Graphical user interface, text

Description automatically generated

**IST 894 Capstone Experience - Dr. Bartolacci**

Semester Long Group Project – Analyzing Data with Elasticsearch and Kibana

Submitted by: Matt Neale and Scott Finlon

mtn5179@psu.edu and smf261@psu.edu

**Table of Contents**

[**Section 1: Lab Introduction 3**](#_Toc89365110)

[**Section 1.1: Lab Purpose 4**](#_Toc89365111)

[**Section 1.2: Estimated Time for Lab Completion 6**](#_Toc89365112)

[**Section 1.3: Network Diagram 7**](#_Toc89365113)

[**Section 2: Step by Step Lab Instructions 7**](#_Toc89365114)

[**Section 2.1: Installing Elasticsearch and Kibana 8**](#_Toc89365115)

[**Section 2.2: Downloading and Importing Logs 14**](#_Toc89365116)

[**Section 2.3: Analyzing Data with Elasticsearch 17**](#_Toc89365117)

[**Section 3: Verification of Lab Completion 21**](#_Toc89365118)

[**Section 3.1: Verification of Elasticsearch and Kibana Installation 22**](#_Toc89365119)

[**Section 3.2: Verification of Data Import 28**](#_Toc89365120)

[**Section 3.3: Verification of Data Visualization and Expected Answers 30**](#_Toc89365121)

[**Section 4: References 37**](#_Toc89365122)

# **Section 1: Lab Introduction**

Data is at the forefront of every decision made within the field of cybersecurity. Mainly, data is transmitted and stored in log files that give security professionals a better idea of what activity has been performed on the system and which users attempted to access them. Nevertheless, log data is useless without the proper tools and organization. This is because system administrators often err on the side of caution and configure the system to log a wide variety of different authorizations, connections, and activities on the system. This lab introduces the concept of log analysis through the application of the Elasticsearch and Kibana tools that collectively makeup the Elastic Stack. During this lab, students will begin to understand the importance of log analysis and visualization by examining authentication logs collected from a webserver. Leveraging the versatility of Elastic, students will see how easily system logs can be transformed into powerful visuals that can significantly improve security visibility across the network (Defining Log Management | The Importance of Logging and Best Practices |Humio, 2021).

At the core of the Elastic Stack is a tool called Elasticsearch. In short, Elasticsearch is a free and open-source software package that enables security engineers to store, search, and examine collected data (Abueg *| Elasticsearch: What It Is, How It Works, And What It’s Used For | KnowI,* 2020). Given Elasticsearch’s storage capability, this tool is often utilized for both real-time security log monitoring and retention to meet the stringent requirements of different standards and regulatory compliance frameworks. Diving deeper into Elasticsearch, one should understand that it works like a generic search engine in which it organizes data into documents (basic unit of Elasticsearch stored as JSON files) and indices (collection of similar documents). This simple organizational structure of Elasticsearch enables it to query data from logs very quickly to speed up the incident response times of security teams. It should also be noted that the organization of data into documents and indices enables Elasticsearch to become highly scalable in large environments (Melo | *A Practical Introduction to Elasticsearch with Kibana | Medium,* 2018). This scalability, combined with the fact that the tool is free to use, makes Elasticsearch a valuable tool for organizations that want to begin building a complex security collection system.

Although the collection and retention capabilities of Elasticsearch are useful for a security professional, the Elastic Stack is made increasingly versatile by Kibana. Similar to Elasticsearch, Kibana is free and open source, once again making it a great implementation for organizations that are looking for cost-effective ways to analyze their data. Ultimately, Kibana is responsible for providing users with a way to view, search, and visualize log data to create easy to continuously updated dashboards and easy-to-read graphs (Elastic | What is Kibana?, n.d.). For example, in this lab, students will be using Kibana to create a bar graph that displays the IP address that was logged the most. In turn, this graph quickly showed users which IP address was attempting to infiltrate the web server through a variety of authentication attempts. Another common use of Kibana as a visualization tool is to create reports that will be presented to senior management. With the necessary data and logs, security engineers can create graphs that sufficiently articulate network activity and leverage this information to gain funding to fuel future projects. Overall, Kibana is not only a great tool for security visualization and monitoring but it is also provides a great opportunity for organizations to create reports and baselines (Sureka | What is Kibana Used for? 10 Important Features to Know | ClarionTech. 2020).

## **Section 1.1: Lab Purpose**

Upon the successful completion of this Elastic lab assignment, the student will learn a variety of important cybersecurity skills that can be directly translated to a real-world technical environment. First and foremost, this lab will provide users with knowledge of different Linux command line executions that teach them how to download and install Linux-based software. For example, the “wget” Linux utility is referenced multiple times that enables users to download software repository GPG keys. This utility, in combination with the “apt-transport-https” command allows users to download software from repositories securely across a network. Aside from these commands, other notable Linux CLI knowledge is presented through the “sudo service [service] start/stop” command syntax and the “curl” command that not only allows users to control the state of a service but to also verify if a web-based service is functioning correctly. Ultimately, the Linux command line tools utilized in this lab provide repeatable commands that can be executed to install a wide variety of different software packages.

In addition to the Linux command line knowledge gained through the tasks of this lab, this assignment also introduces the concept of log collection and analysis. As suggested by the introduction above, the combination of Elasticsearch and Kibana provides security professionals the ability to collect and visualize log data. After completing the initial setup of the two services, the user is asked to download a sample webserver log that displays numerous authentication attempts. While an examination of the raw log content provides a fundamental understanding of the activity that occurred, the real value of Elastic is presented when the user is tasked with creating a visual representation of the data. In order to quickly answer the challenge questions presented in this lab, the user must leverage Kibana’s ability to generate graphs based on the detected log fields. By doing so, one should be able to easily and visibly identify the exact IP addresses that was attempting to authenticate to the server multiple times. The log analysis tasks of this lab are extremely valuable for any information security professional as it allows them to leverage system data to accelerate incident response procedures.

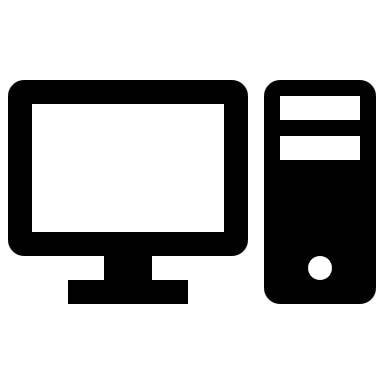
An often-overlooked purpose of any lab that introduces a user to a new piece of software is that the user is able to further explore the service beyond the specifications of the lab instructions. In other words, since this lab assignment provides the student with all of the necessary steps to install and run Elastic, they can use their own logs to create their own dashboards, visuals, and reports. For example, a user can try out different Elastic visualization lenses that best fit a particular dataset and can create different graphs that display different statistics. Additionally, because the Elastic service is installed within an isolated virtual environment, students can test new implementations without worrying about making damaging alterations to a production environment. This idea is encouraged throughout the lab, such as when the lab asks the user how they can change the listening port of the Elasticsearch service. If users are able to successfully change this listening port, they will learn a valuable lesson about configuration files and service maintenance.

## **Section 1.2: Estimated Time for Lab Completion**

One of the best characteristics of the Elastic Stack is that it can be implemented into almost any network environment in a relatively short amount of time. With that said, in order to take full advantage of the Elastic Stack, users must not only understand how to import data into the system but must also understand how to use the web-based Elastic tool to understand, inspect, and visualize this data. As a result, the estimated time of completion for this lab, including the necessary background research on the Elastic Stack, is around **1 hour**. It should be noted that some students may take longer than the estimated time if they decide to further explore the Elastic visualization lenses to analyze their own data or logs. Given the popularity and versatility of Elastic, it is a great idea to further explore the capabilities of Elastic to ensure that the tasks outlined in this lab can be repeated within a real technical environment.

## **Section 1.3: Network Diagram**

For the purposes of this lab assignment, all log collection and analysis can be performed on a single Kali Linux system within the “Cyber Basics” virtual environment. This is because the Elasticsearch and Kibana services are capable of running locally on a system without utilizing a lot of system resources. In a real-world environment, users will often see a dedicated Elastic server in which logs are securely transmitted from different systems.



**Computer Name:** Kali

**IP Address:** 10.1.143.131 (Loopback: 127.0.0.1)

**Elasticsearch service:** TCP/9200

**Kibana service:** TCP/5601 (URL: http://www.localhost:5601)

# **Section 2: Step by Step Lab Instructions**

To complete this lab exercise demonstrating the configuration and utilization of the Elasticsearch service, students will use the “Cyber Basics” virtual environment within the U.S. Cyber Range. It should be noted that Elasticsearch and Kibana can also be installed on other Operating Systems, such as Windows, but the setup will require a different process with different commands. The tasks below provide a step-by-step walkthrough that provides all of the necessary information required to get started with Elasticsearch.

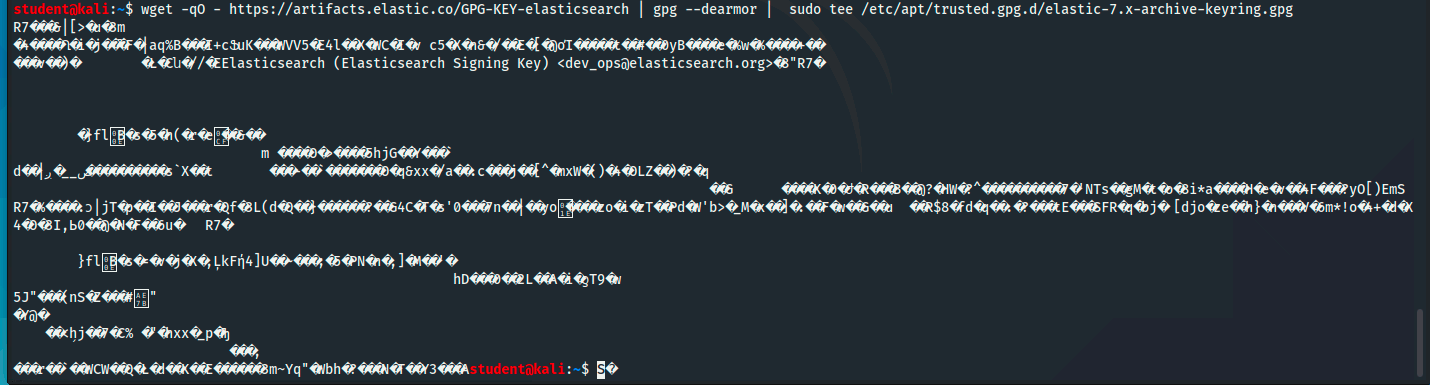
## **Section 2.1: Installing Elasticsearch and Kibana**

The first task of this lab focuses on the installation and initial configuration of both the Elasticsearch analytics tool and the Kibana visualization resource. To get started, power on the Kali Linux virtual machine and login with a username of **student** and a password of **student** (if needed).

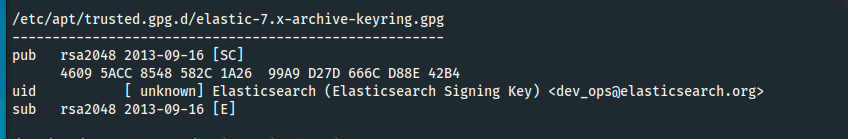
As with a majority of Linux applications and services, the installation of the Elasticsearch and Kibana can be done completely from the Linux terminal. To open the Linux terminal, open the ‘**Terminal Emulator**’ from the Kali Linux dock (along the bottom of the screen) or navigate to the “Applications Menu” and select “Terminal Emulator.” Once the Terminal is open, change your permissions to the root user using the `**sudo su –`** command. If prompted, provide the sudo password (**student**) to complete the privilege escalation process.

Graphical user interface

Description automatically generated

Because we will be downloading Elasticsearch and Kibana from the Elastic repository, we must first import the repository’s GPG key to our machine. This process utilizes the **wget** Linux utility to download the GPG key from an Elastic URL that can then be used for file integrity purposes. Once the key is retrieved, you can use GPG to unpack the key from ASCII armor, and then pipe that to tee which will write the output to a specified file. The full command is this: `**wget -qO - https://artifacts.elastic.co/GPG-KEY-elasticsearch | gpg --dearmor | sudo tee /etc/apt/trusted.gpg.d/elastic-7.x-archive-keyring.gpg**`.

It will print the binary output of the de-armored key to the screen but creates the key file “/etc/apt/trusted.gpg.d/elastic-7.x-archive-keyring.gpg” which can be verified by running by running `**sudo apt-key list**` and looking for the entry for elastic-7.x.



After the repository key is imported into your system, you must then install the apt-transport-https to the system. This package allows APT transport to download packages via HTTPS. Using a command of `**sudo apt install apt-transport-https`** installs the package to the sytstem (Note: if you receive an error during this installation, try running the **apt-get update** command).Text

Description automatically generated

Now that we have our environment ready, we can begin downloading Elasticsearch itself. We need to add Elasticsearch’s APT repository to our trusted sources list by running `**echo "deb https://artifacts.elastic.co/packages/7.x/apt stable main” | sudo tee /etc/apt/sources.list.d/elastic-7.x.list`**. This command writes the URL of their repository to a file in our /etc/apt/sources.list.d directory. The completion of this step can be verified by navigating to the new directory, ‘**cd /etc/apt/sources.list.d’** and then viewing the “elastic-7.x.list” file ‘**cat elastic-7.x.list**’. If the file is not present, try checking the syntax of the **tee** command to ensure that the file path and file name is correct.

Text

Description automatically generated

With the completion of the above step, you have successfully completed all of the prerequisites for an Elasticsearch installation. In order to install Elasticsearch and Kibana, we need to update our local apt sources by running ‘**sudo apt update.’** This command retrieves available installation packages from the /etc/apt/sources folder including the Elasticsearch one that we just created. Once the update is complete, install Elasticsearch using ‘**sudo apt-get install elasticsearch’**. It is worth noting that these two commands can be combined using the && characters as follows: `**sudo apt update && sudo apt-get install elasticsearch`**.

Text

Description automatically generated

Kibana is an open-source frontend application that sits on top of the Elastic Stack. To install Kibana, run ‘**sudo apt-get install Kibana’**.

Text

Description automatically generated

To enable these services by default, you will need to run `**sudo systemctl enable kibana**` and `**sudo systemctl enable elasticsearch**`.

A screenshot of a computer

Description automatically generated with medium confidence

While they are now enabled to start on boot, we still have to start them up manually for the first time because services are not started upon installation. To start the Elasticsearch and Kibana services, run `**sudo service elasticsearch start`** as well as `**sudo service kibana start`.**



The final step of the Elasticsearch/Kibana installation process is to ensure that the services are enabled and functioning. This can be done by sending an HTTP request to the Elasticsearch web service. All new instances of Elasticsearch will attempt to bind to port 9200. Therefore, you can issue a command of ‘**curl -X GET “localhost:9200/’** to verify that your Elasticsearch is running correctly on your local machine. You should receive an output similar to the screenshot below.

Text

Description automatically generated

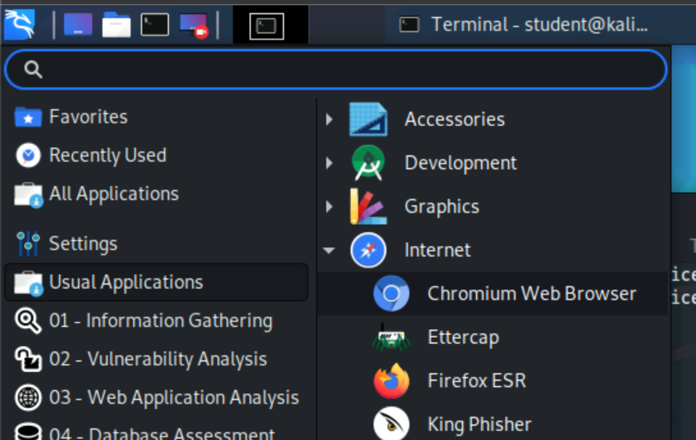
Although a similar process can be used to ensure that Kibana is functioning correctly, you can also use a web browser to connect to the Elastic Kibana GUI over port 5601. After providing a URL of “**127.0.0.1:5601**”, you should see the page depicted below.

Graphical user interface, application, website

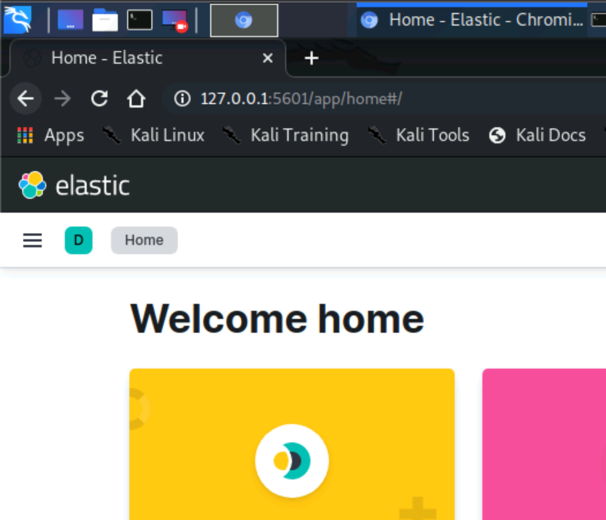
Description automatically generated

## **Section 2.2: Downloading and Importing Logs**

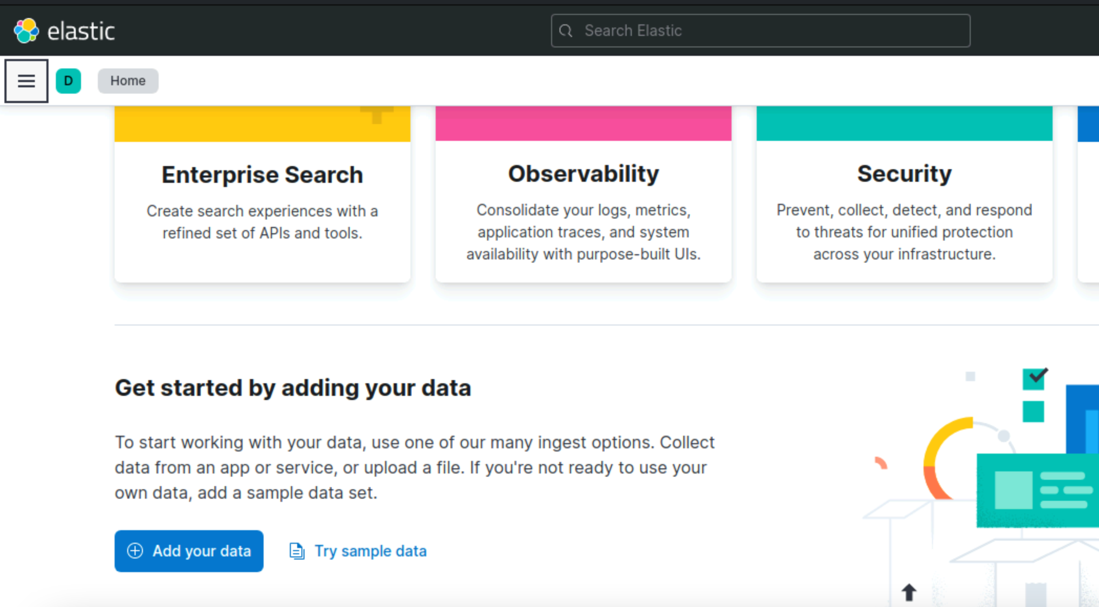
Now that Elasticsearch and Kibana are installed and the services started, we can add some data. First, we need to open a terminal window and type ‘**cd ~/Desktop**’ and then ‘w**get https://www.explodingwoodchucks.com/auth.log**’ to download a sample web server authentication log. In this environment, Firefox has some display issues so you need to click ‘**Usual Applications**’ then ‘**Internet**’ and finally ‘**Chromium Web Browser**’.



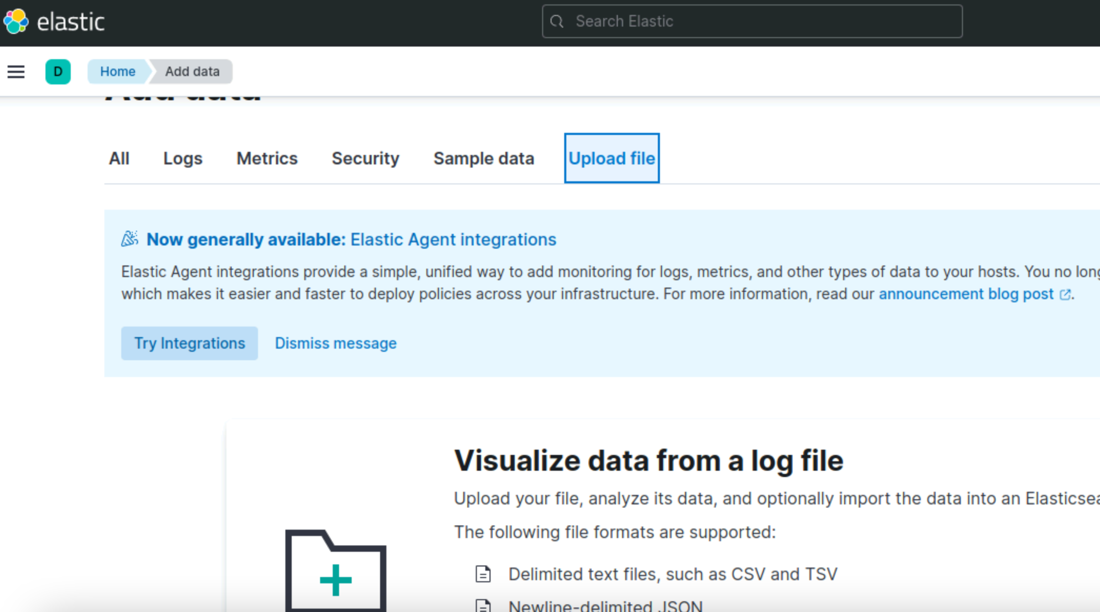
Once the Chromium browser opens, type in the URL ‘**http://127.0.0.1:5601**’ to get to the Kibana web interface.

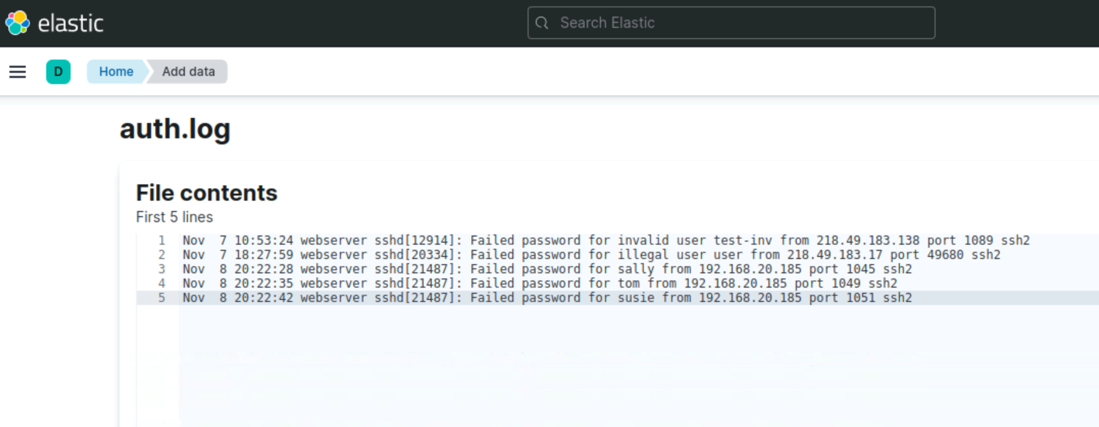


Scroll down the front page, and click the blue button for ‘**Add your data**’.

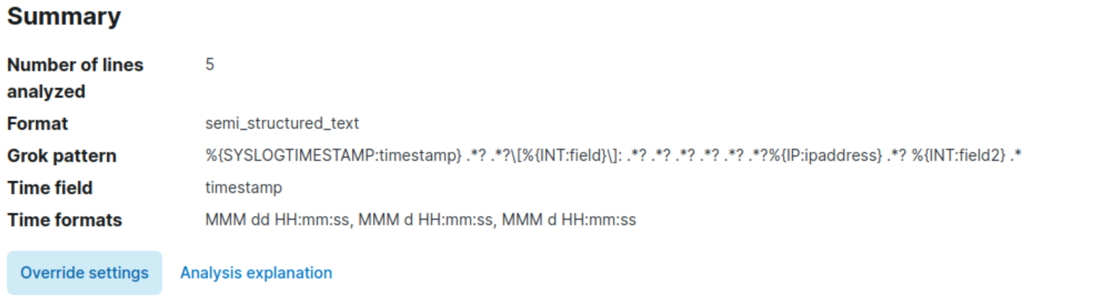


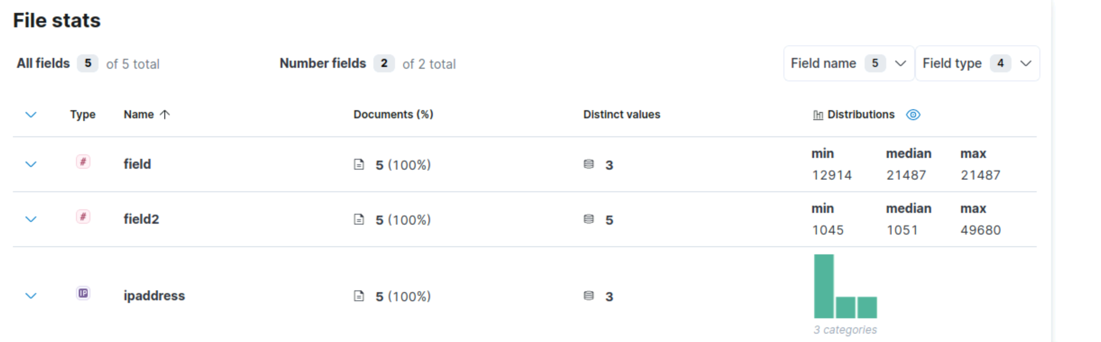
Click on ‘**Upload file,**’ scroll down to where it says ‘**Select or drag a file**’ and select the **auth.log** filethat was downloaded on your Desktop. If successful, you will see the contents of the file displayed on the screen.

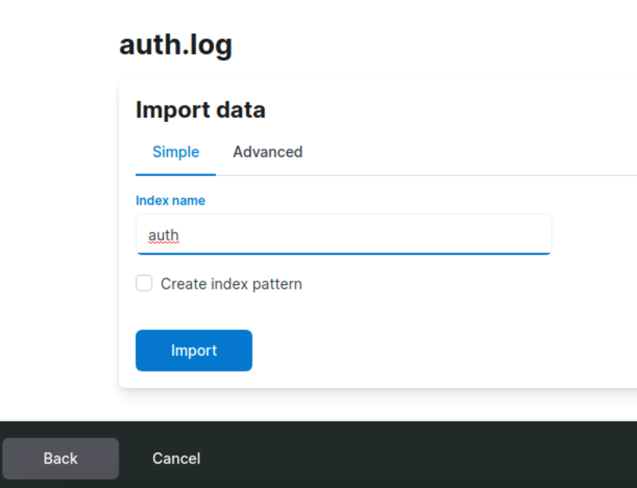


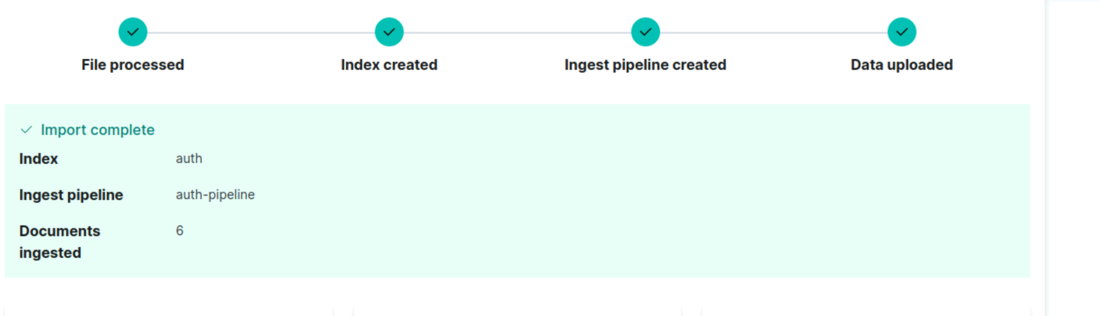


As you scroll down the page, you’ll see a summary of the file which includes: how many lines were analyzed and imported, what format it detected, the pattern it used to identify fields, which field indicates the time, and what the timestamp format is. Next, you’ll see overall stats related to the data itself and the fields it identified. Finally, it will ask you what index you would like to ingest this data to, you can type ‘auth’ and click import and you should then see a screen with four green checkmarks and a message that says ‘Import complete’





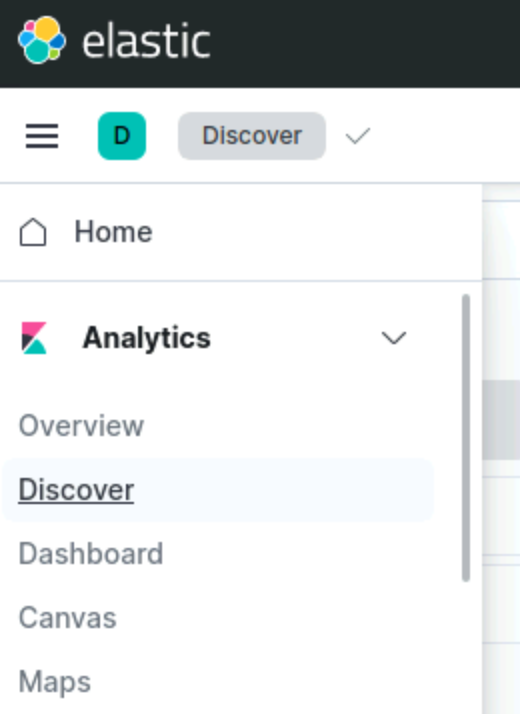




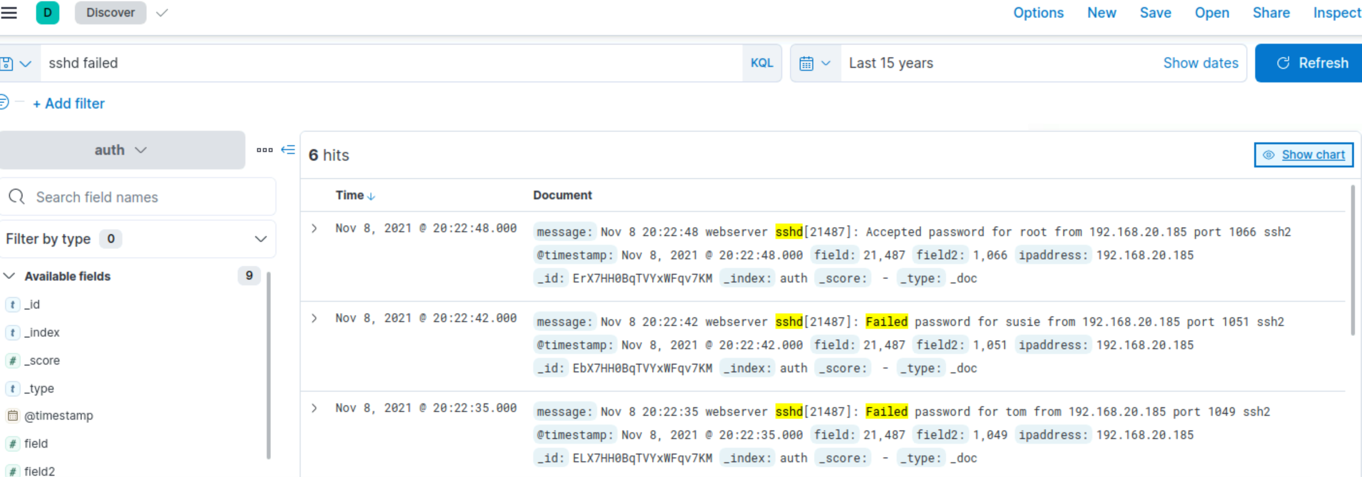
## **Section 2.3: Analyzing Data with Elasticsearch**

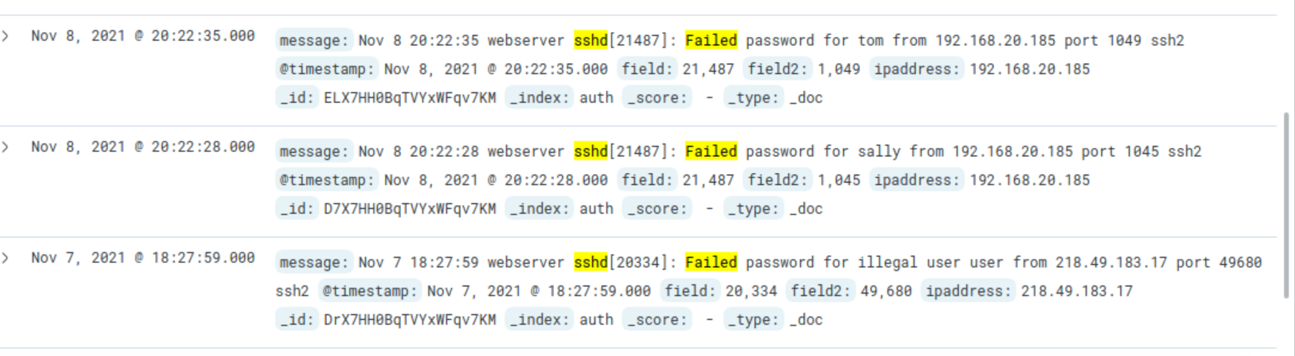
Kibana has two methods for querying data. You can use the Kibana Query Language(KQL) (Kibana Query Language | Kibana Guide [7.15] | Elastic, n.d.) or you can opt out of that and use Lucene Query Syntax (Lucene Query Syntax | Kibana Guide [7.15] | Elastic, n.d.). Lucene is a Java based searching and indexing library written by Apache, where KQL is purpose-built for Kibana and allows you to perform free-text searches and searches based on specific fields.

To start analyzing the data, click the hamburger icon and then click ‘**Discover**’

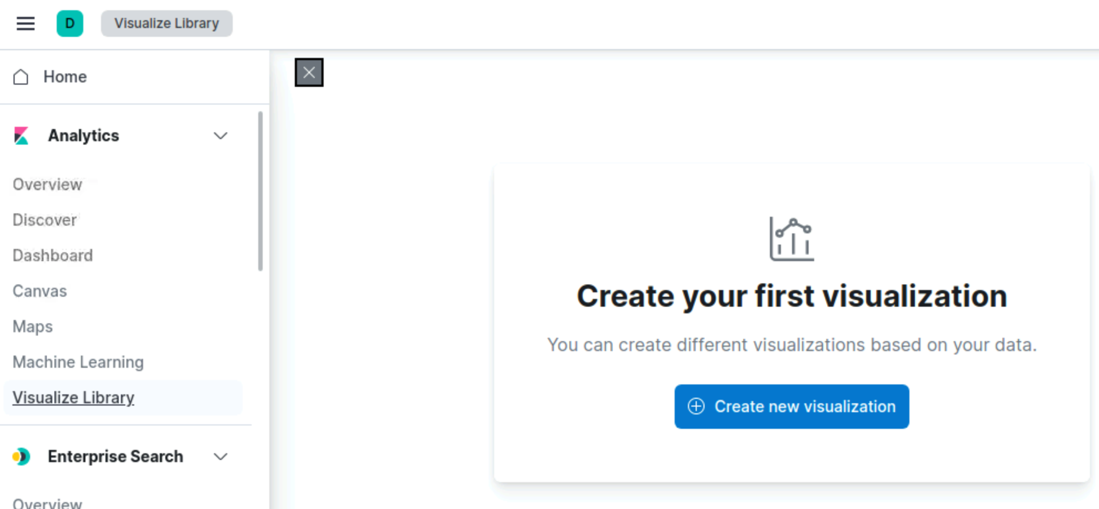


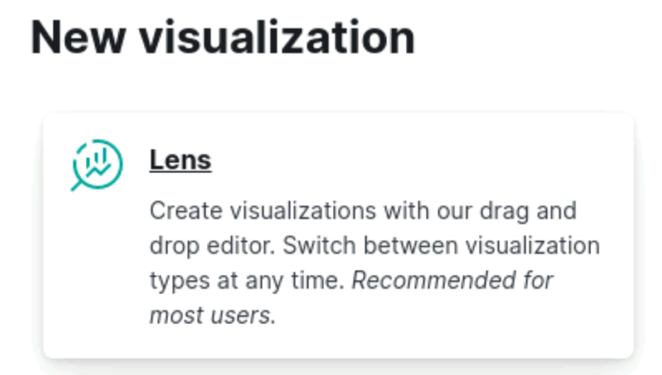
You can type multiple words into the Discovery query bar like ‘**sshd failed**’. As reference, there is an implicit logical ‘OR’ between any words. Therefore, if you want to search for terms in conjunction with another put an ‘AND’ explicitly in the bar (ex: ‘**sshd AND failed**’. Note: you might have to increase the time of the search window depending on the time frame of the log in relation to your current time.

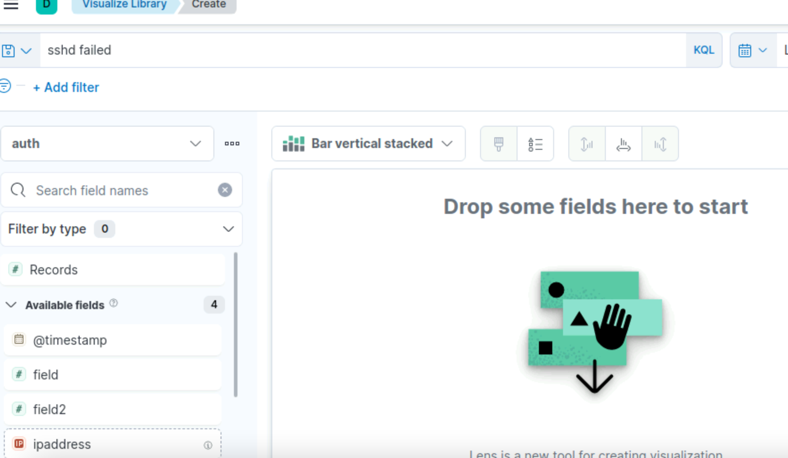


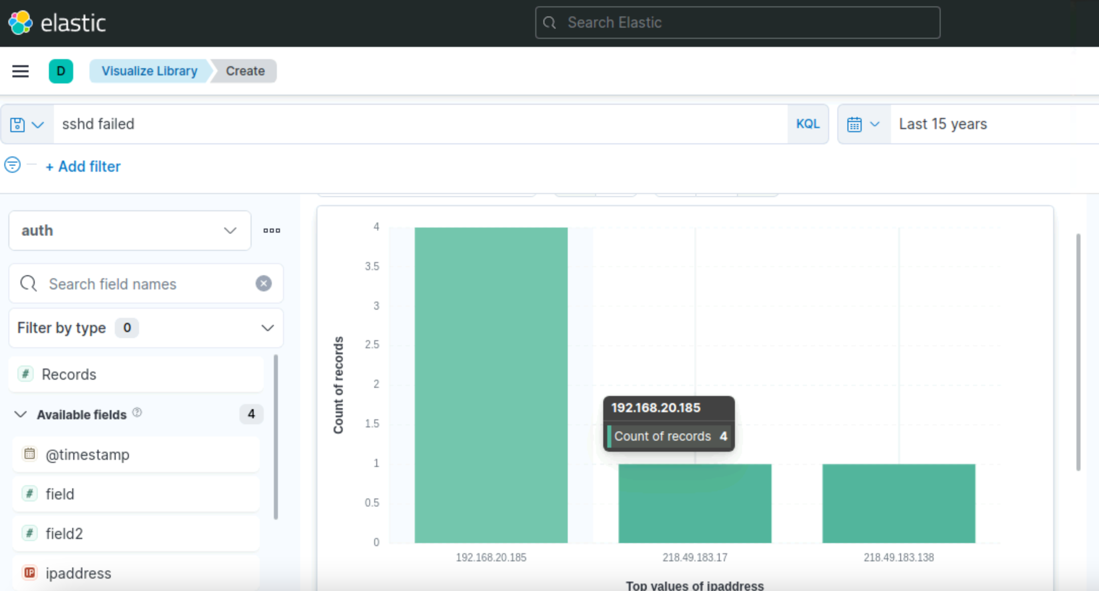


To visualize the data, click the hamburger again and select ‘**Visualize Library**’. Next, choose ‘**Create new visualization**’ and then click ‘**Lens**’ from the choice of selections. From there, you are free to drag and drop a field that you’d like to visualize (such as the ‘**ipaddress**’ field) where you will see a bar chart that shows the count of records.

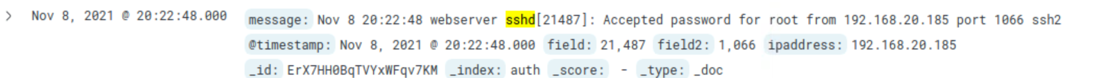








Looking at the data visualization, you can see that a specific IP address is in the log several times, and when you take a closer look at the data you see that one IP address failed several times, but then was successful.



Using your newly created data visualization of the auth.log file, answer the questions below in your lab report. It is important for security professionals to create visualizations that make it easy to pinpoint anomalies and activities that deviate from the network’s/system’s normal baseline. Before looking at the data itself, the first question below challenges you to use your prior knowledge of network ports and services to ensure that Elasticsearch can be integrated into almost any technological environment.

**Question 1: If you already have a service operating on port 9200, can you still use the Elasticsearch service? If so, please provide the steps that you would take. (Hint: Conduct some research on how to change the Elasticsearch service port).**

**Question 2: What was the IP address that showed up the most in the log?**

**Question 3: What user account did they successfully log in as?**

**Question 4: What query did you use to find the successful authentication for that IP address?**

# **Section 3: Verification of Lab Completion**

When creating a highly technical and comprehensive hands-on lab exercise, it is critical that the step-by-step instructions sufficiently promote learning to ensure that the user could repeat the tasks in a corporate environment. As a result, this section of our group project provides proof that our cookbook-style directions are both complete and easy to follow. For example, this section will provide screenshots and descriptions of this Elasticsearch/Kibana lab that mimics the steps that a student would take.

## **Section 3.1: Verification of Elasticsearch and Kibana Installation**

The first step of this lab requires the student to log into the U.S Cyber Range lab environment and power on the Kali Linux machine within the “Cyber Basics” module.

Graphical user interface, text, application

Description automatically generated

Figure : Powering on the Cyber Basics Environment

After the user connects to the Kali Linux machine, they are able to navigate to the Application Menu located in the top left corner of the screen. This application menu defaults to the “Favorites” tab where the user will be able to open the “Terminal Emulator” application. This application brings up the Linux command line terminal where the user is instructed to change their privileges to the root user by executing the ‘**sudo su –**‘ command. Figure 2 verifies that the privileges have been escalated given that the prompt reads “root@kali” after execution.

A picture containing text, screen

Description automatically generated

Figure : Escalating to the Root User in Kali Linux

Having successfully escalated the privileges to root, the user is able to begin installing the Elasticsearch and Kibana services. First, this requires the user to provide a command of **‘wget -qO - https://artifacts.elastic.co/GPG-KEY-elasticsearch | gpg --dearmor | sudo tee /etc/apt/trusted.gpg.d/elastic-7.x-archive-keyring.gpg’** that imports the repository’s GPG key into the machine’s store. Essentially, this command provides three different actions: get the GPG key (using ‘wget’), unpack the key (using ‘gpg –dearmor’), and write the output to a file (using ‘tee’). This command utilizes the Linux pipe character (|) to combine a variety of commands into a singular execution.

Text

Description automatically generated

Figure : Importing GPG Keys

As suggested by the lab instructions, the user is able to verify that the key has been successfully imported and written using the ‘**sudo apt-key list**’ as shown in Figure 4.

Text

Description automatically generated

Figure : Viewing GPG Keys in Kali Linux

Once the key is installed, the next step in the Elasticsearch/Kibana process is to add the APT transport service. This service is required whenever users want to download software packages over HTTPS (TCP/443). To install the service, a simple command of ‘**sudo apt install apt-transport-https**’ should be provided.

Text

Description automatically generated

Figure : Installing the APT Transport Service

The Elasticsearch APT repository must be added to the machine’s trusted sources list (using a command of ‘**echo "deb https://artifacts.elastic.co/packages/7.x/apt stable main” | sudo tee /etc/apt/sources.list.d/elastic-7.x.list**`) before the services are installed. Once again, this command makes use of the “tee” command to write the URL of the repository to a file. The user can verify that the file was written by navigating to the correct directory (using ‘**cd**’) and viewing the file (using ‘**nano**’).

Graphical user interface

Description automatically generated

Figure : Adding the Elasticsearch Repository to the Machine's Trusted Sources List

When all of the prerequisites are complete, the user is able to install both Elasticsearch and Kibana. With adequate knowledge of the Linux && characters one is able to install both services with one command (‘**sudo apt update && sudp apt-get install elasticsearch && sudo apt-get install kibana**’).

Text

Description automatically generated

Figure : Installing Elasticsearch and Kibana

Both Elasticsearch and Kibana have different ways for the user to verify that the services were successfully installed. Regardless, the services must first be started from the command line as they do not run when installed. After issuing command of ‘**sudo service elasticsearch start**’ and ‘**sudo service kibana start**,’ the user can execute the ‘**curl -X GET “localhost:9200/”**’ command to verify if Elasticsearch is running. This is because the Elasticsearch service binds to port 9200 (Elastic n.d.). Further, the user can ensure that Kibana is running by using a web browser to connect to a URL of ‘**localhost:5601**.’ Figures 8 and 9 show that the Kali Linux machine is capable of running both services successfully.

Graphical user interface, text

Description automatically generated

Figure : Verification of Elasticsearch Installation

Graphical user interface, application, website

Description automatically generated

Figure : Verification of Kibana Installation

## **Section 3.2: Verification of Data Import**

Aside from the installation of Elasticsearch and Kibana, the second major task of this lab was to download and import logs into the services for visualization. Obviously, the most important step here is to import a log that contains important information about the network or system. The authors of this lab created a log called “auth.log” and hosted it on a website at the following URL: “https://www.explodingwoodchucks.com/auth.log.” In order to download this log to the system, the user must first navigate to their Desktop directory (using a command of ‘**cd ~/Desktop**’) and then use the wget Linux command by providing the URL described above. As reference, the user can also issue the “ls” command within the Desktop directory to ensure that the file is present.

Text

Description automatically generated

Figure : Downloading Logs for Visualization

After the “auth.log” file is successfully downloaded, the remainder of the import process is quite simple. As stated earlier, the user can use a Chromium-based web browser to navigate to the Kibana instance (‘**localhost:5601**’ or ‘**http://localhost:5601**’). On the Elastic homepage, an ‘**Add data**’ button is present that allows the user to select a data source. By selecting the ‘**Upload file**’ option, the user is able to select the “auth.log” file from their machine and import it into Elastic as seen in Figure 11 below.

A screenshot of a computer

Description automatically generated

Figure : Importing Data into Elastic

One of the most useful aspects of Elastic is that it provides the contents and statistics of the imported log file. Figure 12 shows that the “auth.log” file is comprised of different logged events that mostly alert of failed SSH password attempts. Additionally, there are different fields in the log file such as “timestamp” and “ipaddress.” After thoroughly examining the log file, the user must select the ‘**Import**’ button located in the bottom left corner of the screen. It is important to note that Elastic will also ask for an index name. This name should provide some indication of what the log contains.

A screenshot of a computer

Description automatically generated

Figure : Importing the auth.log File

Graphical user interface, text, application

Description automatically generated

Figure : Verification of Successful auth.log Import

## **Section 3.3: Verification of Data Visualization and Expected Answers**

Although importing log files into Elastic is relatively straightforward, it is very beneficial to security engineers as the tools are capable of structuring and visualizing log data. Once Elastic is done importing the data, the user can navigate to the ‘**Discover**’ tab located under “Analytics.” This brings up the structured and organized contents of the “auth.log” file. For this exercise, the user will most likely need to modify the date filter before viewing the contents of the log. As suggested by the lab instructions, a search for ‘**sshd failed**’ highlights entries in which a user attempted to access the SSH service of the web server with an incorrect password. Figure 14 shows failed passwords for the “susie,” “tom,” and “sally,” users from an IP address of 192.168.20.185.

A screenshot of a computer

Description automatically generated

Figure : Searching the auth.log File with Elastic

While searching for key words in a log file may be useful at times, visualizing the data often allows users to quickly pinpoint areas of interest. To visualize the “auth.log” file, the user should navigate to the ‘**Visualize Library**’ tab within the Analytics page and select the ‘**Create new visualization**’ option. In turn, the user can select the ‘**Lens**’ option from the list and drag valuable auth.log fields (such as ‘**ipaddress**’) to the visualization area. The default bar graph that appears shows that the IP address of 192.168.20.185 is by far the most prevalent IP address in the log.

Graphical user interface, application

Description automatically generated

Figure : Visualizing the auth.log File by IP Address

Elastic makes it very easy for users to change their visuals. For example, a Donut chart with percentages can be used as an alternative to the bar chart in Figure 15 to adequately display the number of occurrences for each address.

Graphical user interface, chart

Description automatically generated

Figure : Changing Visualization Formats in Elastic

As a conclusion to this lab, the instructions present four different questions about the Elastic framework as well as the contents of the “auth.log” file. Once again, it is important to verify that these answers can be answered appropriately to ensure that students develop an understanding of the tools. It is recommended that instructors look for answers similar to those presented below.

**Question 1: If you already have a service operating on port 9200, can you still use the Elasticsearch service? If so, please provide the steps that you would take. (Hint: Conduct some research on how to change the Elasticsearch service port).**

Often times, third-party software developers select a random high network port for their web-based service to operate in order to ensure that the service does not attempt to bind to a port that is already open and listening. With that said, if a user already has a service operating on port 9200, they can alter the configuration of Elasticsearch to change the listening port. The first step to changing the Elasticsearch listening port is to turn off the service using a command of ‘**sudo service Elasticsearch stop**.’ Turning off the service allows the user to make changes to the configuration without causing issues to the tool. After turning Elasticsearch off, the user should open the Linux command line and escalate their privileges to the root user with ‘**sudo su -**.’ Next, the user should navigate to the “etc/elasticsearch” directory as it contains the “elasticsearch.yml” configuration file (Networking | Kibana Guide [7.15] | Elastic, n.d.). By using the “cat” (‘**cat elasticsearch.yml**’) or “nano” (‘**nano elasticsearch.yml**’, the user can remove the comment on the line that reads “http.port” and enter the port that they want Elasticsearch to listen on. The figures below verify that this process works with a port number of 9201.

A screenshot of a computer

Description automatically generated

Figure : Changing the Elasticsearch Listening Port

Text

Description automatically generated

Figure : Verifying Elasticsearch on New Port

**Question 2: What was the IP address that showed up the most in the log?**

As stated above, the IP address that showed up the most in the auth.log file was 192.168.20.185. Looking at Figure 19 below, one can see that this IP address was logged 4 different times.

Chart

Description automatically generated

Figure : Logs per IP Address

**Question 3: What user account did they successfully log in as?**

Using the Elastic search query, it can be determined that the user with the 192.168.20.185 IP address was able to successfully log in as the ‘**root**’ user. Because the auth.log file is not very long, it is relatively easy to use keyword queries to locate the user account that was accessed.

Graphical user interface, text, application

Description automatically generated

Figure : Finding Successful Log Ins

**Question 4: What query did you use to find the successful authentication for that IP address?**

As detailed in the lab instructions, the Elastic query function has an implicit logical “OR” in between each word. Therefore, in order to produce the most detailed search query to find the user account that was accessed, a user can use “AND” in between each word. For example, a query of ‘**Accepted AND password AND 192.168.20.185**’ returns the exact log entry that shows the “**root**” user account was successfully authenticated.

Graphical user interface, text, application, email

Description automatically generated

Figure : Building an Elastic Search Query

# **Section 4: References**

Abueg, R. *Elasticsearch: What It Is, How It Works, And What It’s Used For | KnowI.* (2020). Retrieved December 1, 2021, from <https://www.knowi.com/blog/what-is-elastic-search/>

*Defining Log Management | The Importance of Logging and Best Practices | Humio* (2021.). Retrieved December 2, 2021, from <https://www.humio.com/glossary/log-management/>

*Install Elasticsearch with Debian Package | Elasticsearch Guide [7.15] | Elastic*. (n.d.). [Learn/Docs/Elasticsearch/Reference/7.15]. Retrieved November 28, 2021, from <https://www.elastic.co/guide/en/elasticsearch/reference/current/deb.html>

*Kibana Query Language | Kibana Guide [7.15] | Elastic*. (n.d.). [Learn/Docs/Kibana/Reference/7.15]. Retrieved December 1, 2021, from <https://www.elastic.co/guide/en/kibana/current/kuery-query.html>

*Lucene query syntax | Kibana Guide [7.15] | Elastic*. (n.d.). [Learn/Docs/Kibana/Reference/7.15]. Retrieved December 1, 2021, from <https://www.elastic.co/guide/en/kibana/current/lucene-query.html>

Melo, V. *A Practical Introduction to Elasticsearch with Kibana | Medium.* (2018). Retrieved December 1, 2021, from <https://medium.com/@victorsmelopoa/an-introduction-to-elasticsearch-with-kibana-78071db3704>

*Networking | Elasticsearch Guide [7.15] | Elastic.* (n.d.). [Learn/Docs/Elasticsearch/Reference/7.15]. Retrieved November 30, 2021, from <https://www.elastic.co/guide/en/elasticsearch/reference/current/modules-network.html>

Sureka, A. *What is Kibana Used for? 10 Important Features to Know | ClarionTech.* (2020). Retrieved December 1, 2021, from <https://www.clariontech.com/platform-blog/what-is-kibana-used-for-10-important-features-to-know>

*What is Kibana? | Elastic.* (n.d.) Retrieved December 1, 2021, from <https://www.elastic.co/what-is/kibana>