Justin Goncalves
8/28/24

Commonwealth Bank Cybersecurity Program

Table of Contents

Program Overview:

Welcome to the Commonwealth Bank Introduction to Cybersecurity Virtual Job Simulation! We are so excited to have you here!

We're building a courageous, trusting culture and supporting growth at pace. We believe Cyber is everyone's business and invest in our people and communities to develop global Cyber capabilities. We're part of Technology and are responsible for the world-leading applications across every single aspect of CommBank – from innovative product platforms for our customers, to essential tools within our business.

During this program, you will get the opportunity to step into the shoes of a Commonwealth Bank Cybersecurity team member and complete tasks that replicate the work that our Cybersecurity team does every day. You'll learn how to create and visually represent a reliable dataset, present your findings to stakeholders and respond to an incident as a cybersecurity analyst.

We hope this program provides a great resource for you to up-skill and strengthen your resume as you explore career options and a potential career at Commonwealth Bank!
Skills you will learn and practice: Splunk Basics, Data Analysis, Data Visualization, Incident Triage, Detection and Response, Data Protection, Password Security, Compliance Knowledge, Penetration Testing

Task One: Data Analysis
Analyse and visualise cyber data to uncover trends and patterns
What you'll learn
- Gain hands-on experience in data analysis, especially in the context of cyber data.
- Understand the importance of data visualization in uncovering trends and patterns.
- Learn to use Splunk, a data analysis tool, for creating dashboards and visual representations of data.
- Develop skills in interpreting data fields and identifying key insights.

What you'll do
- Install and set up Splunk Enterprise, import the provided dataset and explore it using Splunk.
- Create a comprehensive dashboard with charts and tables to visualize fraud-related data for submission.

Task Two: Incident Response
Mitigate cyber incidents by analyzing, responding, and recovering for your organization.
What you'll learn
- Understand the nature of a cyber incident based on provided incident timeline and descriptions.
- Learn about various types of cyber attacks and their characteristics.
- Understand the steps involved in incident response and recovery.

What you'll do
- Identify the type of cyber attack that occurred based on the incident details.
- Outline the next steps to be taken as a cyber security analyst, including containment, resolution, and recovery measures.
- Provide a list of actions to contain, resolve, and recover from the incident.
- Describe post-incident activities and measures to prevent similar incidents in the future.

Task Three: Security Awareness
Secure the bank, one password at a time.
What you'll learn
- Identify the type of cyber attack that occurred based on the incident details.
- Outline the next steps to be taken as a cyber security analyst, including containment, resolution, and recovery measures.
- Provide a list of actions to contain, resolve, and recover from the incident.
- Describe post-incident activities and measures to prevent similar incidents in the future.

What you'll do
- Research ACSC's best practices for secure passwords.
- Design a visually clear and appealing infographic using suitable software (e.g., Canva).
- Create an easy-to-understand PDF infographic for fellow employees, emphasizing password security.

Task Four: Penetration Testing
Uncover and exploit web application weaknesses: a comprehensive penetration testing report
What you'll learn
- Gain understanding of penetration testing principles and techniques.
- Learn to identify and exploit vulnerabilities in web applications.
- Understand the importance of creating a comprehensive penetration testing report.

What you'll do
- Create an account on HackThisSite and complete all 11 levels of the "Basic" web challenge.
- Document a Penetration Testing Report including an executive summary, scope, vulnerability descriptions, key findings, and security recommendations for each level.
- Apply the knowledge gained from the challenge to real-world scenarios and improve penetration testing skills.

Task 1: Data Analysis

Review the scenario below. Then complete the tasks and activites.

Scenario

As a cyber security generalist at Commonwealth Bank, it is important to be aware of the increasing rate and complexity of financial fraud and the need for effective defence solutions. Financial fraud poses a significant challenge for financial institutions, and it is important for Commonwealth Bank to stay up to date with the latest fraud detection technologies and strategies to minimise risk. Protecting against and responding to fraud is a major responsibility for you and your team. By detecting and stopping fraud, the bank can protect its customers, employees and reputation while also enhancing the resilience of its financial system.

To help with this task, you will be using a tool called Splunk to visually represent the given data. Representing data in a visual format, also known as data visualisation, makes it easier for the data analytics team to understand and gain insights. Visual data is a universal, fast and effective way to communicate information.

You will be building a dashboard to make it easier to identify patterns and trends in the given dataset. The dashboard will provide crucial reporting and metrics information that can aid in identifying and detecting fraud. By using this dashboard, the team will be able to quickly identify any suspicious activity and take the necessary steps to prevent fraud from occurring. Overall, the goal of this task is to use data visualisation and a dashboard to make it easier to detect fraud and protect Commonwealth Bank and its customers from financial loss.

About the dataset

Data was collected and structured by the Fraud team. This dataset consists of payments from various customers made in different periods and amounts.

The feature columns include:

- **Step**: This feature represents the month from the start of the simulation. The steps represent four months that the simulation ran virtually.
    - 0: May
    - 1: June
    - 2: July
    - 3: August
- **Customer**: Customer ID
- **Age**: Categorised age
    - 0.0: <= 18
    - 1.0: 19 - 25
    - 2.0: 26 - 35
    - 3.0: 36 - 45
    - 4.0: 46 - 55
    - 5.0: 56 - 65
- **Gender**: Gender of the customer
    - F: Female
    - M: Male
- **PostcodeOrigin**: The postcode of origin/source.
- **Merchant**: The merchant's ID.

- **Category**: Category of the purchase.
- **Amount**: Amount of the purchase.
- **Fraud**: Target variable that shows if the transaction is fraudulent - 1 or non-fraudulent - 0.

Note: Dataset was randomly generated and created for your virtual experience.

Here is your task

Data was collected and structured by the Fraud team. This dataset consists of payments from various customers made in different periods and amounts.

Your first task is to analyze and visualize the data you have prepared in a software tool called Splunk.

Here are the steps you need to take:

1. Download the 60-day free trial of Splunk Enterprise. (The link is provided in the Resources section below.)
2. Install Splunk Enterprise on your computer.
3. Using the "prepared_data" file in the Resources section, import this file into Splunk.
4. Study the file using the "Interesting Fields" section in Splunk. This tells you about the data you're using.
5. Create a dashboard to include the following charts/tables:
    a. Count by Category, Fraudulent transactions, Age and Merchant.
    b. Fraud detected by Age, Category, Step (month) and Gender.
    c. Which gender performed the most fraudulent activities and in what category?
    d. Which age group performed the most fraudulent activities and to what merchant?
6. Export the dashboard as a PDF and upload it below as your submission for this task.

SIEM Data Analysis and Visualization

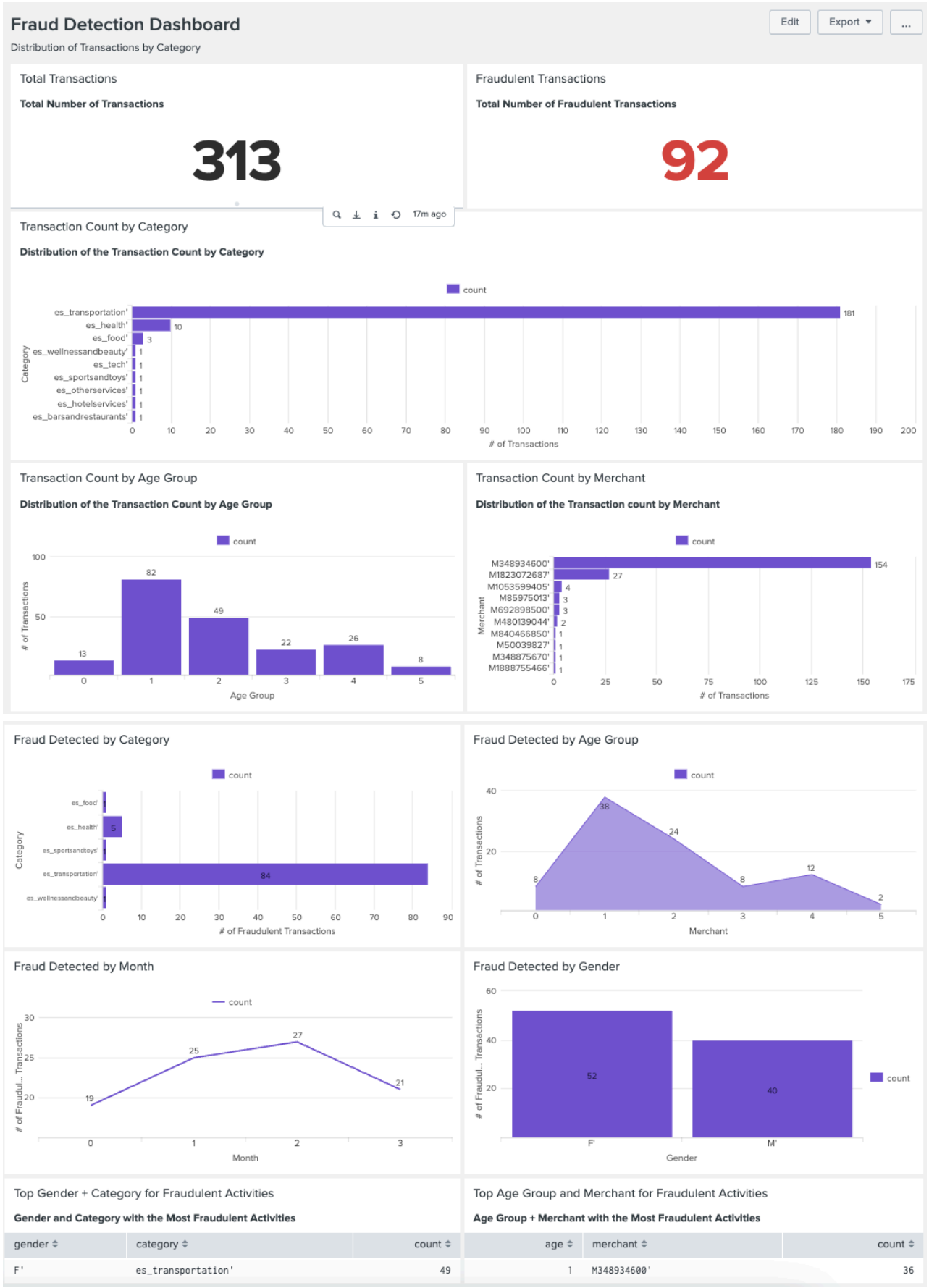
To begin my work on the Commonwealth Bank Cybersecurity Program task, I first installed and set up Splunk Enterprise, the required software for this program. Once Splunk was ready, I uploaded and imported the "prepared_data" file into the platform. This step required careful attention to detail, particularly during the field mapping process, to ensure that all data fields, such as category, age, merchant, and fraud indicators, were accurately recognized by Splunk. Successfully importing the data laid the groundwork for the subsequent analysis and visualization tasks.

With the data imported, I turned my attention to analyzing the file using Splunk's "Interesting Fields" section. This feature was invaluable as it allowed me to quickly identify key fields that would be crucial for my analysis, such as *category*, *age*, *merchant*, and *fraud*. I carefully studied the distribution and structure of the data, which informed my decisions on how to approach the creation of the dashboard. Understanding these fields was critical, as it ensured that the queries I created would yield meaningful insights and that the visualizations I developed would accurately represent the data's underlying patterns.

With a clear understanding of the data, I proceeded to create the dashboard, carefully following the requirements. I started by creating panels for Total Transaction Count and Fraudulent Transaction Count to provide a high-level overview of the dataset at a quick glance. These panels set the stage for deeper analysis by showing the overall number of transactions and how many of those were fraudulent. I then created Transaction Count by Category, Age Group, and Merchant panels to visualize the distribution of transactions across different categories, demographics, and merchants. These panels provide context for where most of the transactions were occurring, which is important for comparing them against fraudulent activity. Moving on to fraud-specific analysis, I developed panels for Fraud Detected by Category, Age Group, Month, and Gender. These visualizations were designed to highlight the areas most affected by fraud, enabling me to identify patterns and trends in fraudulent behavior. Finally, I included targeted panels for Top Gender + Category for Fraudulent Activities and Top Age Group and Merchant for Fraudulent Activities. These panels provided actionable insights by pinpointing the specific gender, category, age group, and merchant combinations most prone to fraud. I chose these panels to ensure a comprehensive analysis that would not only identify where fraud was happening but also who was most involved, thereby offering valuable insights for potential preventive measures.

See Data Visualization Dashboard Below

Splunk Fraud Detection Dashboard



Task 2: Incident Response

Review the scenario below. Then complete the tasks and activites.

Scenario

As a member of the cyber security division, your team must handle this incident and the team lead has assigned the issue to you. Below is the timeline of events:

- 10:30 a.m. – The IT Service Desk receives a report from one of your colleagues at the bank that they have received an email from HR telling all employees to update their timesheets in the company's support portal so the timesheets can be approved on time by their line managers against the next pay day. The colleague clicked the link in the email that opened what looked like the portal. However, following the employee's input of the user credentials, an unfamiliar error page appeared like the one below.

- 2:00 p.m. – Eight more reports of emails similar to the one reported earlier are received by the IT Service Desk. Upon further investigation, it was found that 62 colleagues across the Risk Department received the same email over the course of two days. The emails directed the users to a fake website to steal their usernames and passwords and download a harmful program.

- 3:50 p.m. – The IT Service Desk receives calls and emails from more colleagues that the file-shares are not opening and they receive an error when trying to open a Word document they have always been able to open.

Here is your task

In addition to the background information above, study the links in the Resources to learn how to provide solutions to the following questions. Answer the questions in the text input below.

Hint: From the links below, you should look for types of cyber security attacks and the steps to take after an incident.

1. What kind of attack has happened and why do you think so?
2. As a cyber security analyst, what are the next steps to take? List all that apply.

3. How would you contain, resolve and recover from this incident? List all answers that apply.
4. What activities should be performed post-incident?

Please note that the scenario described in this module is fictional and was created just for your virtual experience.

Here are some resources to help you:

1. Top 10 Common types of Cybersecurity Attacks (infocyte.com)
2. 11 Types of Phishing + Real-Life Examples (pandasecurity.com)
3. 8 Critical steps to take after a ransomware attack: Ransomware response guide for businesses - Emsisoft | Security Blog
4. Battling Ransomware: How to Respond to a Ransomware Incident (forbes.com)
5. Frequently Asked Questions - Ransomware | Information Security Office (berkeley.edu)
6. What to do before and after a cybersecurity breach? | american.edu

Incident Report

To create the incident report, I began by carefully analyzing the situation using the timeline of events and the resources provided, including research links, to ensure a thorough understanding of the attack. I identified the type of attack—a phishing attempt likely coupled with malware—by examining the patterns and effects reported by the affected employees. In addressing each question, I focused on immediate actions such as isolating compromised systems and disabling affected accounts, while also considering long-term strategies to prevent future incidents. By systematically considering the technical aspects of the response and leveraging the research resources, I was able to craft a detailed and actionable report that not only addresses the current incident but also sets the stage for stronger cybersecurity practices moving forward.

Justin Goncalves
Commonwealth Bank Cybersecurity Analyst
8/28/2024

Incident Report

Phishing Attack and Malware Incident:

A phishing attack combined with a potential malware infection has occurred. The attack began with phishing emails that appeared to be from the HR department, asking employees to update their timesheets through a provided link. When users clicked on the link, they were directed to a fake portal designed to steal their login credentials. The presence of an unfamiliar error page following credential input suggests the attacker's intent to harvest user information for later exploitation. The subsequent reports of inaccessible file-shares and errors with Word documents indicate that the phishing email likely delivered a malicious payload, which is now affecting the network's functionality.

As a cybersecurity analyst, the next steps involve containment, investigation, eradication, and recovery. Initially, I would isolate the affected systems to prevent the spread of malware. Disabling compromised accounts, particularly those where users clicked the phishing link, is crucial to limit unauthorized access. Following this, I would conduct an investigation to identify all recipients of the phishing email and assess the scope of the attack. Analyzing any malware samples collected during the investigation will help us understand the threat and inform our response. Once the investigation is complete, I will proceed with eradicating the malware, restoring affected files, and securing user accounts through mandatory password resets. Finally, monitoring the network for any signs of lingering threats will be essential to ensure full recovery.

To contain, resolve, and recover from this incident, a thorough and systematic approach is essential. The first priority is containment, which involves isolating all systems that interacted with the phishing email or are showing signs of infection. This critical step prevents the malware from spreading further and causing more damage. Once containment is achieved, the focus shifts to resolving the issue and recovering affected systems. The following actions should be taken:

- **Disabling compromised accounts** to immediately stop any unauthorized access and initiating a mandatory password reset for all affected users.
- **Deploying antivirus and antimalware tools** across the network to thoroughly scan and remove the malicious software from all infected systems.

- **Restoring affected files from backups** to ensure that the system is free from any remnants of the malware and that all necessary data is intact and accessible.
- **Re-enabling secured user accounts** only after verifying that the systems are completely clean and secure, ensuring that users can safely resume their work.
- **Monitoring the network continuously** to detect any signs of lingering threats or suspicious activity, ensuring that the organization remains secure and that normal business operations can safely continue.

These steps are designed to comprehensively address the immediate threat while ensuring the integrity of the organization's systems and data. By following this approach, the organization can quickly recover from the incident while minimizing the impact on its operations.

Post-incident, it is imperative to focus on long-term preventive measures that will reduce the likelihood of similar attacks occurring in the future. A proactive and strategic approach to cybersecurity is necessary to strengthen the organization’s defenses. The following activities should be prioritized:

- **Conducting phishing awareness training** for all employees to help them recognize and avoid similar threats in the future. This training should be comprehensive and include:
    - Real-world examples of common phishing tactics that employees might encounter.
    - Clear guidelines for identifying suspicious emails and reporting them promptly to the IT Service Desk.
- **Reviewing and enhancing email security measures** to identify and address any vulnerabilities that allowed the phishing emails to bypass existing defenses. This review should lead to:
    - The implementation of multi-factor authentication (MFA) for all accounts to add an additional layer of security.
    - Improvements in spam filtering and email scanning protocols to better detect and block phishing attempts before they reach employees.
- **Documenting the incident and response efforts** in a detailed report that provides valuable insights for management and serves as a reference for future security planning. This report should include lessons learned from the incident and recommendations for ongoing security improvements.

These post-incident activities are crucial not only for addressing the current issue but also for building a stronger, more resilient cybersecurity posture for the organization. By taking these steps, the organization can significantly reduce its risk of future incidents and ensure that employees are better equipped to handle potential threats.

Task 3: Security Awareness

Review the scenario below. Then complete the tasks and activites.

Scenario

Security awareness is the knowledge and understanding of potential security threats and best practices for protecting yourself and your organisation against those threats. It's about being aware of the different types of cyber threats, such as phishing scams and malware, and understanding how to avoid them. It also means staying up to date with the latest security trends and information, so you can make informed decisions and take actions to keep yourself and your organisation safe.

Security awareness also includes knowing how to create strong and secure passwords, as well as understanding the best practices for keeping software and systems updated. Additionally, it involves understanding the legal and regulatory requirements related to information security. Overall, security awareness is important because it helps individuals and organisations stay vigilant against potential threats and take proactive steps to protect themselves and their assets.

One way to raise security awareness is by creating an infographic. An infographic can serve as a reminder to employees about the importance of secure passwords and other best practices for information security. It can also help to reinforce the organisation's commitment to security and raise awareness about the specific steps that should be taken to protect sensitive information.

Moreover, an infographic can be used as a tool for education, particularly in the aftermath of a security incident like the one in the previous task. It can help employees understand how to prevent similar incidents from happening in the future, serving as an effective tool for incident response management and employee engagement in security awareness. This is what you are going to be doing in this task, creating an infographic to raise security awareness on password security among your peers based on Australian Cyber Security Centre (ACSC) advice.

Here is your task

To design an effective infographic to raise security awareness on password security among fellow employees, follow these steps:

1. Research the best practices for creating secure passwords according to the ACSC website.
2. Choose software or online tool to design the infographic (e.g Canva).
3. Use simple and clear visuals to represent the key points and statistics on password security.
4. Incorporate the ACSC advice on creating strong passwords.
5. Include tips on how to manage passwords.
6. Make the infographic visually appealing and easy to understand for the audience.

Remember that the target audience for this infographic are your fellow employees and the format in which this should be delivered is PDF. You can find a “Sample Infographic” in the Resources section to help you get started.

Security Awareness Infographic

For this task, I conducted thorough research based on the Australian Cyber Security Centre's (ACSC) guidelines to ensure that the information provided is both accurate and practical. The focus was on delivering clear and actionable tips that employees can easily implement to enhance their password security. I chose to highlight the importance of passphrases, unpredictability, and multi-factor authentication, as these are key elements in protecting sensitive information. For the design, I utilized Canva, a tool I am personally proficient in and have experience with, to create a visually appealing and easy-to-follow layout. Bold headings and concise explanations were used to ensure that the message is both engaging and memorable. My goal was to create a resource that not only educates but also empowers employees to take proactive steps in securing their digital world.

Task 4: Penetration Testing

Review the scenario below. Then complete the tasks and activites.

Scenario

As a cybersecurity generalist at CommBank, it's important to have a basic understanding of penetration testing. Penetration testing is a way to check the security of computer systems and networks by simulating an attack. This helps identify weaknesses in the system and evaluate the effectiveness of security measures. By regularly doing this, organisations can find and fix potential security problems before they can be exploited by bad people.

In this task, you will be completing the “Basic” web challenge from HackThisSite.org, which is an online platform that provides a safe and legal environment for students like you to improve their cyber security skills through a variety of challenges.

The challenge is divided into 11 levels and each level ranges from easy to difficult. The purpose of this challenge is to test your skills and knowledge in identifying vulnerabilities and exploiting them. By completing this challenge, you will gain a better understanding of how to identify and exploit vulnerabilities in web applications. Additionally, you will also learn how to apply this knowledge to real-world scenarios, which will help you improve your penetration testing skills. How exciting! Look at it like a game that helps you learn about web security.

After completing the challenge, you will need to create a pentest report detailing what you found and learned, and give recommendations for how to better secure the web application. A penetration testing report is like a summary of the results of a security test. It shows any weaknesses or problems that were found during the test and suggests ways to fix them. This report is important because it helps organisations understand where they need to improve their security and how to do it. It also helps them comply with laws and regulations related to security. In simple terms, a pentest report is like a report card for a company's security and helps them pass security inspections. By having a good security posture, organisations can prevent data breaches, protect sensitive information, and maintain compliance with regulations and industry standards.

Here is your task

To complete this task:

1. Go to HackThisSite and create an account.
2. On the left-hand side, Click on the “Challenges” section and select “Basic” (or click here).
3. Complete all levels from Basic Level 1 to 11.
4. After completing all levels, document a Penetration Testing Report that includes an executive summary, scope of web application tested, vulnerability description and key findings for each level, as well as recommendations on how to better secure the web application.
5. Additional resources are provided in the Resources for help, which will be especially useful if you have no prior experience with pentesting.

It's important to note that this is a legal and safe environment for individuals to improve their cyber security skills, and all activities should be done in accordance with the website's terms of service and ethical guidelines.

Resources for the task:

1. How to View the HTML Source Code of a Web Page (computerhope.com) - Hint for Level 1, 2 & 3
2. How to view source code – ViewSourcePage.com - Hint for Level 1, 2 & 3
3. How to Edit Any Web Page in Chrome (or Any Browser) (howtogeek.com) - Hint for Level 4 & 5
4. ASCII Table - ASCII Character Codes, HTML, Octal, Hex, Decimal - Hint for Level 6.
5. Linux and Unix cal command tutorial with examples | George Ornbo (shapeshed.com) - Hint for Level 7
6. cal command in Linux with Examples - GeeksforGeeks - Hint for Level 7
7. Linux Commands Cheat Sheet | Red Hat Developer - Hint for Level 7
8. Server-Side Includes (SSI) Injection | OWASP Foundation - Hint for Level 8 & 9
9. View, edit, and delete cookies - Chrome Developers - Hint for Level 10
10. Apache HTTP Server Tutorial: .htaccess files - Apache HTTP Server Version 2.4 - Hint for Level 11

Justin Goncalves
Commonwealth Bank Cybersecurity Analyst
8/28/2024

Penetration Testing Report

Executive Summary

The goal of this penetration testing exercise was to identify, exploit, and document various vulnerabilities within a web application. The web application consisted of multiple levels, each representing a unique challenge designed to test different aspects of web security. The primary objectives were to assess the application's security posture, learn more about penetration testing, and gain practical experience in identifying and exploiting vulnerabilities.

During this exercise, several critical vulnerabilities were identified, ranging from weak input validation and insecure cryptology practices to flawed authentication mechanisms and improper file/directory permissions. The testing involved advanced techniques such as JavaScript and cookie tampering, Server Side Includes (SSI) injection, directory traversal, and the exploitation of Apache configurations. Each level provided insights into common security issues and highlighted the importance of implementing robust security measures.

Scope

The penetration testing focused on the following key areas:

1. **Input Validation**:
   - During testing, we extensively explored the application's ability to validate user inputs. By manipulating input fields, we identified several vulnerabilities where the application failed to properly sanitize and validate inputs, leading to potential exploits such as command injection and unauthorized access. These tests demonstrated how inadequate input validation can expose the application to a wide range of attacks, allowing malicious inputs to be processed by the server.
2. **Cryptology**:
   - The scope included analyzing the cryptographic methods used within the application. We reverse-engineered an encryption pattern to decrypt a password, which involved recognizing the systematic transformation of characters based on an observed algorithm. This process highlighted the significance of understanding encryption methods and the risks posed by weak or predictable encryption practices.
3. **Authentication and Permissions**:
   - We tested the robustness of the application’s authentication mechanisms and file/directory permissions. This involved manipulating cookies and URLs to bypass authentication and access restricted areas. The tests revealed how flaws in the authentication process and improper permission settings could be exploited to gain unauthorized access to sensitive information. Our exploration of these mechanisms illustrated the potential for security breaches if authentication and permission controls are not properly implemented.
4. **Server Side Logic and Scripting**:
   - The application’s server-side logic and scripting were scrutinized for vulnerabilities. Through techniques such as Server Side Includes (SSI) injection, we executed commands and traversed directories beyond the intended access levels. These tests exposed weaknesses in how server-side

scripts were handled, demonstrating the risks associated with improperly secured server-side logic, which could allow attackers to execute unauthorized commands or access protected files.

5. **Directory Traversal**:
   - Directory traversal techniques were employed to explore how the application managed file paths and access controls. By manipulating URLs and input fields, we were able to navigate through the server's file system and access directories that should have been restricted. These tests highlighted the application's susceptibility to directory traversal attacks, showing how an attacker could exploit these vulnerabilities to access sensitive files or directories.

Vulnerability Description and Key Findings

Level 1
- Vulnerability: Password stored directly in the HTML source code.
- Exploitation: By viewing the HTML source code, the password was immediately visible within a hidden input field. This allowed for the password to be easily copied and pasted into the login form without needing to bypass any security measures. The ease of access highlighted a significant flaw in how sensitive information was being stored client-side.
- Recommendation: Avoid storing sensitive information in client-side code.

Level 2
- Vulnerability: Missing password file allowed an empty password to be accepted.
- Exploitation: The password validation script was dependent on an external file that was missing. By submitting the form without entering any password, the script processed the empty input as valid, granting access. This oversight in file management led to a critical security gap.
- Recommendation: Implement proper error handling and input validation.

Level 3
- Vulnerability: Password stored in an accessible file (password.php).
- Exploitation: By directly navigating to the file's URL (password.php), the password was exposed in plaintext. This exploitation demonstrated the danger of storing sensitive data in publicly accessible files, where a user could bypass the main form entirely by directly accessing the file.
- Recommendation: Restrict access to sensitive files and use server-side validation.

Level 4
- Vulnerability: Password recovery system allowed email address modification.
- Exploitation: The password recovery form included the recipient's email address within the HTML code. By modifying the email address field in the browser's developer tools, I was able to redirect the password reminder to my own email. This manipulation bypassed the intended security controls and exposed the password to unauthorized parties.
- Recommendation: Validate and secure input fields on the server side.

Level 5
- Vulnerability: Password recovery system allowed email address modification.
- Exploitation: Similar to Level 4, I altered the hardcoded email address within the HTML source code. By redirecting the password reminder to my own email, I successfully intercepted the password. This repeated vulnerability demonstrated the importance of server-side input validation to prevent unauthorized email redirection.
- Recommendation: Secure input fields and ensure server-side validation.

Level 6
- Vulnerability: Weak encryption algorithm allowed reverse-engineering.
- Exploitation: The password was encrypted using a pattern-based method that incremented characters based on their position. By analyzing the pattern using known inputs and outputs, I was able to reverse-engineer the encryption process and decrypt the password. This demonstrated how predictable encryption patterns can be exploited.
- Recommendation: Use strong, industry-standard encryption methods.

Level 7
- Vulnerability: Command injection allowed directory listing.
- Exploitation: The input field for the calendar command was vulnerable to command injection. By appending "2024; ls -l" to the input, I was able to execute additional commands, which listed the directory contents and revealed the password file. This exploitation demonstrated the risks of allowing unfiltered input to interact with system commands.
- Recommendation: Implement input sanitization and restrict command execution.

Level 8
- Vulnerability: SSI injection allowed directory traversal and file access.
- Exploitation: By injecting a Server Side Includes (SSI) command into the input field, I was able to traverse the directory structure and list files outside the intended directory. This allowed me to identify and access the file containing the password, bypassing any intended restrictions. The SSI injection demonstrated the dangers of allowing unchecked user input in web applications.
- Recommendation: Secure input fields and restrict SSI commands.

Level 9
- Vulnerability: Directory traversal via SSI allowed access to restricted files.
- Exploitation: Similar to Level 8, I used an SSI command to list and access files in a restricted directory. By specifying the exact path, I was able to bypass the application's intended restrictions and retrieve the password from a protected file. This exploitation highlighted the need for stronger input validation and directory permissions.
- Recommendation: Implement input validation and restrict directory access.

Level 10
- Vulnerability: Cookie manipulation allowed unauthorized access.
- Exploitation: By using JavaScript to alter the cookie's authorization value from "no" to "yes", I was able to bypass the password requirement entirely. This manipulation of client-side cookies demonstrated the risks of relying on cookies for critical security checks without proper server-side validation.
- Recommendation: Secure cookies and implement server-side validation.

Level 11
- Vulnerability: Hidden directory and file accessible through Apache's .htaccess configurations.
- Exploitation: After identifying a pattern in song titles related to Elton John, I navigated through several directories by appending letters to the URL. Eventually, I accessed an .htaccess file that revealed the presence of a hidden file named "DaAnswer." By replacing the .htaccess in the URL with "DaAnswer," I accessed a page that provided the password. This exploitation underscored the importance of securing .htaccess files and preventing directory traversal.
- Recommendation: Secure .htaccess files and hidden directories to prevent unauthorized access.

Recommendations

1. Implement Strong Input Validation: Ensure that all input fields are properly validated on the server side to prevent unauthorized access and exploitation.
2. Use Strong Encryption Methods: Replace weak encryption algorithms with industry-standard encryption methods to protect sensitive information.
3. Enhance Authentication and Permissions: Implement robust authentication mechanisms and secure file/directory permissions to prevent unauthorized access.
4. Secure Server-Side Logic and Scripts: Regularly audit and secure server-side scripts to prevent SSI injection and other logical vulnerabilities.
5. Restrict Directory Access: Use proper directory permissions and restrict access to sensitive files and directories to prevent directory traversal attacks.
6. Harden Apache Configurations: Review and secure Apache configurations, including .htaccess files, to prevent exploitation and unauthorized access.

Conclusion

The penetration testing exercise revealed critical vulnerabilities across several aspects of the web application, including input validation, cryptology, authentication and permissions, server-side logic, and directory traversal. Each level demonstrated weaknesses that could be exploited, emphasizing the importance of robust security measures. By addressing these issues through recommended improvements, the application can greatly reduce its risk of exploitation and protect sensitive data. This exercise not only identified areas for enhancement but also provided valuable hands-on experience, reinforcing the need for continuous security assessments and proactive mitigation to maintain a secure environment.

Personal Reflection

Throughout the Commonwealth Bank Introduction to Cybersecurity program, I have gained invaluable insights and practical experience that will be instrumental in my future career as a cybersecurity analyst. The program was meticulously designed to simulate the responsibilities of a cybersecurity professional, offering a comprehensive overview of various facets of the field. From data analysis using Splunk to incident response, security awareness, and penetration testing, each task presented a unique set of challenges, helping me to build a well-rounded skill set that will be invaluable in my future career as a cybersecurity analyst.

The first two tasks focused on data analysis using Splunk and incident response. In the data analysis task, I learned how to use Splunk to visualize and interpret fraud-related data, which sharpened my ability to detect patterns and identify potential security threats. The incident response task further deepened my understanding of how to effectively manage and contain cyber incidents, providing valuable insights into the critical steps involved in mitigating risks and restoring normalcy after an attack. These tasks taught me the importance of proactive monitoring and the need for swift, decisive actions in the face of security breaches.

The third and fourth tasks centered on security awareness and penetration testing. Designing an infographic on password security for the security awareness task helped me appreciate the importance of clear communication in cybersecurity, particularly when conveying complex information to a non-technical audience. The penetration testing task was very challenging, yet rewarding, as it allowed me to explore various vulnerabilities in a web application, including server-side scripting, directory traversal, and cryptology. These tasks not only honed my technical skills but also reinforced the importance of user education and robust security measures in protecting an organization’s digital assets.

Overall, the Commonwealth Bank Cybersecurity Program offered the perfect balance between difficulty and enjoyment. Each task pushed me to think critically and adapt to different scenarios, all while reinforcing the skills and knowledge needed to excel in cybersecurity. It provided me with a practical and engaging way to deepen my cybersecurity knowledge while having fun tackling real-world problems. The program has equipped me with the skills and confidence necessary to excel in my future career, ensuring that I am well-prepared to contribute to the security of any organization I join.

Certificate of Completion

Commonwealth Bank

Forage

Inspiring and empowering future professionals

Justin Goncalves

Introduction to Cybersecurity Job Simulation

Certificate of Completion

August 28th, 2024

Over the period of August 2024, Justin Goncalves has completed practical tasks in:

Data analysis
Incident response
Security awareness
Penetration testing

Tom Brunskill
CEO, Co-Founder of Forage

Enrolment Verification Code onz4Zu5z8qj6coHbJ | User Verification Code rCsmcFfq94jiPnW64 | Issued by Forage