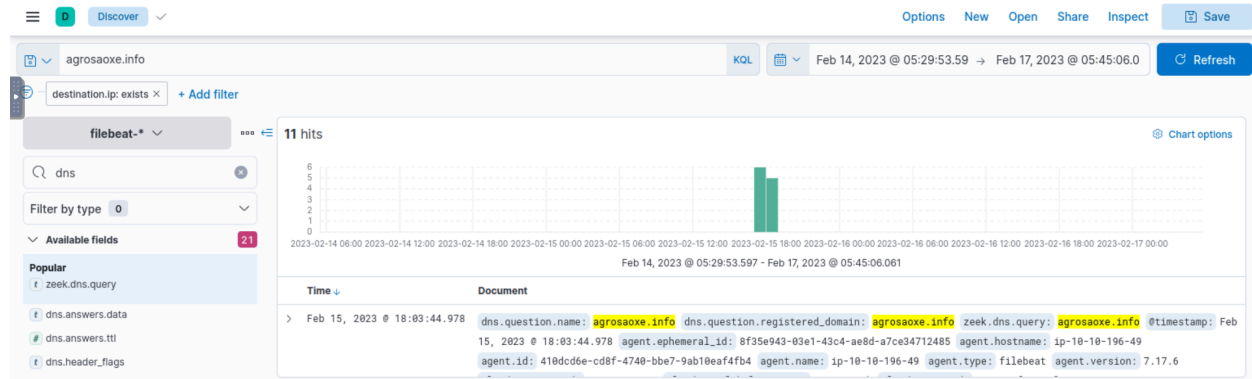
Note that you need to remove the defang in your search query.

Lastly, the events are stored in the filebeat-\* index. Ensure the search query is between 02/14/2023 and 02/17/2023. Good luck!

\*\*\*\*\*

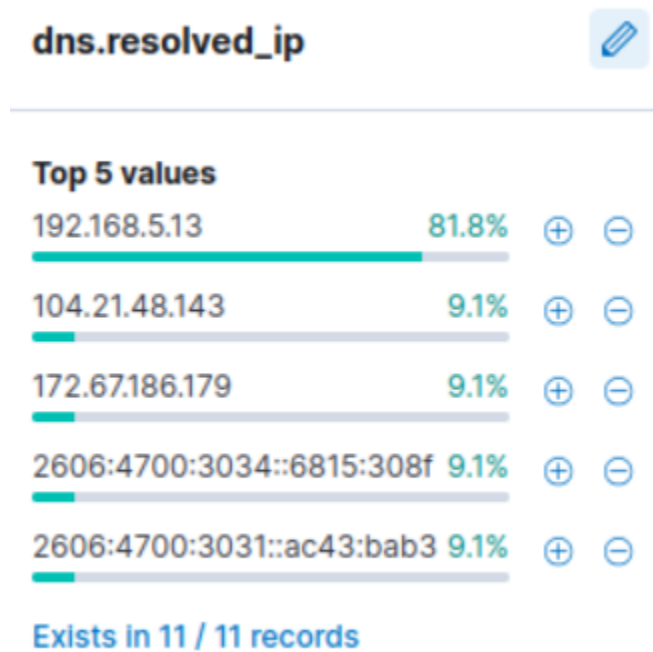
Answer the questions below:

How many DNS queries to agrosaoxe[.]info have been created?



Answer: 11

Before deploying the sinkhole configuration, what IPv4 addresses are resolved by agrosaoxe[.]info? (format: IPs in ascending order)



Answer: 104.21.48.143,172.67.186.179

What is the IP address used for DNS Sinkhole?

dns.resolved\_ip



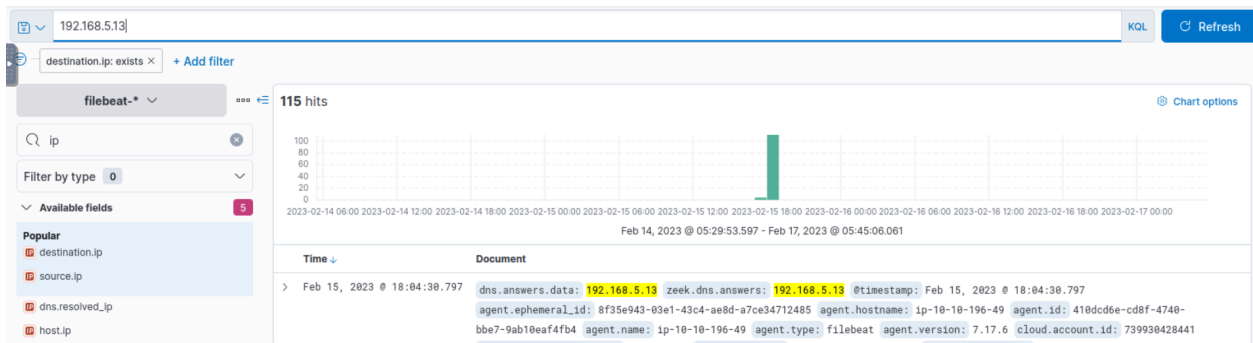
### Top 5 values

192.168.5.13	81.8%	⊕	⊖
104.21.48.143	9.1%	⊕	⊖
172.67.186.179	9.1%	⊕	⊖
2606:4700:3034::6815:308f	9.1%	⊕	⊖
2606:4700:3031::ac43:bab3	9.1%	⊕	⊖

Exists in 11 / 11 records

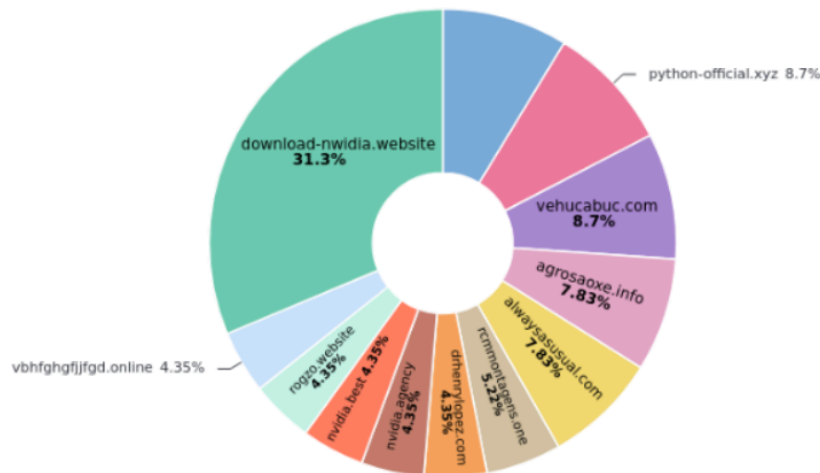
Answer: 192.168.5.13

How many hits were caused by connections to sinkholed domains?



Answer: 115

How many unique domains have been sinkholed?



Answer: 12

## Intelligence-Driven Detection

You have successfully deployed preventive mechanisms to mitigate known IOCs in your infrastructure. To maximise the capabilities of your detection and response, you are now tasked to improve the detection capabilities of your tooling.

We have started utilising Threat Intelligence from the previous task to prevent potential compromises from malicious actors. Now, we will leverage Threat Intelligence IOCs to know if something suspicious is happening in our infrastructure effectively.

### Optimizing Detection Capabilities

Implementing detection based on IOCs may be pretty straightforward, as one may think we can deploy a blocklist rule for known malicious IOCs. An example set of detection use cases is listed below for each common Threat Intel IOC.

IoC	Detection Use Case
<b>IP Address</b>	<p>Connections via Firewall logs wherein the direction of the connection dictates the potential root cause:</p> <ul style="list-style-type: none"> <li>- Egress connection to a malicious IP indicates a potential execution of malware, thus communicating with a threat actor's IP address.</li> <li>- Ingress connection from a malicious IP dictates an intrusion attempt from malicious actors, showing traces of the pre-exploitation phase.</li> </ul>
<b>URL</b>	Connections via Proxy logs wherein the HTTP method dictates the



	nature of the connection: <ul style="list-style-type: none"> <li>- HTTP GET requests indicate a potential download of malicious files or access to a phishing website.</li> <li>- Moreover, HTTP POST requests indicate a potential submission of credentials or exfiltration of stolen files.</li> </ul>
<b>Domain</b>	Malicious domains seen in DNS logs directly indicate a malicious activity in either of the following: <ul style="list-style-type: none"> <li>- The domain hosts malware or additional files for its execution chain.</li> <li>- The domain is a phishing website.</li> <li>- The domain is being used for a C2 connection.</li> </ul>

The scenarios above depict the usage of publicly available IOCs to hit any suspicious connections across different data sources such as Firewalls, DNS and Proxy servers. However, this kind of setup may require continuous fine-tuning of rules to accommodate the growth of IOCs.

We can combine some prevention techniques discussed in Task 3 to detect suspicious traffic.

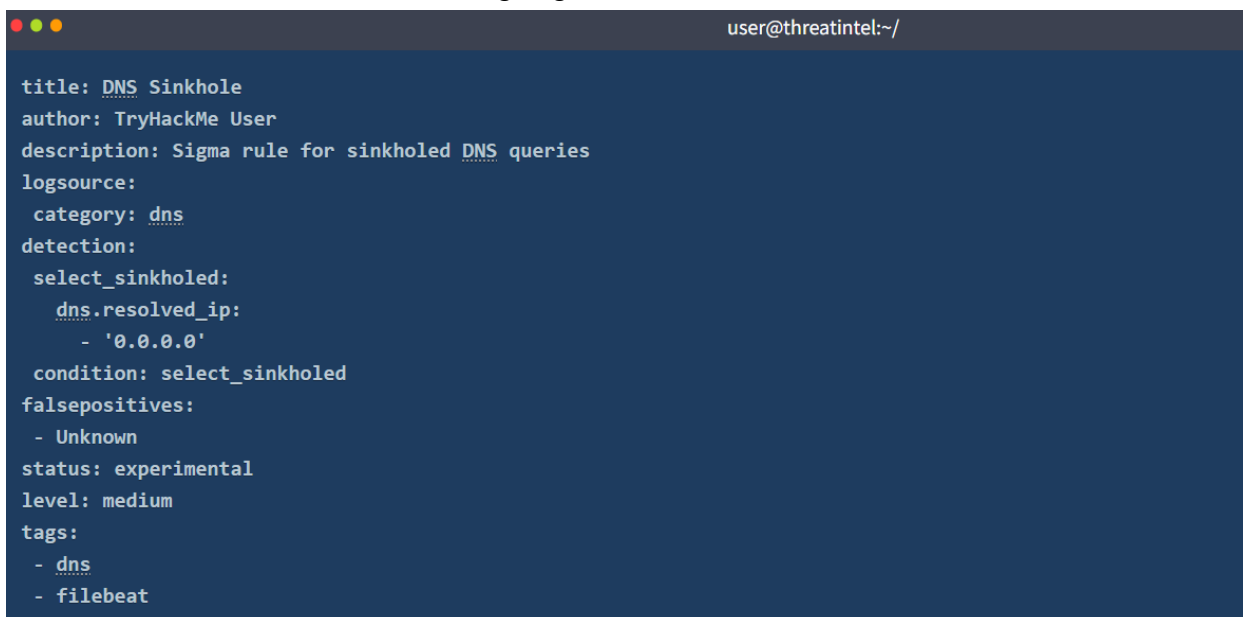
<b>Prevention Technique</b>	<b>Detection Use Case</b>
<b>DNS Sinkhole</b>	Domains resolving a loopback (127.0.0.1 or 0.0.0.0) may indicate a connection to a malicious domain based on DNS' sinkhole blocklist configuration.
<b>Firewall IP Blocking</b>	Blocked connections to and from a specific IP address may indicate malicious activity and is worthy of investigation. Logs generated attributed to IOC blocking gives more context about the connection.
<b>Proxy Blocking</b>	Blocked web connections may indicate a malicious attempt to access malware or a phishing site. The Proxy service could provide more information if it has tagging capabilities to reflect malicious connections via tags.
<b>Mail Gateway Blocking</b>	Emails blocked based on the email sender's domain may indicate a spam attempt from a malicious sender.

By doing so, the fine-tuning detection rules only rely on blocklist updates from prevention tactics. Hence, this way introduces an optimal way to prevent and detect malicious activity based on Threat Intelligence IOCs.

## Sigma Rules Revisited

As discussed throughout the Detection Engineering Module, Sigma is an open-source generic signature language to describe log events in a structured format. This allows for quick sharing of detection methods by security analysts.

In this task, we will use the following Sigma rule to hunt for sinkholed domains.

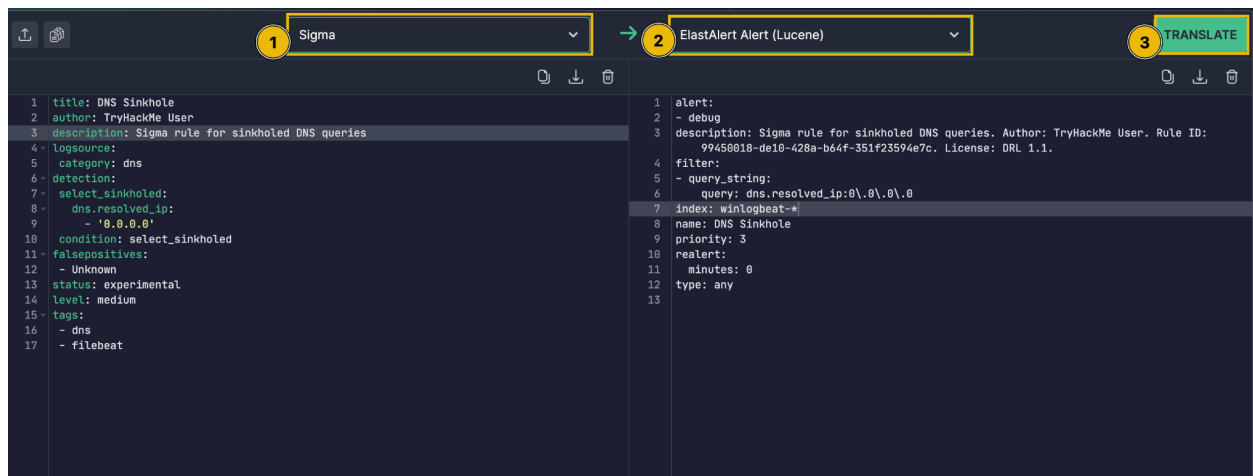
A terminal window with a dark blue background and white text. The window title bar shows three colored dots (red, yellow, green) on the left and the text 'user@threatintel:~/ ' on the right. The terminal content is a Sigma rule configuration in YAML format.

```
title: DNS Sinkhole
author: TryHackMe User
description: Sigma rule for sinkholed DNS queries
logsource:
  category: dns
detection:
  select_sinkholed:
    dns.resolved_ip:
      - '0.0.0.0'
  condition: select_sinkholed
falsepositives:
  - Unknown
status: experimental
level: medium
tags:
  - dns
  - filebeat
```

The Sigma rule above hunts for DNS queries resolving 0.0.0.0. As discussed previously, such cases may indicate a connection to a known suspicious domain based on DNS Sinkhole configuration.

## Playing with ElastAlert and Uncoder.io

To emulate sample detection, we will use Uncoder.io to translate the previously mentioned Sigma rule into ElastAlert. Ensure that the conversion is set from Sigma to ElastAlert before clicking Translate.



Before we use the generated rule, let's have a quick run-through about ElastAlert.

ElastAlert is an open-source framework for alerting on anomalies, spikes, or other patterns of interest found in data stored in Elasticsearch. It integrates with Elasticsearch, Kibana, and other tools in the Elasticsearch ecosystem and can be configured to send alerts to various external services such as Email, Slack, PagerDuty, and more.

The resulting ElastAlert rule from Uncoder.io contains the following information:

Field	Definition
<b>alert</b>	The Alerter type to use. The value debug will log the alert information at the info level.
<b>filter</b>	A list of Elasticsearch query filters. The current query searches for all domains resolving to 0.0.0.0.
<b>index</b>	The name of the index that will be searched. In our current context, the rule will scan the contents of the filebeat-* index.
<b>realert</b>	This option allows you to ignore repeating alerts for some time. The value minutes: 0 will generate all alerts despite its redundancy.
<b>type</b>	The rule type to use. The value will generate an alert for every successful query return.

Now that we have introduced ElastAlert, access the machine via SSH using the provided credentials in Task 2 (user:tryhackme) and navigate to the ~/elastalert directory. You may see that the directory contains a config file and a subdirectory.

```
user@threatintel:~/elastalert$ ls
config.yaml  rules
user@threatintel:~/elastalert$ ls rules/
sinkhole.yaml
```

The config.yaml file contains all the configurations needed to connect and query to our Elasticsearch instance, while the ~/elastalert/rules directory contains a placeholder rule. You may populate this rule with the translated Sigma to ElastAlert rule from Uncoder.io.

Note: After copying the Elastalert rule generated by Uncoder.io, execute the following to clean the syntax and point ElastAlert to the right index.

- In the description field, remove the string starting from Author: until the end of the line. This is being removed to make the syntax of the ElastAlert rule valid.
- In the index field, replace winlogbeat-\* with filebeat-\*

The ElastAlert rule should be similar to the one below after executing the steps mentioned above.

```
user@threatintel:~/elastalert/rules$ cat sinkhole.yaml
alert:
- debug
description: Sigma rule for sinkholed DNS queries.
filter:
- query_string:
    query: dns.resolved_ip:"0.0.0.0"
index: filebeat-*
name: dns_sinkhole
priority: 3
realert:
  minutes: 0
type: any
```

After configuring the sinkhole.yaml rule, navigate back to the elastalert directory and start executing ElastAlert.

```
user@threatintel:~/elastalert/rules$ cd ~/elastalert
user@threatintel:~/elastalert$ elastalert --start 2023-02-16T00:00:00 --verbose 2>&1 | tee output.txt
```

The elastalert command above can be broken down as:

- ElastAlert executes the rule we configured starting from 02/16/2023 until the present.
- It also provides verbose output.
- Lastly, the snippet uses 2>&1 | tee output.txt to write the results into output.txt. Note that file descriptors were used since the output is being written at the INFO level.

Once the command is executed, you may need to wait a few seconds to finish the initial run. The following string indicates that the execution is finished: X query hits (X already seen), X matches, X alerts sent

After seeing the string above, you can now stop the execution of the elastalert command with CTRL + C.

To complete the task, review the results written in ~/elastalert/output.txt to answer the questions below.

\*\*\*\*\*

**Answer the questions below:**

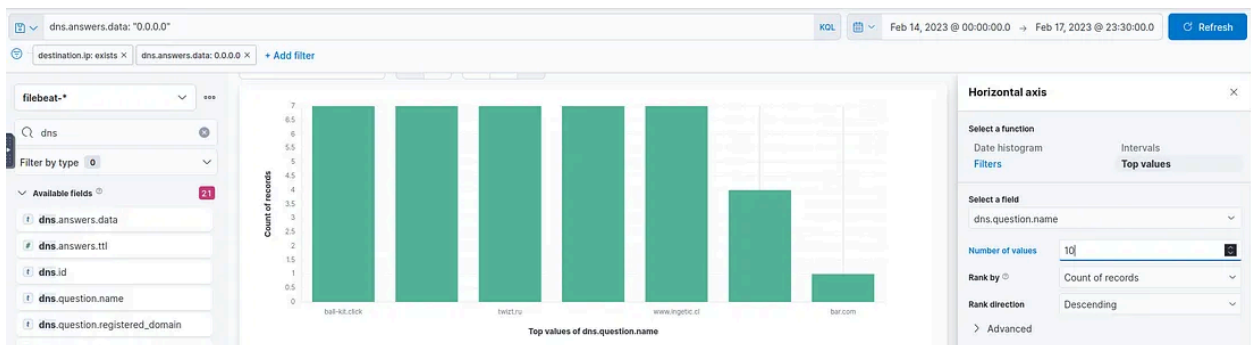
**What is the value of the alert field in the converted ElastAlert rule?**

Answer: **debug**

**How many alerts were generated by the rule?**

Answer: **40**

**How many unique domains were sinkholed via 0.0.0.0?**



Answer: **7**

**What is the sinkholed domain that has .ru TLD?**

To get the answer I ran `cat output.txt | grep '.ru'` and got

```
:false,\"RA\":false,\"Z\":0,\"answers\":[\"0.0.0.0\"],\"TTLs\":[3600.0],\"rejected\":false},  
  \"query\": \"twizt.ru\",  
INFO:elastalert:Backward configuration change check run at 2023-02-26 07:40 UTC
```

Answer: **twizt.ru**

## Conclusion

Congratulations! You have completed learning and improving the Security Operation's capabilities using Threat Intelligence.

In the previous tasks, we have learned the following:

- The difference between Threat Intelligence Producers and Consumers.
- Knowing your current needs and capabilities is vital to maximising the usage of Threat Intelligence.
- Implementation of Intelligence-driven prevention and detection.
- Revisiting Sigma, Uncoder and Elastalert for the example tooling of intelligence-driven detection.

To conclude, this room focused on leveraging Threat Intelligence to improve the overall capabilities of a Security Operations Center, ranging from detection capabilities to preventive mechanisms that mitigate entirely known threats.

In this world of ever-evolving threats, we need to utilise every ounce of knowledge to protect our organisation, including all the intelligence provided by organisations that serve as Producers and bits of meaningful feedback from Consumers. Through collaboration, we can protect and improve the security posture of every organisation.