EncryptEdge Labs

Cybersecurity Analyst Internship

Task Report

atalmamun@gmail.com

Task No: 29

Credit: Offensive Security

EncryptEdge Labs

Table of Contents

1.0 EncryptEdge Labs Internship Task Report

1.1 Introduction

The Capstone Project titled *"Integrated Incident Response and Emerging Threats Defense"* is designed to demonstrate practical knowledge and hands-on skills acquired throughout the cybersecurity internship program. This project simulates real-world cyber incidents and requires a complete incident response cycle, from detection to recovery. It also includes the identification and mitigation of emerging threats through the analysis of recent Common Vulnerabilities and Exposures (CVEs). By engaging with a series of guided labs and simulations, the project emphasizes critical thinking, technical investigation, and professional reporting—core skills for any cybersecurity analyst.

1.2 Objective

The primary objective of this capstone project is to provide hands-on experience in:

- Executing full-cycle incident response (IR) processes including identification, containment, eradication, and recovery.
- Investigating and mitigating known vulnerabilities using up-to-date CVE data.
- Strengthening detection and prevention strategies to minimize the risk of future cyber threats.
- Developing clear, actionable, and professional documentation of cybersecurity incidents and mitigation strategies.

1.3 Requirements

To successfully complete the capstone project, the following requirements must be met:

- Completion of all assigned modules in the RangeForce platform, covering both incident response and CVE-based threat mitigation.
- Accurate documentation of each exercise, including indicators of compromise (IoCs), detection and containment steps, tools used, and strategic recommendations.
- Submission of a comprehensive incident simulation report and CVE defense summary.
- Inclusion of screenshots as proof of lab completions and key milestones.
- Use of appropriate tools such as a text editor or word processor for reporting and a screenshot tool to capture evidence of task execution.

2.0 Phase 1: Incident Response Execution

This phase focuses on applying fundamental and advanced incident response (IR) techniques through interactive lab modules. Each task simulates a real-world security incident and provides hands-on experience with detection, containment, eradication, and recovery processes. Documentation of each exercise includes the use of tools, identified Indicators of Compromise (IoCs), and recommended mitigation strategies.

2.1 Incident Handling Fundamentals

Objective:
To understand the full incident response lifecycle, including the key stages: identification, detection, containment, eradication, recovery, and post-incident reporting.

Activity Summary:
I completed the **"Incident Handling"** module in the RangeForce platform. This foundational lab introduced the theoretical and practical aspects of handling incidents effectively in an enterprise environment.

Deliverable – Summary:
The incident response lifecycle is a structured approach to managing cybersecurity incidents. The **detection phase** involves identifying unusual or malicious activity through alerts and log analysis. **Containment** limits the spread of an attack by isolating affected systems. **Eradication** focuses on removing the threat from the environment, and **recovery** ensures that systems are restored to normal operations securely. Post-incident activities involve documenting lessons learned and improving response plans for future incidents.

2.2 Responding to Remote System Attacks

Objective:
To detect and respond to an unauthorized remote system attack using appropriate tools and methodologies.

Activity Summary:
I completed the **"Remote System Attacks"** module. The lab simulated a scenario where an attacker gained unauthorized access to a remote system. I was tasked with identifying the indicators of compromise and containing the threat.

Deliverable – Summary:
During the lab, I detected IoCs such as unusual login attempts, command execution via reverse shells, and suspicious network traffic. I used command-line tools and log analysis to identify the compromised host and isolate it from the network. Containment

actions included disabling affected user accounts and blocking the attacker's IP address at the firewall. A key lesson learned was the importance of monitoring remote access logs and setting up alerts for anomalous behavior.

2.3 Malware Incident Response Exercise

Objective:
To investigate and neutralize a malware infection, documenting detection methods, IoCs, and containment techniques.

Activity Summary:
I completed the **"Malware Incident Response Exercise"** module. The lab involved analyzing a compromised host suspected of malware infection and taking necessary actions to eliminate the threat.

Deliverable – Summary:
I identified malware-related IoCs such as rogue processes, registry modifications, and persistence mechanisms. Tools like process monitors and antivirus scanners were used for detection. I terminated malicious processes and removed registry entries to contain the infection. Mitigation strategies included deploying endpoint protection software, conducting system-wide scans, and ensuring regular patch management. In a corporate setting, user education and strict email filtering would help prevent such incidents.

2.4 Perimeter Compromise Simulation

Objective:
To detect and defend against a perimeter breach, followed by implementing stronger security controls.

Activity Summary:
I completed the **"Perimeter Compromise Challenge"** module. This simulated an attack on the network perimeter, where vulnerabilities in external services were exploited.

Deliverable – Summary:
The initial detection involved observing failed login attempts and port scans. I used firewall logs and intrusion detection system (IDS) alerts to identify the attack vector. The containment involved blocking suspicious IPs and tightening firewall rules. Post-event, I recommended updating perimeter device firmware, applying network segmentation, and configuring intrusion prevention systems (IPS) to monitor traffic proactively.

2.5 Independent Malware Incident Challenge

Objective:
To apply incident response skills independently in a complex malware attack scenario.

Activity Summary:
I completed the **"Malware Incident Response Challenge"** module without step-by-step guidance, simulating a real-world independent response scenario.

Deliverable – Summary:
IoCs identified included command-and-control (C2) traffic, unauthorized scheduled tasks, and modified system files. I contained the malware by isolating the system, eradicating malware artifacts, and restoring clean configurations. The main challenge was pinpointing the malware's persistence mechanism, which I overcame by analyzing startup scripts and scheduled tasks. This exercise reinforced the importance of thorough system inspection and using layered defenses.

3.0 Phase 2 – CVE Analysis and Remediation

This phase focuses on identifying, analyzing, and mitigating real-world vulnerabilities (CVEs) affecting widely used software and operating systems. Each module highlights a specific vulnerability, its impact, and appropriate remediation techniques. Through hands-on exercises, I learned how to assess risks, detect exploits, and implement long-term mitigation strategies in alignment with security best practices.

3.1 Overview of CVEs

Objective:
To understand the purpose and structure of CVEs and the importance of CVSS scores in assessing vulnerability severity.

Activity Summary:
I completed the **"CVE Overview"** module in RangeForce. The module explained how CVEs are documented, categorized, and scored using the Common Vulnerability Scoring System (CVSS).

Deliverable – Summary:
The CVSS rating system evaluates vulnerabilities based on several factors, including access complexity, required privileges, and potential impact on confidentiality, integrity, and availability. Scores range from 0.0 (low severity) to 10.0 (critical). High and critical vulnerabilities require immediate attention as they can be exploited remotely or lead to system compromise without user interaction.

3.2 WinRAR Vulnerability – CVE-2023-38831

Objective:
To analyze and mitigate a critical vulnerability in WinRAR that allows arbitrary code execution.

Activity Summary:
I completed the **"CVE-2023-38831 WinRAR - Arbitrary Code Execution"** module. This vulnerability allows attackers to craft malicious archive files that execute hidden payloads when opened.

Deliverable – Summary:
I analyzed the exploit by inspecting the behavior of specially crafted .RAR files. The detection involved monitoring file execution paths and identifying embedded scripts. Mitigation steps included updating WinRAR to the latest patched version and using endpoint protection tools to flag suspicious archive behavior. As a preventive measure,

users should avoid opening unknown archive files and administrators should enforce file-type filtering on email gateways.

3.3 Windows Search Exploit – CVE-2023-36884

Objective:
To investigate and remediate a remote code execution vulnerability related to Windows Search functionality.

Activity Summary:
I completed the **"CVE-2023-36884 Windows Search - Remote Code Execution"** module. The lab demonstrated how a specially crafted shortcut or document could trigger execution of malicious code via the Windows Search protocol.

Deliverable – Summary:
Detection was performed by analyzing the use of search-ms URI handlers in document files. The mitigation strategy involved disabling vulnerable protocol handlers via Group Policy and applying Microsoft's security patch. To prevent future exploitation, it's essential to monitor for abnormal usage of URI schemes and restrict user permissions on endpoint devices.

3.4 Log4Shell Vulnerability – CVE-2021-44228

Objective:
To investigate the Log4Shell vulnerability affecting Java-based applications and deploy mitigation strategies.

Activity Summary:
I completed the **"CVE-2021-44228 Log4Shell"** module, which simulated the exploitation of a logging library to perform remote code execution (RCE).

Deliverable – Summary:
The detection involved examining logs for suspicious JNDI strings and LDAP callbacks.

I mitigated the risk by updating the Log4j library to a secure version and removing vulnerable classes. To prevent re-exploitation, I recommended continuous vulnerability scanning, proper input validation in applications, and using a Web Application Firewall (WAF) to block exploit attempts.

3.5 PrintNightmare Exploit – CVE-2021-34527

Objective:
To analyze and remediate a vulnerability in the Windows Print Spooler service that enables privilege escalation.

Activity Summary:
I completed the **"CVE-2021-34527 PrintNightmare"** module, where the lab simulated a local privilege escalation scenario through the print spooler.

Deliverable – Summary:
I detected exploit attempts by monitoring for abnormal spooler activity and privilege changes. Containment involved disabling the print spooler service where unnecessary and applying security patches from Microsoft. I also recommended implementing strict privilege management and auditing all print-related services in the environment.

3.6 Detecting Ransomware Variants (REvil)

Objective:
To detect and respond to ransomware activities, focusing on REvil variants.

Activity Summary:
I completed the **"REvil Detection and Response"** module, which provided insights into detecting and responding to ransomware behavior.

Deliverable – Summary:
Detection techniques included monitoring file encryption patterns, command-line activity, and creation of ransom notes. I practiced isolating infected systems and

recovering data from backups. For proactive defense, I suggested implementing real-time endpoint monitoring, network segmentation, and frequent user training on phishing awareness.

4.0 Phase 3 – Integrated Incident Simulation and Reporting

This phase integrates all previously learned incident response (IR) and vulnerability management concepts into a comprehensive incident simulation. The goal is to demonstrate a coordinated response to a multi-vector attack affecting both the network perimeter and internal applications. It also includes a strategic review of vulnerabilities (CVEs) with long-term defense recommendations.

4.1 Comprehensive Incident Simulation

Objective:
To simulate a coordinated response to a multi-faceted cyber attack, applying the full incident response lifecycle—detection, containment, eradication, and recovery—across affected systems and services.

Activity Summary:
Using insights from prior modules, I simulated a complex attack involving both network intrusion and application-layer exploitation. The attack vectors included unauthorized remote access, exploitation of known CVEs (e.g., Log4Shell and PrintNightmare), and lateral movement to compromise internal systems.

Deliverable – Simulated Incident Report:

- **Detection:**
  Initial signs of compromise were detected via abnormal network traffic and user behavior anomalies. IDS alerts and log correlation revealed access attempts on

exposed web services and unusual PowerShell activity.

- **Containment:**
  The compromised systems were isolated from the network. Access controls were tightened, malicious accounts were disabled, and suspicious IP addresses were blocked at the firewall.

- **Eradication:**
  Malware components were removed, vulnerable software was updated or replaced, and persistence mechanisms (e.g., scheduled tasks, registry modifications) were eliminated.

- **Recovery:**
  Clean system backups were restored, user passwords were reset, and services were brought back online in a phased manner after verification. Logs were monitored post-recovery for reinfection attempts.

- **Lessons Learned:**
  The simulation highlighted the importance of early detection, centralized logging, timely patching, and effective containment policies. A coordinated response across IT, security, and incident response teams is essential to minimize damage and downtime.

- **Recommendations for Enhancing Organizational Defenses:**

    - Implement automated alerting for critical vulnerabilities.
    - Apply Zero Trust principles across internal networks.
    - Conduct regular security awareness training and red team exercises.
    - Maintain a tested and documented incident response plan.

4.2 Final CVE Review and Proactive Measures

Objective:
To reflect on the CVEs addressed during the project and develop proactive strategies for long-term security posture improvement.

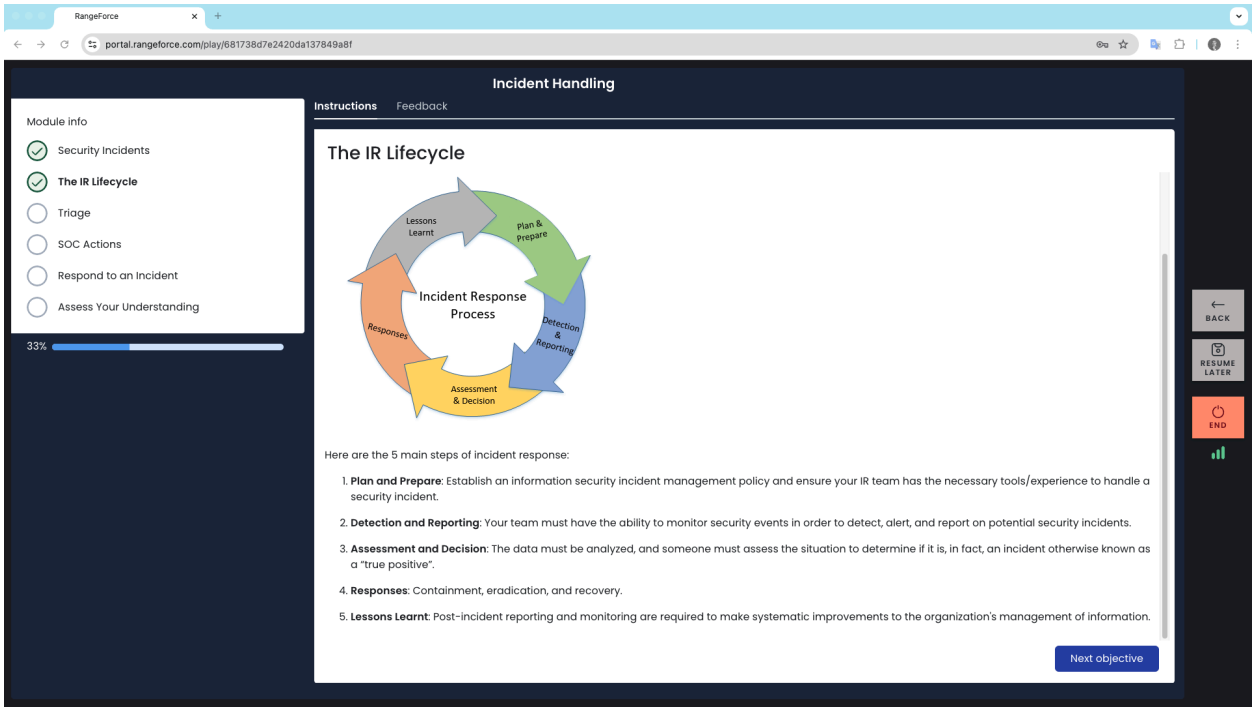
Deliverable – CVE Defense Summary:

CVE	Vulnerability Type	Mitigation Steps	Proactive Measures
CVE-2023-38831 (WinRAR)	Arbitrary Code Execution	Updated WinRAR, restricted archive file execution	Educate users on safe file handling, use sandbox environments
CVE-2023-36884 (Windows Search)	Remote Code Execution	Disabled vulnerable protocols, applied patches	Audit protocol usage, restrict URI handling via GPO
CVE-2021-44228 (Log4Shell)	Remote Code Execution	Upgraded Log4j, monitored logs for JNDI payloads	Regular dependency scanning, input sanitization
CVE-2021-34527 (PrintNightmare)	Privilege Escalation	Disabled Print Spooler where unnecessary, patched systems	Least privilege enforcement, restrict service usage
REvil Ransomware	Ransomware	Detected via behavior analysis, isolated system, restored from backup	Endpoint detection & response (EDR), frequent backups, phishing awareness

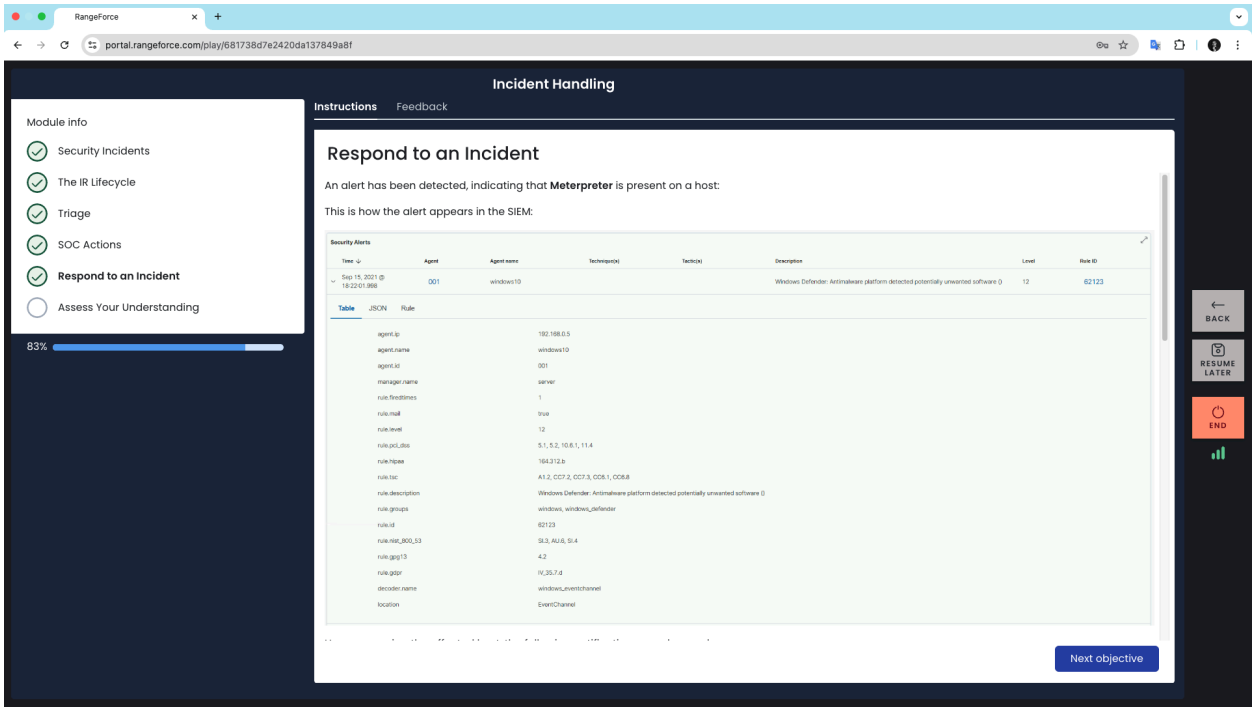
5.0 Lab Completion Screenshots: Proof of Completion

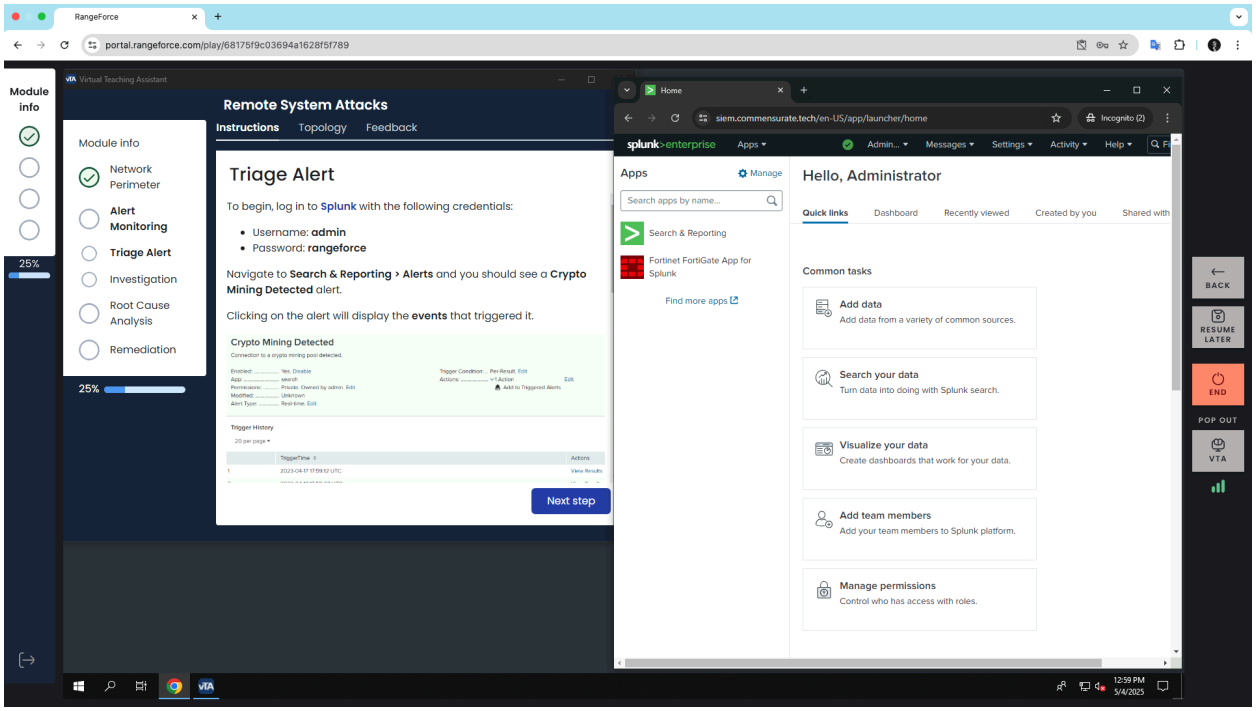
This section includes screenshots captured during the completion of each lab exercise as evidence of active participation, successful execution, and critical milestones achieved throughout the capstone project. Each screenshot reflects either the final step of a module, key configurations made, indicators of compromise (IoCs) identified, or successful mitigation applied.

5.1 Incident Response

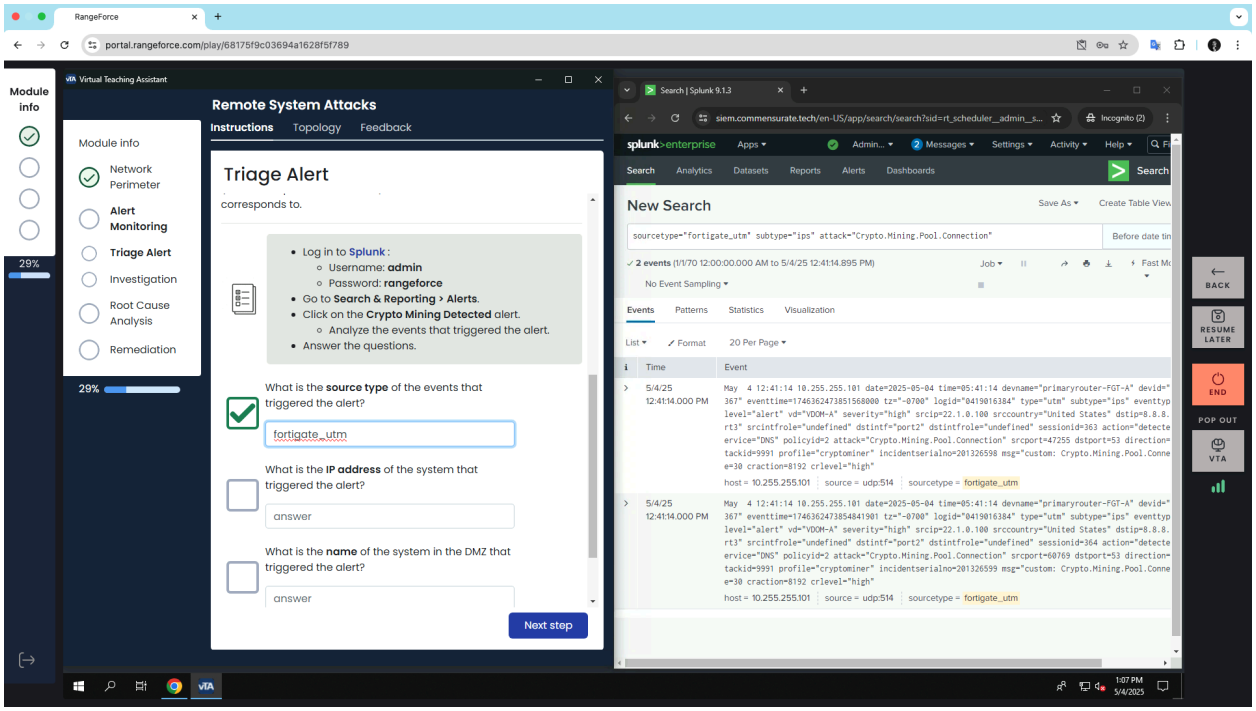


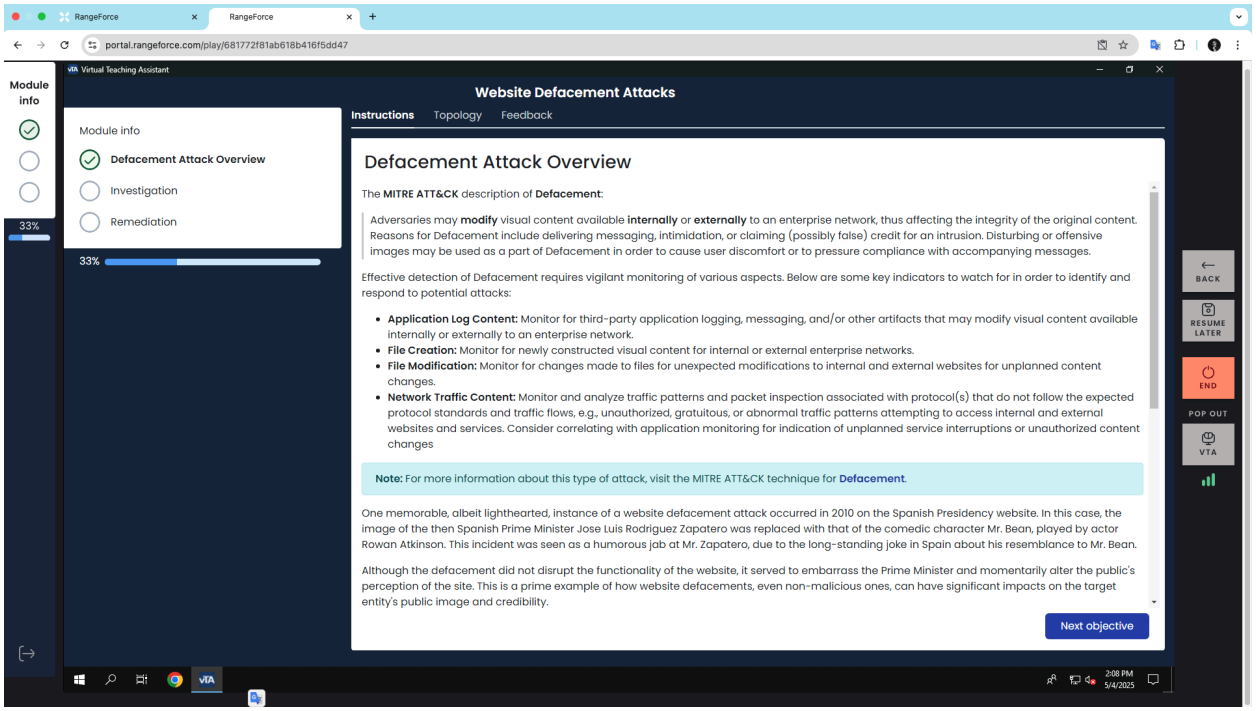
EncryptEdge Labs

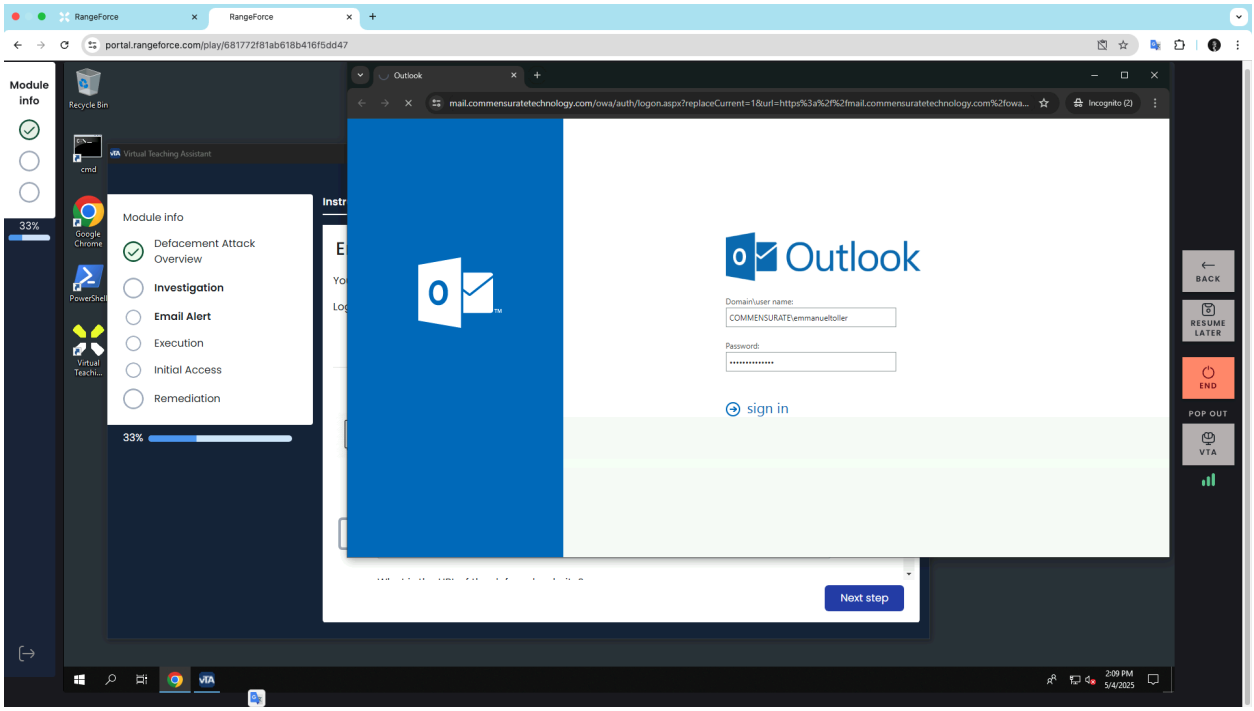


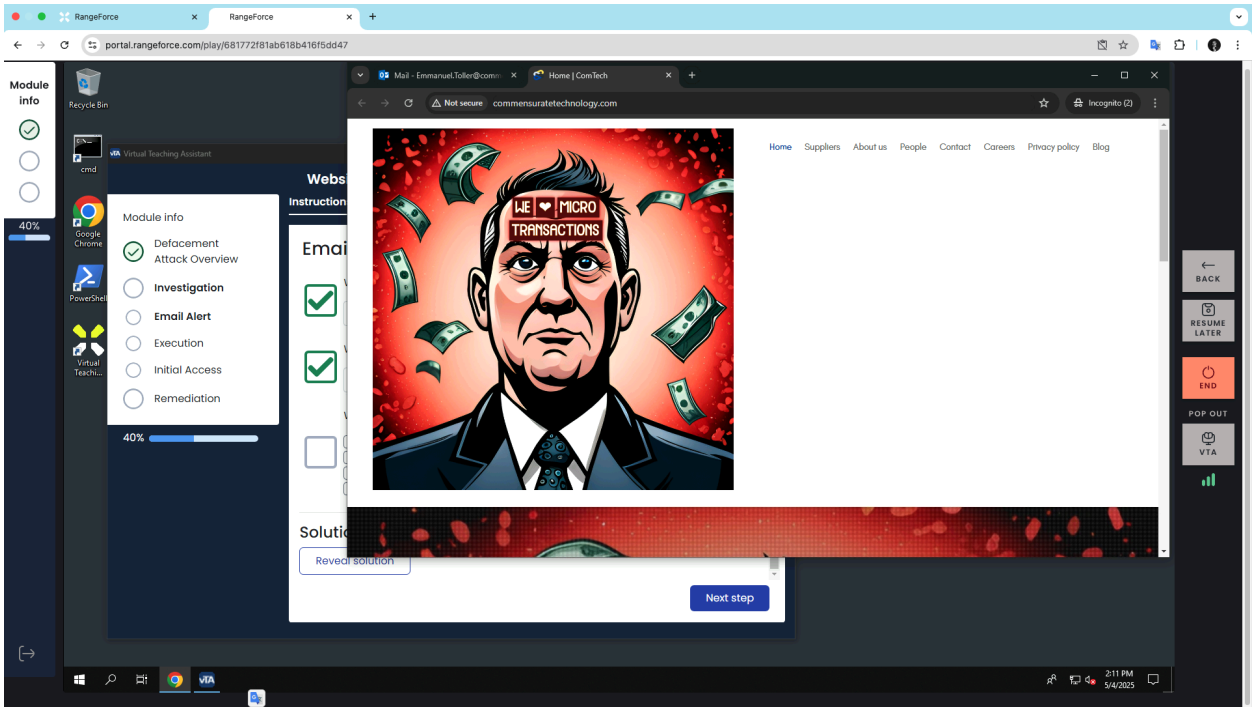


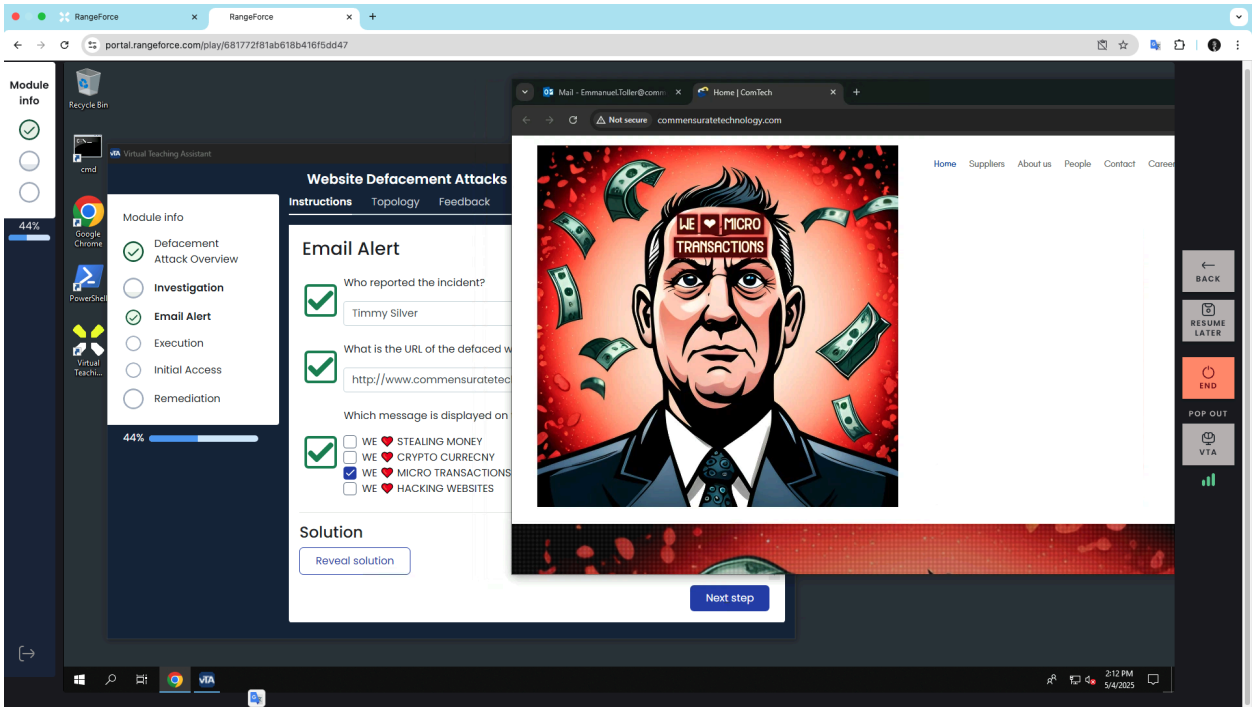
EncryptEdge Labs

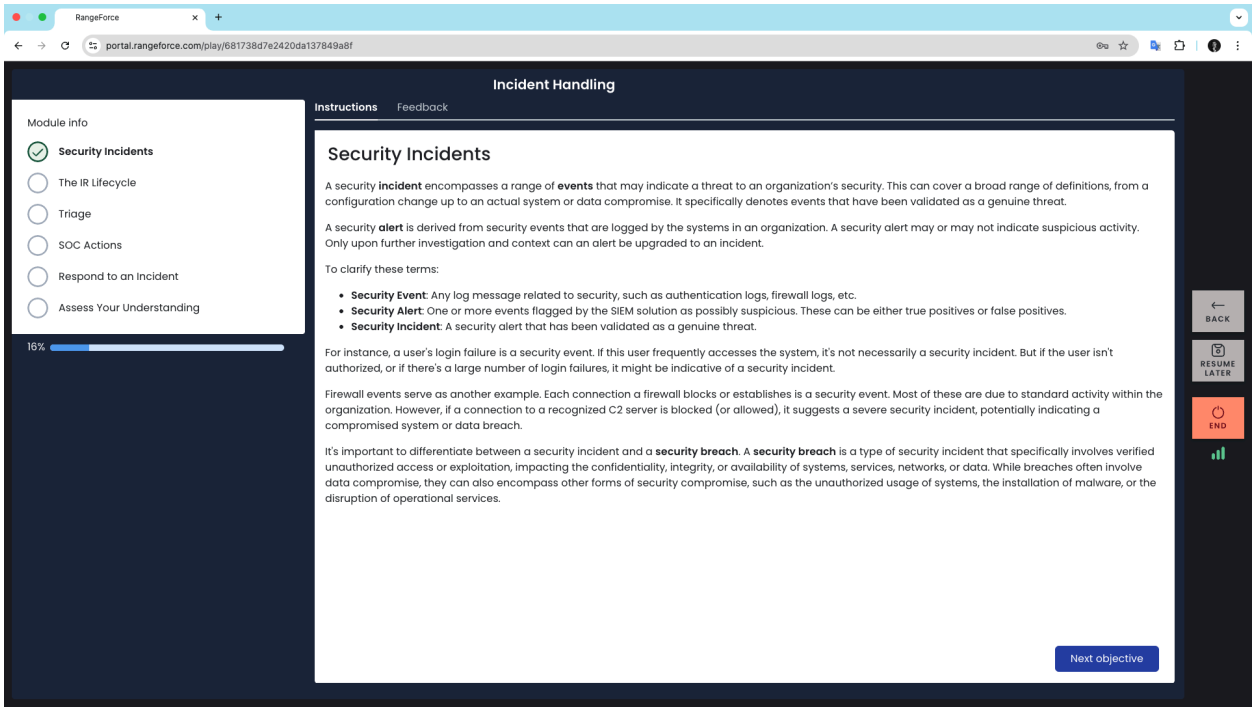




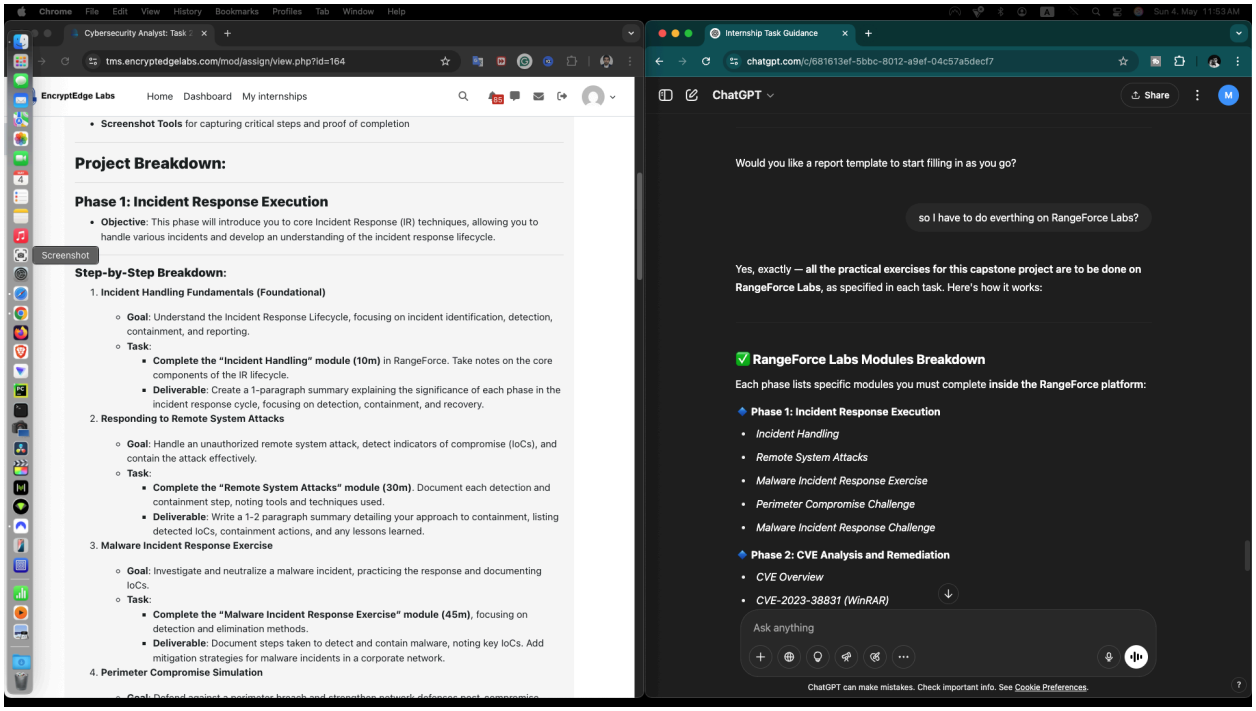

RangeForce
portal.rangeforce.com/play/681772f81ab618b416f5dd47
Module info
Defacement Attack Overview
Investigation
Email Alert
Execution
Initial Access
Remediation
33%
Outlook
mail.commensuratetechnology.com/owa/auth/logon.aspx?replaceCurrent=1&url=https%3a%2f%2fmail.commensuratetechnology.com%2fowa...
Domain\user name:
COMMENSURATE\emmanueltoller
Password:
sign in
Next step
BACK
RESUME LATER
END
POP OUT
VTA

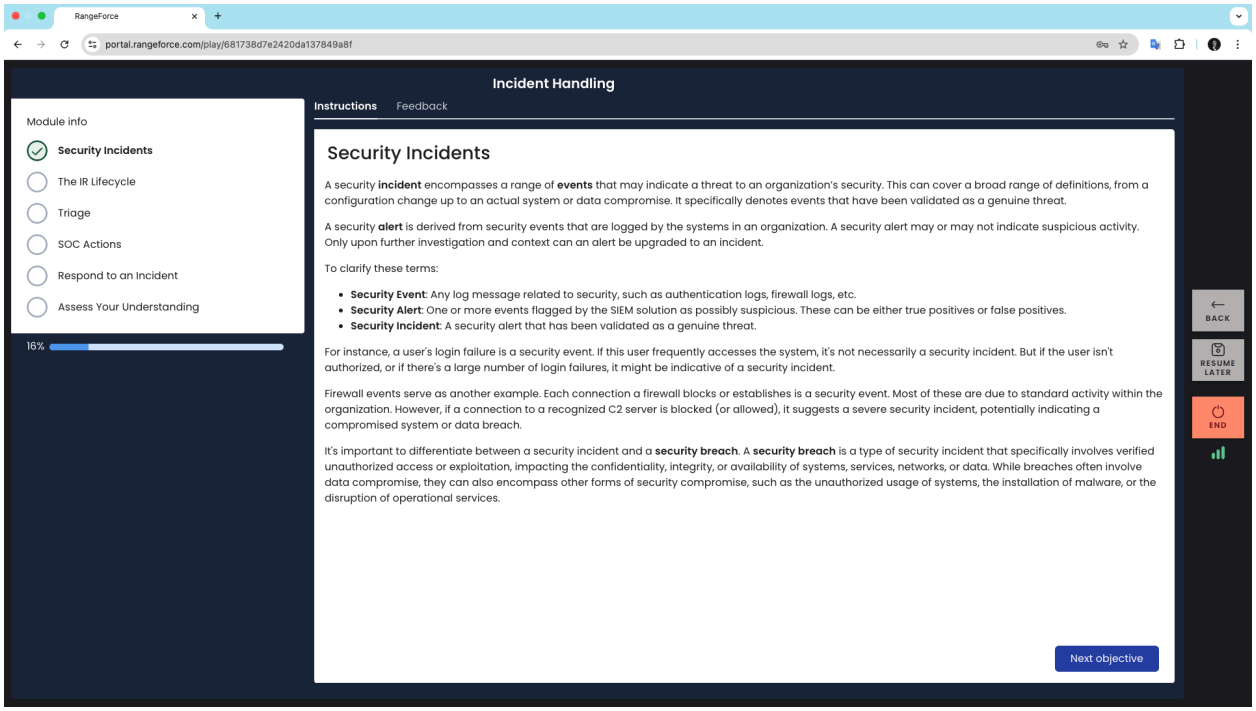

RangeForce
portal.rangeforce.com/play/681772f81ab618b416f5dd47
Module info
Defacement Attack Overview
Investigation
Email Alert
Execution
Initial Access
Remediation
40%
commensuratetechnology.com
Home
Suppliers
About us
People
Contact
Careers
Privacy policy
Blog
WE ❤ MICRO TRANSACTIONS
Reveal solution
Next step
BACK
RESUME LATER
END
POP OUT
VTA

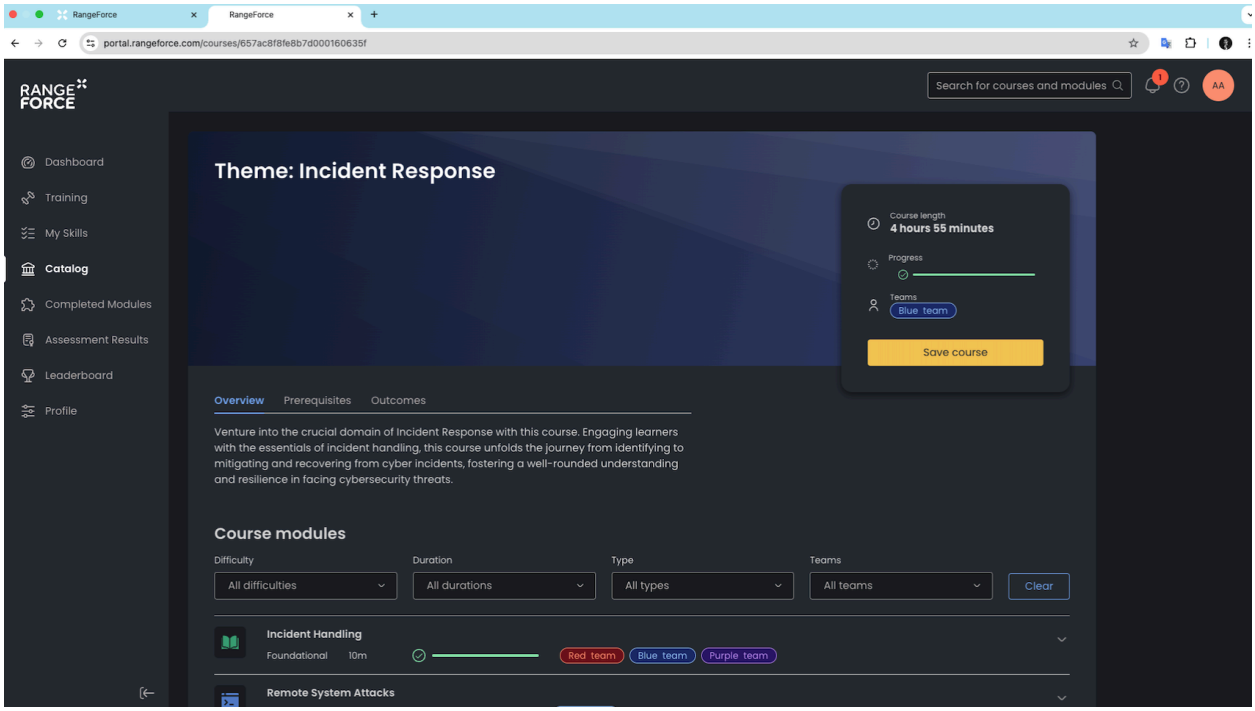

RangeForce
portal.rangeforce.com/play/681772f81ab618b416f5dd47
Module info
Website Defacement Attacks
Instructions
Topology
Feedback
Module info
Defacement Attack Overview
Investigation
Email Alert
Execution
Initial Access
Remediation
44%
Email Alert
Who reported the incident?
Timmy Silver
What is the URL of the defaced w
http://www.commensuratetec
Which message is displayed on
WE ❤ STEALING MONEY
WE ❤ CRYPTO CURRECNY
WE ❤ MICRO TRANSACTIONS
WE ❤ HACKING WEBSITES
Solution
Reveal solution
Next step
Mail - EmmanuelToller@comm
Home | ComTech
Not secure
commensuratetechnology.com
Home
Suppliers
About us
People
Contact
WE ❤ MICRO TRANSACTIONS
BACK
RESUME LATER
END
POP OUT
VTA


RangeForce
portal.rangeforce.com/play/681738d7e2420da137849a8f
Incident Handling
Instructions
Feedback
Module info
Security Incidents
The IR Lifecycle
Triage
SOC Actions
Respond to an Incident
Assess Your Understanding
16%
Security Incidents
A security incident encompasses a range of events that may indicate a threat to an organization's security. This can cover a broad range of definitions, from a configuration change up to an actual system or data compromise. It specifically denotes events that have been validated as a genuine threat.
A security alert is derived from security events that are logged by the systems in an organization. A security alert may or may not indicate suspicious activity. Only upon further investigation and context can an alert be upgraded to an incident.
To clarify these terms:
Security Event: Any log message related to security, such as authentication logs, firewall logs, etc.
Security Alert: One or more events flagged by the SIEM solution as possibly suspicious. These can be either true positives or false positives.
Security Incident: A security alert that has been validated as a genuine threat.
For instance, a user's login failure is a security event. If this user frequently accesses the system, it's not necessarily a security incident. But if the user isn't authorized, or if there's a large number of login failures, it might be indicative of a security incident.
Firewall events serve as another example. Each connection a firewall blocks or establishes is a security event. Most of these are due to standard activity within the organization. However, if a connection to a recognized C2 server is blocked (or allowed), it suggests a severe security incident, potentially indicating a compromised system or data breach.
It's important to differentiate between a security incident and a security breach. A security breach is a type of security incident that specifically involves verified unauthorized access or exploitation, impacting the confidentiality, integrity, or availability of systems, services, networks, or data. While breaches often involve data compromise, they can also encompass other forms of security compromise, such as the unauthorized usage of systems, the installation of malware, or the disruption of operational services.
Next objective
BACK
RESUME LATER
END

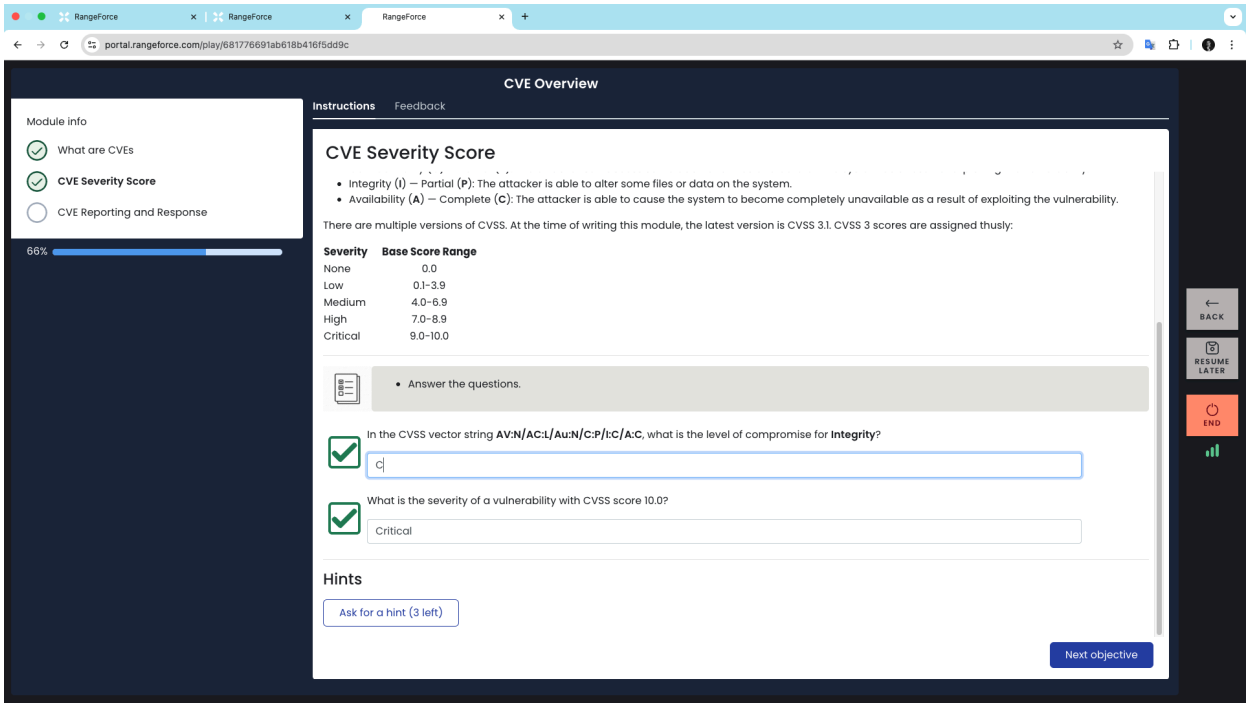
EncryptEdge Labs



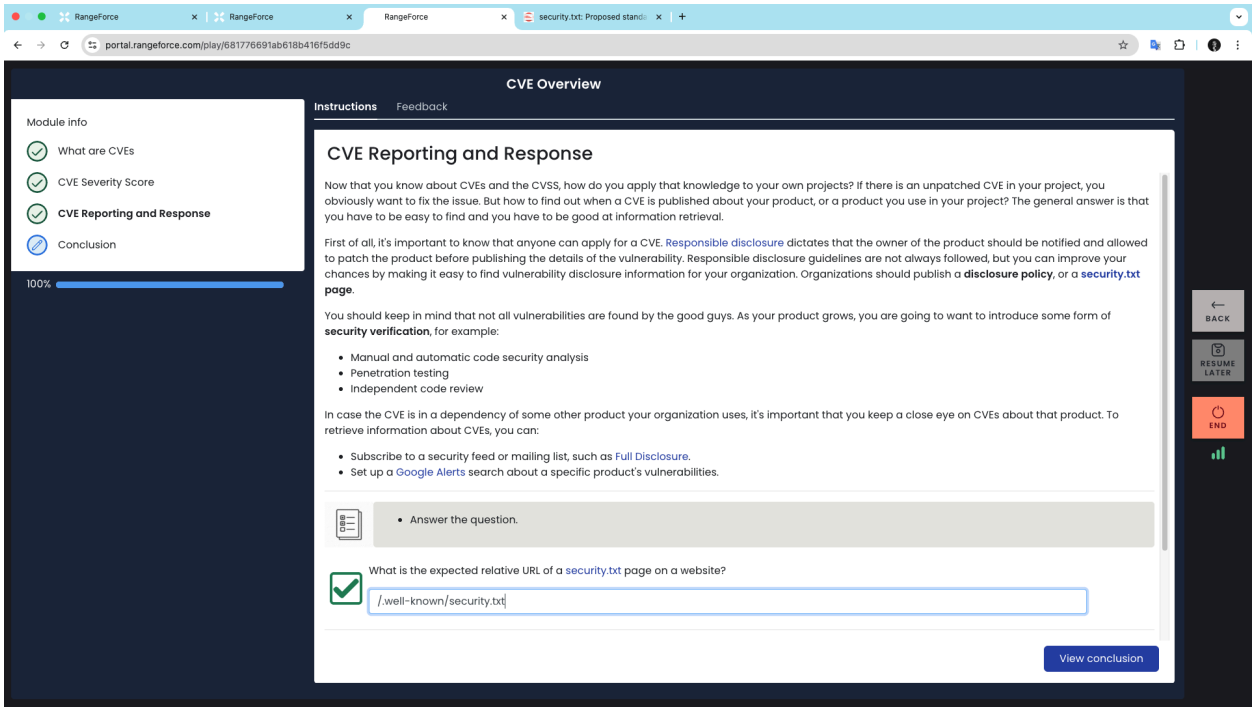


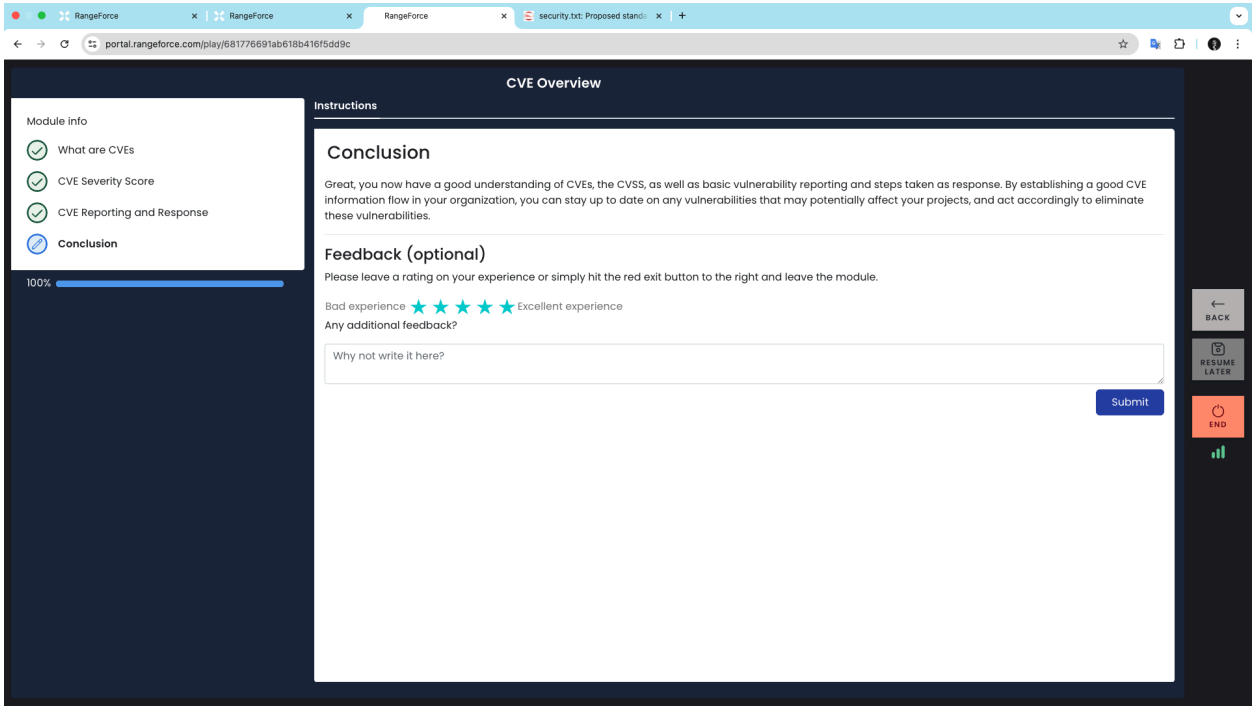


5.2 CVEs and Emerging Threats

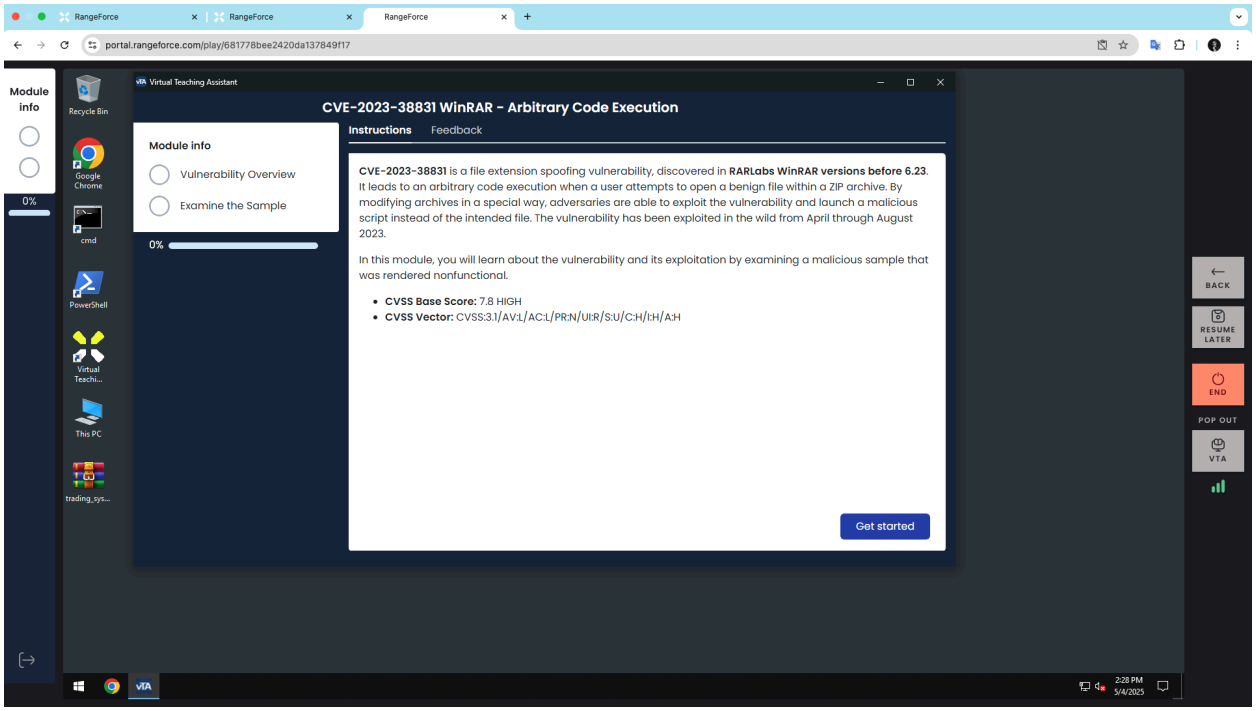


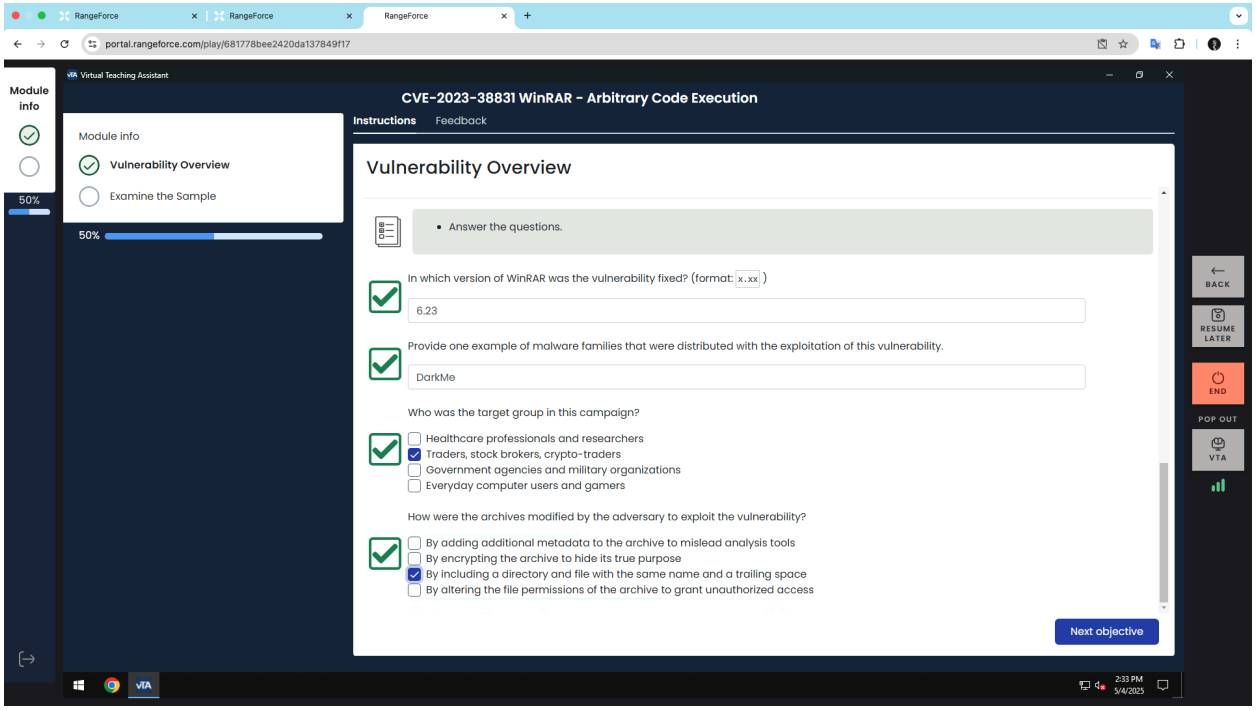

EncryptEdge Labs


RangeForce
RangeForce
RangeForce
security.txt: Proposed standa
portal.rangeforce.com/play/681776691ab618b416f5dd9c
CVE Overview
Module info
What are CVEs
CVE Severity Score
CVE Reporting and Response
Conclusion
100%
Instructions
Feedback
CVE Reporting and Response
Now that you know about CVEs and the CVSS, how do you apply that knowledge to your own projects? If there is an unpatched CVE in your project, you obviously want to fix the issue. But how to find out when a CVE is published about your product, or a product you use in your project? The general answer is that you have to be easy to find and you have to be good at information retrieval.
First of all, it's important to know that anyone can apply for a CVE. Responsible disclosure dictates that the owner of the product should be notified and allowed to patch the product before publishing the details of the vulnerability. Responsible disclosure guidelines are not always followed, but you can improve your chances by making it easy to find vulnerability disclosure information for your organization. Organizations should publish a disclosure policy, or a security.txt page.
You should keep in mind that not all vulnerabilities are found by the good guys. As your product grows, you are going to want to introduce some form of security verification, for example:
Manual and automatic code security analysis
Penetration testing
Independent code review
In case the CVE is in a dependency of some other product your organization uses, it's important that you keep a close eye on CVEs about that product. To retrieve information about CVEs, you can:
Subscribe to a security feed or mailing list, such as Full Disclosure.
Set up a Google Alerts search about a specific product's vulnerabilities.
Answer the question.
What is the expected relative URL of a security.txt page on a website?
/.well-known/security.txt
View conclusion
BACK
RESUME LATER
END

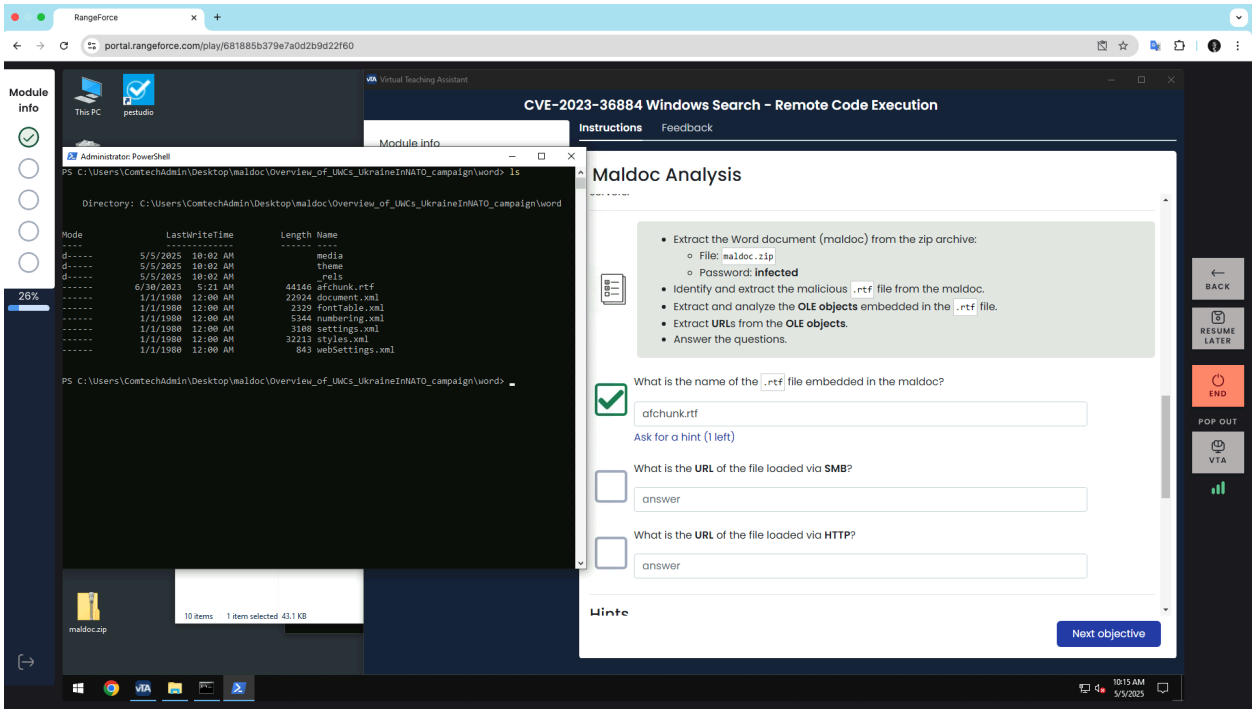

RangeForce
RangeForce
RangeForce
security.txt: Proposed standa
portal.rangeforce.com/play/681776691ab618b416f5dd9c
CVE Overview
Module info
What are CVEs
CVE Severity Score
CVE Reporting and Response
Conclusion
100%
Instructions
Conclusion
Great, you now have a good understanding of CVEs, the CVSS, as well as basic vulnerability reporting and steps taken as response. By establishing a good CVE information flow in your organization, you can stay up to date on any vulnerabilities that may potentially affect your projects, and act accordingly to eliminate these vulnerabilities.
Feedback (optional)
Please leave a rating on your experience or simply hit the red exit button to the right and leave the module.
Bad experience
Excellent experience
Any additional feedback?
Why not write it here?
Submit
BACK
RESUME LATER
END

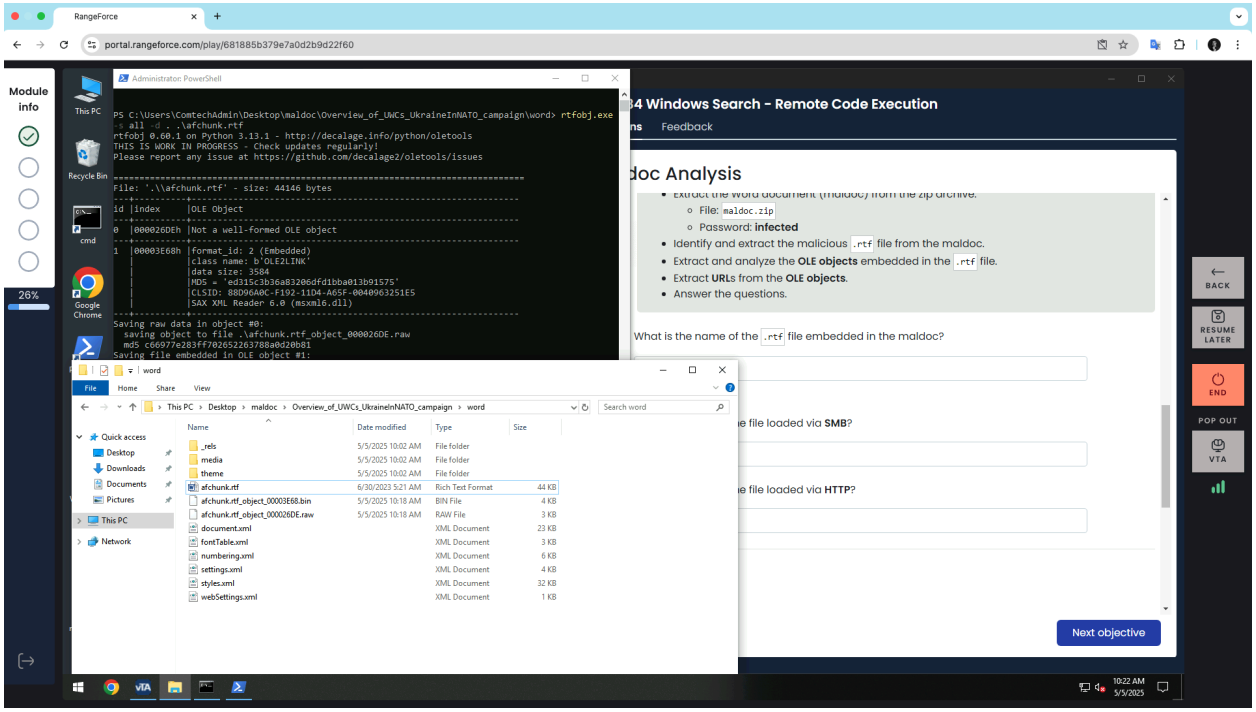
EncryptEdge Labs

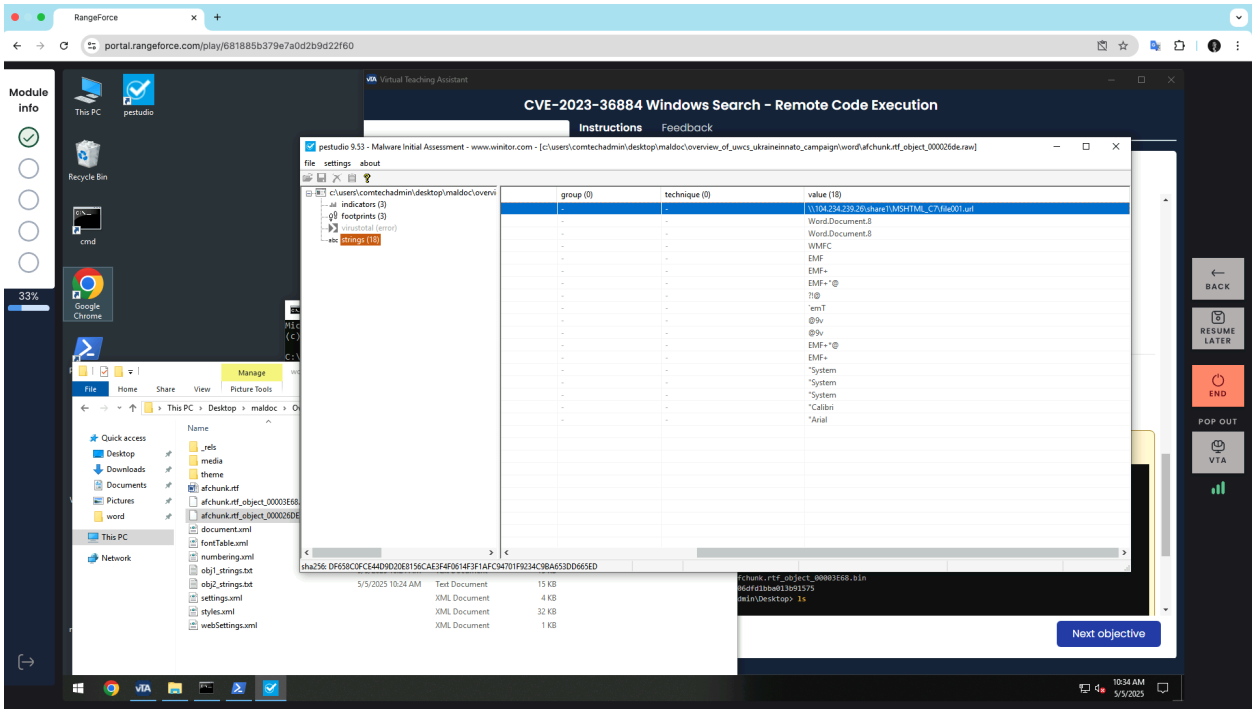


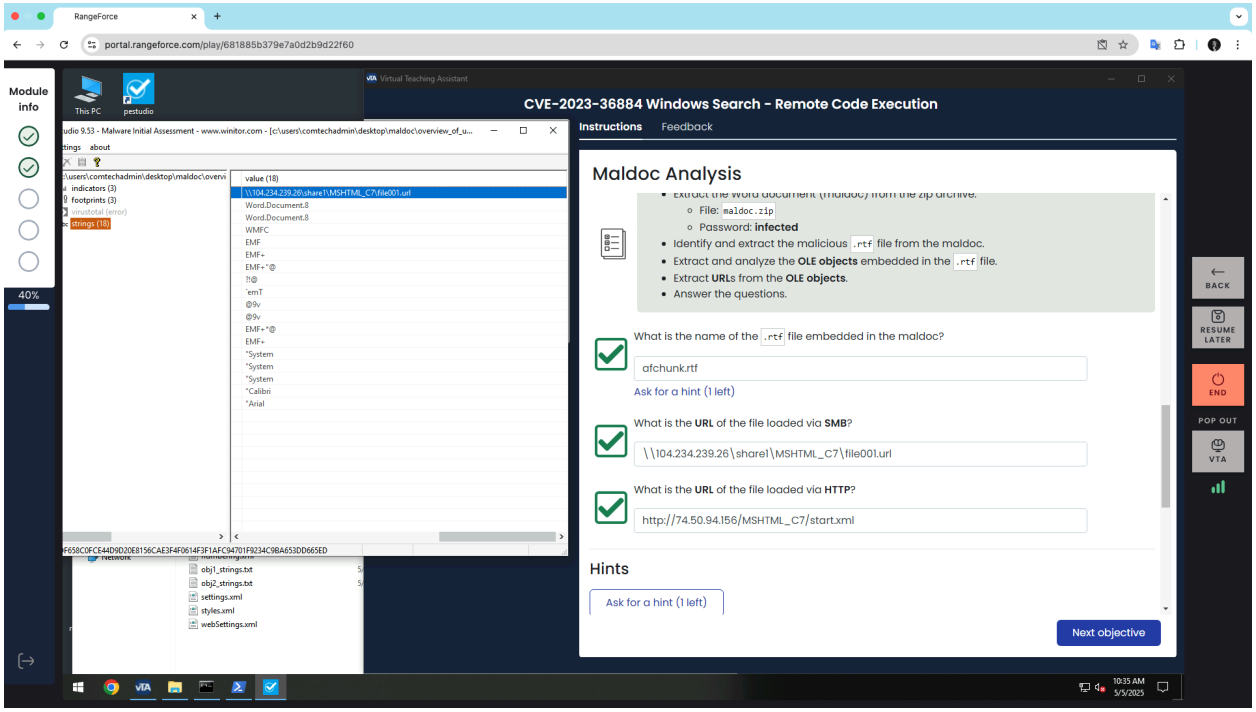


EncryptEdge Labs

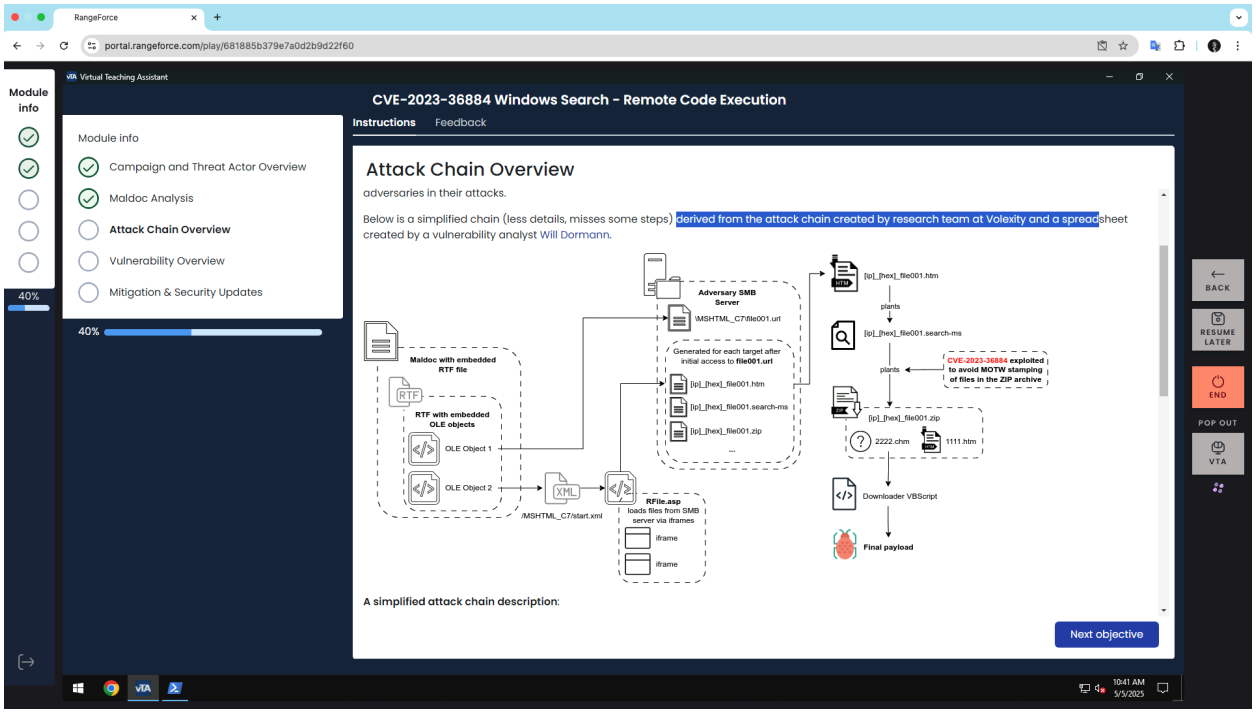


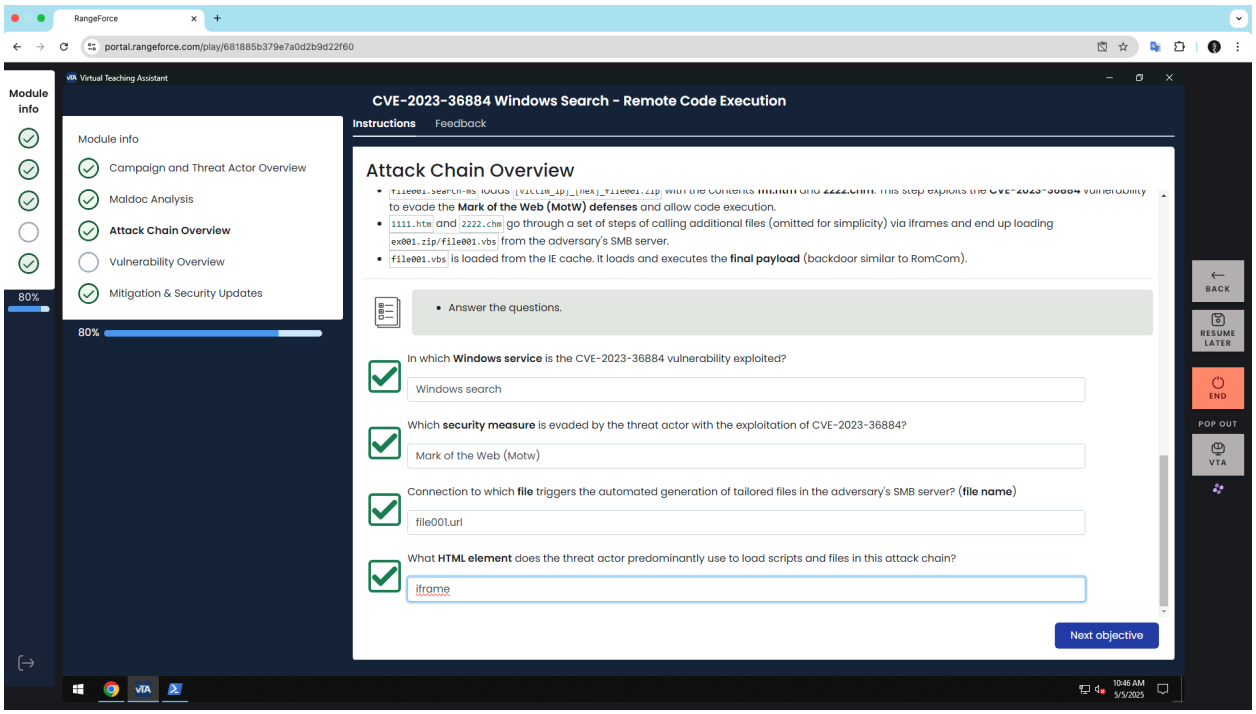



RangeForce
portal.rangeforce.com/play/681885b379e7a0d2b9d22f60
Module info
33%
CVE-2023-36884 Windows Search - Remote Code Execution
Instructions
Feedback
Next objective
BACK
RESUME LATER
END
POP OUT
VTA

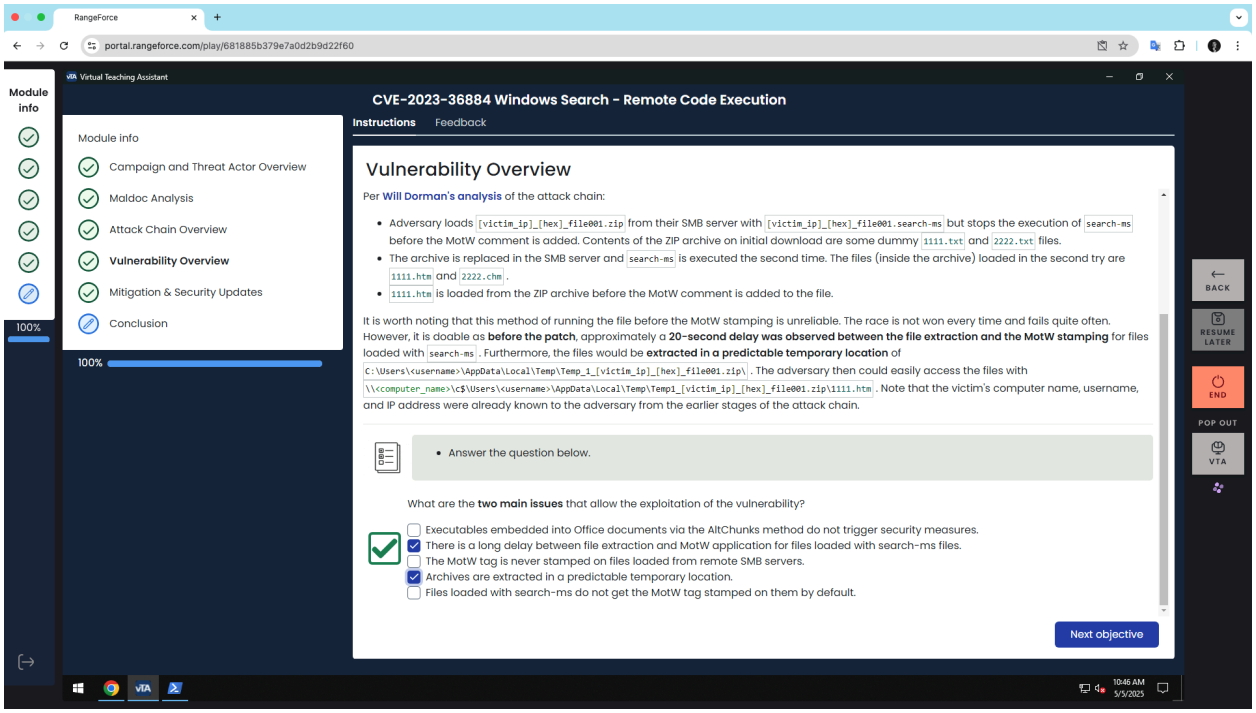

RangeForce
portal.rangeforce.com/play/681885b379e7a0d2b9d22f60
Module info
40%
CVE-2023-36884 Windows Search - Remote Code Execution
Instructions
Feedback
Maldoc Analysis
File: maldoc.zip
Password: infected
Identify and extract the malicious .rtf file from the maldoc.
Extract and analyze the OLE objects embedded in the .rtf file.
Extract URLs from the OLE objects.
Answer the questions.
What is the name of the .rtf file embedded in the maldoc?
afchunk.rtf
Ask for a hint (1 left)
What is the URL of the file loaded via SMB?
\\104.234.239.26\share1\MSHTML_C7\file001.url
What is the URL of the file loaded via HTTP?
http://74.50.94.156/MSHTML_C7/start.xml
Hints
Ask for a hint (1 left)
Next objective
BACK
RESUME LATER
END
POP OUT
VTA

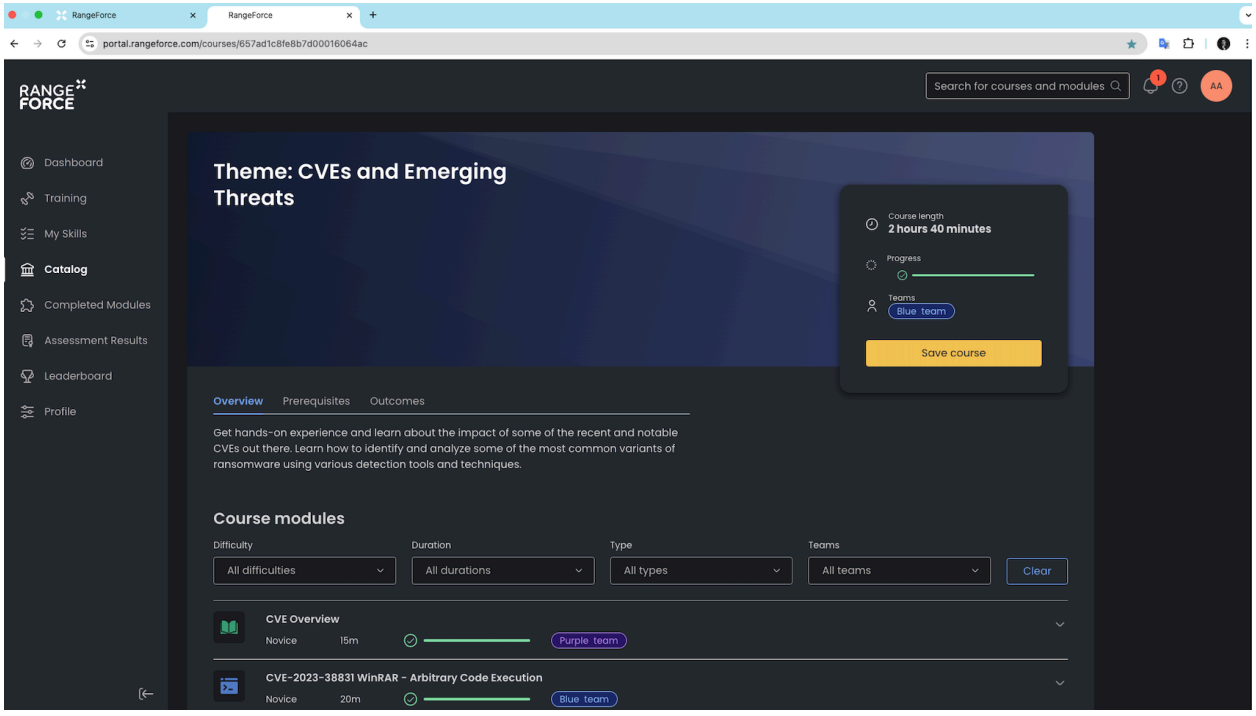
EncryptEdge Labs





EncryptEdge Labs





EncryptEdge Labs

This Internship Task report was developed on [May, 05, 2025]

By:

atalmamun@gmail.com