*Network Security Monitoring*

*(NSM*)

***Phase3  
Prepared by:  
Lelyan Saadeh***

***Date:***

***July26,2024***

***Prepared For :  
Etienne Pitouti***

**Table of Contents**

1. **Executive Summary …………………………………………… Page3**
2. **Introduction ……………………………………………………. Page3**
3. **Identification of Malicious Traffic …………………………… Page4**

* **Unauthorized Cryptocurrency Mining Incident**

1. **Analysis Steps Taken to Identify the Incident ……………….. Page5**

* **Traffic Analysis**
* **Initial Observations**
* **Detailed Analysis of Protocols and Traffic Patterns**
* **In-depth analysis of Initial Traffic Observations**
* **Possible Use of Domain Generation Algorithms**
* **HTTP POST Request Analysis:**

1. **Implications …………………………………………………….. Page12**
2. **Other Possible Incidents Considered …………………………. Page13**

* **Distributed Denial-of-Service (DDoS) Attack**
* **Internal Misuse**

1. **Risk Analysis of Threats to Company Assets ………………… Page14**

* **Unauthorized Use of System Resources**
* **Exposure of Sensitive Credentials**
* **Unauthorized Data Access and Exploitation**
* **Network Performance Degradation**

1. **Asset List ………………………………………………………… Page18**
2. **Summary ………………………………………………………… Page18**
3. **Glossary …………………………………………………………. Page19**
4. **References ……………………………………………………….. Page19**

**Executive Summary**

This report provides a comprehensive examination of network traffic captured from a client site using Wireshark, with a focus on identifying and addressing malicious activities. The primary concern identified is an Unauthorized Cryptocurrency Mining Attack, where an attacker covertly utilizes a device’s resources for cryptocurrency mining without the owner’s permission.

The report details the methodology used to identify this attack, evaluates its implications, and offers recommendations for prevention. To facilitate understanding and remediation, a thorough asset list of the discovered devices is included. This structured approach ensures clarity regarding the threat and outlines actionable steps to enhance network security.

**Introduction**

This report presents an in-depth investigation of network traffic captured from a client site using Wireshark. The client has reported experiencing bandwidth issues, and given their history of inadequate security practices, there is considerable concern that these issues may be attributed to malicious activities.

The objective of this analysis is to identify any unauthorized activities impacting network performance. By scrutinizing the traffic data, the goal is to uncover potential threats, assess their effects on the network, and provide recommendations for addressing and preventing similar issues in the future.

**Identification of Malicious Traffic**

**NAME: Unauthorized Cryptocurrency Mining Incident**

**Definition:**

An Unauthorized Cryptocurrency Mining Incident happens when someone secretly uses a computer or network to mine cryptocurrency, like Bitcoin, without the owner's permission. This type of attack hijacks computing resources to solve complex problems necessary for mining, which requires a lot of processing power.

Usually, these incidents are carried out through malicious software—such as trojans or malware—that’s secretly installed on the victim’s system. Once installed, this software takes over the computer’s CPU or GPU to perform the mining tasks. The impact can be quite noticeable: the computer may slow down, electricity bills may increase, and the hardware might be strained. Victims might notice their computers running more slowly, their power bills going up, and a general drop in system performance.

To detect these unauthorized mining activities, security experts need to carefully analyze network traffic using tools like Wireshark. By examining the traffic, they can spot unusual patterns or connections to IP addresses known for cryptocurrency mining pools. Identifying these signs helps uncover hidden mining operations and allows for quick action to protect the system’s performance and security.

**Analysis Steps Taken to Identify the Incident**

**Traffic Analysis**

Introduction:

Traffic analysis is a key part of keeping a network secure because it helps spot unusual activity and potential security threats. It all starts with figuring out what normal network traffic looks like. This means understanding the usual volume of traffic, the types of data being sent, and how it's distributed across different ports and IP addresses. By establishing this "baseline," you have a reference point to spot anything that seems out of the ordinary, which might suggest unauthorized activity.

Tools like Wireshark are essential for this kind of analysis. They give you detailed insights into network traffic through stats and visualizations. For instance, Wireshark can show you how different protocols are used, the number of packets being transferred, and the rate at which data moves across the network. If you notice a big jump in traffic on unusual ports or a sudden increase in encrypted data, it might be a sign of something suspicious.

Wireshark also offers visual tools like graphs and charts to make it easier to understand these patterns. Time-based graphs can reveal traffic spikes, while protocol hierarchies and flow diagrams show how data travels through the network. Plus, its filtering features let you focus on specific types of traffic, helping you dig deeper into any potential issues.

Additionally, examining specific protocols like TCP, TLS, and HTTP is essential. TCP manages data packet transmission, so any irregular patterns or unusual connection attempts may indicate problems. TLS encrypts data to protect it from unauthorized access; unexpected increases in encrypted traffic or connections to unknown certificate authorities might suggest security concerns. HTTP, used for web traffic, should be monitored for unusual requests or data loads. Analyzing these protocols helps you gain a clearer understanding of network activities and identify potential issues before they escalate.

**Detailed Steps:**

* **Initial Observations**
* Initial Setup

Opening the File:

I began by opening the uvw.pcap file using Wireshark, a renowned network protocol analyzer. This tool provided a detailed view of the raw network traffic data captured in the file, allowing me to investigate the various aspects of the network communication.

* Evaluating Statistics and Hierarchy

1. **Navigating Wireshark’s Statistics:**

* After loading the file, I accessed the "Statistics" menu in Wireshark. This menu is essential for obtaining a comprehensive overview of the network traffic captured in the pcap file. It offers various tools to analyze and interpret the data effectively.

1. **Using Statistical Tools:**

* Protocol Hierarchy:
* I utilized the Protocol Hierarchy tool to break down the captured traffic by protocol. This view provided insight into the distribution of different network protocols within the dataset. Notably, I observed that TCP constituted 99.74% of the packets, indicating a high dependency on TCP for network communication.
* **Conversations Statistics:**
* also reviewed the Conversations statistics to examine the interactions between different network endpoints. This tool helped me identify which IP addresses were communicating with each other and the volume of data exchanged. It was particularly useful for pinpointing IP addresses with high traffic volumes and understanding the nature of their communication.
* **Detailed Analysis of Protocols and Traffic Patterns**
* Protocols Analysis:
* I found that a staggering 99.74% of the packets were using TCP. TCP is known for its reliable and ordered communication, so seeing such a high percentage is a bit unusual. It suggests that the network is heavily dependent on TCP, which could be a sign of potential issues or misuse.
* Additionally, I noticed that 77.21% of the traffic was encrypted with TLS. While TLS is crucial for keeping data secure while it's being transmitted, having so much of the traffic encrypted means that a lot of the communications are hidden from view.
* Conversation Analysis:
* By examining the Conversations statistics, I identified IP addresses involved in substantial data transfers. For instance, IP addresses 50.7.161.218 and 38.229.70.33 were responsible for large volumes of packets and data, which warranted further investigation to ensure their activities were legitimate.
* High Traffic Volume:
* Traffic from IP Address 172.16.253.129: I noticed that this IP address was generating unusually high volumes of traffic, with significant packet and byte counts across various ports.
* Traffic Volume and Unusual Ports:
* The analysis of high traffic volumes from specific IP addresses, such as 172.16.253.129, highlighted unusual patterns that suggested potential issues or anomalies.
* I also detected traffic using non-standard ports, such as port 9001, which is not typically associated with regular internet traffic. This finding prompted a closer examination of the services or activities using these ports.
* Pattern and Anomaly Detection:
* I observed repetitive high-volume traffic patterns between IP addresses 172.16.253.129 and 188.173.32.149, indicating ongoing data exchanges or possible targeted attacks.
* Traffic Examination:

The IP address 188.173.32.149 looks suspicious because it’s generating a lot of traffic—1,376 packets and 217,451 bytes—with a uniform distribution across multiple ports. This pattern suggests that there might be automated or scripted activity happening. Moreover, all the traffic from this IP is completely filtered, meaning it’s being blocked by security measures. This combination of high traffic, consistent patterns, and complete blockage raises red flags and calls for a closer look to understand what’s going on.

* **In-Depth Analysis of Initial Traffic Observations**
* **Possible Use of Domain Generation Algorithms (DGAs):**

To start the analysis, I focused on TCP and TLS packets due to their high volume in the network traffic. With TCP making up an impressive 99.74% of the packets, it was clear that this protocol was heavily utilized. to isolate these protocols and streamline my analysis, I applied Wireshark filters specifically for TCP (tcp) and TLS (tls). This allowed me to focus on traffic using these protocols and exclude other types of traffic that might dilute the insights.

In addition to examining TCP traffic, I also focus in on IP addresses that showed significant data transfers, specifically 50.7.161.218 and 38.229.70.33. These IPs had been identified earlier as having large data volumes, which often signals potential issues or unusual activities. To analyze traffic related to these IPs more effectively, I used filters to isolate packets involving these addresses. The filters used were ip.addr == 50.7.161.218 and ip.addr == 38.229.70.33, allowing me to focus solely on the communications associated with these IPs.

**Analyzing TCP Streams for Patterns**

While analyzing the traffic within the TCP stream between the internal IP address 172.16.253.129 and the external IPs 38.229.70.33 and 50.7.161.218, I noticed some intriguing patterns in the domain names and data. These patterns strongly suggest the use of Domain Generation Algorithms (DGAs). DGAs are techniques employed by cybercriminals to generate numerous domain names that change frequently, making it challenging to track and block these domains due to their random and rapidly evolving nature.

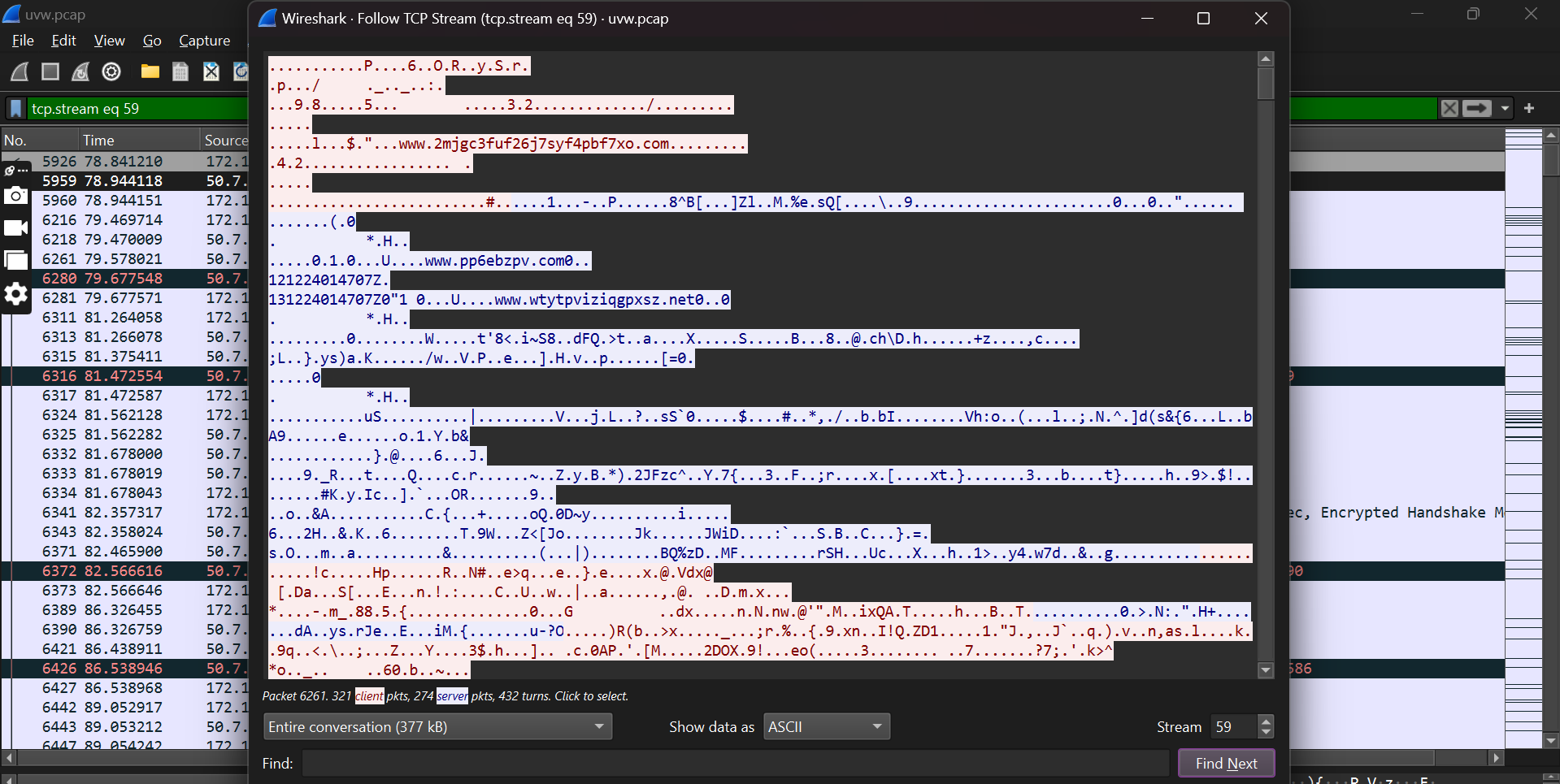


Figure 1: Traffic Analysis Highlighting Domain Generation Algorithms (DGAs) Patterns

The domains I identified display clear signs of Domain Generation Algorithms (DGAs). These domains are characterized by their random and seemingly nonsensical names, such as `www.pp6ebzpv.com0`, `www.wtytpviziqgpxsz.net0`, and `www.pts5agysxnvyyvbysfv.net`. The random and frequently changing nature of these domains makes them hard to blacklist or detect using traditional security measures. This pattern strongly suggests that DGAs are being employed to support ongoing malicious activities, as they help attackers evade detection and maintain covert operations.

Moreover, I observed similar DGA patterns across more than five different IP addresses, all of which exhibited random and frequently changing domain names. This widespread use of DGAs further reinforces the likelihood of sophisticated and coordinated malicious activities.

Additionally, the encrypted and obfuscated data patterns linked with these domains hint at potentially illicit activities, such as unauthorized cryptocurrency mining. The use of encryption and obfuscation serves to hide the true nature of the communication, making detection and response more challenging. This sophisticated approach not only helps in evading immediate detection but also indicates a more advanced attempt to maintain covert operations.

* **HTTP POST Request Analysis:**

The high proportion of encrypted traffic 77.21% TLS suggested that a lot of the network's communication was securely hidden. Given this, it seemed like critical or sensitive activities could be buried within this encrypted data.

Although HTTP traffic was relatively minimal at just 1.93%, HTTP POST requests are often used for transmitting sensitive information like login details, form submissions, and transactions. This made them a prime candidate for further investigation.

After applying the filter (http.request.method == "POST"), I observed that all HTTP POST communications were exclusively between the internal IP 172.16.253.129 and the external IP 188.173.32.149. This finding was particularly noteworthy since 188.173.32.149 had already been flagged as suspicious. By focusing on these specific POST requests, I was able to concentrate on the data exchanged between these two IPs, which allowed me to narrow down the investigation and identify potential unauthorized activities more effectively.

The analysis of the HTTP POST requests uncovered several key insights into the suspicious activity:

* Basic Authentication Exposure: The HTTP POST request used Basic Authentication with credentials encoded in Base64 (dXNlcjU6VUI5N2FkMg==), which decodes to user5:U597ad2. Base64 encoding is not secure, it can be easily reversed to reveal the actual credentials. This vulnerability is a significant security risk, as anyone intercepting the traffic could decode and misuse these credentials.
* Mining Software Identification: The request headers indicated that the client was using cgminer version 2.7.5, a popular cryptocurrency mining software. The headers also included X-Mining-Extensions and X-Mining-Hashrate, showing that the client was actively mining with a reported hashrate of 1,000,000 hashes per second.
* Mining Work Request: The request body contained a JSON payload ({"method": "getwork", "params": [], "id":0}) that is typically used to request new mining work.
* Server Response Analysis: The server responded with an HTTP status line HTTP/1.1 200 OK, which means the request was successfully processed. The response also included a JSON payload with mining-related data, such as hash1, data, target, and midstate.

In analyzing the HTTP POST requests, two major security issues stand out. First, Basic Authentication credentials were encoded in Base64. While this encoding method disguises the data, it does not secure it—anyone intercepting the traffic can easily decode these credentials. For instance, `user5:U597ad2` is vulnerable to unauthorized access, posing a significant risk to system security.

The more alarming discovery is the presence of cgminer version 2.7.5, a cryptocurrency mining software. The request headers revealed a high hashrate of 1,000,000 hashes per second and included specialized mining extensions, indicating intensive use of network resources. Cgminer connects to mining pools, performs complex computations to validate transactions, and competes to add them to the blockchain. This process is resource-intensive and can severely impact network performance.

The HTTP POST requests also showed that the mining software was actively requesting new mining tasks, with server responses including mining-related data such as hash1, data, target, and midstate. This suggests a persistent and ongoing mining operation.

The presence of cgminer indicates that network resources are being misused, likely due to a compromised system. This not only affects network performance but also points to deeper security issues, potentially involving malware. Immediate action is required to identify how the mining software was introduced, remove associated threats, and strengthen security measures to prevent further unauthorized activities.

* **Implications:**

The hacker’s primary goal was to exploit the client’s network resources for unauthorized cryptocurrency mining. By installing `cgminer`, a well-known mining tool, they leveraged the client’s computational power and network bandwidth to mine cryptocurrencies like Bitcoin. Cryptocurrency mining is incredibly resource-intensive, involving complex calculations to solve cryptographic puzzles and validate transactions on the blockchain. This process consumes a lot of bandwidth and processing power, which can severely degrade system performance and limit bandwidth availability for legitimate users.

The situation worsened due to a significant security flaw related to Basic Authentication. Basic Authentication, which relies on Base64 encoding, is inherently insecure because it only disguises the credentials rather than encrypting them. If someone intercepts these credentials, they can be easily decoded, potentially giving unauthorized access to sensitive systems. This vulnerability likely enabled the attacker to gain access, install the mining software, and operate it without being detected. The client’s lack of fundamental security practices, such as properly securing authentication mechanisms, has significantly contributed to the ease with which the attack was executed.

To further conceal their malicious activities, the hacker used Domain Generation Algorithms (DGAs). DGAs create numerous random, ever-changing domain names, which makes it difficult for traditional security systems to detect and block malicious traffic. The mining software relies on these constantly shifting domains to stay in touch with its command and control servers or mining pools. By continuously changing domain names, DGAs obscure the mining activity, making it tough for network administrators to identify and block the malicious traffic. This highlights a significant lapse in the client’s security awareness and readiness, as they failed to anticipate and counter such sophisticated evasion techniques.

In summary, the combination of DGAs and cryptocurrency mining software shows a well-planned and deliberate attack strategy. The DGAs hide the mining activities by frequently changing communication endpoints, while the mining software exploits the client’s resources for the attacker’s gain. This coordinated approach not only worsens the bandwidth issues but also reveals a serious security breach. The client’s poor security practices and lack of effective response to known threats have allowed this situation to escalate.

**Other Possible Incidents Considered and Why They Were Eliminated**

1. **Distributed Denial-of-Service (DDoS) Attack**

A Distributed Denial-of-Service (DDoS) attack floods a network or server with excessive traffic from multiple sources to cause disruptions, slowdowns, or even outages.

**Why It Was Eliminated:**

I quickly ruled out a DDoS attack once we dug into the details. Unlike DDoS attacks, which flood a system with a huge volume of generic traffic to cause disruptions, our analysis showed something quite different. We found `cgminer` and specific mining-related headers in HTTP POST requests, which clearly pointed to cryptocurrency mining activities. This was a much more precise clue compared to the broad, indiscriminate traffic of a typical DDoS attack. Furthermore, the traffic we observed was heavily encrypted and focused on ports tied to mining operations, not the chaotic traffic patterns of a DDoS. The presence of Domain Generation Algorithms (DGAs) reinforced this by indicating sophisticated mining operations, rather than a simple traffic overload. So, the evidence strongly indicated that cryptocurrency mining was the real issue here, not a DDoS attack.

1. **Internal Misuse**

Internal misuse involves legitimate users within the organization abusing their access privileges for unauthorized activities, such as running mining software on company devices**.**

**Why It Was Eliminated:**

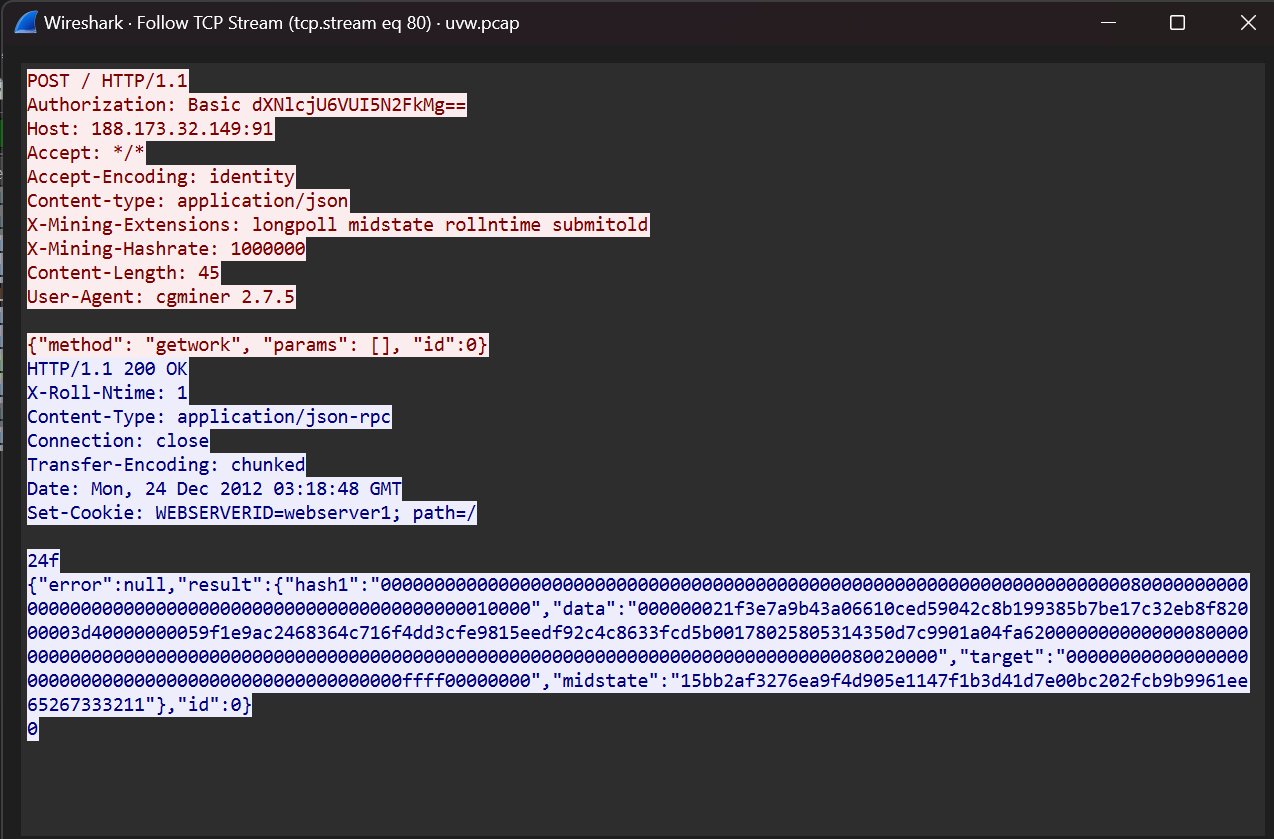
The analysis of the network traffic highlighted clear external indicators of malicious activity, specifically pointing to external IP addresses like 188.173.32.149 being involved in cryptocurrency mining. This focus on external IPs, along with the presence of mining software such as `cgminer`, strongly suggests an external attack rather than misuse by internal users. The traffic patterns observed were consistent with external attackers deploying mining software, and there was no direct evidence implicating internal users. The absence of any signs of internal involvement further confirms that the issue is related to an external attack rather than actions by employees.

**Risk Analysis of Threats to Company Assets**

1. **Unauthorized Use of System Resources**

Unauthorized cryptocurrency mining, often referred to as "cryptojacking," can severely impact system performance. When mining software like cgminer is detected on a device, it signifies that the system's CPU or GPU is being commandeered to mine cryptocurrency without the owner's consent. This hidden mining operation can cause significant slowdowns, elevate energy costs, and potentially damage the hardware over time.

Evidence: The presence of mining headers in HTTP POST requests and the detection of software such as cgminer are strong indicators that the system is being misused for unauthorized mining activities.



Real-Life Example: A notable case occurred in 2018 with Coinhive malware, which infected numerous websites to exploit visitors' CPUs for mining Monero cryptocurrency. This led to widespread slowdowns and performance issues on the affected sites,

ultimately diminishing user experience and increasing operational costs for the site operators.

Recommendation:

* Install Security Software: Use up-to-date antivirus and endpoint protection to block unauthorized apps.
* Monitor Performance: Track CPU and GPU usage for unusual spikes.
* Conduct Regular Audits: Frequently check for unauthorized software and vulnerabilities.

1. **Exposure of Sensitive Credentials**

Base64 encoding isn’t a secure method for protecting credentials because it can be easily decoded. When credentials are included in HTTP POST requests, they’re at risk of being intercepted and accessed by unauthorized parties. This issue was starkly illustrated by the Equifax breach, which exposed the severe impact of weak authentication and inadequate data protection practices.

Evidence: Base64-encoded credentials in HTTP POST requests are vulnerable since they can be decoded with minimal effort, allowing unauthorized access.

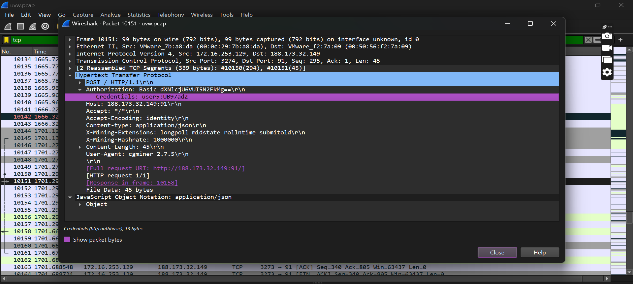
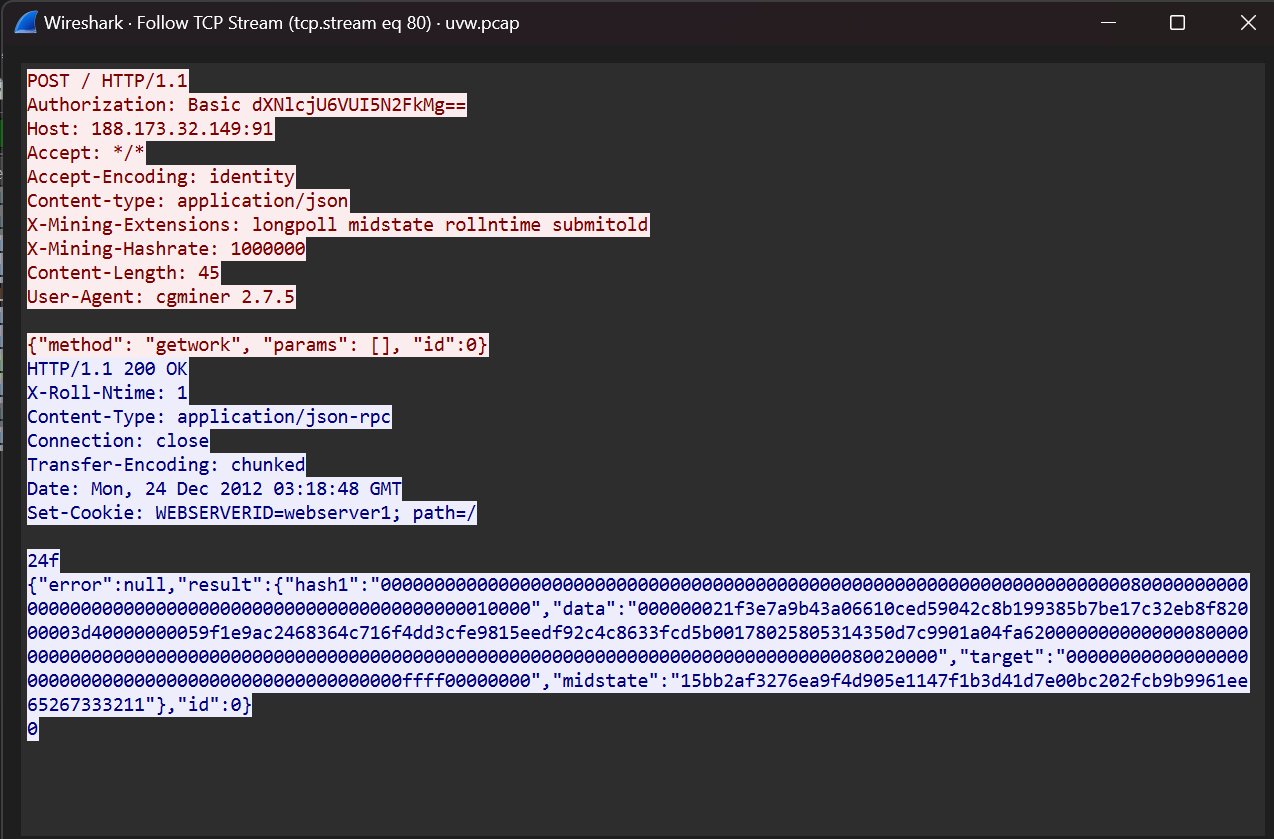


Figure 2: Figure 3: Base64-Encoded Credentials in HTTP POST Requests

Real-Life Example: The 2017 Equifax breach is a prime example where the personal information of around 147 million people was compromised due to insufficient protection and encryption of sensitive data. Attackers exploited these weaknesses to gain access to valuable information.

Recommendation:

* Use Stronger Authentication: Move from Basic Authentication to more secure methods like OAuth or multi-factor authentication (MFA).
* Encrypt TheData: Ensure all data in transit is encrypted with HTTPS to protect credentials from interception.
* Update Security Policies Regularly: Keep security policies and training current to emphasize the importance of secure authentication methods.

1. **Unauthorized Data Access and Exploitation**

Unauthorized data access and exploitation are major concerns when suspicious domains created by Domain Generation Algorithms (DGAs) are detected. DGAs are techniques used by malware to generate numerous domains that frequently change, making it tough to identify and block these malicious channels. Attackers leverage DGAs to maintain hidden communication channels with compromised systems, which enables them to exfiltrate data or issue commands while dodging traditional security measures.

Evidence: Domains such as `www.2mjgc3fuf26j7syf4pbf7xo.com` and `www.pp6ebzpv.com` appear to be generated by DGAs. These domains are frequently used by malware to establish covert communication channels that are challenging to detect and disrupt. This technique is a clear indication of efforts to evade traditional security measures and maintain unauthorized access.

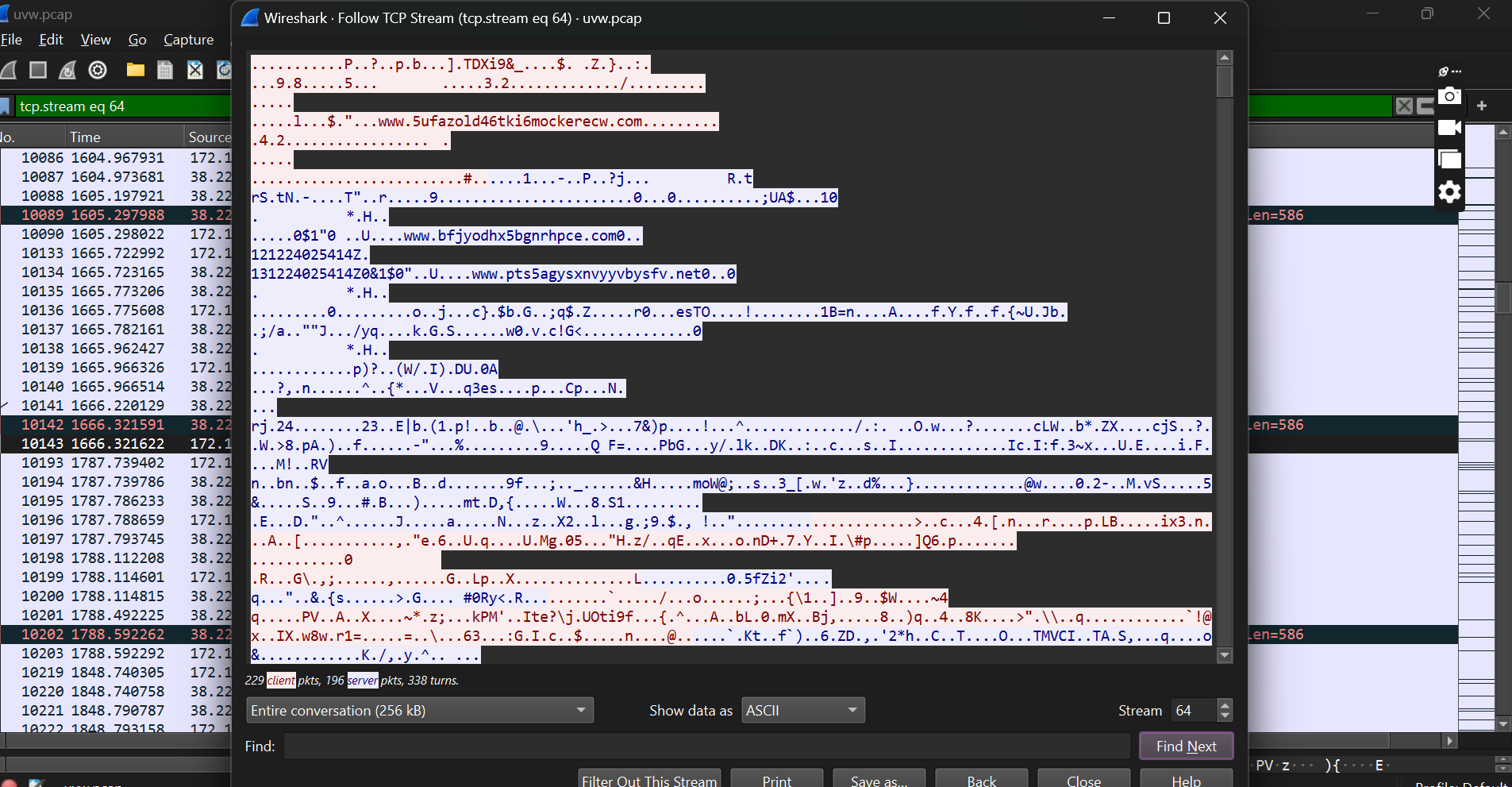
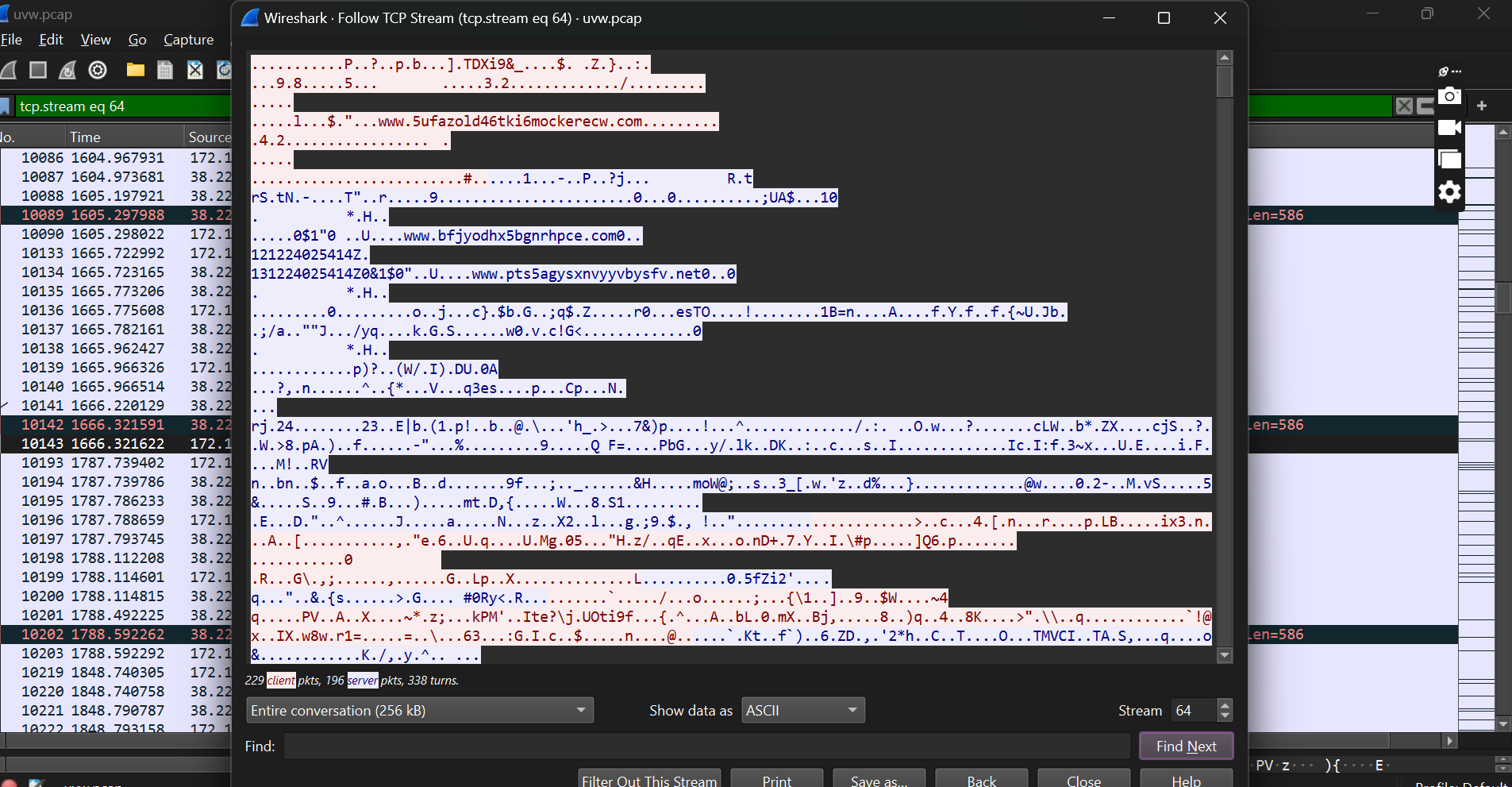


Figure 4 : Figure 5: Suspicious Domains Generated by DGAs for Unauthorized Access

Real-Life Example: The 2014 Sony Pictures breach is a notable example. Attackers used DGAs to create command-and-control channels, which allowed them to steal sensitive corporate data and disrupt operations. This breach led to the leak of confidential information, including unreleased films and private emails, causing significant financial and reputational damage to Sony.

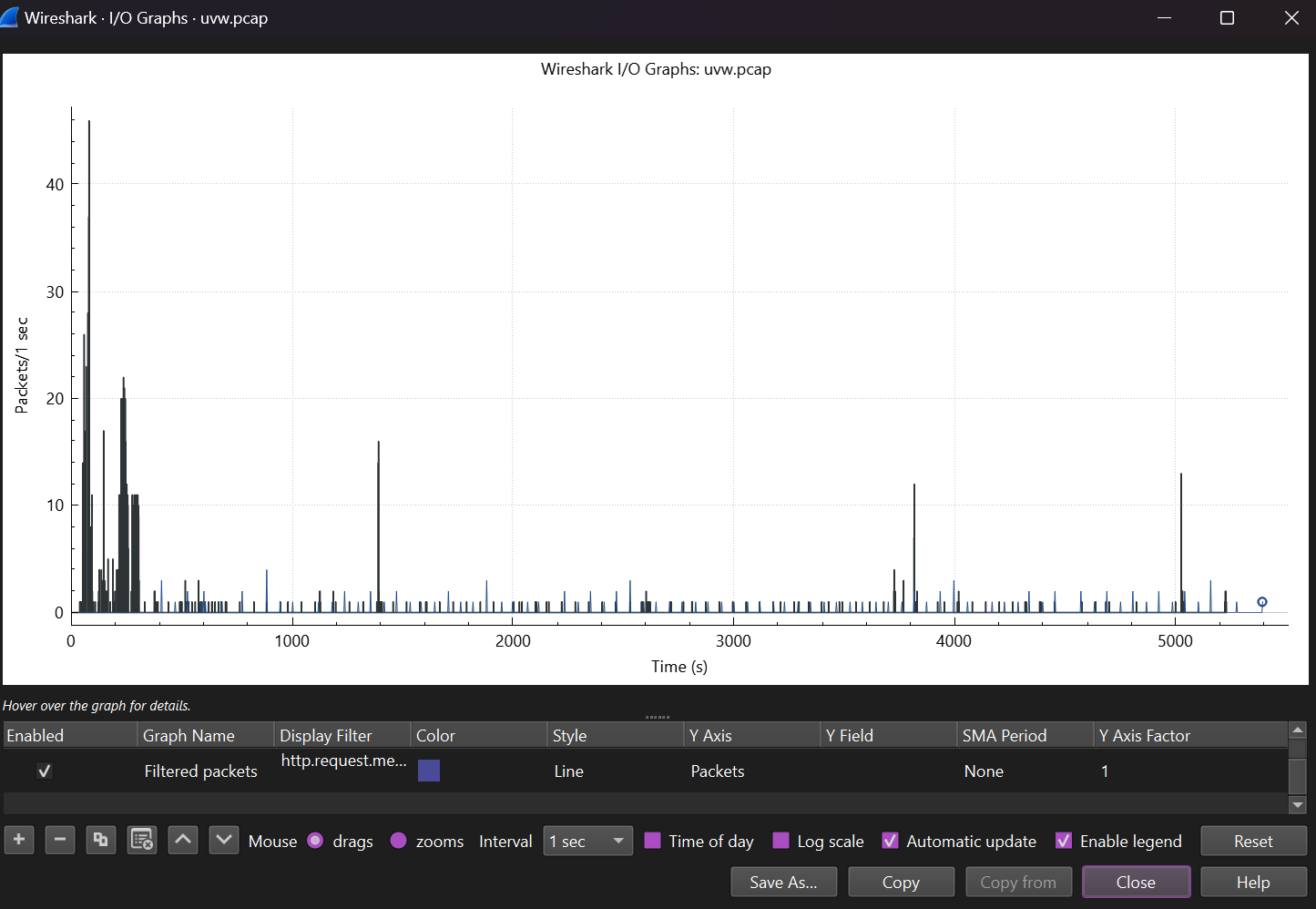
Recommendation:

* Use Advanced Threat Detection: Implement systems capable of identifying and blocking communication with suspicious or DGA-generated domains.
* Segment The Network: Divide the network into segments to restrict access to sensitive data and limit the impact of a breach. Apply the principle of least privilege to control access.
* Keep Everything Updated: Regularly update all software and systems to address vulnerabilities that could be exploited by attackers.

1. **Network Performance Degradation**

Unauthorized cryptocurrency mining can significantly degrade network performance and reliability. When mining activities generate high volumes of traffic, it strains the network, leading to slower speeds and reduced overall performance.

**Evidence:** Analysis of the captured traffic reveals a consistent pattern of high-volume HTTP POST requests linked to mining activities. This constant and substantial bandwidth consumption is a clear indicator of how mining operations can monopolize network resources, adversely affecting performance.



Real-Life Example: In 2018, a New Jersey school district experienced severe network slowdowns due to a cryptojacking attack, which used their network to mine cryptocurrency and caused significant performance issues.

Recommendation:

* Use Traffic Management Tools: Implement tools to prioritize critical applications and manage bandwidth effectively, even with high mining traffic.
* Monitor Network Traffic Regularly: Continuously check for unusual traffic patterns or spikes to address issues before they severely impact performance.
* Educate Users: Train users to recognize and avoid potential security threats, such as cryptojacking, to minimize the risk of performance-degrading malware.

**Asset list**

|  |  |  |  |
| --- | --- | --- | --- |
| **Ip Address** | **Machine Name** | **MAC Address** | **OS** |
| 172.16.253.129 | DellXT | 00:0c:29:7b:a8:da | Windows XP |
| 172.16.253.254 | Not given | 00:50:56:f6:be:3c | MSFT 5.0  (likely Windows Server 2000/2003) |

**SUMMARY:**

This report presents a detailed analysis of network traffic captured from a client site using Wireshark. The client has reported bandwidth issues, and given their history of poor security practices, there is a high likelihood of malicious activity.

Upon analyzing the capture file `uvw.pcap`, I identified a significant issue: Unauthorized Cryptocurrency Mining. It appears that an attacker has been covertly using the network’s resources to mine Bitcoin without the owner's consent.

The traffic analysis uncovered several concerning patterns. I observed unusually high volumes of TCP and TLS traffic, substantial data transfers to specific IP addresses, and unusual port usage. These anomalies, coupled with evidence of Domain Generation Algorithms (DGAs) and mining software, suggest a sophisticated and unauthorized mining operation.

The implications of this activity are severe. It is placing a substantial strain on system resources, increasing energy costs, and compromising overall network security. The attacker exploited vulnerabilities such as Basic Authentication with Base64 encoding, which is relatively easy to compromise. Additionally, the use of DGAs allowed the attacker to frequently change domain names, thereby evading detection.

Other potential issues were considered, such as a Distributed Denial-of-Service (DDoS) attack but were ruled out due to the specific nature of the traffic. Similarly, internal misuse was dismissed, as the suspicious activity was linked to external IP addresses and mining software.

The risk analysis highlights the significant problems caused by this unauthorized mining: resource depletion, exposure of sensitive credentials, unauthorized data access, and degradation of network performance. To address these issues, I recommend strengthening authentication methods, enhancing threat detection capabilities, employing traffic management tools, and regularly monitoring network activity.

The report concludes with a comprehensive asset list, providing a clear overview of the affected devices and their characteristics. This detailed analysis, along with my actionable recommendations, aims to bolster network security and prevent similar incidents in the future.

**Glossary**

* **cgminer: An open-source cryptocurrency mining software used for mining digital currencies like Bitcoin. It can operate on various hardware setups and requires significant computational power.**
* **Hashrate: A measure of the computational power used in cryptocurrency mining. It indicates how many hashes (cryptographic calculations) can be performed per second.**
* **X-Mining-Extensions Header: A custom HTTP header used in mining software to specify additional features or capabilities that enhance mining performance.**
* **X-Mining-Hashrate Header: A custom HTTP header used in mining software to report the current hashrate, or computational power, of the mining operation.**
* **JSON (JavaScript Object Notation): A lightweight data interchange format that is easy for humans to read and write and easy for machines to parse and generate. Often used in web applications for transmitting data.**
* **Mining Pool: A group of cryptocurrency miners who combine their computational resources to increase their chances of solving cryptographic puzzles and earning rewards.**
* **Malware: Malicious software designed to damage, disrupt, or gain unauthorized access to computer systems. The presence of mining software can be an indicator of potential malware.**
* **HTTP POST Request: An HTTP request method used to send data to a server, typically involving sensitive information like login credentials or form submissions. The data is included in the body of the request.**
* **Base64 Encoding: A method for encoding binary data into ASCII text, commonly used for data transfer. It is not a security measure, as encoded data can be easily decoded.**

**References**

* **“The Evolution of Cryptojacking: From Coinhive to a Cybersecurity Plague”**, V. Constantinescu, Bitdefender, Dec. 29, 2023. [Online]. Available: <https://www.bitdefender.com/blog/hotforsecurity/the-evolution-of-cryptojacking-from-coinhive-to-a-cybersecurity-plague/>. [Accessed: Jul. 23, 2024].
* **"2017 Equifax Data Breach,"** *Wikipedia*, Jun. 29, 2024. [Online]. Available: <https://en.wikipedia.org/wiki/2017_Equifax_data_breach#:~:text=Private%20records%20of%20147.9%20million,the%20public%20until%20September%202017>. [Accessed: Jul. 23, 2024].
* **"2014 Sony Pictures Hack,"** *Wikipedia*, Jul. 11, 2024. [Online]. Available: <https://en.wikipedia.org/wiki/2014_Sony_Pictures_hack>. [Accessed: Jul. 23, 2024].
* "Domain generation algorithm," Wikimedia, 22 July 2023. [Online]. Available: <https://en.wikipedia.org/wiki/Domain_generation_algorithm>. [Accessed: 23 July 2024].
* D. Smallwood, "Hacker Pig Latin: A Base64 Primer for Security Analysts," Dark Reading, Jan. 21, 2021. [Online]. Available: <https://www.darkreading.com/cybersecurity-analytics/hacker-pig-latin-a-base64-primer-for-security-analysts>. [Accessed: Jul. 23, 2024].
* "CGMiner," BitcoinWiki, Dec. 21, 2023. [Online]. Available: <https://bitcoinwiki.org/wiki/cgminer#:~:text=CGMiner%20is%20the%20most%20popular,you%20can%20discover%20in%20BitcoinWiki>. [Accessed: Jul. 23, 2024].
* ChatGPT conversation regarding cryptojacking and security practices," OpenAI, Jul. 23, 2024.
* "PCAP Analysis Laboratory," Dynamite AI, [Online]. Available: <https://lab.dynamite.ai/pcaps>.
* "Cryptojacking," Wikipedia, The Free Encyclopedia, 5 July 2024. [Online]. Available: <https://en.wikipedia.org/wiki/Cryptojacking>. [Accessed: 26 July 2024]