Phishing Email Detection

My project called ‘Phishing Email Detection’ was created with the purpose of monitoring emails and reporting spam/phishing emails. Our team’s goal was to develop an automated email forensics tool that analyzes emails based on information from the email headers. In cybersecurity, phishing and other variations such as vishing are some of the most successful attacks. I aimed to simplify forensic email analysis by using Microsoft Graph API to pull message data, parse headers, and evaluate security indicators such as SPF, DKIM, and DMARC values.

Technical Implementations

The project was built in Python 3 and used several libraries such as Requests for API calls to Microsoft graph, Beautiful Soup to parse HTML email bodies, JSON for report formatting, and Dotenv for credential management. The authentication was handled through OAuth 2.0 using environment variables in a .env file for the Client ID, Tenant ID, and the Scopes. The header analysis checked SPF and DKIM records for spoofing or validation of errors. The body analysis looked for suspicious links, errors, or attachments which were evaluated for malicious content using Virus Total.

Results

The tool successfully identified phishing indicators in several test emails. We sent emails using malicious links and spoofed senders. Legitimate emails were correctly rated at low risk, showing high true positive rates. The program correctly gave an output in a JSON format displaying the email details to the user alongside its risk verdict.

Lessons Learned & Conclusion

I learned a lot about email forensics while working on this project throughout the semester. We faced many challenges such as learning how to use OAuth authentication and managing tokens. It was my first time creating an azure application and setting permissions to my app to provide access to its job. What worked well was the Graph API integration, which provided stable data access and consistent results. Once I was able to successfully retrieve the headers from the email and dive into what information was important from the

headers, it helped me to develop a working model that would successfully search for indicators of a phishing email.

In the future I want to implement Gmail connectivity as I feel it's much wider used than Outlook. I would also like to create some sort of GUI dashboard and ACL. The dashboard would be nice to have a cleaner view of the data that my program is intercepting, and the ACL couldn't automatically block its from sending me mail if phishing is detected.