# **SSCP – System Security Certified Practitioner Labs**

## Lab 4-01: Audit an Organization's Security for Phishing Attacks

### Case Study

CyberShield Marketing Solutions, a leading social media marketing firm, recognized the escalating threat of phishing attacks targeting employees engaged on platforms such as Facebook and YouTube. With the rise in phishing incidents leading to financial fraud, the company sought a proactive approach to assess and mitigate the risks associated with social engineering attacks. To enhance cybersecurity awareness and resilience, CyberShield conducted a comprehensive audit of phishing threats within its organization. By leveraging ethical hacking techniques and phishing simulation tools, the company identified vulnerabilities, trained employees, and reinforced its security framework. This initiative successfully strengthened the firm's defense against phishing attacks, ensuring a safer digital environment for its workforce.

### Business Challenge

CyberShield Marketing Solutions faced a critical challenge in protecting its employees from phishing attacks, which were increasingly used to execute financial fraud on social media platforms. Given the nature of their work, employees regularly interacted with links, messages, and attachments, making them prime targets for social engineering tactics. The company needed to assess the risk of different phishing attacks, estimate potential losses, and implement a structured awareness program to prevent employees from falling victim to scams. Additionally, they required a reliable mechanism to audit phishing vulnerabilities and ensure compliance with industry security best practices.

### Solution

To address these challenges, CyberShield Marketing Solutions hired a System Security Certified to conduct a security audit and identify vulnerabilities within its network. The firm implemented OhPhish, a web-based phishing simulation tool designed to evaluate employees' susceptibility to phishing attacks. Through controlled phishing campaigns, the organization assessed employee responses to deceptive emails and fraudulent links. The platform collected real-time data, providing Management Information System (MIS) reports and actionable insights based on user roles, departments, and designations. By analyzing these trends, the company developed targeted security awareness training programs to educate employees on recognizing and mitigating phishing threats. This proactive approach significantly reduced the organization's exposure to phishing attacks, reinforcing its cybersecurity posture and fostering a culture of vigilance against social engineering tactics.

|  |
| --- |
| **Procedure** |
| 1. Before starting this lab, you must activate your **OhPhish** account. Go to the **Click here** hyperlink in the OhPhish notification above the **My Courses** section.    2. You will be redirected to the OhPhish **Sign Up** page. Enter the remaining personal details, check the **I'm not a robot** checkbox, and click the **Complete Signup** button.    3. Open your email account during the registration process. Open an email from **OhPhish**, and click the CLICK HERE TO LOGIN button in the email.    4. The **OhPhish** login page will appear. Log in using the credentials received in the email.    5. You will be redirected to the **Reset Password** page. Enter the new password in both fields and click the **Reset Password** button to reset the password.    6. Once you log in to your OhPhish account, you will be redirected to the OhPhish **Dashboard**. Then, click on the **Entice to Click** option.    7. The **Create New Email Phishing Campaign** form appears. Enter any name in the Campaign Name field (here, **Test - Entice to Click**). In the **Select Template Category** field, select **Coronavirus/COVID-19** from the drop-down list. In the **Select Country** field, leave the default option selected (**All**). In the **Select Template** field, click the **Select Template** button and select **Corona Virus Advisory** from the drop-down list.    8. Leave fields such as **Sender Email**, **Sender Name**, **Subject**, **Select Time Zone**, **Expiry Date**, and **Schedule Later** set to their default values, as shown in the screenshot. In the **Import users** field, click **Select Source**.    9. **Import Users** pop-up appears. Click to select the **Quick Add** option from the list of options.    10. The **Import Users Info** pop-up appears; enter the details of the employee and click **Add**.    11. Similarly, you can add the details of multiple users. Here, we added two users. Add the users' details and click **Import**.    12. In the **Batch Count** and **Batch Interval** fields, set the values to **1**. Leave the **Landing Page** field set to its default value. Scroll down to the end of the page and click **Create** to create the phishing campaign.    13. **Add to your Whitelist** pop-up appears; click **Done**.    14. The confirmation pop-up appears; click **SURE**.    15. A countdown timer appears, and the phishing campaign initiates in ten seconds. The **Alert!** pop-up appears, indicating successful initiation of a phishing campaign; click **OK**.    16. Open the phishing email on the victim's PC. In this case, we use **Windows Server 2019** as a victim.    17. Click on [Ctrl+Alt+Delete](https://labclient.labondemand.com/Instructions/11b98422-7c52-4514-97c5-fcb8d6f711e8?rc=10) to activate it. By default, the **Administrator** profile is selected; enter the password into the machine and press **Enter** to log in.    18. Open any web browser (**Mozilla Firefox**) and then open the email client provided while creating the phishing campaign (here, **Gmail**). After you log in to your **Gmail** account, search for an email with the subject **COVID-19 Advisory** in the **Inbox**. Click on the **Safety Measures** link in the email.    19. If a **Suspicious link** pop-up appears, click **Proceed**. The landing page **Oh You've been Phished** appears, as shown in the screenshot.    20. Go back to the **Windows 10** machine. Click on the **Test – Entice to Click** campaign present on the **OhPhish Dashboard**.    21. The **Campaign Detailed Report** page displays the Campaign Details and **Summary** sections. In the **Campaign Summary** section, you can observe that the values of **No. of targets who have clicked the link (defaulters)** and **No. of Targets who have opened the mail** are both **1** (here, we have opened only one email account).    22. Click **Home** in the left pane to navigate to the OhPhish **Dashboard**. In the OhPhish **Dashboard**, click on the **Send Attachment** option.    23. The **Create New Email Phishing Campaign** form appears. Enter any name in the Campaign Name field (here, **Test – Send to Attachment**). In the **Select Template Category** field, select **Office Mailers** from the drop-down list. In the **Select Country** field, leave the default option selected (**All**). In the **Select Template** field, select the **PF Amount Credited** option from the drop-down list and then click the **Select** button. Leave fields such as **Sender Email**, **Sender Name**, **Subject**, **Select Time Zone**, **Expiry Date**, and **Schedule Later** set to their default values, as shown in the screenshot. In the **Attachment** field, enter any name (here, **Additional Information**).    24. Click the **Select Source** button under the **Import users** field.    25. **Import Users** pop-up appears. Click to select the **Quick Add** option from the list of options.    26. The **Import Users Info** pop-up appears; enter the details of the employee and click **Add**.    27. Similarly, you can add the details of multiple users. Here, we added two users. Add the users' details and click **Import**. In the **Batch Count** and **Batch Interval** fields, set the values to **1**. Leave the **Landing Page** field set to its default value. Scroll down to the end of the page and click **Create** to create the phishing campaign.    28. **Add to your Whitelist** pop-up appears; click **Done**. The **confirmation** pop-up appears; click **SURE**. A countdown timer appears, and a phishing campaign initiates in ten seconds. The **Alert!** pop-up appears, indicating successful initiation of a phishing campaign; click **OK**.    29. Go back to the **Windows Server 2019** victim machine. In the **Gmail** account opened previously, navigate to the **Inbox** folder. You will find an email from **HR – ABP News**, as shown in the screenshot. Click on the **EPF – KYC Documents Upload Centre** hyperlink present in the email.    30. If a **Suspicious** link pop-up appears, click **Proceed**. You will be redirected to the **Oh You've been Phished** landing page, as shown in the screenshot.    31. Go back to the **Windows 10** machine. Click on the **Test – Send to Attachment** campaign present on the **OhPhish Dashboard**.    32. The **Campaign Detailed Report** page displays the Campaign Details and **Summary** sections. In the **Campaign Summary** section, you can observe that the value of **No. of targets who have clicked the link (defaulters)** is **1**. Click on the **1** icon to see the defaulter.    33. The **Campaigns Users** page appears, displaying the details of the defaulter, such as **Risk Score**, **Credentials**, **IP Address**, **Location**, etc., as shown in the screenshot.    34. Click to expand the **Reports** section in the left pane and select the **Executive Summary Report** option.    35. The **Campaign Report** page appears; select any phishing campaign from the drop-down list (here, **Test – Send to Attachment**) and click on the **Export** icon to export the report.    36. The **Opening Phishing-Simulation-Test** window appears; select the **Save File** radio button and click **OK**.    37. The file is downloaded to the default location (here, **Downloads**). Navigate to the download location and double-click the **Phishing-Simulation-Test---Send-Attachment** file to open it.    38. The executive phishing report appears in the document, as shown in the screenshot. |

## Lab 4-02: Identify XSS Vulnerabilities In Web Applications

### Case Study

SecureCam Solutions, a renowned security testing company, was tasked with assessing the security of a client's web application against Cross-Site Scripting (XSS) vulnerabilities. With the increasing exploitation of XSS attacks to steal user credentials, inject malicious scripts, and compromise web applications, the firm recognized the need for a thorough penetration test. By leveraging a combination of manual testing and automated tools, SecureCam Solutions identified vulnerabilities within the application, categorized their severity, and provided actionable remediation strategies. This proactive security assessment helped the client strengthen its application defenses, ensuring enhanced protection against XSS-based threats.

### Business Challenge

The client’s web application was at risk of XSS attacks, which could lead to severe consequences such as data breaches, session hijacking, and unauthorized script execution. Given the widespread use of dynamic content and user-input fields, detecting and mitigating XSS vulnerabilities was a critical security concern.

### Solution

To address these challenges, SecureCam Solutions assigned a penetration tester to perform a comprehensive XSS vulnerability assessment of the web application. The security expert conducted manual testing by injecting custom JavaScript payloads into input fields to evaluate the application’s response. In addition, the penetration tester utilized XSpear, an advanced automated tool, to accelerate the testing process and generate detailed reports on discovered vulnerabilities.

|  |
| --- |
| **1. Using Manual Payloads**  Go to the website <http://testphp.vulnweb.com/>. It is a vulnerable website for pentesting purposes.  Search anything in the search bar. If it reflects on the page, it would mean a **Reflected XSS** vulnerability is present.    Navigate to the **signup** tab on the left and enter the credentials to hunt the XSS vulnerability. You can try this on all the input fields on the website, such as feedback forms, chat boxes, comment boxes, or any other input field.    Now, enter some information in the fields and click the **update** button. It shows the name on the top, which means this input field is also vulnerable.    Inject JavaScript code in the vulnerable input field to execute that code. Here is a simple payload used that is **<script> alert(1) </script>.**    Click **update** to run the code. It executes the **alert function**.    You can try any other payload, such as **<script> alert(document.cookie) </script>,** to retrieve the cookies.    The code is executed and shows the login value or session ID.    **2. Using XSpear Tool**  It is an automated XSS vulnerability detection tool; Install it using the **gem install XSpear** command.    Explore the options using the **XSpear -h** command.    Use the **XSpear -u "example.com/search.php?parameter=value" -d "searchFor=PostData"** command. Post Data is the value you put in the input field; it can be any value.    It shows the critical levels (Low, Medium, High) of the vulnerabilities, payloads, and their descriptions.    You can also see the **High-Risk** vulnerabilities.    You can use the payloads it has given to exploit the website. |

## Lab 5-01: Simulate Security Incidents in a Controlled Lab Environment Using HTTP RAT Trojan

### Case Study

CyberSec Labs, a cybersecurity research and training organization, aimed to enhance its ability to detect, analyze, and mitigate real-world cyber threats. As part of its security incident simulation, the team set up a controlled lab environment to test the effectiveness of Remote Access Trojans (RATs). By deploying an HTTP RAT Trojan, the team established a remote connection between a Windows 7 machine (10.10.50.202) as the attacker system and a Windows Server 2016 machine (10.10.50.211) as the target system. This exercise provided valuable insights into the behavior of malicious payloads, the attack lifecycle, and potential detection and response strategies. The simulation helped CyberSec Labs refine its cybersecurity training programs and develop advanced defensive techniques.

### Business Challenge

Organizations face a growing threat from Remote Access Trojans (RATs), which cybercriminals use to gain unauthorized access, exfiltrate sensitive data, and execute malicious commands on compromised systems. To strengthen incident response capabilities, CyberSec Labs needed to simulate real-world cyber threats in a controlled lab environment without risking actual systems. This involved understanding the infection mechanism of HTTP RAT Trojans, including payload execution and remote command control. Additionally, the organization aimed to develop and test defense strategies to detect, contain, and mitigate Trojan-based attacks. Furthermore, CyberSec Labs sought to train cybersecurity professionals in handling Advanced Persistent Threats (APTs) and remote exploitation techniques, ensuring they could effectively respond to evolving cyber threats.

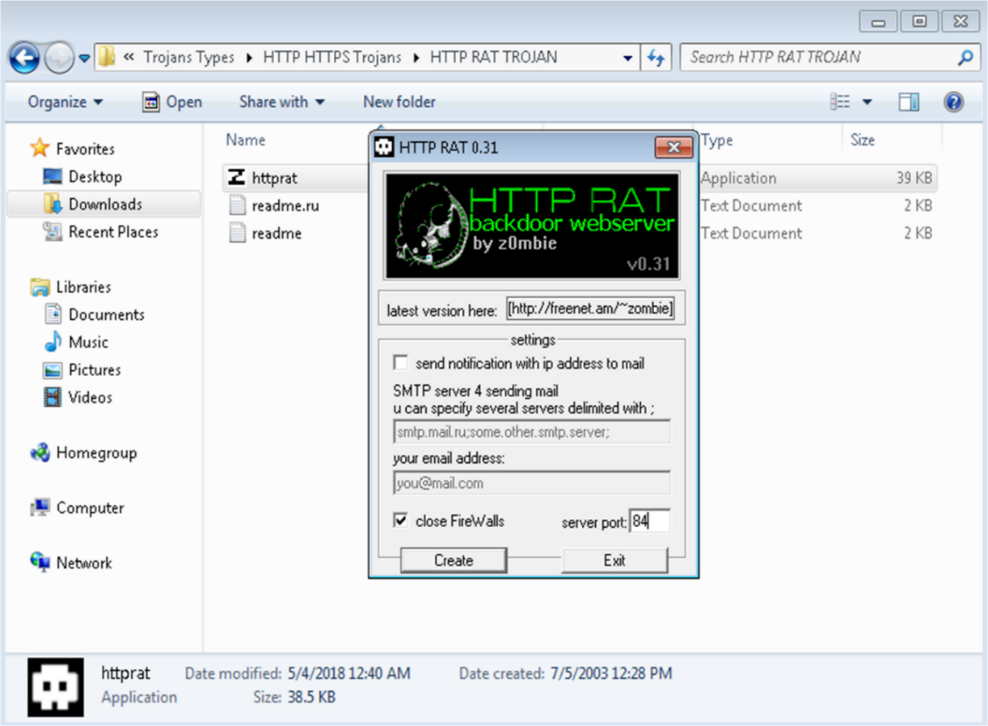
### Solution

To achieve these objectives, CyberSec Labs conducted a controlled simulation using an HTTP RAT Trojan in a secure test environment. The penetration testing team configured the attack machine (Windows 7, IP: 10.10.50.202) to act as the RAT server and created and deployed the HTTP RAT payload on the target machine (Windows Server 2016, IP: 10.10.50.211). Once the Trojan file was executed, remote access to the compromised server was established, allowing the team to monitor the attack lifecycle, including command execution, system control, and data exfiltration techniques. To strengthen detection and response strategies, the team analyzed various detection methods, such as antivirus solutions, endpoint security measures, and network monitoring tools. Additionally, they developed remediation strategies, including threat hunting, RAT removal, and incident response best practices. Through this simulation, CyberSec Labs successfully enhanced its understanding of Remote Access Trojan (RAT) attacks, improved its security training framework, and reinforced its defensive capabilities against real-world cyber threats.

**Configuration and Procedure:**

Use a Windows 7 machine to run the HTTP RAT Trojan.

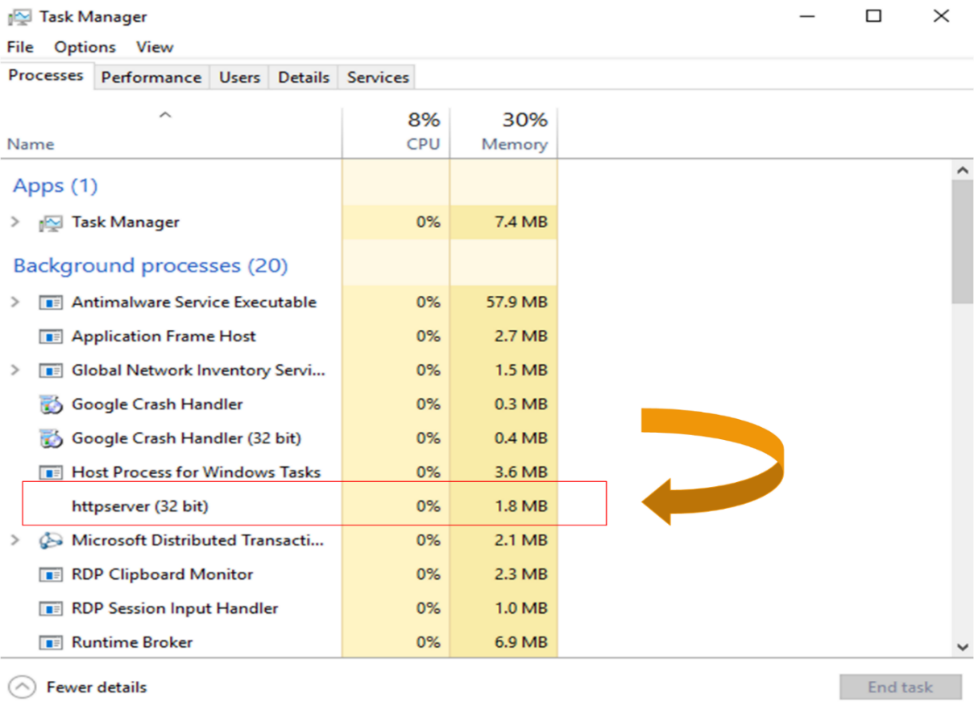
1. "send a notification with IP address to mail" should be unchecked.
2. Configure Port.
3. Click **Create**.



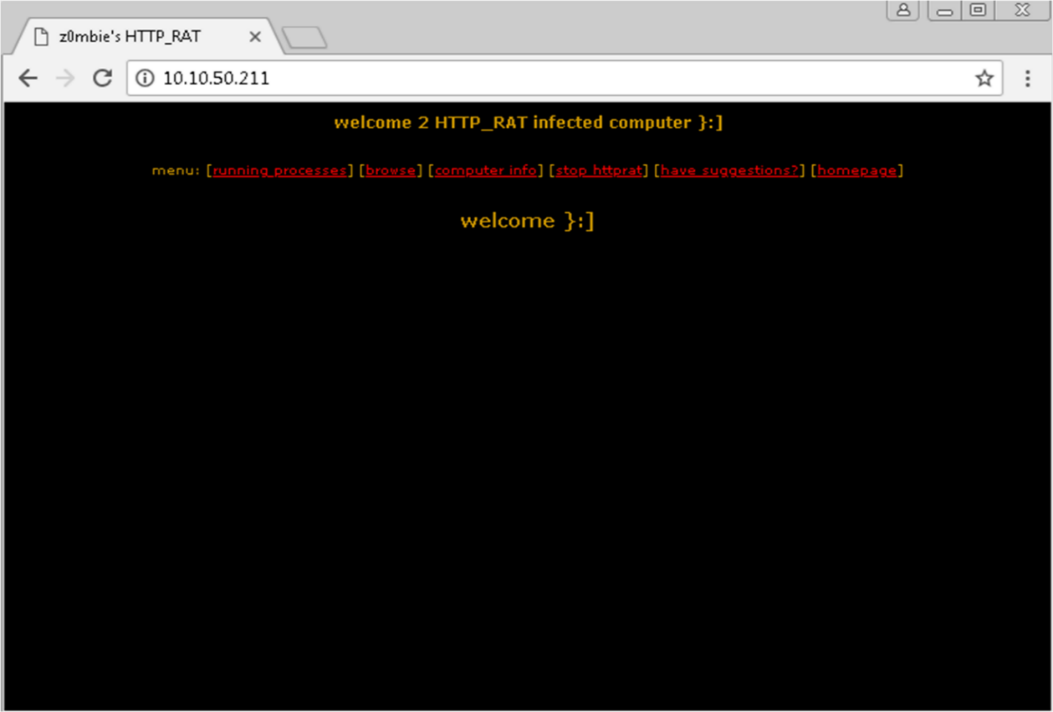
You will see a new executable file in the default directory where the application is installed. Transfer this file to the victim's machine.



1. Run the file after logging in to the victim's machine (in our case, Windows Server 2016).
2. Check the task manager to see if a process is running; An HTTP Server task will appear to be in progress.

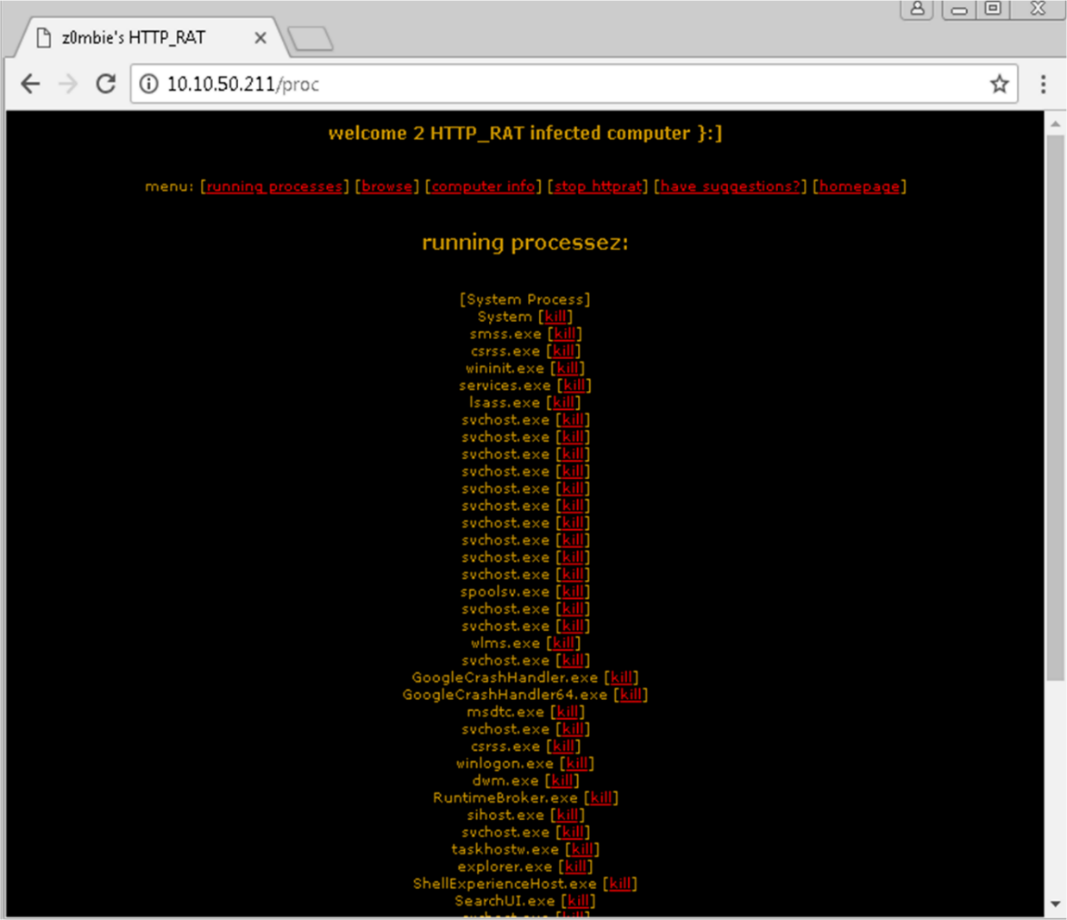


1. Go back to Windows 7.
2. Open a Web browser.
3. In our case, go to the victim's computer's IP address: 10.10.50.211.



From the victim's computer, the HTTP connection is active. You can look at processes that are running and browse drives. Using this tool, you can also check the victim's computer information.

1. Click **Running Processes**.



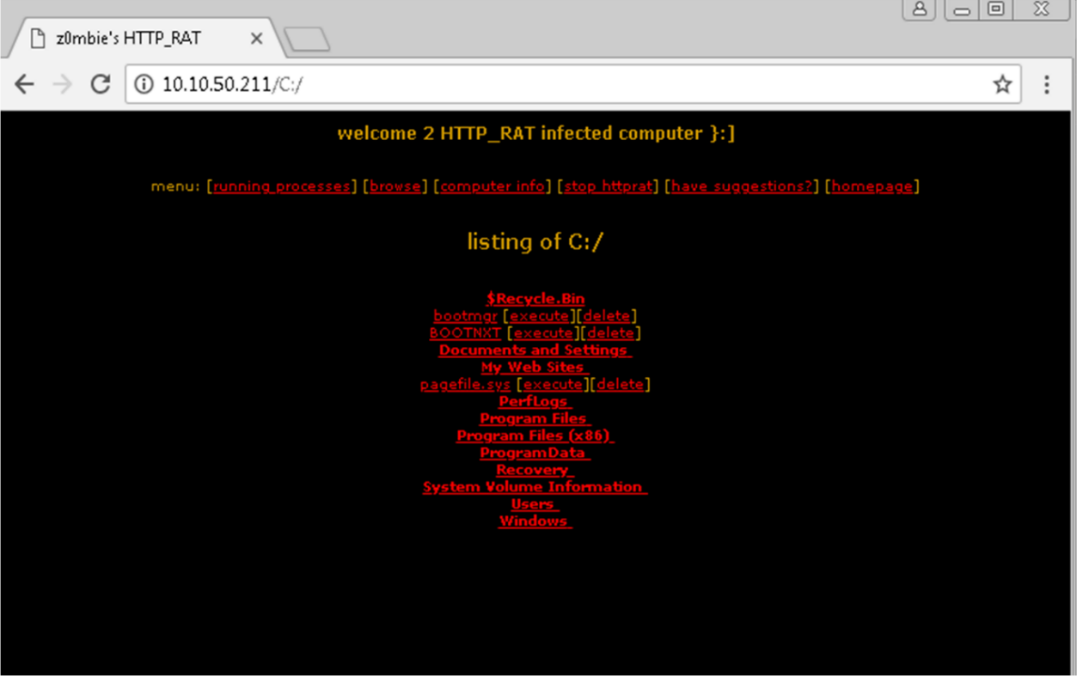
The above output shows the "running process" of the victim's machine.

1. Click **Browse**.



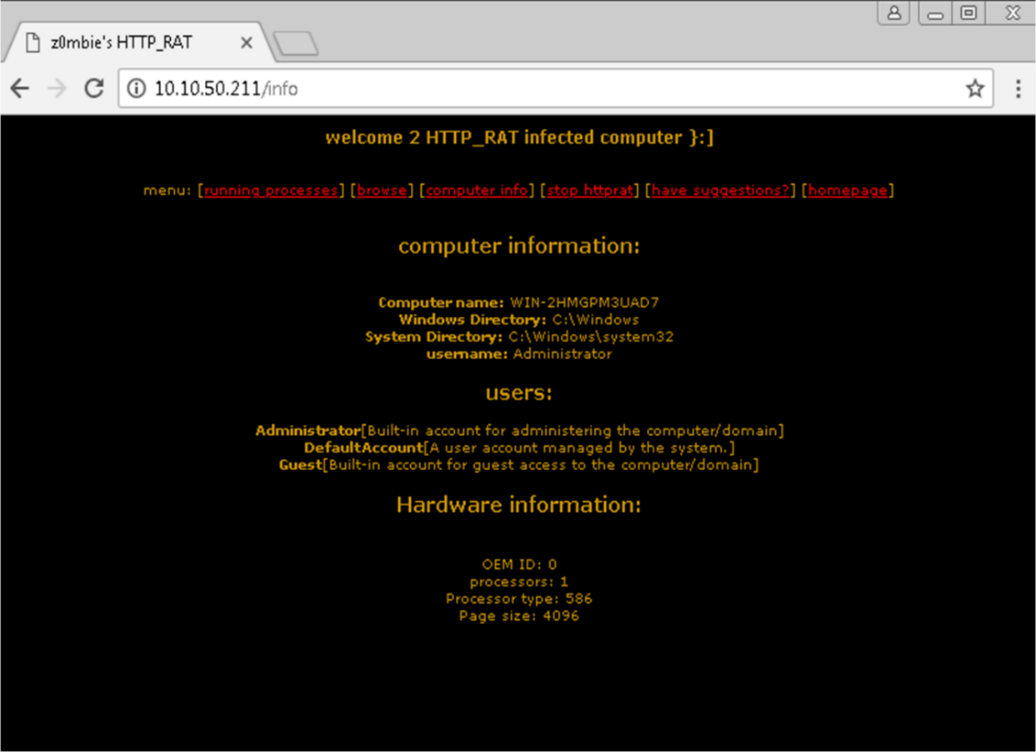
The output shows drives.

1. Click **Drive C**.



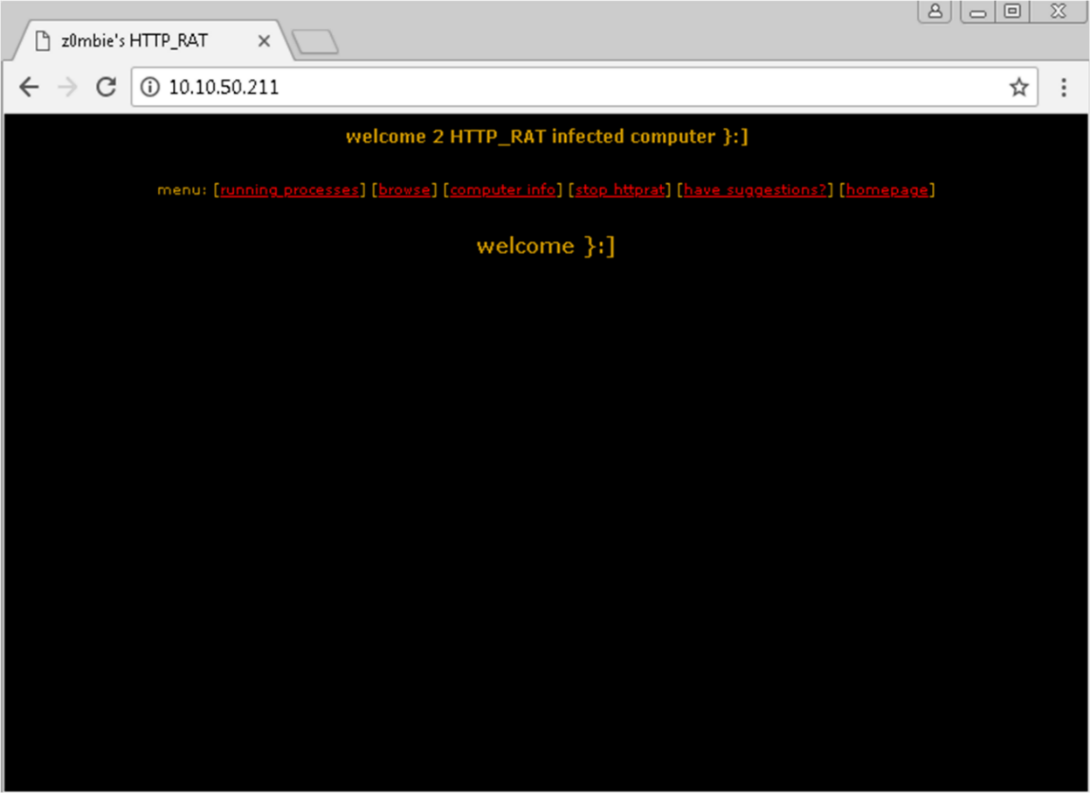
Output showing C drive.

1. Click **Computer Information**.

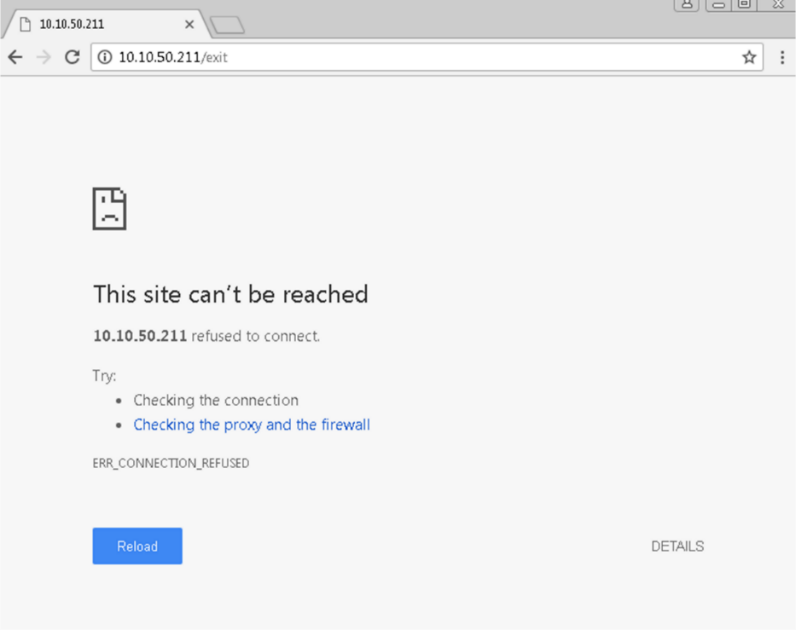


The output shows computer information.

1. To terminate the connection, click **Stop\_httpRat**.

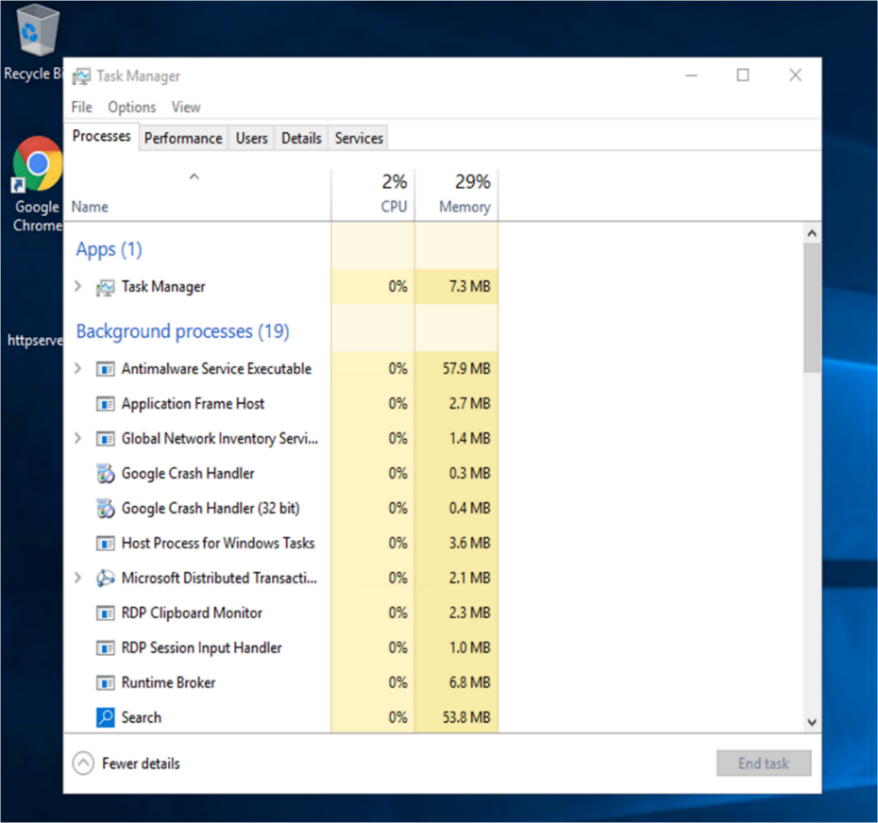


1. Refresh the browser.



The connection is successfully terminated.

1. Open Windows Server 2016 and check the running processes.



The HTTP server process is terminated.

## Lab 5-02: Monitoring a TCP/IP Connection Using CurrPort Tool

### Case Study

To strengthen network security and enhance incident response capabilities, SecureMac Solutions conducted a simulation to monitor and analyze TCP/IP connections using the CurrPort tool. By re-executing the HTTP RAT Trojan on a Windows Server machine (10.10.50.211), the team aimed to detect unauthorized remote access and terminate the malicious connection. This exercise provided critical insights into identifying suspicious network activity, analyzing active TCP/IP connections, and applying effective countermeasures to mitigate threats. By leveraging CurrPort, a lightweight network monitoring tool, SecureMac Solutions successfully demonstrated how security professionals can track and eliminate unauthorized network access in real-time.

### Business Challenge

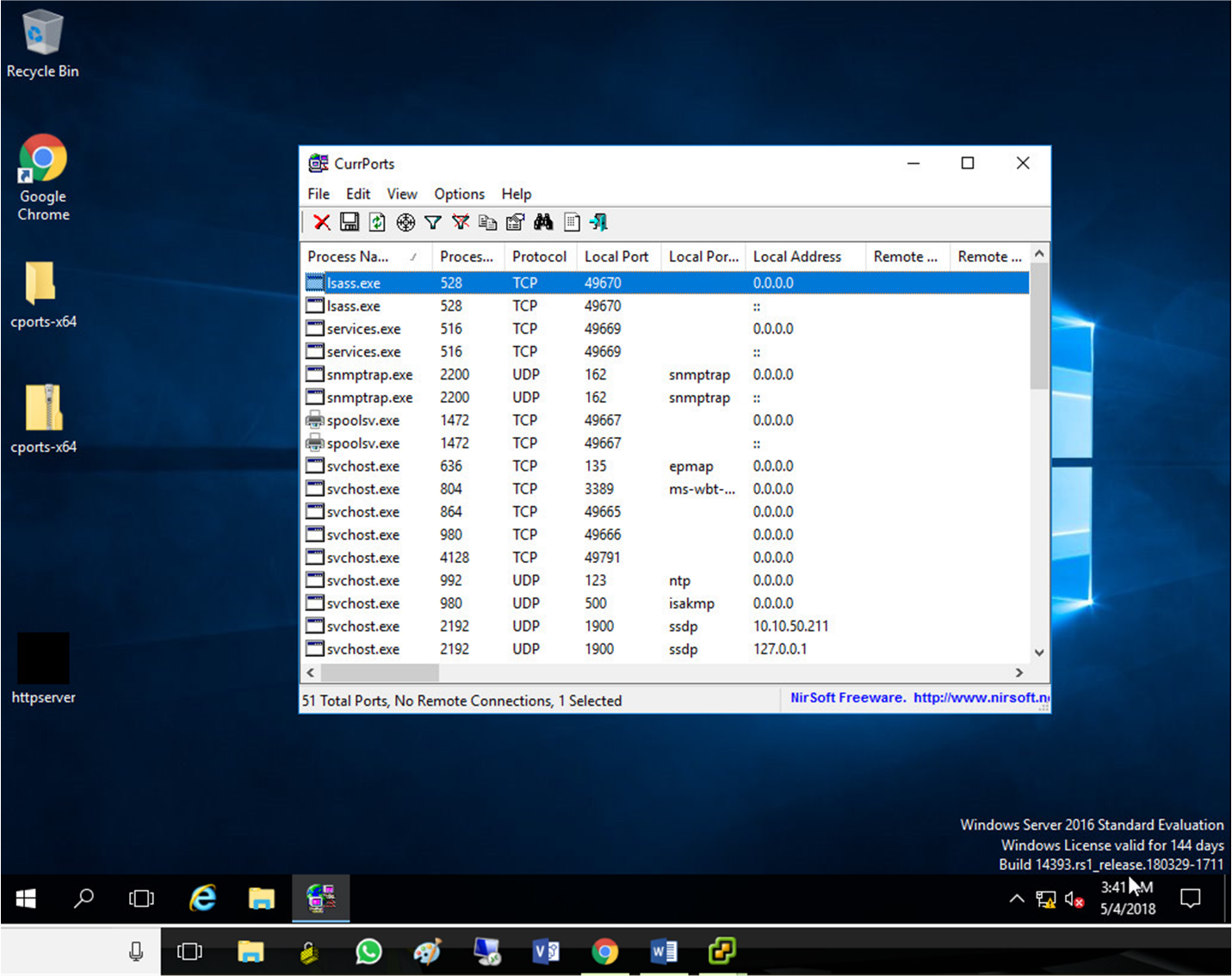
Organizations face an increasing risk of malicious remote access attempts, often facilitated through Remote Access Trojans (RATs) and other stealthy malware. Cybercriminals exploit these threats to establish covert connections, exfiltrate sensitive data, and execute malicious commands on compromised systems. To enhance network security monitoring, SecureMac Solutions aimed to detect unauthorized TCP/IP connections established by malware, analyze connection details such as source and destination IP addresses, ports, and processes, and terminate suspicious network activity to prevent further exploitation. Additionally, the organization sought to train security professionals in proactive network monitoring and incident response techniques, ensuring they could effectively mitigate evolving cyber threats.

### Solution

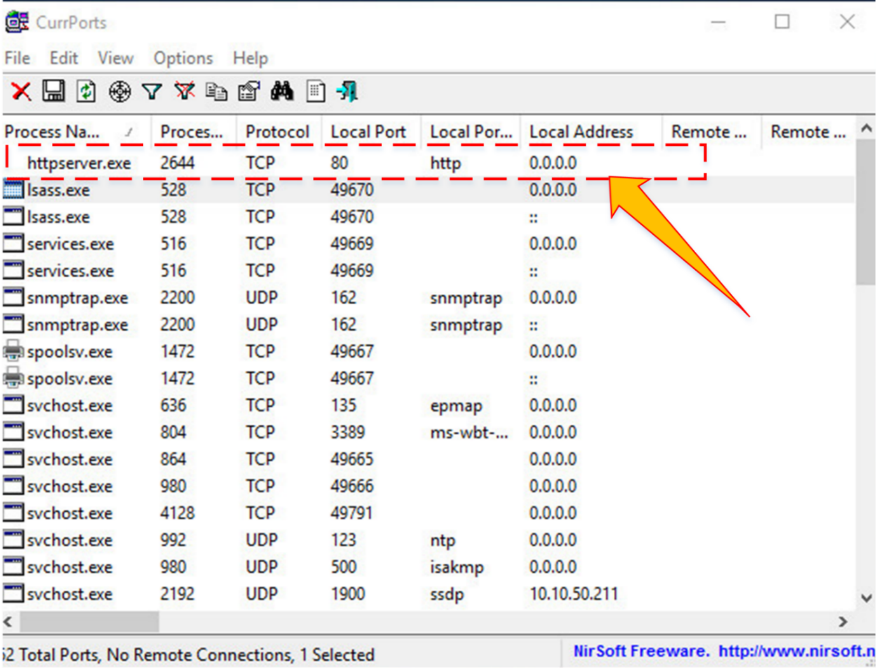
To address these challenges, SecureMac Solutions executed a structured approach using the CurrPort tool to monitor active TCP/IP connections. The team first re-executed the HTTP RAT Trojan on the Windows Server machine (10.10.50.211) to simulate an attack scenario. Using CurrPort, they examined real-time network connections, identifying the malicious HTTP RAT process based on its source IP, destination IP, and port number. Once detected, the team manually terminated the connection and ensured that no residual malicious activity remained. This exercise enhanced the organization's ability to identify, track, and eliminate unauthorized network connections, reinforcing its overall cybersecurity posture and incident response capabilities against evolving threats.

**Configuration:**

1. On Windows Server 2016, launch the application Currports and monitor the processes.



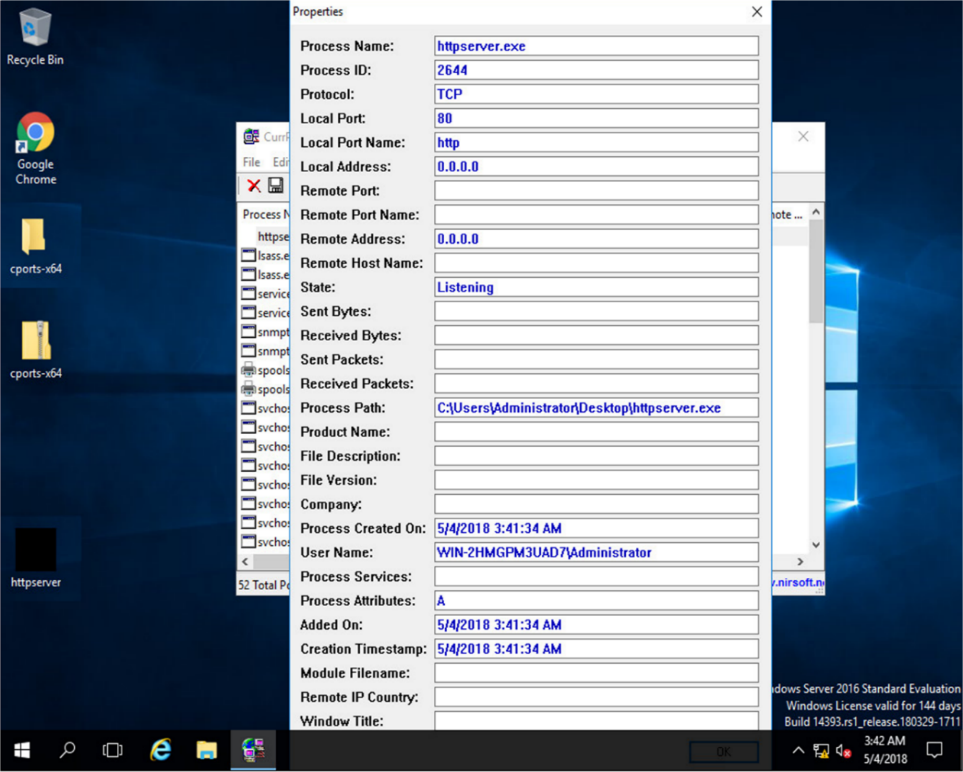
1. Execute the HTTP Trojan that was developed in the previous lab.



A new process is added.

The IP address, the local and remote ports, the process name, and the protocol can all be observed.

1. For more details, right-click **httpserver.exe** and go to **Properties**.



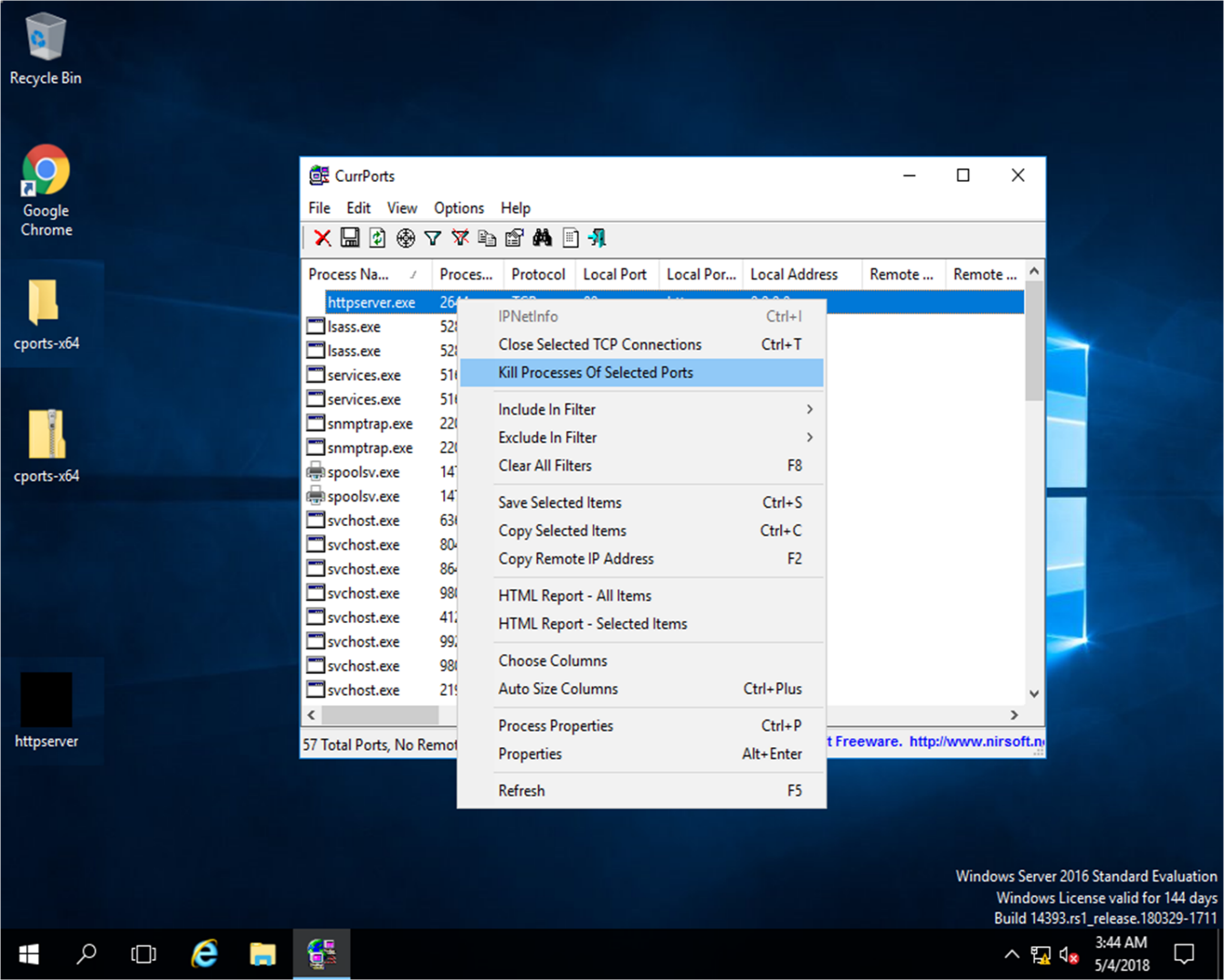
Properties show more details about the TCP connection.

1. Go to a Windows 7 machine and initiate the connection mentioned in the previous lab using a web browser.

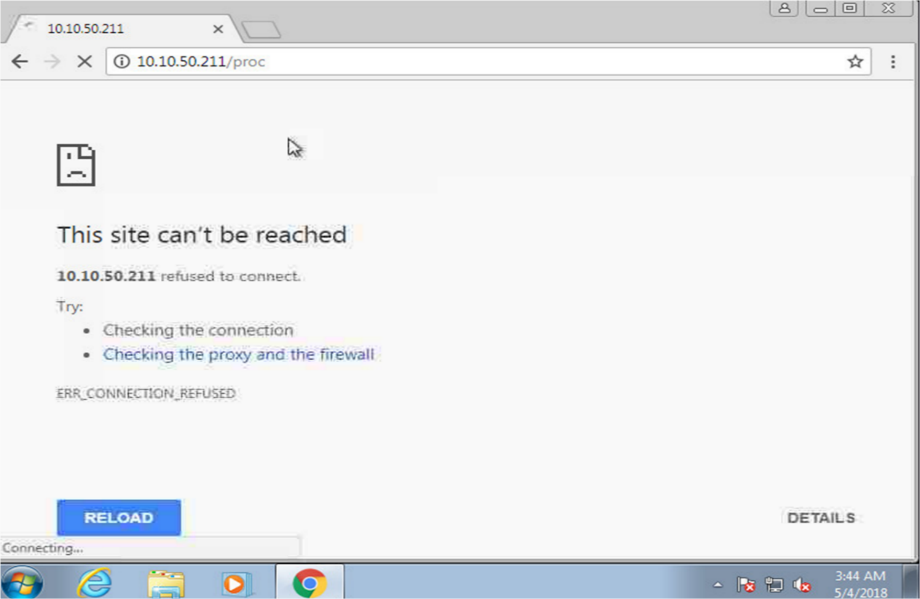


The connection is successfully established.

1. Go back to the Windows Server 2016. Kill the connection.



1. Retry establishing the connection from Windows 7 to confirm.



## Lab 5-03: Perform Static & Dynamic Malware Analysis

### Case Study

With the increasing risk of malware infections through downloaded files, organizations must proactively analyze potential threats before they can compromise IT infrastructure. LisaTech A software development company experienced a security breach when an employee unknowingly downloaded a Trojan-infected file, allowing a hacker to gain control of the system. The attacker created unauthorized directories and files, potentially leading to further exploitation. To prevent future malware intrusions, the company sought the expertise of cybersecurity specialists to implement static and dynamic malware analysis, ensuring that all files were thoroughly examined before or after being downloaded.

### Business Challenge

The organization faced a growing threat from malware-laden downloads, as employees frequently accessed files from the internet. Cybercriminals exploited this vulnerability through social engineering attacks, embedding Trojans and viruses within seemingly legitimate files. Once executed, these malicious files granted attackers unauthorized access to corporate systems, posing severe risks such as data breaches, system manipulation, and potential financial losses. To enhance security, the company needed to identify and analyze malicious files before execution, implement a robust malware analysis process to detect threats, protect IT infrastructure from future malware attacks, and train employees to recognize and avoid suspicious downloads.

### Solution

To mitigate these risks, the company enlisted a System Security Certified to assess vulnerabilities and strengthen its malware defense strategy. Before downloading any file, static malware analysis was performed to examine its code, structure, and potential indicators of compromise without executing it. If an employee mistakenly downloaded a suspicious file, dynamic malware analysis was conducted by executing the file in an isolated environment to monitor its behavior, network activity, and system modifications. By implementing these preventive and reactive measures, the organization successfully secured its IT resources, preventing further cyber threats and ensuring that employees could safely access files without the risk of malware infections.

**Note:** Before starting this lab, download HxD editor, FileHash, Process Hacker, and TCP View on Windows PC using the links below.

<https://mh-nexus.de/downloads/HxDSetupENU.zip>

<https://bit.ly/3HCbVxB>

<https://bit.ly/3si9g5W>

<https://bit.ly/3JaGZFg>

|  |
| --- |
| **Procedure** |
| **Static Malware Analysis**  1. Right-click on the **my\_app.exe** file used in Lab 7-04. Click on the **properties.**    2. Check the file type; it can be an exe or text file.    3. Open the **HxD editor** and drag & drop the **my\_app.exe** file. You should see that decoded code on the file. It also shows that the file is not executable.    4. Click on the **Files Hashes** tab. You should see the MD5 and SHA-1 hashes of the file. Copy the **MD5 hash** of the file.    5. Open the browser and search the Virus Total. Click on the **Search,** paste the **MD5 hash** of the file, and press **enter**.    6. You should see the detailed analysis of the file **my\_app.exe**.    **Dynamic Malware Analysis**  1. Open the **Process Hacker**. You should see the malware running in the background.    2. Open the **Task Manager**. You should see the description of the malware.    3. Open the **TCP View**. You should see the remote IP of the malware.    4. Open the browser and search the **Hybrid Analysis**. Drag and drop the **malware file**. It will take time to generate a detailed analysis of the file. |

## Lab 6-01: Implement Encryption and Decryption Using Cryptographic Algorithms

### Case Study

CipherSolutions Inc., a cybersecurity firm, sought to enhance data security for its clients amidst increasing cyber threats and regulatory demands. They devised a comprehensive strategy to implement encryption and decryption using cryptographic algorithms to meet this challenge. Analyzing client requirements and industry best practices, they identified suitable algorithms based on key length and attack resistance factors. They then developed a modular framework supporting diverse algorithms such as AES and RSA, ensuring compatibility and customization. Prioritizing performance optimization, they minimized computational overhead through algorithmic improvements and parallel processing techniques. Additionally, compliance with TLS/SSL and PKCS standards bolstered interoperability and client confidence in data security. CipherSolutions Inc. successfully fortified its clients' data against evolving cyber risks through these efforts.

### Business Challenge

CipherSolutions Inc., a leading provider of cybersecurity solutions, faced the pressing need to bolster the security of its clients' sensitive data. With increasing cyber threats and stringent regulatory requirements, the company recognized the imperative to implement robust encryption and decryption mechanisms across its products and services. However, the challenges were multifaceted. The company needed to ensure compatibility with various platforms, maintain performance efficiency, and adhere to industry standards while safeguarding data integrity and confidentiality.

### Solution

CipherSolutions Inc. tackled the challenges by thoroughly analyzing client needs and industry standards to select appropriate cryptographic algorithms. They then developed a modular framework supporting various algorithms, ensuring compatibility and customization. Performance optimization through algorithmic improvements and parallel processing minimized computational overhead. Compliance with TLS/SSL and PKCS standards further ensured interoperability and boosted client confidence in data security.

|  |
| --- |
| **Note:** These algorithms, AES and RSA, can be implemented in any programming language, whether C#, C++, Python, or any other.  **Code Explanation:**  This Python script demonstrates encryption and decryption using two popular cryptographic algorithms: AES (Advanced Encryption Standard) and RSA (Rivest-Shamir-Adleman).  **Libraries Used:**  The script imports necessary libraries like **os** for **system operations, random** and **string** for random data generation, and **cryptography** for cryptographic operations.  **AES Encryption and Decryption:**  The script defines functions for AES encryption and decryption.   * generate\_AES\_key(): Generates a random 256-bit AES key. * encrypt\_AES(message, key): Encrypts a message using AES encryption with a randomly generated initialization vector (IV). * decrypt\_AES(ciphertext, iv, key): Decrypts a ciphertext using the provided IV and AES key.   **RSA Key Generation and Encryption:**   * Functions are provided for RSA key generation and encryption. * generate\_RSA\_key(): Generates an RSA private key with a key size of 2048 bits. * encrypt\_RSA(message, public\_key): Encrypts a message using RSA encryption with a given public key. * decrypt\_RSA(ciphertext, private\_key): Decrypts a ciphertext using the provided RSA private key.   **Main Execution:**  In the main section:   * Random AES and RSA keys are generated. * User input is taken for messages to be encrypted. * Messages are encrypted using both AES and RSA algorithms. * Original messages, encrypted ciphertexts, and decrypted plaintexts are printed.   **Printing Results:**  Finally, the script prints the original, encrypted, and decrypted messages for both AES and RSA.  This script is a practical demonstration of encryption and decryption using AES and RSA algorithms in Python, offering insights into each process step.      **CODE:**  **import os**  **import random**  **import string**  **from cryptography.hazmat.primitives.ciphers import Cipher, algorithms, modes**  **from cryptography.hazmat.backends import default\_backend**  **from cryptography.hazmat.primitives import padding, serialization, hashes**  **from cryptography.hazmat.primitives.asymmetric import rsa**  **from cryptography.hazmat.primitives.asymmetric.padding import PKCS1v15**  **def generate\_AES\_key():**  **return os.urandom(32)**  **def encrypt\_AES(message, key):**  **iv = os.urandom(16)**  **cipher = Cipher(algorithms.AES(key), modes.CFB(iv), backend=default\_backend())**  **encryptor = cipher.encryptor()**  **padder = padding.PKCS7(algorithms.AES.block\_size).padder()**  **padded\_data = padder.update(message.encode('utf-8')) + padder.finalize()**  **ciphertext = encryptor.update(padded\_data) + encryptor.finalize()**  **return ciphertext, iv**  **def decrypt\_AES(ciphertext, iv, key):**  **cipher = Cipher(algorithms.AES(key), modes.CFB(iv), backend=default\_backend())**  **decryptor = cipher.decryptor()**  **unpadder = padding.PKCS7(algorithms.AES.block\_size).unpadder()**  **decrypted\_data = decryptor.update(ciphertext) + decryptor.finalize()**  **unpadded\_data = unpadder.update(decrypted\_data) + unpadder.finalize()**  **return unpadded\_data.decode('utf-8')**  **def generate\_RSA\_key():**  **private\_key = rsa.generate\_private\_key(public\_exponent=65537, key\_size=2048, backend=default\_backend())**  **return private\_key**  **def encrypt\_RSA(message, public\_key):**  **ciphertext = public\_key.encrypt(message.encode('utf-8'), PKCS1v15())**  **return ciphertext**  **def decrypt\_RSA(ciphertext, private\_key):**  **plaintext = private\_key.decrypt(ciphertext, PKCS1v15())**  **return plaintext.decode('utf-8')**  **aes\_key = generate\_AES\_key()**  **rsa\_private\_key = generate\_RSA\_key()**  **rsa\_public\_key = rsa\_private\_key.public\_key()**  **plain\_text\_aes = input("Enter a message to encrypt with AES: ")**  **cipher\_text\_aes, iv = encrypt\_AES(plain\_text\_aes, aes\_key)**  **print(f"Original message: {plain\_text\_aes}")**  **print(f"Encrypted message with AES: {cipher\_text\_aes.hex()}")**  **decrypted\_text\_aes = decrypt\_AES(cipher\_text\_aes, iv, aes\_key)**  **print(f"Decrypted message with AES: {decrypted\_text\_aes}")**  **plain\_text\_rsa = input("Enter a message to encrypt with RSA: ")**  **cipher\_text\_rsa = encrypt\_RSA(plain\_text\_rsa, rsa\_public\_key)**  **print(f"Original message: {plain\_text\_rsa}")**  **print(f"Encrypted message with RSA: {cipher\_text\_rsa.hex()}")**  **decrypted\_text\_rsa = decrypt\_RSA(cipher\_text\_rsa, rsa\_private\_key)**  **print(f"Decrypted message with RSA: {decrypted\_text\_rsa}")** |

## Lab 6-02: Set up And Configure SSL/TLS For Secure Communications

### Case Study

XYZ Corporation is a global financial institution providing online banking services to millions of customers worldwide. With the increasing prevalence of cyber threats, ensuring secure communication between clients and servers has become a top priority. However, the company faces challenges setting up and configuring SSL/TLS for secure communications across its online banking platform.

### Business Challenge

XYZ Corporation's current communication infrastructure faces significant security risks, leaving sensitive customer data vulnerable to interception and breaches. This threatens regulatory compliance, particularly in the highly regulated financial industry, where failure to meet data security standards can result in severe penalties and reputational damage. Additionally, configuring SSL/TLS for secure communications also presents challenges compounded by resource constraints. Despite recognizing the importance of secure communication protocols, XYZ lacks the expertise and resources for effective deployment and maintenance. Finding a streamlined solution is imperative to address these vulnerabilities efficiently.

### Solution

To enhance security, XYZ Corporation should prioritize the implementation of SSL/TLS certificates. This involves obtaining certificates from trusted Certificate Authorities (CAs) to authenticate the server's identity and encrypt communication between clients and servers. Configuring the server to support only secure cipher suites and protocols, such as TLS 1.2 or higher, mitigates cryptographic vulnerabilities and ensures robust encryption. Regular updates and patch management are essential to keeping SSL/TLS libraries and server software up-to-date with the latest security patches, reducing the risk of exploitation by cyber attackers. Deployment of a Web Application Firewall (WAF) adds an extra layer of defense by inspecting and filtering incoming web traffic, detecting and blocking malicious requests attempting to exploit SSL/TLS protocols or web applications. Continuous monitoring tools should be implemented to track SSL/TLS traffic for anomalies and security incidents, with regular security audits conducted to assess the effectiveness of SSL/TLS configurations and identify areas for improvement. This comprehensive approach ensures a secure and compliant communication environment for XYZ Corporation's sensitive data and transactions.

|  |
| --- |
| 1. Open your Kali Linux terminal. To install an **SSL** scan on your machine, use **sudo apt install SSLscan.**      1. Now, you can see all the configurations of any website by using an **SSL** scan. Use the command **sslscan** ***website URL.***      1. After this, you will get all the information and configurations of SSL/TLS on the website will get the following things. |

## Lab 6-03: Explore Cryptographic Tools and Libraries (e.g., OpenSSL) in a Lab Environment.

**Case Study**

In response to mounting data security concerns, ABC Company proactively initiated efforts to reinforce its security measures against unauthorized access and breaches while concurrently ensuring adherence to stringent regulatory mandates. However, the intricacies of seamlessly integrating cryptographic solutions into pre-existing systems presented a formidable challenge. To surmount this hurdle, ABC Company explored cryptographic tools and libraries within a specialized laboratory environment, primarily emphasizing OpenSSL. ABC Company garnered invaluable technical insights through methodical experimentation encompassing rigorous encryption and decryption testing, meticulous key management, proficient certificate handling, comprehensive performance evaluation, and thorough security auditing. Consequently, the company effectively bolstered its data security posture, achieved regulatory compliance, and seamlessly integrated cryptographic solutions into its operational framework, reinforcing its resilience against evolving threats within the digital realm.

### Business Challenge

A business operating in a data-driven environment faces several data security, compliance, and integration complexity challenges. Firstly, safeguarding sensitive information against unauthorized access and breaches is paramount. Secondly, meeting regulatory compliance standards such as GDPR, HIPAA, or PCI DSS requires robust cryptographic solutions. Lastly, integrating cryptographic tools seamlessly into existing systems without disruption poses a significant challenge.

### Solution

To address these challenges, the business established a controlled lab environment to explore cryptographic tools, focusing on OpenSSL. The exploration encompassed encryption and decryption testing across various data types, key management, certificate handling, performance evaluation, and security auditing. This comprehensive approach enabled the business to enhance data security, achieve compliance, and seamlessly integrate cryptographic solutions into its operations.

|  |
| --- |
| **Note:** You need to download OpenSSL first. To download OpenSSL, visit the OpenSSL website and navigate to the Downloads section. Choose the appropriate version for your operating system and select the corresponding installation package. Download the package and verify its integrity. Install OpenSSL following the instructions, then verify the installation by checking the version in the terminal or command prompt using the command openssl version.   1. Open the command prompt in your windows and type **openssl** to see all the menus in openssl.      1. Now, add a create new txt file; you can use **echo “*anything*” >*name of the file.***       As you can see the text in the file.   1. Now, for encrypting this file, use the command **openssl *any enc version* –e –in *filename***   **-out *the file name*** you want to save ( -e means encryption, -in means input,-out means output); enter the password and retype it (any password).    In the above image, These are some encryption versions.     1. You can check the encryption file by typing the output file name.      1. Now for decryption, use the command as same as point 3 **openssl *any enc version* –e –in *filename* -out *the file name*** use set password. Also, use another file name for precise results*.*      1. Now check the file by typing the name; both have the same text After decryption. |

## Lab 6-04: Practice Generating and Managing Digital Certificates

### Case Study

XYZ Company, a prominent e-commerce platform, faces security threats and compliance challenges due to the sensitive data it handles. They need help managing the digital certificate lifecycle across various servers and endpoints. To tackle this, XYZ implements a robust PKI, automates certificate lifecycle management, and adopts a centralized certificate management platform. Integration with SIEM systems enhances security monitoring, while regular audits ensure compliance. This comprehensive approach strengthens XYZ's cybersecurity posture, reduces risks, and fosters stakeholder trust.

### Business Challenge

XYZ Company, a leading e-commerce platform, faces significant security threats due to the sensitive nature of the data it handles, including customer information and financial transactions. The risk of unauthorized access and data breaches is a constant concern. Compliance with industry regulations such as PCI DSS (Payment Card Industry Data Security Standard) and GDPR (General Data Protection Regulation) requires robust security measures, including correctly managing digital certificates. Failure to comply could result in hefty fines and damage to the company's reputation. With many digital certificates deployed across various servers, endpoints, and cloud services, XYZ needs help managing the entire certificate lifecycle, including issuance, renewal, revocation, and monitoring for expiration dates.

### Solution

To address these challenges, XYZ implements a robust Public Key Infrastructure (PKI) to manage its digital certificates effectively. This includes establishing a trusted Certificate Authority (CA) within the organization to issue and manage certificates, ensuring the authenticity and integrity of sensitive data. Automated tools and solutions are adopted for certificate lifecycle management, automating the issuance, renewal, and revocation processes, reducing the likelihood of human error, and ensuring that certificates are always up to date. XYZ also invests in a centralized certificate management platform that provides a single pane of glass view for all certificates deployed across the organization. This platform enables administrators to track certificate usage, monitor expiration dates, and respond quickly to security incidents. Additionally, XYZ integrates its certificate management platform with Security Information and Event Management (SIEM) systems to enhance security monitoring. This enables real-time monitoring of certificate-related events and anomalies, enabling proactive threat detection and response. Regular audits and compliance checks ensure that certificate management practices align with industry regulations and internal security policies. Any deviations or vulnerabilities are promptly addressed to maintain a secure environment.

|  |
| --- |
| **Note:** You can perform this task in Windows using the Command Prompt (CMD) with similar commands:   * **Press Win + R, type certmgr. msc, and press Enter to open the Certificate Manager.** * **In the Certificate Manager window, go to Action > All Tasks > Import.** * **Follow the Import Wizard to import the certificate from the file you received.** * **Choose the option to import the certificate into the Local Machine's Trusted Root Certification Authorities store.** * **Complete the wizard and the certificate will be imported successfully.**  1. Open your terminal and install **OpenSSL** using the command **sudo apt install openssl.** If you have already installed skip this. 2. Now use the command **openssl genpkey -algorithm RSA -out private.key,** This command generates an RSA private key and saves it to a file named private.key using OpenSSL.      1. Now use the command **openssl req -new -key private.key -out csr.pem,** This command generates a CSR named csr.pem using the private key generated in the previous step (fill all the things according to your demand). Set a password for a private file, and do not share the password with anyone.      1. You can submit the generated CSR (csr.pem) to a CA of your choice, such as Let's Encrypt or another trusted CA, either manually through their website or using automated tools like Certbot. Once the CA verifies your CSR, they will issue a digital certificate. Download the certificate and install it on your server or device. 2. Use the command **openssl x509 -inform pem -in certificate.crt -out certificate.pem,** This command converts the certificate from .crt format to .pem format. 3. You can verify the installed certificate **using openssl x509 -noout -text -in certificate.pem**, This command will display detailed information about the installed certificate. The **crs.pem** file contains your certificate and the **private.key** file contains your password. |

## Lab 7-01: Configure and Secure Network Devices

### Case Study

CyberDefend Solutions is a premier cybersecurity firm specializing in providing comprehensive security solutions to businesses worldwide. Founded in 2010, CyberDefend Solutions has earned a reputation for excellence in delivering cutting-edge cybersecurity strategies and services. With a team of experienced cybersecurity professionals and network engineers, CyberDefend Solutions is dedicated to helping businesses mitigate cyber risks and protect their digital assets.

### Business Challenge

One of CyberDefend Solutions' clients, a medium-sized marketing agency named MarketBoost Inc., faced a critical cybersecurity challenge. MarketBoost Inc.'s network infrastructure consisted of a single router, a switch, and three PCs used by their marketing team. The marketing agency had recently experienced a security incident, resulting in unauthorized access to sensitive client data and potential exposure of confidential marketing strategies. Concerned about the potential impact on client relationships and reputation, MarketBoost Inc. sought CyberDefend Solutions' expertise to secure their network devices and prevent future cybersecurity incidents.

You are recruited as a Network Engineer for CyberDefend Solutions. Your task is to install and configure firewalls on PCs to enhance network security.

### Solution

In this lab, you will configure firewall software on each PC to monitor and control inbound and outbound traffic. Create firewall rules to block potentially malicious connections and restrict access to sensitive data.

A network security device, whether in hardware or software form, is a firewall that controls inbound and outbound traffic according to a defined set of security rules. It determines whether to accept, reject, or block specific traffic based on these rules.

1. Configure Router

2. Assign IPv4 addresses to PC1, PC2, and PC3

3. Configure Firewall on PCs

4. Verification

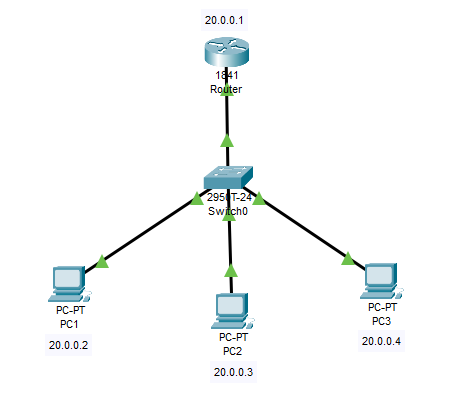


Figure 7-05: Lab Topology

### Configuration & Verification

|  |
| --- |
| **//Configure Router**  Router>en  Router#config t  Router(config)#int fa0/0  Router(config-if)#ip add 20.0.0.1 255.0.0.0  Router(config-if)#no shut  exit    **//Assign IPv4 to PC1**    **//Assign IPv4 to PC2**    **//Assign IPv4 to PC3**    **//Firewall Configuration on PC1**    **//Firewall Configuration on PC2**    **//Firewall Configuration on PC3**    **//Verify PC1**  C:/>ping 20.0.0.3  C:/>ping 20.0.0.4    Check the web browser by typing the IP address into the URL bar.  20.0.0.3    **//Verify PC2**  C:/>ping 20.0.0.2  C:/>ping 20.0.0.4    Check the web browser by typing the IP address into the URL bar.  20.0.0.4    **//Verify PC3**  C:/>ping 20.0.0.2  C:/>ping 20.0.0.3    Check the web browser by typing the IP address into the URL bar.  20.0.0.2 |

## Lab 7-02: Hacking a Wi-Fi Protected Access Network using Aircrack-ng

### Case Study

CyberShield Security Solutions, a cybersecurity firm specializing in network security assessments, conducted an ethical hacking simulation to evaluate the security of a Wi-Fi network secured with WPA encryption. Wireless networks, while essential for connectivity, remain vulnerable to password-based attacks if weak or easily guessable credentials are used. To assess potential security risks, the penetration testing team captured 802.11 wireless network packets and saved them for analysis. Using Common User Passwords Profiler (Cupp) and Aircrack-ng, they created a custom password list and attempted to crack the WPA passphrase. This simulation provided critical insights into wireless security weaknesses, reinforcing the importance of strong authentication mechanisms and secure password policies to protect against unauthorized access.

### Business Challenge

Wireless networks are a prime target for cybercriminals, as weak passwords and improper security configurations can allow unauthorized access to sensitive information. Attackers exploit WPA security flaws to intercept data, infiltrate networks, and compromise connected systems. CyberShield Security Solutions identified several key security challenges, including the need to capture and analyze wireless network traffic to detect vulnerabilities, assess WPA password strength against brute-force and dictionary attacks, and strengthen authentication measures to prevent unauthorized access. Additionally, the company emphasized the importance of educating employees and clients on using strong, complex passwords and enabling Multi-Factor Authentication (MFA) to enhance overall network security.

### Solution

To mitigate these risks, the penetration testing team at CyberShield Security Solutions utilized Aircrack-ng to capture 802.11 network packets and extract the WPA handshake data required for password cracking. They then employed Cupp to generate a customized password list based on user-specific information, improving the efficiency of dictionary attacks. By executing Aircrack-ng against the captured data, they successfully demonstrated how weak passwords could be exploited. This exercise emphasized the importance of implementing strong passphrase policies, conducting regular security audits, and adopting advanced security protocols, such as WPA3 encryption and enterprise authentication methods, to mitigate wireless security risks effectively.

1. Capture some WLAN packets using the filter **eth.add==aa:bb:cc:dd:ee** and save the file.
2. Go to a Kali Linux terminal.
3. Change the directory to the desktop.

root@kali:~# **cd Desktop**

1. Download the **Cupp** utility to create a wordlist.

root@kali:~# git clone [https://github.com/chetan3 1295/cupp.git](https://github.com/chetan31295/cupp.git)



1. Change the directory to /Desktop/Cupp.

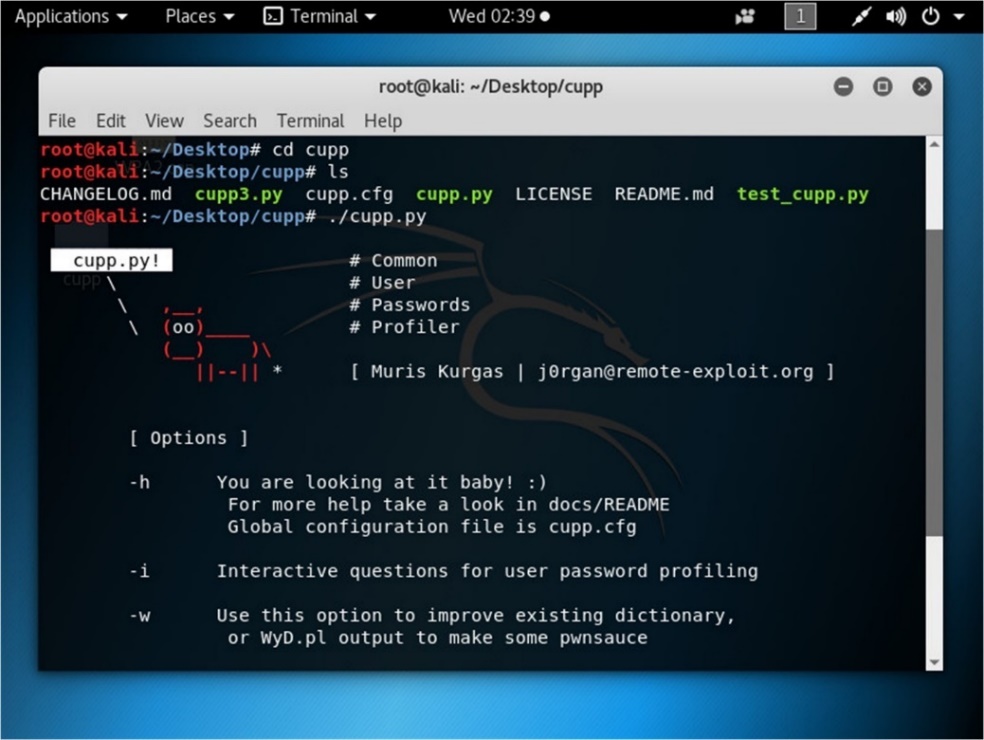
root@kali:~/Desktop# **cd cupp**

1. List the folders in the current directory.

root@kali:~/Desktop/cupp# **ls**

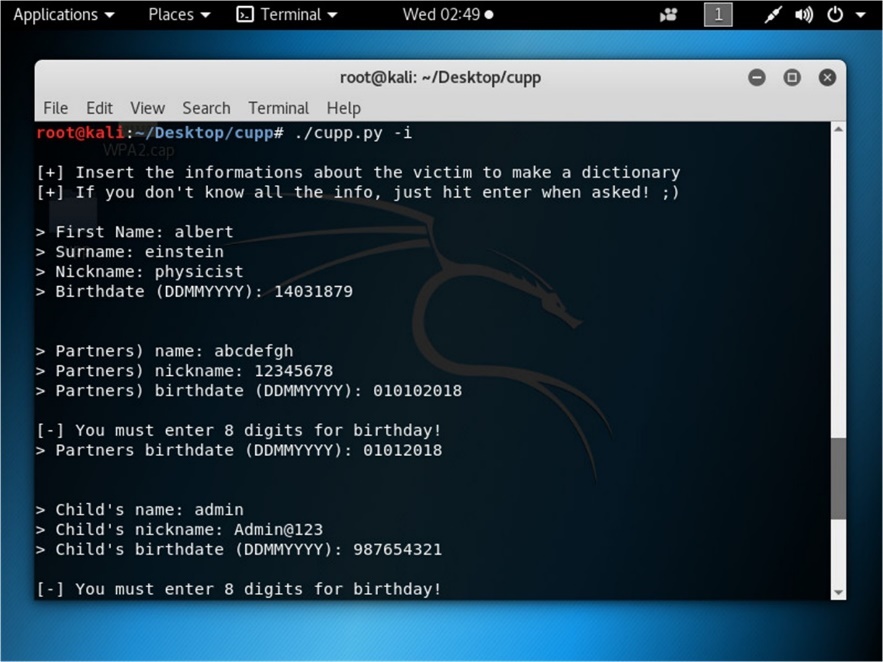
1. Run the utility **cupp.py**

root@kali:~/Desktop/cupp# **./cuppy.py**

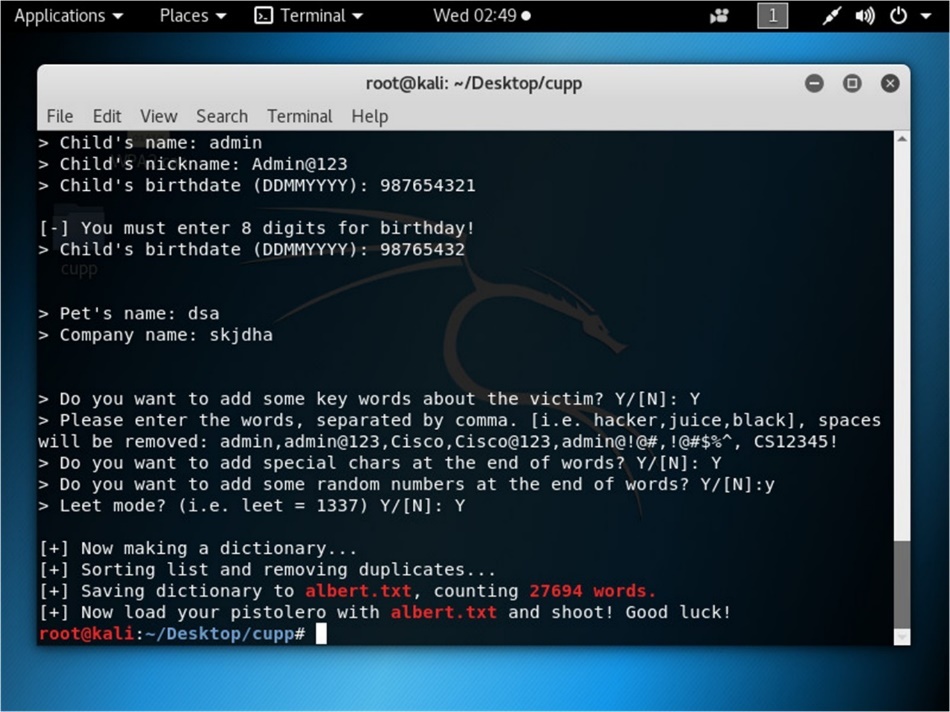


1. Use an interactive question for user password profiling.

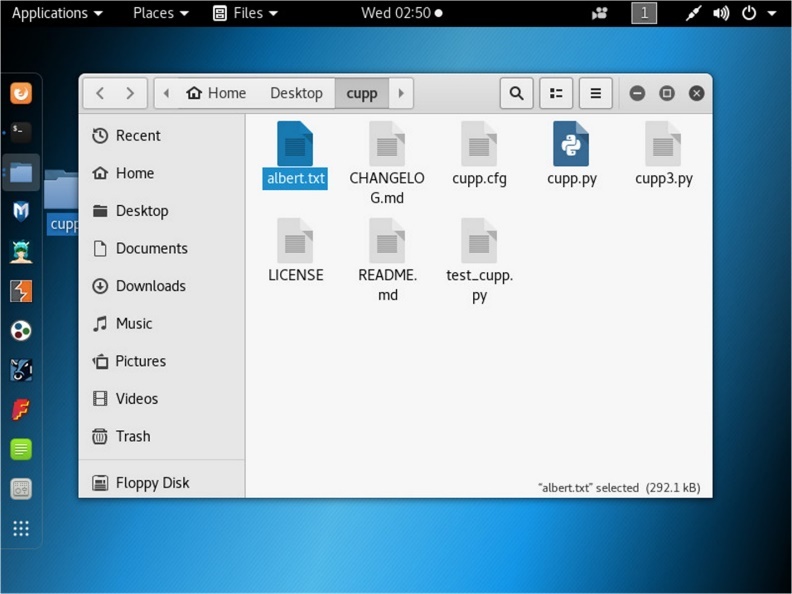
root@kali:~/Desktop/cupp# **./cupp.py -i**



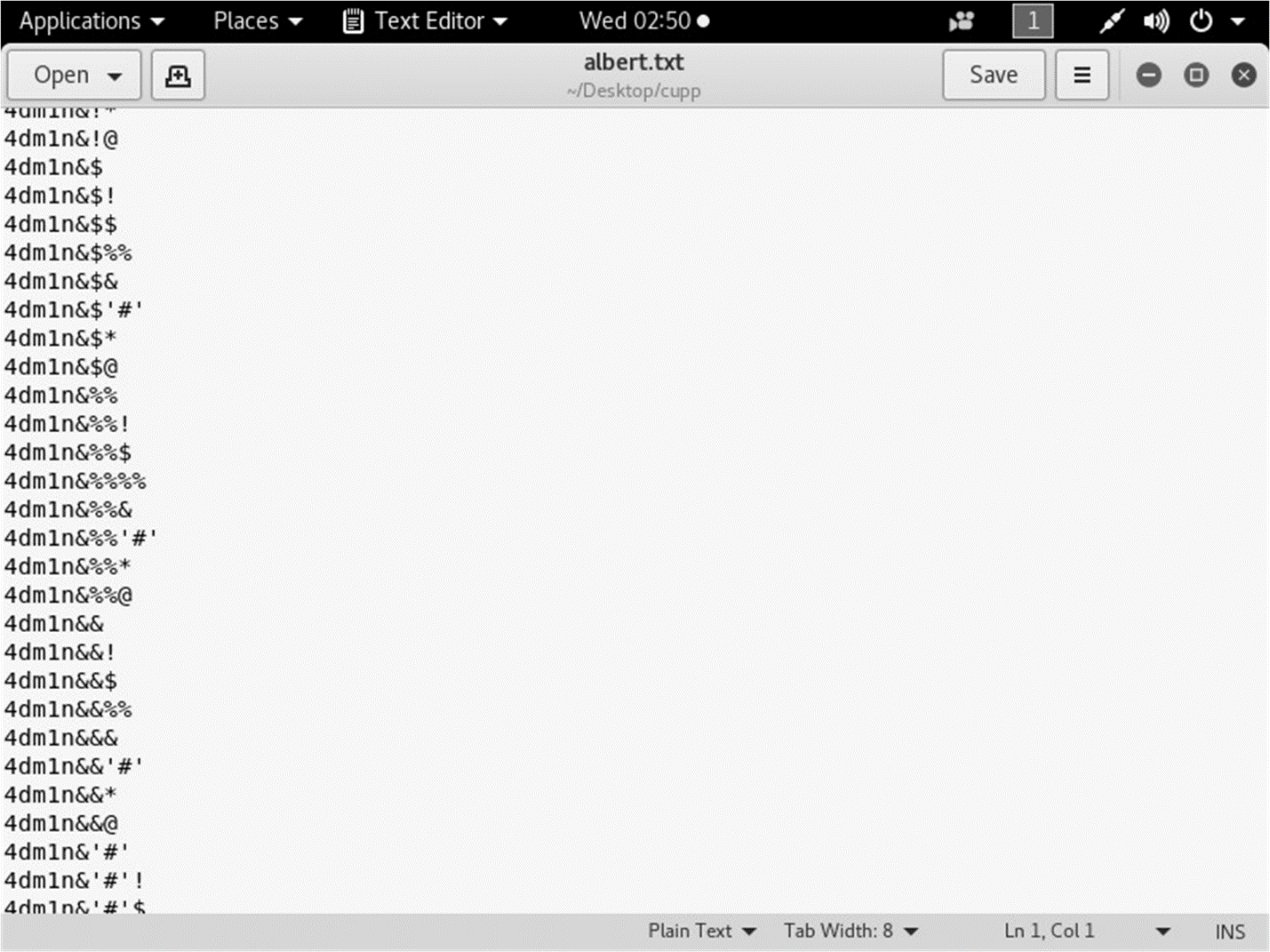
1. Provide the closest information about the target. It will increase the chances of successful cracking.
2. You can add keywords.
3. You can add special characters.
4. You can add random numbers.
5. You can enable the Leet mode.



1. After successful completion, you will find a new text file named the first name you typed in the interactive option. This file will contain many possible combinations. As shown in the figure below, the albert.txt file has been created in the current directory.



1. You can check the file by opening it.

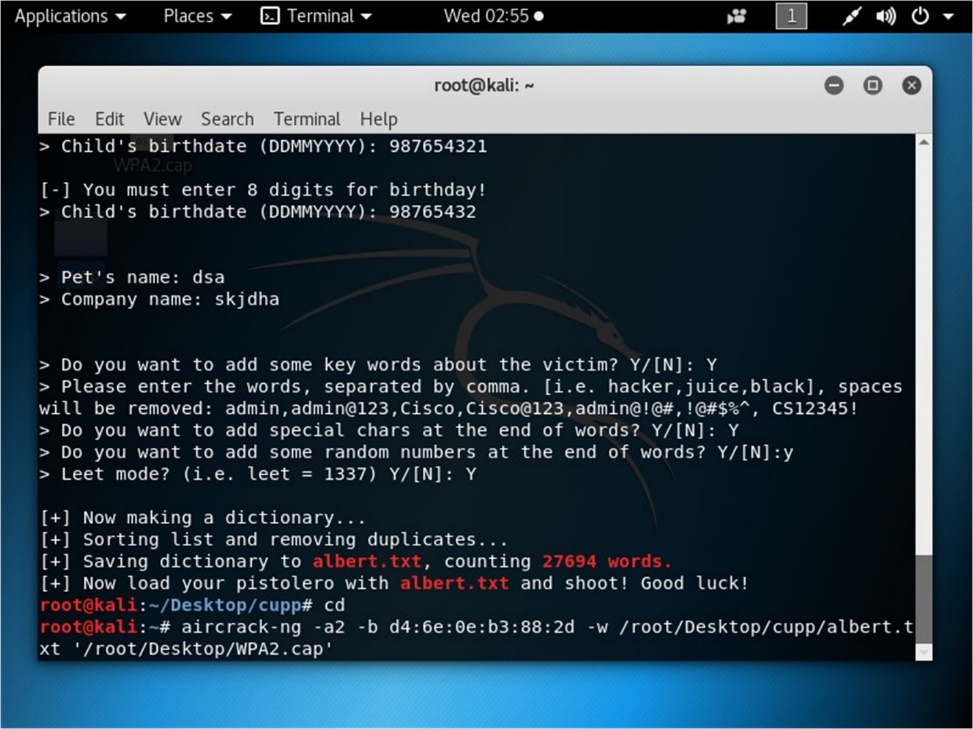


1. Now, crack the password using Aircrack-ng with the help of the password file created.

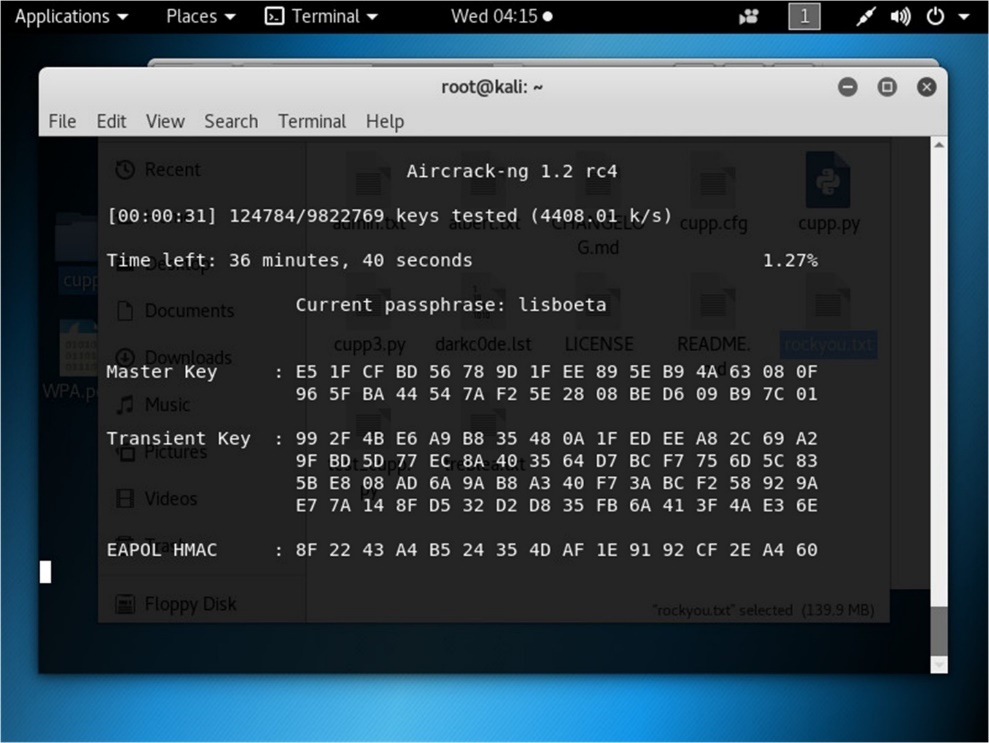
root@kali:~ # cd

root@kali:~ # **aircrack-ng –a2 –b <*BSSID of WLAN Router*> -w /root/Desktop/cupp/Albert.txt ‘/root/Desktop/WPA.cap’**

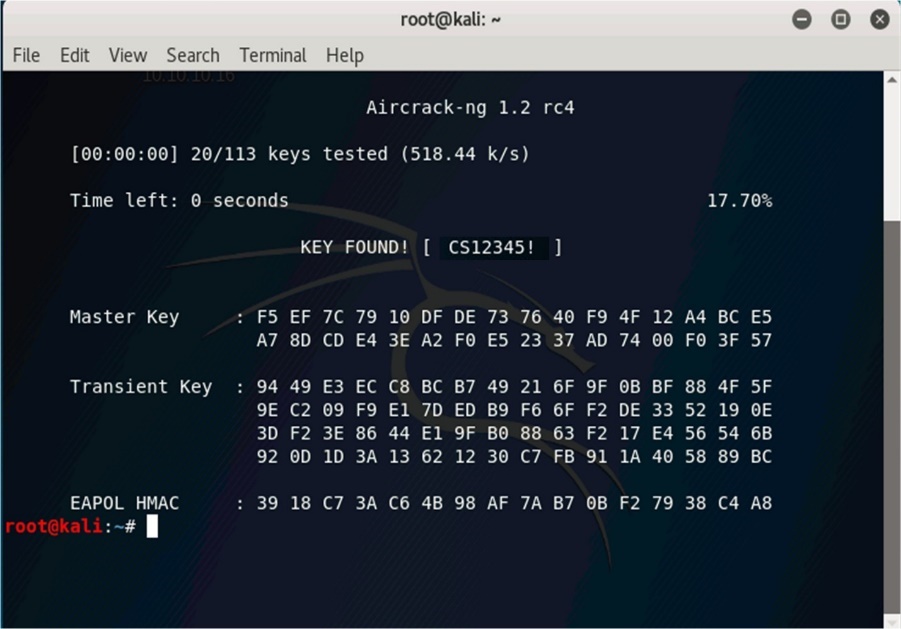
WPA.cap is a captured packet file.



1. This will start the process, and all keys will be checked.



1. The result will either show you the key or will refuse to crack from the dictionary.



## Lab 7-03: Set up VPNs (Virtual Private Networks) for Secure Remote Access

### Case Study

CW is a cybersecurity consultancy company that offers security services such as PCI DSS implementation and audit, ISMS - ISO 27001, vulnerability assessment, and penetration testing. Some employees work remotely. The head office, located in City 1, houses a centralized data center with various servers, including web, file, and database servers etc.

### Business Challenge

The organization requires a secure method for employees to access the company's servers from their respective locations to the head office. As sensitive client data is frequently exchanged and accessed by staff, it is essential to ensure data security and maintain productivity levels.

As a network security administrator in CW, you have to create a Virtual Private Network (VPN) to enable secure remote connectivity, allowing employees to access sensitive information securely.

### Solution

In this lab, we will configure a private network on a router for secure remote access.

***Configuring VPN On Router***

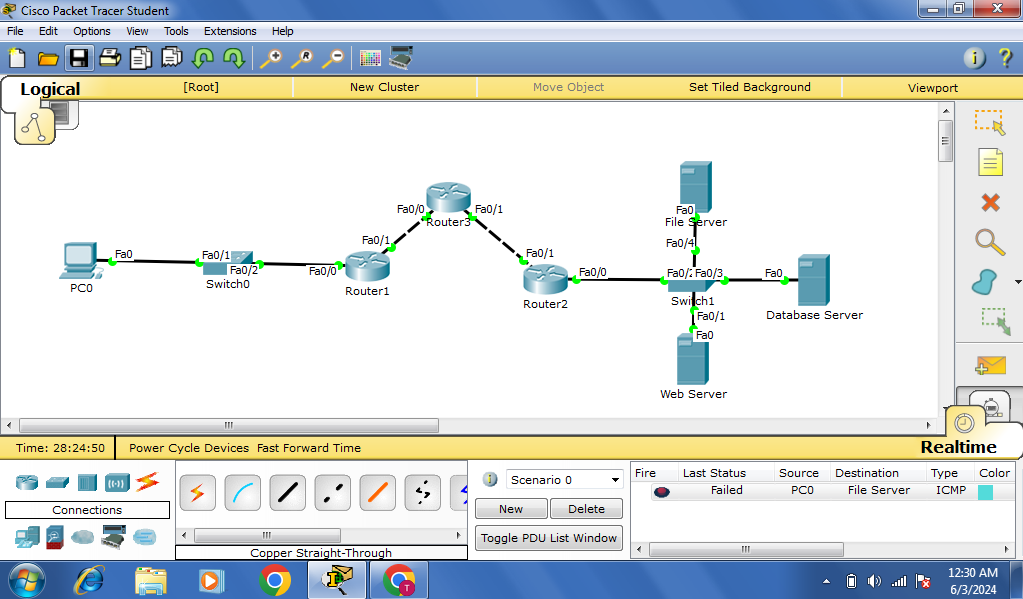


Figure 7-07: Lab Topology

|  |
| --- |
| **// Configure Interfaces on Router 1**  Router> enable  Router# configure terminal  Router(config)# hostname Router1  Router1(config)# interface fastethernet0/0  Router1(config-if)# ip address 192.168.1.1 255.255.255.0  Router1(config-if)# no shutdown  Router1(config-if)# exit  Router1(config-if)# interface fastethernet0/1  Router1(config-if)# ip address 1.1.1.2 255.0.0.0  Router1(config-if)# no shutdown  Router1(config-if)# exit  Router1(config)# exit    **//Configure Interfaces on Router 2**  Router> enable  Router# configure terminal  Router(config)# hostname Router2  Router2(config)# interface fastethernet0/0  Router2(config-if)# ip address 193.168.1.1 255.255.255.0  Router2(config-if)# no shutdown  Router2(config-if)# exit  Router2(config-if)# interface fastethernet0/1  Router2(config-if)# ip address 2.1.1.2 255.0.0.0  Router2(config-if)# no shutdown  Router2(config-if)# exit  Router2(config)# exit    **// Configure Interfaces on Router 3**  Router> enable  Router# configure terminal  Router(config)# hostname Router2  Router3(config)# interface fastethernet0/0  Router3(config-if)# ip address 1.1.1.1 255.0.0.0  Router3(config-if)# no shutdown  Router3(config-if)# exit  Router3(config-if)# interface fastethernet0/1  Router3(config-if)# ip address 2.1.1.1 255.0.0.0  Router3(config-if)# no shutdown  Router3(config-if)# exit  Router3(config)# exit    **// Check Routing Table of R1**  Router1# show ip route    **// Check Routing Table of R2**  Router2# show ip route    **// Check Routing Table of R3**  Router3# show ip route    **// Configure Static Routing on Router 1**  Router1# configure terminal  Router1(config)# ip route 193.168.1.0 255.255.255.0 1.1.1.1  Router1(config)# ip route 2.0.0.0 255.0.0.0 1.1.1.1  Router1(config)# exit    **// Configure Static Routing on Router 2**  Router2# configure terminal  Router2(config)# ip route 192.168.1.0 255.255.255.0 2.1.1.1  Router2(config)# ip route 1.0.0.0 255.0.0.0 2.1.1.1  Router2(config)# exit    **// Configure Static Routing on Router 3**  Router1# configure terminal  Router1(config)# ip route 192.168.1.0 255.255.255.0 1.1.1.2  Router1(config)# ip route 193.168.1.0 255.255.255.0 2.1.1.2  Router1(config)# exit    **// Check Routing Table of R1**  Router1# show ip route    **// Check Routing Table of R2**  Router2# show ip route    **// Check Routing Table of R3**  Router3# show ip route  **// Configure Tunnel Interface on Router 1**  Router1# configure terminal  Router1(config)# interface tunnel 1  Router1(config-if)# ip address 3.1.1.1 255.0.0.0  Router1(config-if)# tunnel source fastethernet0/1  Router1(config-if)# tunnel destination 2.1.1.2  Router1(config-if)# exit  Router1(config)# exit    **// Configure Tunnel Interface on Router 2**  Router1# configure terminal  Router2(config)# interface tunnel 1  Router2(config-if)# ip address 3.1.1.2 255.0.0.0  Router2(config-if)# tunnel source fastethernet0/1  Router2(config-if)# tunnel destination 1.1.1.2  Router2(config-if)# exit  Router2(config)# exit    **// Check Routing Table of R1**  Router1# show ip route    **// Check Routing Table of R2**  Router2# show ip route    **// Configure Tunnel Routing on Router 1**  Router1# configure terminal  Router1(config)# ip route 193.168.1.0 255.255.255.0 3.1.1.2  Router1(config)# exit    **// Configure Tunnel Routing on Router 2**  Router2# configure terminal  Router2(config)# ip route 192.168.1.0 255.255.255.0 3.1.1.1  Router2(config)# exit    **// Check Routing Table of R1**  Router1# show ip route    **// Check Routing Table of R2**  Router2# show ip route    Hence, the tunnel is configured. |

## Lab 7-04: Perform Wireless Traffic Analysis

### Case Study

NetSecure Solutions, a cybersecurity firm specializing in network security assessments, was tasked with monitoring wireless network traffic to evaluate potential vulnerabilities and ensure compliance with security policies. Understanding wireless network behavior is crucial for identifying unauthorized access, misconfigurations, and encryption weaknesses. The security team aimed to gather essential network details, including SSID, encryption algorithms, operating country, and network type (enterprise or personal). By leveraging traffic analysis tools, the company sought to enhance its wireless security posture and protect against potential cyber threats.

### Business Challenge

Wireless networks present unique security challenges, as they are more susceptible to interception, unauthorized access, and data leaks compared to wired networks. Organizations must continuously monitor network traffic to detect anomalies, assess encryption protocols, and ensure compliance with security best practices. NetSecure Solutions identified several key challenges in this regard, including the need to capture and analyze wireless network traffic to gather essential network details, identify security algorithms and encryption standards to assess protection levels, and dist.inguish between enterprise and personal network configurations to enforce appropriate security policies. Additionally, the company recognized the importance of filtering out irrelevant traffic to improve efficiency in network monitoring and analysis.

### Solution

To address these challenges, the security team at NetSecure Solutions deployed Wireshark, a powerful traffic analysis tool, to capture and examine wireless network traffic. By analyzing the collected data, they successfully extracted key network details such as SSID, encryption protocols, channel numbers, and network type. Advanced filtering techniques were used to eliminate unnecessary traffic, allowing for more precise monitoring. This process enabled the organization to detect security vulnerabilities, verify encryption strength, and enhance overall network security. Through continuous monitoring and traffic analysis, NetSecure Solutions strengthened its wireless security framework, ensuring secure communication and protection against potential cyber threats.

|  |
| --- |
| **Wireshark**  Open Wireshark on your Linux machine and start capturing the packets.  **1. Identify Channel**  **wlan.ssid == "Home\_Network"**  Apply this filter to display the frames coming from the router with the SSID **Home\_Network**. Under the radio, the information section shows the channel of the wireless network, that is, **channel 6**.    **2. Identify the Encryption and the Network Type**  Select a packet, and check Group Cipher Suite and Pairwise Cipher Suite (cipher is a term in cryptography); you will notice that both suites use **AES encryption**. WPA2 uses AES encryption.  The authentication mode is **PSK (Pre-Shared Key),** which means the access point is located in a small office or home network,    **3. Identify WPS Setup**  **wlan.ssid contains "Amazon Wood" && wlan.wfa.ie.type == 0x4**  Apply this filter to check whether WPS is set up. It allows IoT devices to connect to the network without entering long passwords because they do not have keyboards. This information is found under the **tag parameter**.    **4. Search by MAC Address**  **wlan.ta == MAC\_Address || wlan.ra == MAC\_Address**  **ta** means transmitter's address, and **ra** means receiver's address.    **5. List All Packets Sent By The Host**  **wlan.fc.type\_subtype == 0x20 && wlan.ra == e8:de:27:16:87:18**  e8:de:27:16:87:18 is the AP address; change it.  If the station (host in the network) sends any data to the access point, it sets the BSSID in the **wlan.ra** field and has the frame control **type\_subtype** set to **0x20**. As a result, you can use this filter to get a list of all data packets sent from the station (host) to the access point.    **6. Identify the Country**  Country information is found in the Country Information parameter. It is broadcast in all the frames (beacons). It provides the standard two-character country code, e.g., US, UK, IN. |

## Lab 7-05: Create A Rogue Access Point To Capture Data Packets

### Case Study

A cafe owner was concerned about his employees' internet usage during working hours and wanted to ensure they were only accessing websites relevant to their job responsibilities. To monitor network activity, he sought a solution that would allow him to track internet traffic without disrupting daily operations. To meet this requirement, ProTech Security Solutions was hired to implement a network monitoring strategy using a rogue access point to capture data packets and analyze employees’ internet usage.

### Business Challenge

Ensuring that employees adhere to internet usage policies while maintaining network security is a significant challenge for business owners. Traditional network monitoring tools may not provide sufficient visibility into user activity, making it difficult to detect unauthorized browsing or potential security threats. The cafe owner faced several key challenges, including the need to monitor employees’ internet activity without interrupting their workflow, capture and analyze network traffic to identify unauthorized browsing, and ensure data integrity and security while implementing monitoring solutions. Additionally, deploying a seamless and undetectable monitoring method was essential to prevent employees from circumventing restrictions.

### Solution

To address these challenges, ProTech Security Solutions deployed a rogue Access Point (AP) using WiFi Pumpkin 3, a tool designed for wireless network monitoring and penetration testing. The rogue AP was configured to appear as a legitimate WiFi network, allowing employees to connect without suspicion. Once connected, all network traffic passing through the rogue AP was captured and analyzed, enabling the cafe owner to monitor internet activity effectively. Through this approach, the business was able to identify non-work-related browsing, enforce security policies, and ensure compliance with company guidelines. This implementation provided valuable insights into network activity, helping the cafe owner maintain a secure and controlled internet environment within the workplace.

|  |
| --- |
| **1. Install WiFi Pumpkin 3**  Install some packages using the **apt install libssl-dev libffi-dev build-essential** command.    Download Pumpkin using the **git clone** <https://github.com/P0cL4bs/wifipumpkin3.git> command.    Install Python packages using the **apt install python3-pyqt5 hostapd** command.    Install python using the **sudo python3 setup.py install** command.    Type **sudo pumpkin3** to enter the session.    Use the **ap** command to see the default settings of the **rogue AP**.    Enter the **proxies** command to see the available proxies.    Use the following commands to set a **rogue AP**;  **set interface *wlan0***  **set ssid *ssid\_name***  **set proxy *noproxy***  **ignore pydns\_server** (to ignore the DNS-related information)    Use the **start** command to activate the AP.    It shows the information when someone connects to it and uses the Internet. |