CSE 4402: Lab Design Document Cookies

Luke Pepin, Tommy Jiang, Mohamed Makhlouf, Jan Feyen

Abstract ***– In this lab, we will analyze the security and management aspects of cookies in web applications, focusing on authentication (login) cookies. We will discuss how security flags are configured across different sites and their implications for user data protection. We will also evaluate the relationship between cookie lifespan and security, emphasizing the trade-offs between convenience and risk. By automating the data collection process using a Python program, we aim to efficiently gather and analyze cookie data from various websites. Additionally, we will categorize websites into different security tiers (high, mid, low) to understand the varying security practices across different types of entities. Collecting thousands of cookies will ensure a comprehensive and statistically significant dataset.***

### Introduction:

Cookies in web security are data stored on a user’s device that helps websites remember important information about that user, such as login credentials, and maintain session states. They enhance the user experience by tracking user activity but must be securely managed to prevent unauthorized access (e.g., XSS) and/or data breaches. Some protection mechanisms include Security Flags like HttpOnly, Cross-origin policy, and SameSite to mitigate risks.

This lab’s objective is to retrieve and analyze login cookies from secure websites. By examining these cookies and their associated security flags, we aim to understand the purpose of each flag and assess their effectiveness in the current security environment. This includes evaluating how these flags contribute to the security of user authentication processes, protecting against common threats such as cross-site scripting (XSS) and cross-site request forgery (CSRF).

The scope of this lab will cover the retrieval and analysis of multiple different cookies (primarily login) under numerous different protocols. These cookies will be reviewed, documented, and tested based on their security attributes. Categorizing websites into different security tiers will help us understand industry-specific security practices and trends.

### Data Collection Program:

We aim to develop a Python program to collect data on cookie security flags from various websites. Using libraries such as requests and BeautifulSoup, the program will fetch and parse HTML content to extract cookie-related data. This data, stored in CSV and JSON formats, will help us analyze and compare security measures across different sites. By examining security flags like Secure, HttpOnly, and SameSite, we will gain insights into cookie management practices and their effectiveness in mitigating threats such as XSS and CSRF. This automated approach will streamline data collection and provide a comprehensive overview of current web security practices.

**Sample Code:**

import xml.etree.ElementTree as ET

import csv

# Load the XML file

tree = ET.parse('cookies.xml') # replace 'cookies.xml' with your file's name

root = tree.getroot()

# Open a CSV file to write the cookies into

with open('cookies.csv', mode='w', newline='') as csv\_file:

fieldnames = ['Domain', 'Path', 'Name', 'Value', 'Secure', 'HttpOnly', 'Expiry', 'SameSite']

writer = csv.DictWriter(csv\_file, fieldnames=fieldnames)

writer.writeheader()

# Loop through XML to extract relevant data

for cookie in root.findall(".//cookie"): # Adjust the XPath based on your XML structure

domain = cookie.find('domain').text if cookie.find('domain') is not None else ''

path = cookie.find('path').text if cookie.find('path') is not None else ''

name = cookie.find('name').text if cookie.find('name') is not None else ''

value = cookie.find('value').text if cookie.find('value') is not None else ''

secure = cookie.find('secure').text if cookie.find('secure') is not None else 'False'

httponly = cookie.find('httponly').text if cookie.find('httponly') is not None else 'False'

expiry = cookie.find('expiry').text if cookie.find('expiry') is not None else ''

samesite = cookie.find('samesite').text if cookie.find('samesite') is not None else ''

# Write each cookie data to the CSV file

writer.writerow({

'Domain': domain,

'Path': path,

'Name': name,

'Value': value,

'Secure': secure,

'HttpOnly': httponly,

'Expiry': expiry,

'SameSite': samesite

})

print("Cookies have been written to cookies.csv")

### Identifying Websites:

To identify websites for cookie extraction, we will balance three primary factors: popularity, security requirements, and login necessity. We will use lists like DomCop’s top 10 million websites to select commonly visited sites, ensuring a diverse dataset that reflects a wide range of user interactions and security practices. Both login-required and non-login websites will be included to provide a comprehensive view of cookie management practices. For instance, non-login websites such as Wikipedia, BBC News, and YouTube will be analyzed to understand how they manage cookies without user authentication. On the other hand, login-required websites like Facebook, Snapchat, and Bank of America will be examined to assess the security measures in place for protecting user credentials and session data.

Websites will be categorized into different security tiers based on their perceived security needs. High-tier websites will include financial institutions like banks and investment platforms, which are required by law to implement stringent security measures. Mid-tier websites will cover government, tech, and academic sites, which handle sensitive information but may not be as heavily regulated as financial institutions. Low-tier websites will focus on dining and retail stores, which typically have less stringent security requirements but still need to protect user data. This categorization will help us understand industry-specific security practices and trends, providing insights into how different sectors prioritize and implement cookie security.

By balancing these factors, we aim to create a robust dataset that captures a wide spectrum of cookie management practices. This approach will allow us to identify common trends and outliers, providing a comprehensive understanding of how cookies are used and secured across various types of websites. Additionally, by including both high-traffic and niche websites, we can ensure that our analysis is representative of the broader web ecosystem. We intend to collect thousands of cookies for data analysis, ensuring a comprehensive and statistically significant dataset.

### Focus on Measurements

Our project will focus on key measurements to identify trends in cookie usage and security. We will analyze the configuration of security flags such as Secure, HttpOnly, and SameSite, which are critical for protecting user data and preventing attacks like XSS and CSRF. By examining how these flags are implemented across different websites, we can assess their effectiveness in mitigating common security threats. For example, the Secure flag ensures that cookies are only sent over HTTPS connections, reducing the risk of interception by malicious actors. The HttpOnly flag prevents client-side scripts from accessing the cookie, protecting against XSS attacks. The SameSite flag helps prevent CSRF attacks by restricting how cookies are sent with cross-site requests.

Additionally, we will examine cookie lifespan to understand the balance between user convenience and security risks. Longer cookie lifespans can enhance user experience by reducing the need for frequent logins, but they also increase the risk of unauthorized access if the cookie is compromised. By analyzing the lifespan of cookies across different websites, we can identify best practices for balancing security and usability.

We will also compare common websites to high-security ones, such as banks, to gain insights into overall security practices. This comparison will help us understand the differences in cookie management between sites with varying security requirements. For instance, high-security websites may implement stricter cookie policies and shorter lifespans to minimize risk, while common websites may prioritize user convenience.

By focusing on these measurements, we aim to recognize common practices in cookie management and pinpoint potential vulnerabilities. This analysis will provide valuable insights into the current landscape of web security, highlighting areas where improvements can be made. Ultimately, our findings will contribute to a better understanding of how cookies are used and secured, helping to enhance overall web security practices. Collecting thousands of cookies will allow us to perform a detailed and statistically significant analysis, ensuring our conclusions are robust and reliable.

### Implementation Plan

**Team Roles**

* Python Script: Jan Feyen
* Website Identification: Mohamed Makhlouf
* Measurement Focus: Luke Pepin
* Security Tiers Analysis: Tommy Jiang

#### **Timeline**

* Demo: Week of Oct. 21st
* Submission: Nov. 1st
* Feedback: Within a week
* Class Presentation: Dec. 3rd or 5th
* Final Submission: Dec. 15th

#### **Python Script**

* Goal: Collect security flag data.
* Tools: Python (requests, BeautifulSoup).
* Testing: Compare with browser developer tools.

#### **Website Identification**

* Goal: List popular websites.
* Source: Top websites from Backlinko.
* Categories: Non-login and login-required.

#### **Measurement Focus**

* Key Metrics: Secure, HttpOnly, SameSite flags, cookie lifespan.
* Analysis: Use Excel/Google Sheets for graphs and written analysis.

#### **Security Tiers Analysis**

* Tiers: High (banking), Mid (UConn, social media), Low (Starbucks, Dominos).
* Comparison: Security flags, user base, usage reasons.

#### **Final Report and Presentation**

* Structure: Academic paper style.
* Presentation: Slides, team members present their sections.

### Conclusion

This lab will analyze the security and management of cookies in web applications, focusing on login cookies. By automating data collection with a Python program, we will efficiently gather and analyze cookie data from various websites, categorized into different security tiers. Our findings will highlight the importance of security flags like Secure, HttpOnly, and SameSite in protecting user data and mitigating threats such as XSS and CSRF. We expect to observe significant variations in security practices across different websites and industries, emphasizing the need for standardized best practices.

The analysis of cookie lifespan will reveal a balance between convenience and security, where longer lifespans enhance user experience but increase security risks. Our tiered analysis will provide insights into the operations, challenges, and defenses of entities at different security levels. Overall, this lab will underscore the critical role of security flags in web application security. Future research could explore additional security mechanisms and the use of cookies in emerging web technologies to further enhance web security practices.

### Contacts:

Luke Pepin [LUKE.PEPIN@UCONN.EDU](mailto:LUKE.PEPIN@UCONN.EDU)

Mohamed Makhlouf [MOHAMED.MAKHLOUF@UCONN.EDU](mailto:MOHAMED.MAKHLOUF@UCONN.EDU)

Tommy Jiang [TOMMY.JIANG@UCONN.EDU](mailto:TOMMY.JIANG@UCONN.EDU)

Jan Feyen [JAN.FEYEN@UCONN.EDU](mailto:JAN.FEYEN@UCONN.EDU)