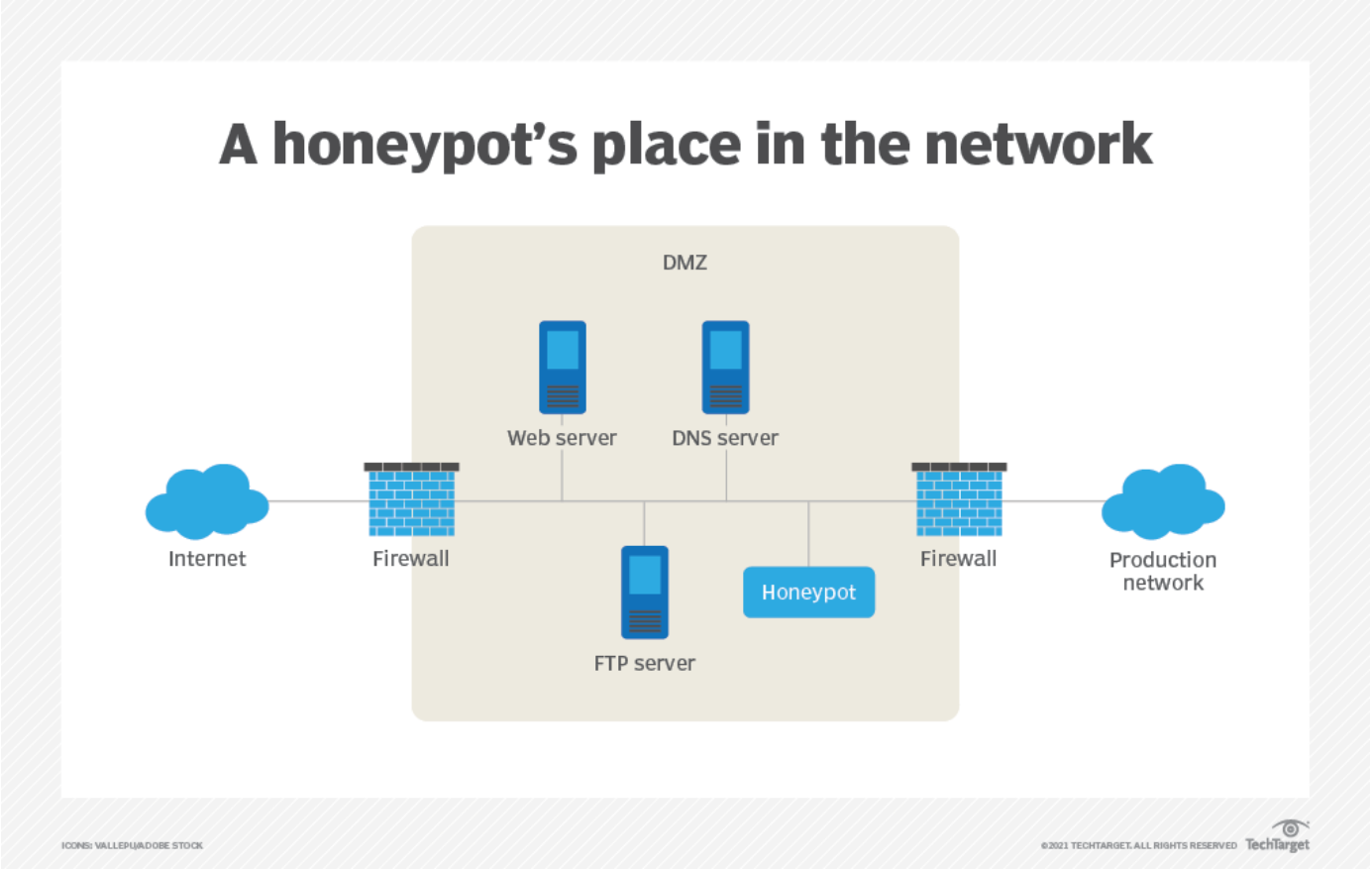
Framework for forensics analysis of the cyber attack on the network using deception technique.



Scenario:

Let's consider a hypothetical crime scene scenario involving an attempted SSH (Secure Shell) attack, and how honeypots and tokens can be used to detect and prevent such an attack.

Scenario: Attempted SSH Attack on a Corporate Server

Background:

In a corporate environment managing sensitive data, a legitimate SSH server is integral for secure remote access. Reports of unusual activity, including multiple failed login attempts and unexpected connection sources, raise suspicions of an attempted SSH attack. To fortify defenses, a cybersecurity analyst implements a honeypot SSH server alongside the authentic one. The honeypot, a decoy server, attracts potential attackers, while both servers are equipped with unique tokens embedded in the login process. This multifaceted defense strategy aims to detect unauthorized access attempts promptly and gather valuable insights into attackers' methodologies for enhanced cybersecurity measures.

Setup:

Legitimate SSH Server: The corporate organization has a legitimate SSH server that employees use for remote access to the internal network.

Honeypot SSH Server: You've set up a honeypot SSH server, which is a decoy server designed to attract and detect attackers. It mimics the appearance of a real SSH server but is isolated from the actual production environment.

Tokens: Tokens are unique identifiers or credentials that are embedded within the SSH server's login process. Legitimate users possess the correct tokens, while attackers are likely to lack them.

High-Level Information Flow Diagram:

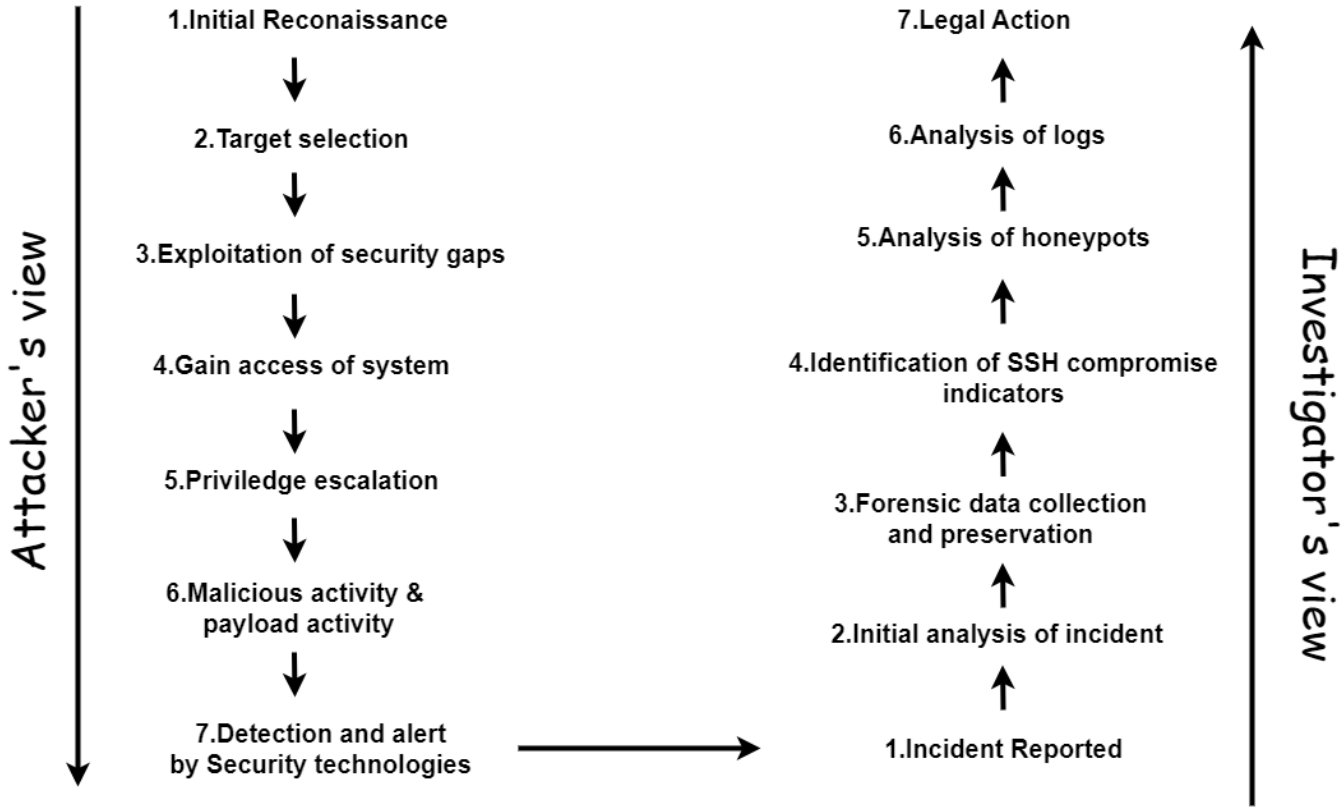


Fig.1 High Level Information Diagram

Resources Required for Investigation:

Resource	Need
Honeypot Server	A dedicated server or virtual machine to act as the honeypot. This can be a separate physical machine or a segmented virtual environment.
Legitimate Server	Ensure the legitimate server is properly configured for security, including the use of strong authentication mechanisms and access controls.
Monitoring and Alerting	Establish automated alerts to notify security personnel of any suspicious activities, login attempts, or deviations from normal behavior.
Honeytrap	Honeytrap is an extensible and opensource framework for running, monitoring, and managing honeypots
Python Scripts	Python script to convert logs gathered by honeytrap into a more usable format and perform some queries itself.
Kibana	Run data analytics at speed and scale for observability, security, and search with Kibana. Kibana gives you the ability to understand your data quickly, spot trends and anomalies at a glance, and route findings to the correct team on the spot.
Skilled Analysts	Trained cybersecurity professionals capable of interpreting alerts, analyzing logs, and responding to potential threats

Data List:

Data Search Lead List:

A data search lead list is a compilation of leads or potential contacts generated through a systematic process of searching for relevant information. This process often involves using various online platforms, databases, or search engines to identify individuals, businesses, or organizations that match specific criteria or keywords. The resulting list serves as a starting point for sales, marketing, or research efforts, helping organizations target and engage with potential customers, partners, or collaborators.

Extracted Data List:

An extracted data list is a collection of information obtained or "extracted" from various sources. This data can be gathered through methods such as web scraping, data mining, or other automated processes. The extracted data list may include details such as names, addresses, contact numbers, email addresses, or any other relevant information based on the purpose of the extraction. Organizations often use extracted data lists for analytics, market research, lead generation, or to populate databases for various business operations.

Data lists

Chain of Custody Process :

In order to preserve digital evidence, the chain of custody should span from the first step of data collection to examination, analysis, reporting, and the time of presentation to the Courts. This is very important to avoid the possibility of any suggestion that the evidence has been compromised in any way.

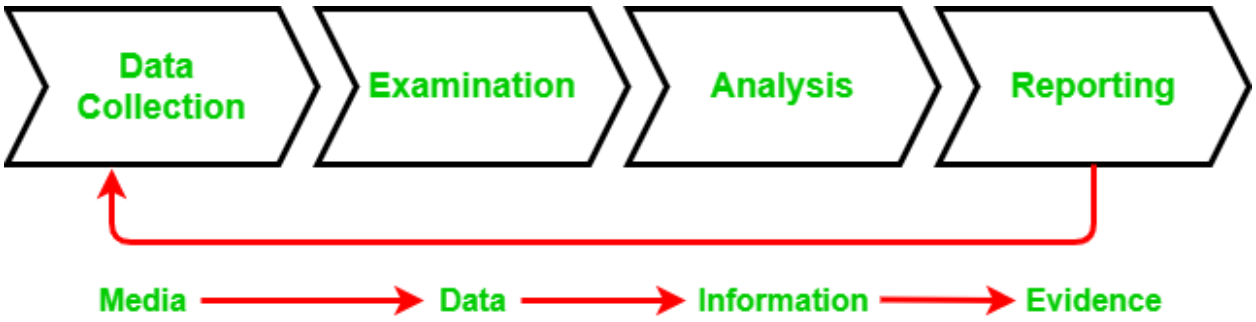


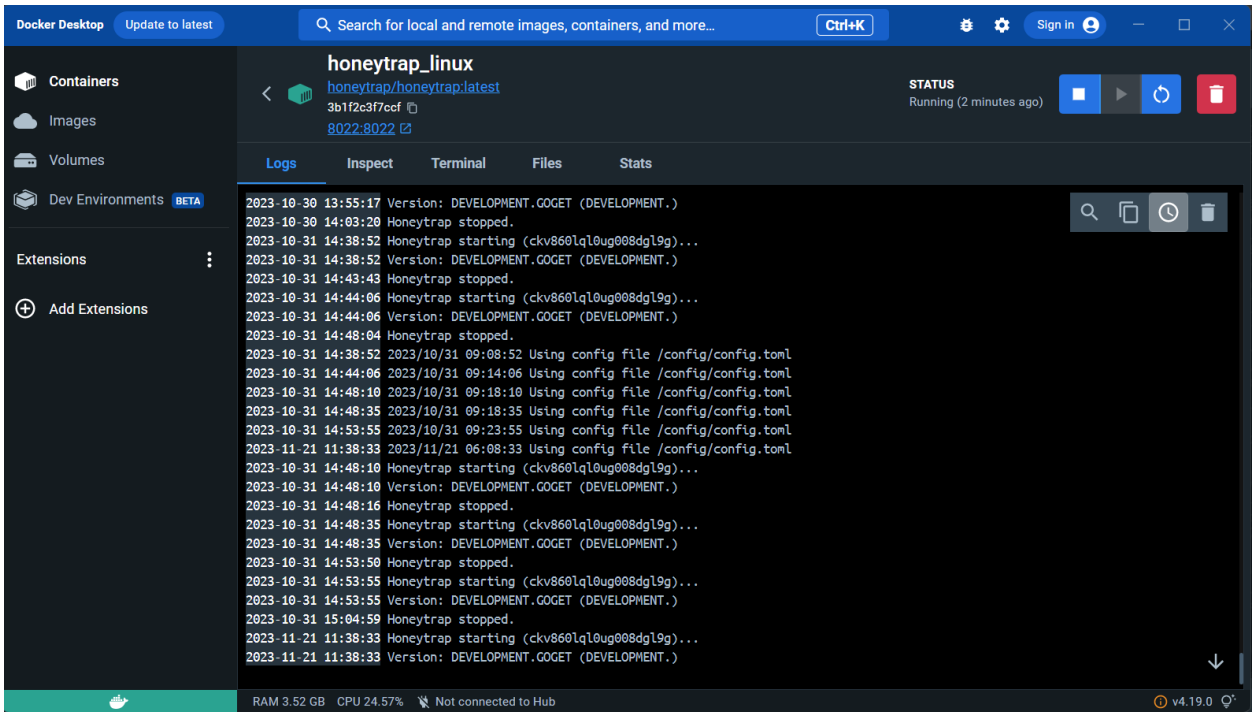
Fig2.,Chain of Custody.

Each stage of the chain of custody in detail:

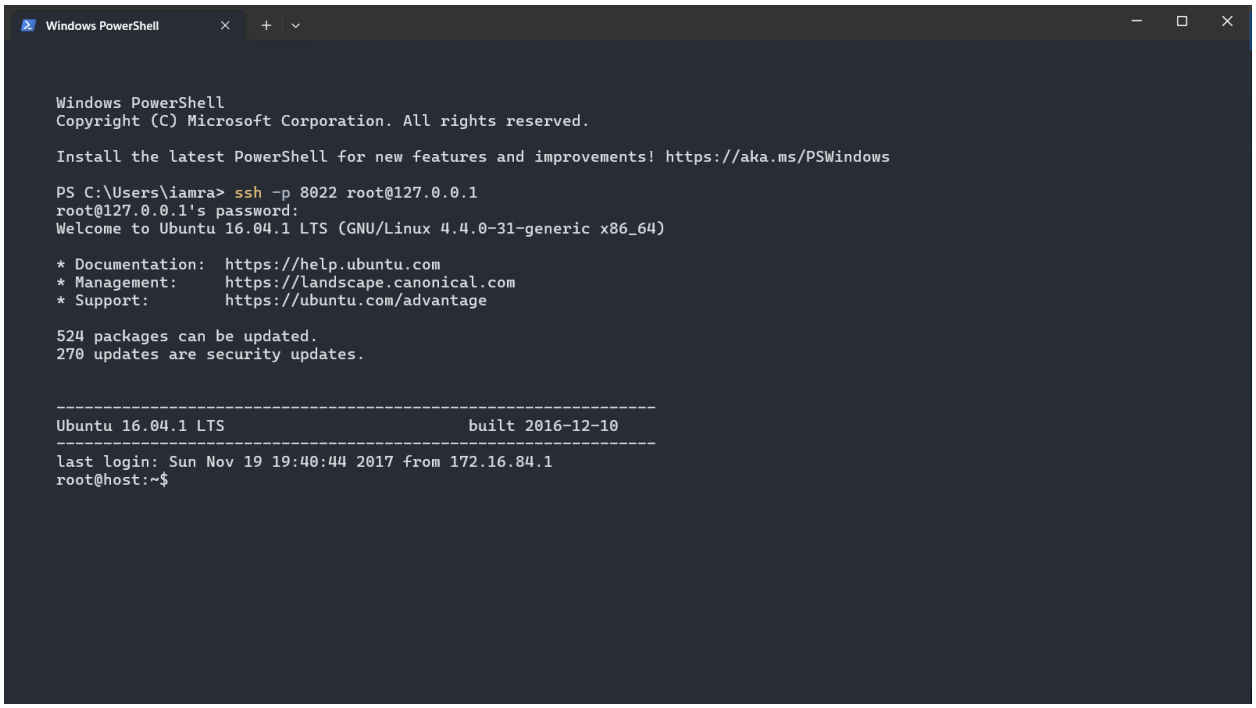
- **Data Collection:** This is where chain of custody process is initiated. It involves identification, labeling, recording, and the acquisition of data from all the possible relevant sources that preserve the integrity of the data and evidence collected.
- **Examination:** During this process, the chain of custody information is documented outlining the forensic process undertaken. It is important to capture screenshots throughout the process to show the tasks that are completed and the evidence uncovered.
- **Analysis:** This stage is the result of the examination stage. In the Analysis stage, legally justifiable methods and techniques are used to derive useful information to address questions posed in the particular case.
- **Reporting:** This is the documentation phase of the Examination and Analysis stage. Reporting includes the following:
    1. Statement regarding Chain of Custody.
    2. Explanation of the various tools used.
    3. A description of the analysis of various data sources.
    4. Issues identified.
    5. Vulnerabilities identified.
    6. Recommendation for additional forensics measures that can be taken.

Artifacts:

Starting the Honeytrap server:



Attacker’s view of the server:



Logs generated by the server:


```
{"category":"heartbeat","date":"2023-11-21T06:10:33.974454008Z","sensor":"honeytrap","sequence":3,"token":"ckv860lql0ug008dgl9g","type":"info"}
{"category":"heartbeat","date":"2023-11-21T06:11:03.987522707Z","sensor":"honeytrap","sequence":4,"token":"ckv860lql0ug008dgl9g","type":"info"}
{"category":"heartbeat","date":"2023-11-21T06:11:33.97528014Z","sensor":"honeytrap","sequence":5,"token":"ckv860lql0ug008dgl9g","type":"info"}
{"category":"heartbeat","date":"2023-11-21T06:12:03.974240777Z","sensor":"honeytrap","sequence":6,"token":"ckv860lql0ug008dgl9g","type":"info"}
{"category":"ssh","date":"2023-11-21T06:12:15.826169327Z","destination-ip":"172.17.0.2","destination-port":8022,"sensor":"services","source-ip":"172.17.0.1","source-port":54692,"ssh.password":"password","ssh.sessionid":"cle4ke5ql0ug00b9o5p0","ssh.username":"root","token":"ckv860lql0ug008dgl9g","type":"password-authentication"}
{"category":"ssh","date":"2023-11-21T06:12:15.855408335Z","destination-ip":"172.17.0.2","destination-port":8022,"sensor":"services","source-ip":"172.17.0.1","source-port":54692,"ssh.payload":"AAAADnh0ZXJtLTI1NmNvbG9yAAAAeAAAAB8AAAKAAAAB4AAAAAuBAAAlgIAAACWAAA==","ssh.request-type":"pty-req","ssh.sessionid":"cle4ke5ql0ug00b9o5p0","token":"ckv860lql0ug008dgl9g","type":"ssh-request"}
{"category":"ssh","date":"2023-11-21T06:12:15.856573719Z","destination-ip":"172.17.0.2","destination-port":8022,"sensor":"services","source-ip":"172.17.0.1","source-port":54692,"ssh.payload":"","ssh.request-type":"shell","ssh.sessionid":"cle4ke5ql0ug00b9o5p0","token":"ckv860lql0ug008dgl9g","type":"ssh-request"}
{"category":"heartbeat","date":"2023-11-21T06:12:33.973738209Z","sensor":"honeytrap","sequence":7,"token":"ckv860lql0ug008dgl9g","type":"info"}
{"category":"ssh","date":"2023-11-21T06:12:58.253582227Z","destination-ip":"172.17.0.2","destination-port":8022,"sensor":"services","source-ip":"172.17.0.1","source-port":49182,"ssh.password":"rr","ssh.sessionid":"cle4kptql0ug00b9o5pg","ssh.username":"root","token":"ckv860lql0ug008dgl9g","type":"password-authentication"}
{"category":"ssh","date":"2023-11-21T06:13:01.901577192Z","destination-ip":"172.17.0.2","destination-port":8022,"sensor":"services","source-ip":"172.17.0.1","source-port":49182,"ssh.password":"pass123","ssh.sessionid":"cle4kptql0ug00b9o5pg","ssh.username":"root","token":"ckv860lql0ug008dgl9g","type":"password-authentication"}
{"category":"heartbeat","date":"2023-11-21T06:13:03.97306965Z","sensor":"honeytrap","sequence":8,"token":"ckv860lql0ug008dgl9g","type":"info"}
{"category":"ssh","date":"2023-11-21T06:13:07.176112065Z","destination-ip":"172.17.0.2","destination-port":8022,"sensor":"services","source-ip":"172.17.0.1","source-port":49182,"ssh.password":"123456","ssh.sessionid":"cle4kptql0ug00b9o5pg","ssh.username":"root","token":"ckv860lql0ug008dgl9g","type":"password-authentication"}
{"category":"ssh","date":"2023-11-21T06:13:13.425695962Z","destination-ip":"172.17.0.2","destination-port":8022,"sensor":"services","source-ip":"172.17.0.1","source-port":56368,"ssh.password":"123456789","ssh.sessionid":"cle4ktdql0ug00b9o5q0","ssh.username":"root","token":"ckv860lql0ug008dgl9g","type":"password-authentication"}
{"category":"ssh","date":"2023-11-21T06:13:17.162426767Z","destination-ip":"172.17.0.2","destination-port":8022,"sensor":"services","source-ip":"172.17.0.1","source-port":56368,"ssh.password":"qwertyu","ssh.sessionid":"cle4ktdql0ug00b9o5q0","ssh.username":"root","token":"ckv860lql0ug008dgl9g","type":"password-authentication"}
```


Python code to format the raw data:

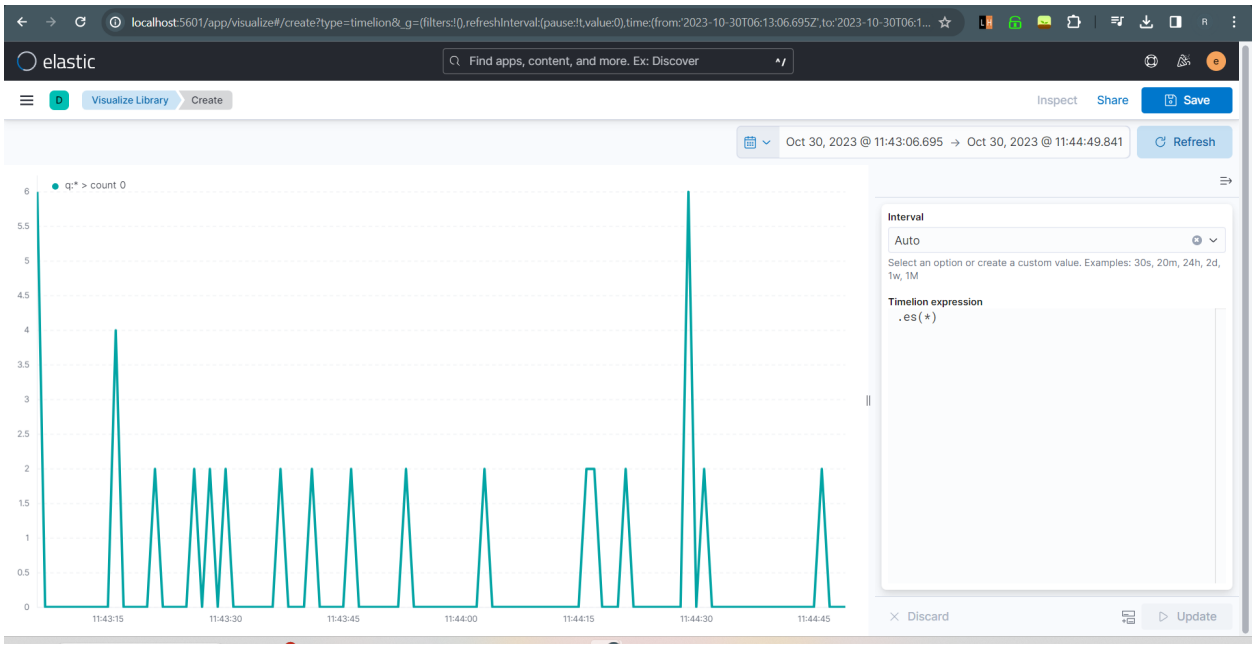
```python
import re
import json

def extract_ip_and_port(line):
    pattern = r'(\d+\.\d+\.\d+\.\d+):(\d+)'
    match = re.search(pattern, line)

    if match:
        ip_address = match.group(1)
        port = match.group(2)
        return ip_address, port

    return None, None

def parse_log(log_file_path, output_json_file):
    with open(log_file_path, 'r', encoding='utf-8') as file:
        log_lines = file.readlines()

    json_objects = []

    for line in log_lines:
        # Check for disconnect event
        if 'honeytrap/server ▶ DEBU' in line and 'Disconnected connection' in line:
            first_ip, second_ip = extract_ip_and_port(line)
            print("Connection from %s:%s disconnected" % (first_ip, second_ip))

        # Extract JSON payload
        try:
            json_data = json.loads(line)
            print("JSON Payload:", json_data)
            json_objects.append(json_data)
        except json.JSONDecodeError:
            pass

    # Write JSON objects to a file with newline delimiter
    with open(output_json_file, 'w', encoding='utf-8') as json_file:
        for obj in json_objects:
            json_file.write(json.dumps(obj))
            json_file.write('\n')

if __name__ == "__main__":
    log_file_path = '/content/log2.txt'  # Replace with your actual log file path
    output_json_file = '/content/output.json'  # Replace with your desired output JSON file path
    parse_log(log_file_path, output_json_file)
```

Formatted data:

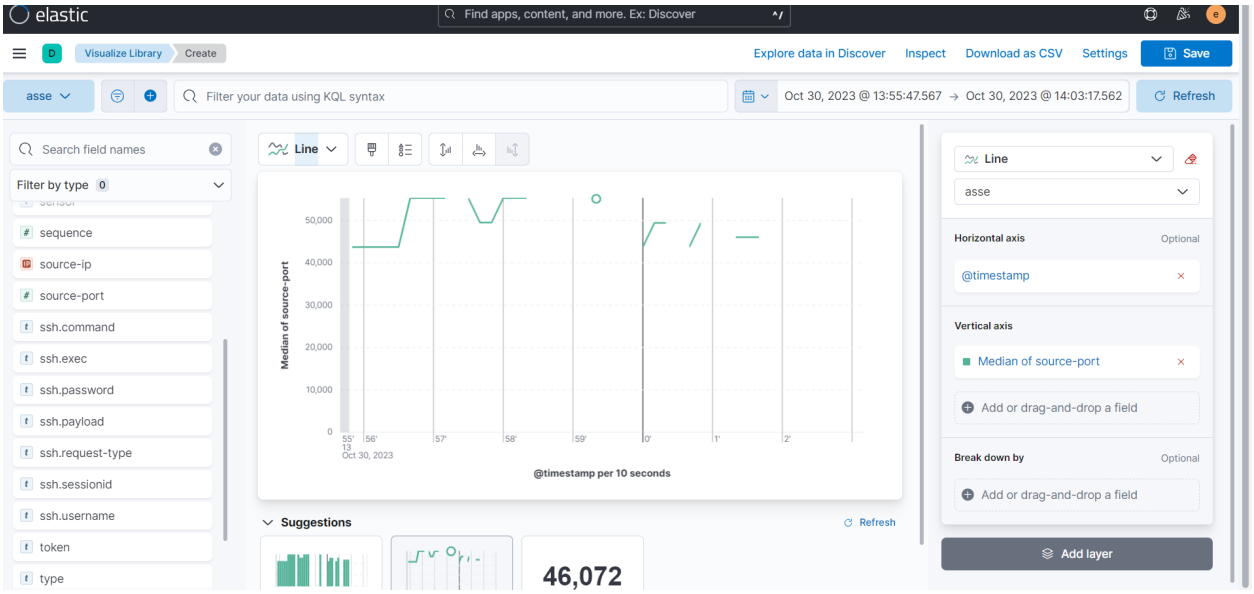

```
JSON Payload: {'category': 'heartbeat', 'date': '2023-10-30T08:25:47.567730696Z', 'sensor': 'honeytrap', 'sequence': 0, 'token': 'ckv860lql0ug008dgl9g', 'type': 'info'}
JSON Payload: {'category': 'ssh', 'date': '2023-10-30T08:25:55.518301893Z', 'destination-ip': '172.17.0.2', 'destination-port': 8022, 'sensor': 'services', 'source-ip': '172.17.0.1',
JSON Payload: {'category': 'ssh', 'date': '2023-10-30T08:25:55.557132209Z', 'destination-ip': '172.17.0.2', 'destination-port': 8022, 'sensor': 'services', 'source-ip': '172.17.0.1',
JSON Payload: {'category': 'ssh', 'date': '2023-10-30T08:26:02.136941252Z', 'destination-ip': '172.17.0.2', 'destination-port': 8022, 'sensor': 'services', 'source-ip': '172.17.0.1',
JSON Payload: {'category': 'ssh', 'date': '2023-10-30T08:26:03.725315629Z', 'destination-ip': '172.17.0.2', 'destination-port': 8022, 'sensor': 'services', 'source-ip': '172.17.0.1',
JSON Payload: {'category': 'ssh', 'date': '2023-10-30T08:26:03.980504814Z', 'destination-ip': '172.17.0.2', 'destination-port': 8022, 'sensor': 'services', 'source-ip': '172.17.0.1',
JSON Payload: {'category': 'ssh', 'date': '2023-10-30T08:26:04.103955932Z', 'destination-ip': '172.17.0.2', 'destination-port': 8022, 'sensor': 'services', 'source-ip': '172.17.0.1',
JSON Payload: {'category': 'ssh', 'date': '2023-10-30T08:26:04.104164782Z', 'destination-ip': '172.17.0.2', 'destination-port': 8022, 'sensor': 'services', 'source-ip': '172.17.0.1',
JSON Payload: {'category': 'ssh', 'date': '2023-10-30T08:26:06.085679326Z', 'destination-ip': '172.17.0.2', 'destination-port': 8022, 'sensor': 'services', 'source-ip': '172.17.0.1',
JSON Payload: {'category': 'ssh', 'date': '2023-10-30T08:26:06.165024429Z', 'destination-ip': '172.17.0.2', 'destination-port': 8022, 'sensor': 'services', 'source-ip': '172.17.0.1',
JSON Payload: {'category': 'ssh', 'date': '2023-10-30T08:26:06.165209049Z', 'destination-ip': '172.17.0.2', 'destination-port': 8022, 'sensor': 'services', 'source-ip': '172.17.0.1',
JSON Payload: {'category': 'ssh', 'date': '2023-10-30T08:26:08.311382622Z', 'destination-ip': '172.17.0.2', 'destination-port': 8022, 'sensor': 'services', 'source-ip': '172.17.0.1',
JSON Payload: {'category': 'ssh', 'date': '2023-10-30T08:26:08.442201413Z', 'destination-ip': '172.17.0.2', 'destination-port': 8022, 'sensor': 'services', 'source-ip': '172.17.0.1',
JSON Payload: {'category': 'ssh', 'date': '2023-10-30T08:26:10.127455896Z', 'destination-ip': '172.17.0.2', 'destination-port': 8022, 'sensor': 'services', 'source-ip': '172.17.0.1',
JSON Payload: {'category': 'ssh', 'date': '2023-10-30T08:26:11.663905575Z', 'destination-ip': '172.17.0.2', 'destination-port': 8022, 'sensor': 'services', 'source-ip': '172.17.0.1',
JSON Payload: {'category': 'ssh', 'date': '2023-10-30T08:26:12.772100091Z', 'destination-ip': '172.17.0.2', 'destination-port': 8022, 'sensor': 'services', 'source-ip': '172.17.0.1',
JSON Payload: {'category': 'ssh', 'date': '2023-10-30T08:26:13.771649304Z', 'destination-ip': '172.17.0.2', 'destination-port': 8022, 'sensor': 'services', 'source-ip': '172.17.0.1',
JSON Payload: {'category': 'ssh', 'date': '2023-10-30T08:26:16.151693741Z', 'destination-ip': '172.17.0.2', 'destination-port': 8022, 'sensor': 'services', 'source-ip': '172.17.0.1',
JSON Payload: {'category': 'heartbeat', 'date': '2023-10-30T08:26:17.565200094Z', 'sensor': 'honeytrap', 'sequence': 1, 'token': 'ckv860lql0ug008dgl9g', 'type': 'info'}
JSON Payload: {'category': 'ssh', 'date': '2023-10-30T08:26:20.289492822Z', 'destination-ip': '172.17.0.2', 'destination-port': 8022, 'sensor': 'services', 'source-ip': '172.17.0.1',
JSON Payload: {'category': 'ssh', 'date': '2023-10-30T08:26:37.995296792', 'destination-ip': '172.17.0.2', 'destination-port': 8022, 'sensor': 'services', 'source-ip': '172.17.0.1', '
JSON Payload: {'category': 'heartbeat', 'date': '2023-10-30T08:26:47.565140403Z', 'sensor': 'honeytrap', 'sequence': 2, 'token': 'ckv860lql0ug008dgl9g', 'type': 'info'}
JSON Payload: {'category': 'ssh', 'date': '2023-10-30T08:26:48.487191535Z', 'destination-ip': '172.17.0.2', 'destination-port': 8022, 'sensor': 'services', 'source-ip': '172.17.0.1',
JSON Payload: {'category': 'ssh', 'date': '2023-10-30T08:26:49.541775328Z', 'destination-ip': '172.17.0.2', 'destination-port': 8022, 'sensor': 'services', 'source-ip': '172.17.0.1',
JSON Payload: {'category': 'ssh', 'date': '2023-10-30T08:26:49.568059386Z', 'destination-ip': '172.17.0.2', 'destination-port': 8022, 'sensor': 'services', 'source-ip': '172.17.0.1',
JSON Payload: {'category': 'ssh', 'date': '2023-10-30T08:26:49.568204574Z', 'destination-ip': '172.17.0.2', 'destination-port': 8022, 'sensor': 'services', 'source-ip': '172.17.0.1',
JSON Payload: {'category': 'ssh', 'date': '2023-10-30T08:26:49.581099909Z', 'destination-ip': '172.17.0.2', 'destination-port': 8022, 'sensor': 'services', 'source-ip': '172.17.0.1',
JSON Payload: {'category': 'ssh', 'date': '2023-10-30T08:26:55.001157444Z', 'destination-ip': '172.17.0.2', 'destination-port': 8022, 'sensor': 'services', 'source-ip': '172.17.0.1',
JSON Payload: {'category': 'ssh', 'date': '2023-10-30T08:26:55.025850827Z', 'destination-ip': '172.17.0.2', 'destination-port': 8022, 'sensor': 'services', 'source-ip': '172.17.0.1',
JSON Payload: {'category': 'ssh', 'date': '2023-10-30T08:26:55.025995606Z', 'destination-ip': '172.17.0.2', 'destination-port': 8022, 'sensor': 'services', 'source-ip': '172.17.0.1',
JSON Payload: {'category': 'ssh', 'date': '2023-10-30T08:26:57.766221154Z', 'destination-ip': '172.17.0.2', 'destination-port': 8022, 'sensor': 'services', 'source-ip': '172.17.0.1', '
JSON Payload: {'category': 'ssh', 'date': '2023-10-30T08:26:59.383578416Z', 'destination-ip': '172.17.0.2', 'destination-port': 8022, 'sensor': 'services', 'source-ip': '172.17.0.1',
JSON Payload: {'category': 'ssh', 'date': '2023-10-30T08:27:01.237362764Z', 'destination-ip': '172.17.0.2', 'destination-port': 8022, 'sensor': 'services', 'source-ip': '172.17.0.1',
JSON Payload: {'category': 'ssh', 'date': '2023-10-30T08:27:07.563167206Z', 'destination-ip': '172.17.0.2', 'destination-port': 8022, 'sensor': 'services', 'source-ip': '172.17.0.1',
JSON Payload: {'category': 'ssh', 'date': '2023-10-30T08:27:09.208761001Z', 'destination-ip': '172.17.0.2', 'destination-port': 8022, 'sensor': 'services', 'source-ip': '172.17.0.1',
JSON Payload: {'category': 'ssh', 'date': '2023-10-30T08:27:14.633986067Z', 'destination-ip': '172.17.0.2', 'destination-port': 8022, 'sensor': 'services', 'source-ip': '172.17.0.1',
JSON Payload: {'category': 'ssh', 'date': '2023-10-30T08:27:16.804899183Z', 'destination-ip': '172.17.0.2', 'destination-port': 8022, 'sensor': 'services', 'source-ip': '172.17.0.1',
JSON Payload: {'category': 'heartbeat', 'date': '2023-10-30T08:27:17.565412042Z', 'sensor': 'honeytrap', 'sequence': 3, 'token': 'ckv860lql0ug008dgl9g', 'type': 'info'}
JSON Payload: {'category': 'ssh', 'date': '2023-10-30T08:27:19.613443581Z', 'destination-ip': '172.17.0.2', 'destination-port': 8022, 'sensor': 'services', 'source-ip': '172.17.0.1',
JSON Payload: {'category': 'ssh', 'date': '2023-10-30T08:27:34.542011434Z', 'destination-ip': '172.17.0.2', 'destination-port': 8022, 'sensor': 'services', 'source-ip': '172.17.0.1',
JSON Payload: {'category': 'ssh', 'date': '2023-10-30T08:27:38.268118444Z', 'destination-ip': '172.17.0.2', 'destination-port': 8022, 'sensor': 'services', 'source-ip': '172.17.0.1',
```

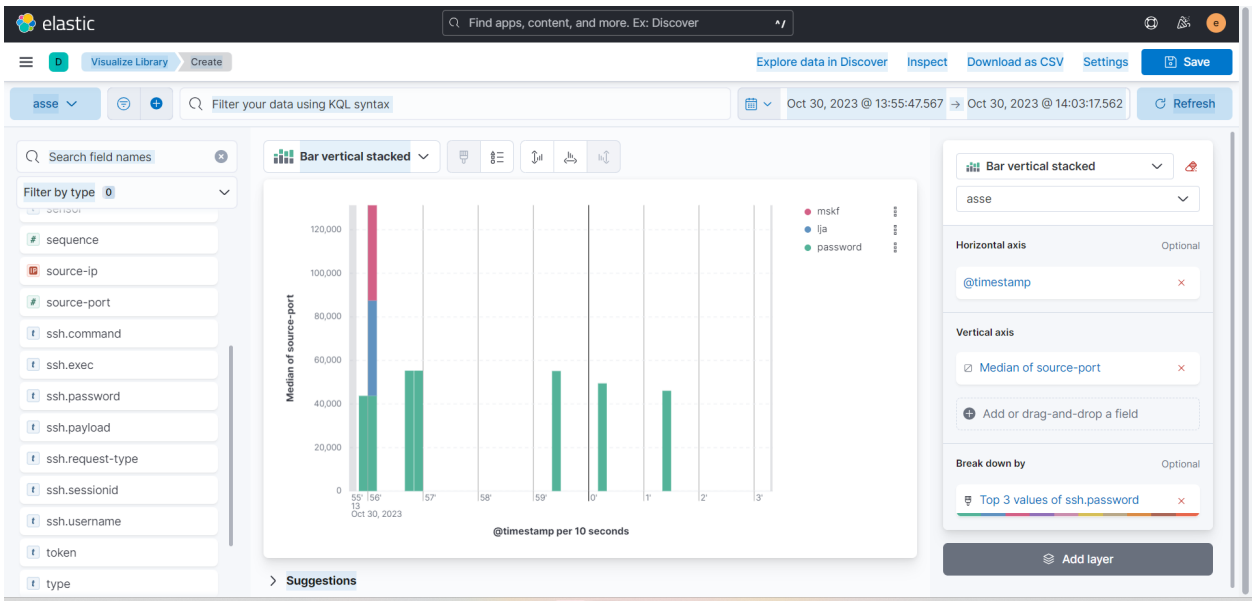

Kibana visualizations:

The graph depicts a chronological analysis of cybersecurity attacks, showcasing the evolving landscape of security incidents over time. The x-axis represents time, segmented into discrete intervals, while the y-axis quantifies the number of attacks or their frequency. The data is sourced from log files or other security-related datasets stored in Elasticsearch, and Kibana is utilized to transform this raw data into meaningful insights.



By visualizing the number of attacks over time using Kibana, organizations gain a powerful tool for monitoring, analyzing, and responding to cybersecurity threats effectively. This graphical representation enhances situational awareness, aids in trend analysis, and empowers security professionals to make informed decisions to fortify their digital defenses.





Examination of Evidence Item 650-457-42715-1
Server logs created by the Honeypot SSH Server

Introduction:

A request for service was issued to conduct a digital forensic examination of the evidence image of the logs created by the honeypot server placed in the organization's database server, evidence item 650-457-42715-1. IP address (172.17.0.1) belonging to a former employee John Doe is suspected of being involved in the illegal access and attempted modification of the database server configurations to give himself elevated privileges.

The organization issued the request for examination to provide any and all information that suggests John Doe either was or was not involved in the attempted exploitation. Whether the access from the given IP was just an innocent mistake or intentional misconduct.

Evidence Summary:

The evidence item 650-457-42715-1 was examined to prove the involvement of John Doe in the illegal access to the server. Certainly, with that context in mind, we will perform a detailed digital forensic examination on the evidence collected from the honeypot logs related to the suspected involvement of the former employee, John Doe. Steps include:

Timeline Analysis: Create a timeline of events based on the timestamps in the logs. This will help establish a chronological order of actions taken by the IP address (172.17.0.1).
The inferred evidence was:

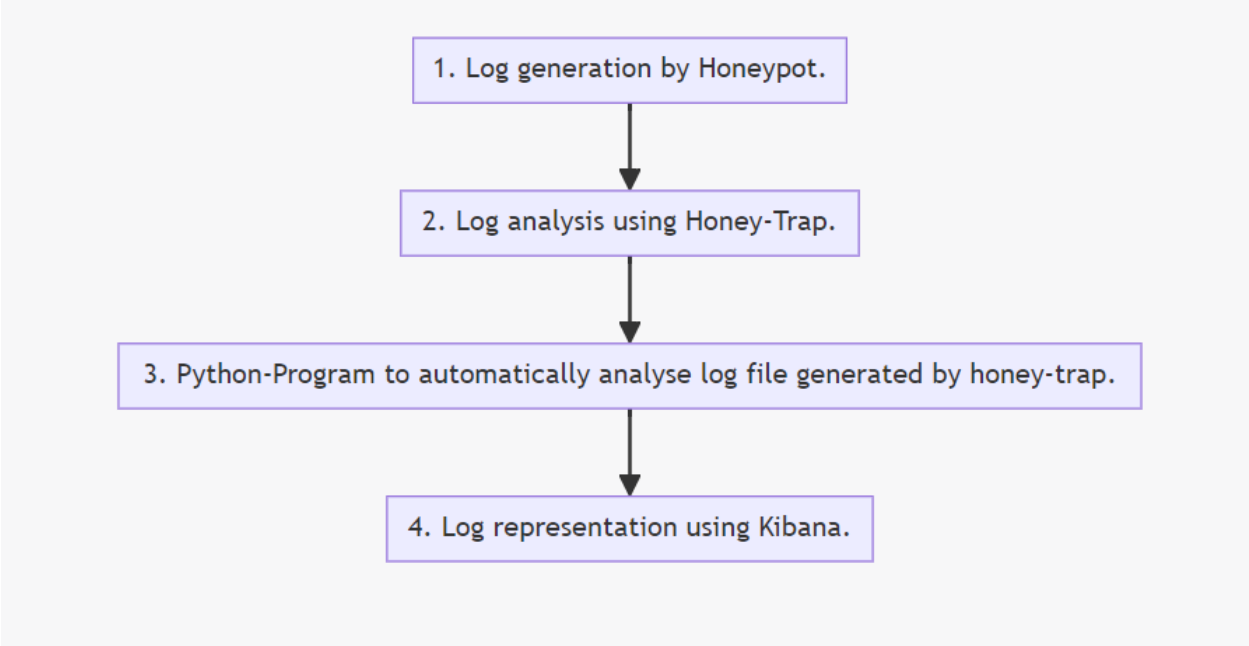
- Lack of Knowledge of Correct Password: The motive to discover or guess the correct password through a brute-force or guessing attack was identified. (Fig.,3)
- Attempted Stealth: Slow keystrokes could be an attempt to avoid detection by security systems that may be designed to identify and block automated login attempts. The individual may be trying to stay under the radar and minimize the risk of being detected. (Fig.,4)
- Information Gathering: The hacker is actively engaged in information gathering, specifically about the existing user accounts on the server. This suggests a strategic approach to understanding the system's user landscape. (Fig.,5)
- Intent for Deeper Access: The attempt to elevate privileges indicates that the hacker is not satisfied with the current level of access. This suggests a motive to gain deeper and potentially more impactful control over the server. (Fig.,6)
- Data Theft or Modification: Changing the database settings suggests an intent to either steal data or modify existing data. This could include altering records, deleting information, or inserting malicious data. (Fig.,7)

- Command Analysis: Review the commands executed on the honeypot server. Look for any attempts to modify database server configurations or gain elevated privileges. (Fig.,8)
- IP Address Attribution: Confirm the attribution of the IP address (172.17.0.1) to John Doe. Cross-reference this information with network access records and any other relevant data. (Fig.,9)

The goal is to provide an objective and thorough analysis of the evidence to determine if John Doe's actions were intentional misconduct or an innocent mistake.

Examination Summary:

The tools and steps used in the investigation were as follows:



Let's break down each step:

1. Log Generation by Honeypot: Honeypot servers are intentionally exposed to attract and log potential attackers. They generate logs containing information about the activities and interactions with the simulated environment.

2. Raw Log Analysis by Investigators: Investigators review the raw logs to gain an initial understanding of the data. This step involves identifying patterns, anomalies, and potential areas of interest within the large dataset.

3. Log Filtering Using Python Code: Python code is employed to filter the raw logs based on specific criteria. This could include filtering by IP addresses, timestamps, or specific keywords that indicate malicious activities. The goal is to reduce the volume of data and focus on relevant information.

4. Analysis of Filtered Data Using Python: Investigators conduct a more in-depth analysis on the filtered data using Python scripts. This step may involve parsing log entries, extracting relevant information, and correlating data points to reconstruct the sequence of events.

5. Representation Using Kibana: Kibana, a data visualization tool, is used to represent the analyzed data in a visually accessible format. Dashboards and visualizations are created to present key findings, trends, and anomalies discovered during the log analysis process.

Chronological Analysis:

1. [Day 1 - 10:00 AM]: IP address 172.17.0.1 (known to belong to John Doe) initiates a connection to the server. The server logs capture 10 consecutive password attempts.

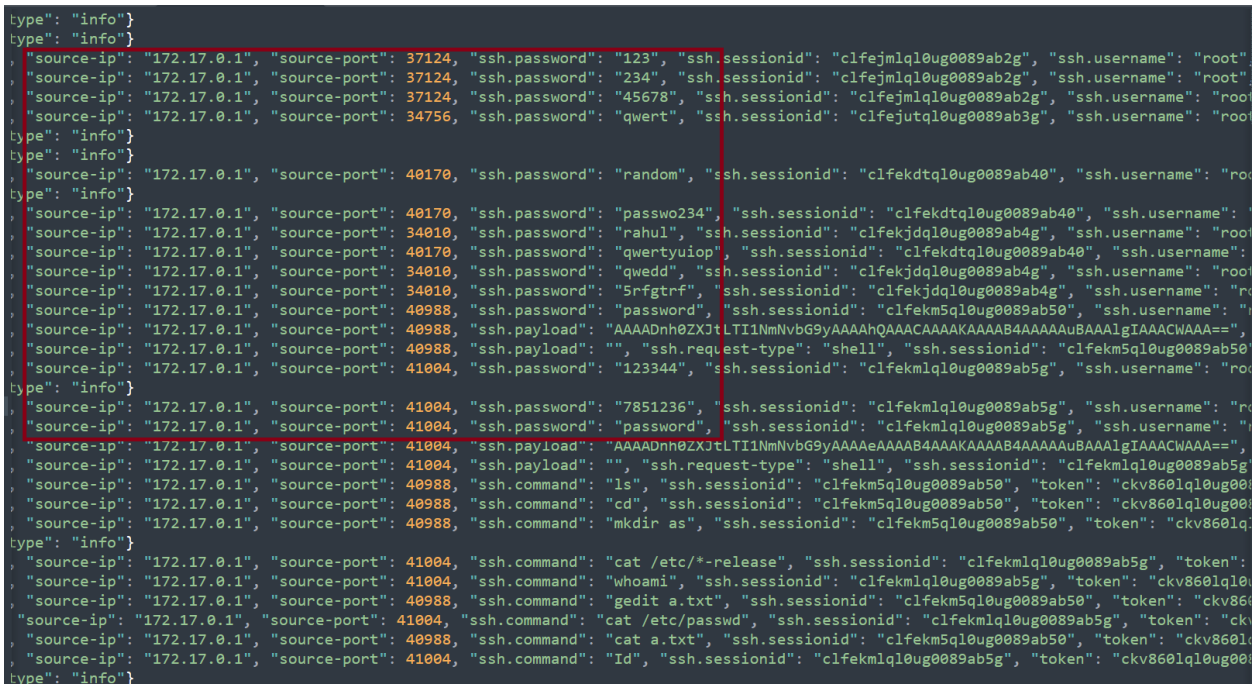


Fig.,3

From the provided information about the hacker's activities on [Day 1 - 10:00 AM], several deductions can be made about the attacker, who is known to be John Doe (IP address 172.17.0.1). Here are some insights:

- Persistence and Determination: The fact that the attacker made 10 consecutive password attempts suggests a high level of persistence and determination to gain unauthorized access to the server. This could indicate a targeted and deliberate effort.

- Lack of Knowledge of Correct Password: The consecutive failed password attempts indicate that the attacker does not possess the correct credentials for the server. This suggests that the motive may be to discover or guess the correct password through a brute-force or guessing attack.

- Manual Entry of Passwords: The slow keystrokes observed during the password attempts suggest that the attacker is manually entering the passwords, rather than using an automated script. This could indicate a more cautious and deliberate approach to avoid detection.

- Potential Insider Threat: Since the IP address is known to belong to John Doe, an insider threat is implied. This raises questions about John Doe's motivations and whether the intent is malicious or if the actions are unintentional (e.g., mistyped passwords).

2. [Day 1 - 10:30 AM]: Slow keystrokes are detected during the login attempts, suggesting manual entry rather than automated script usage.

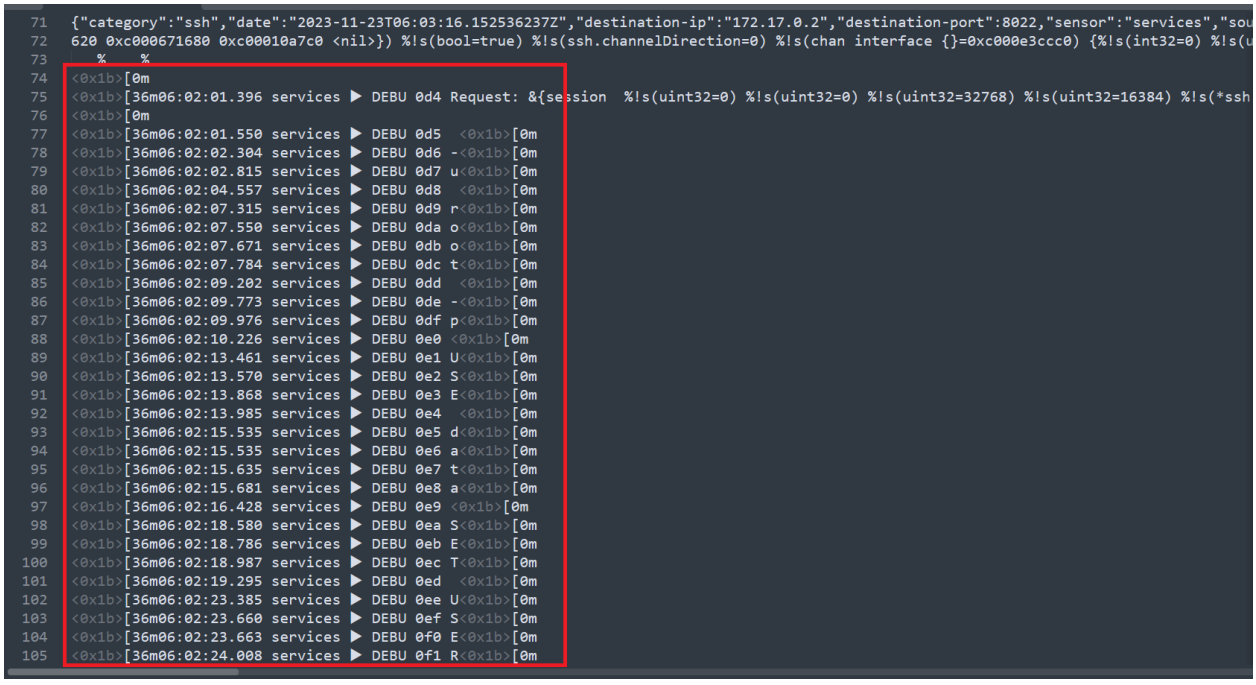


Fig.4

The detection of slow keystrokes during login attempts provides additional insights into the behavior of the individual attempting to access the server. Here are some deductions that can be made:

- Manual Entry: The fact that keystrokes are slow suggests that the login attempts are likely performed manually by a human rather than being automated by a script or tool. This indicates a more deliberate and careful approach to the login process.

- Focused and Targeted Attack: Manual entry of login credentials suggests a more focused and targeted attack. The attacker may be specifically interested in this server or have a particular objective in mind.

- Attempted Stealth: Slow keystrokes could be an attempt to avoid detection by security systems that may be designed to identify and block automated login attempts. The individual may be trying to stay under the radar and minimize the risk of being detected.

- Possibly Insider Threat: The use of slow keystrokes, combined with the knowledge that the IP address belongs to John Doe, might raise suspicions of an insider threat. Insider threats involve individuals with legitimate access attempting unauthorized activities.

3. [Day 1 - 11:15 AM]: The IP address attempts to search for the current user on the server, indicating an attempt to gather information about the existing user accounts.

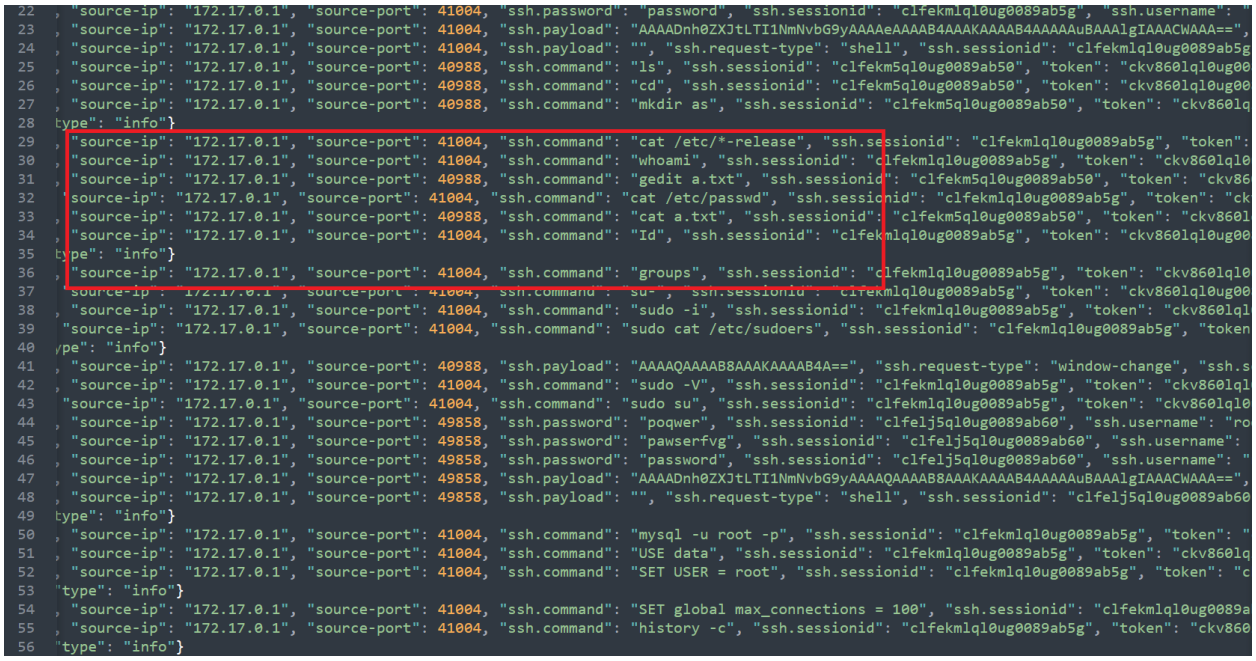


Fig.,5

The point that the IP address attempted to search for the current user on the server provides insights into the hacker's intent and tactics. Here are some deductions that can be made:

- Information Gathering: The hacker is actively engaged in information gathering, specifically about the existing user accounts on the server. This suggests a strategic approach to understanding the system's user landscape.

- Preparation for Privilege Escalation: Gathering information about the existing user accounts is a common precursor to privilege escalation attempts. The hacker may be identifying potential targets for privilege escalation or lateral movement within the network.

4. [Day 1 - 12:00 PM]: Further analysis of the logs reveals an attempt to elevate privileges, suggesting an effort to gain higher-level access on the server.

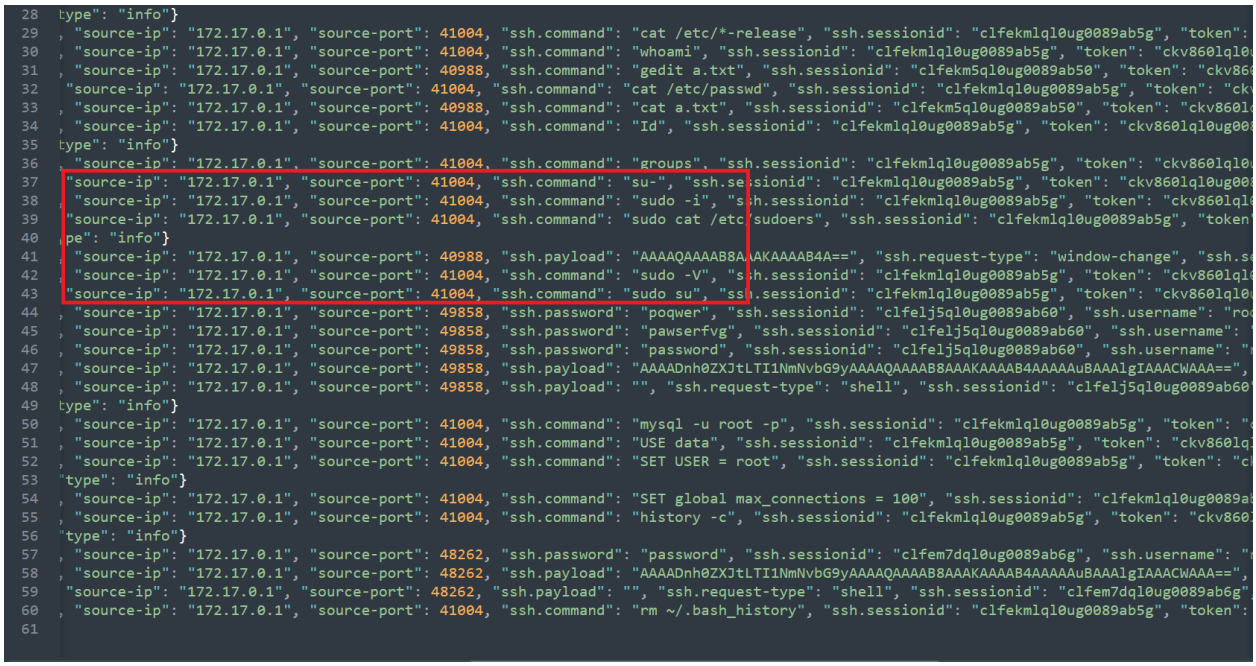


Fig.,6

From the information provided that there was an attempt to elevate privileges on the server, several insights and deductions can be made about the hacker:

- Intent for Deeper Access: The attempt to elevate privileges indicates that the hacker is not satisfied with the current level of access. This suggests a motive to gain deeper and potentially more impactful control over the server.

- Sophistication Level: The act of attempting to escalate privileges requires a certain level of sophistication. It suggests that the hacker is not a novice and has a good understanding of the system's security architecture.

- Objective Beyond Initial Access: The act of privilege escalation often indicates that the hacker has specific objectives beyond the initial unauthorized access. This could include gaining control over critical system functions, accessing sensitive data, or compromising other user accounts.

- Potential Insider Knowledge: The attempt to elevate privileges may also indicate insider knowledge or previous reconnaissance. The hacker might have insights into the server's configuration, security policies, or specific vulnerabilities that could be exploited for privilege escalation.

5. [Day 1 - 1:30 PM]: The IP address makes attempts to change the database name and settings on the server. This indicates an effort to manipulate database configurations.

Fig.,7

The attempt to change the database name and settings on the server suggests several potential motivations and characteristics of the hacker. Here are some deductions that can be made:

- Data Theft or Modification: Changing the database settings suggests an intent to either steal data or modify existing data. This could include altering records, deleting information, or inserting malicious data.

- Targeted Attack: The focus on database configurations indicates a targeted attack rather than random or automated scanning. The hacker seems to have a specific goal related to the server's database.

- Persistence and Sophistication: The attempt to change database settings demonstrates a certain level of persistence and sophistication. It suggests that the hacker is not merely probing for vulnerabilities but actively trying to achieve a specific objective.

- Potential for Long-Term Impact: Database manipulations can have significant and long-term impacts on an organization, affecting data integrity, system performance, and business operations.

6. [Day 1 - 2:45 PM]: Logs show attempts to remove run commands, indicating an effort to cover tracks and erase evidence of activities.

Fig.,8

The attempt to remove run commands suggests several aspects of the hacker's intentions and knowledge:

- Intent to Conceal Activities: The attempt to remove run commands indicates a deliberate effort to conceal the hacker's activities on the server. This suggests a level of awareness about the potential consequences of their actions and a desire to avoid detection.

- Concern for Anonymity: The attempt to remove run commands also reflects a concern for maintaining anonymity. By erasing evidence of their activities, the hacker aims to reduce the likelihood of being traced back to their identity.

7. Day 1 - 4:00 PM]: The same IP address (172.17.0.1) is consistently used throughout the series of activities.

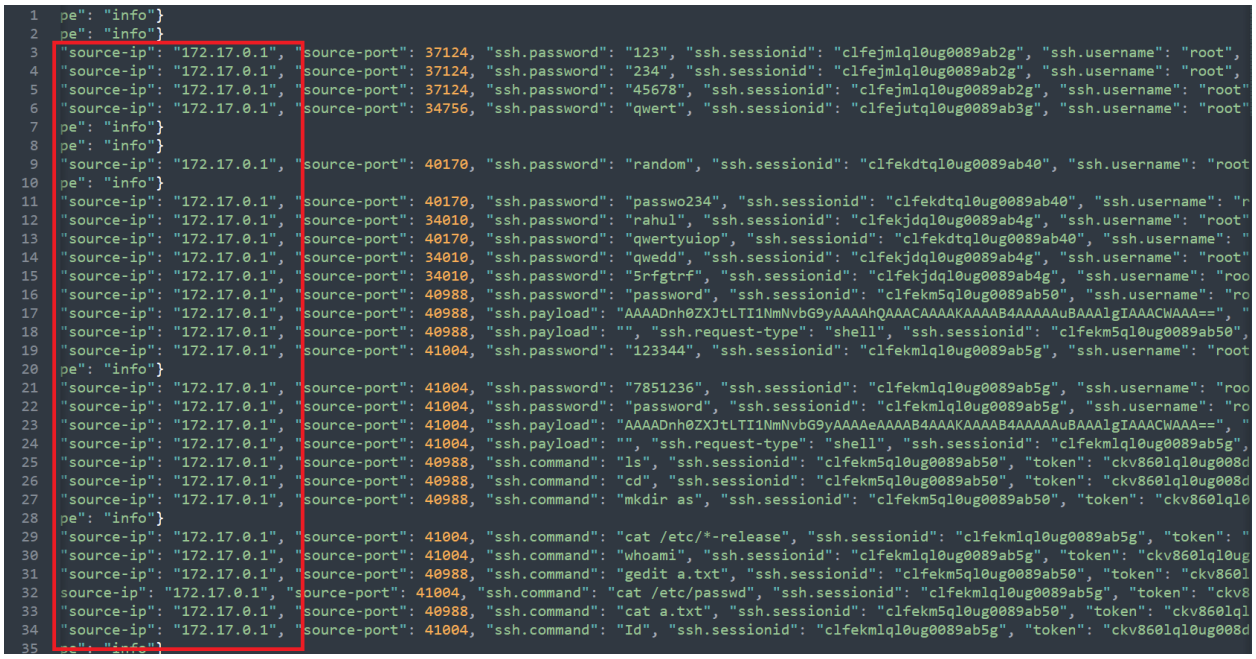


Fig.,9

The consistent use of the same IP address (172.17.0.1) throughout the series of activities provides additional insights into the hacker's behavior and tactics. Here are some deductions that can be made:

- Identity Confirmation: The use of a consistent IP address, especially if it is known to belong to John Doe, helps confirm the identity of the hacker. It strengthens the attribution of the activities to a specific individual.

- Covering Tracks: The consistent use of the same IP address throughout the activities may indicate an attempt to cover tracks. By using a known IP address, the hacker might be trying to blend in with legitimate user activity, making it more challenging to detect and attribute the actions.

This timeline provides a sequential overview of the activities performed by John Doe on the server.

Conclusion:

The chronological analysis of John Doe's activities on the server unveils a pattern of deliberate and sophisticated actions. Starting with 10 consecutive password attempts, John demonstrated persistence and determination, suggesting a targeted effort to gain unauthorized access. The observation of slow keystrokes indicates a manual, cautious approach, possibly to avoid automated detection. Subsequent attempts to search for the current user, elevate privileges, and manipulate database settings showcase strategic objectives beyond initial access, with implications of insider knowledge.

The attempt to erase run commands underscores John's awareness of forensic traces, indicating a desire to cover tracks and maintain anonymity. Importantly, the consistent use of the same IP address throughout the series of activities strengthens the attribution to John Doe, affirming his identity. These insights collectively suggest a sophisticated and potentially malicious intent, raising concerns about insider threats and emphasizing the need for thorough investigation and remediation measures.