CYSE Final Analysis Document

The UML diagrams created provided an organized and structured representative of the cybersecurity system. Key features like a VPN, email security with an internal email system, and even physical security were mapped out in activity diagrams to illustrate the coordination between components like firewalls, antivirus, and VPNs. Network monitoring via an intrusion detection system (IDS) was highlighted in a sequence diagram to show safe communication. Additionally, included in both an activity and use case diagram depicted a potential website to show user and organizational exchanges and user authentication through interactions between users and system components. In trying to enhance system and security, these diagrams visualize how VPNs, internal emails, physical security, firewalls, antiviruses, IDS systems, and websites all function to provide a strong cybersecurity framework.

Threat detection and system performance were greatly improved by the security automation created. The automation of system performance monitoring made sure that resources were used as efficiently as possible and pointed out any errors identified that could turn into attacks. Log analysis automation made it easier to scan system logs and spot any odd activity or patterns. Notifications from alert generation allow for quicker reactions to threats that are identified and decrease the possibility of vulnerabilities being exposed. The vulnerability scanner finds flaws in the system and provides useful information for properly managing risks. Lastly, the traffic monitoring automation analyses network traffic to detect problems and unauthorized attempts. Overall, these automated procedures increase threat detection effectiveness while also gathering information that can be used for risk management and better security.

The log-analysis script successfully sifts through security logs to find any suspicious activity (i.e. failed log-in attempts and instances of unauthorized access), extracting important data like event date, type, and status, and then creating a summary of these events. A challenge, in regards to the log analysis script, is that some logs can be corrupted and/or have incorrect formatting, which can lead to an error. They can also have differing delimiters, but that can be edited within the code if needed.

System-performance is a script that can track OS resources such as CPU, memory, and disk usage at regular intervals and logs the data derived from this tracking into a log in a specified file. When a certain threshold is met, an email alert is generated with smtplib. A large challenge we encountered was successfully trying to set up the email alert system; it would not work unless the SMTP credentials were done properly. It also, unlike the log analysis script, only creates an alert and not a summary, which can be vague to the user. To add, the thresholds mentioned in the script can vary from system to system and might need fine-tuning on the user’s part.

Alert-generation builds on log-analysis, monitoring/reading the logs created by it alongside system-performance and generating an alert in a log if any of the pre-existing thresholds are met or exceeded. Since there was no SMTP involved and it was only simple logging, not many challenges were faced.

The scripts for traffic-monitoring and vulnerability-scan were effective in finding any potential network activity threats by utilizing scapy. Additionally, the former logs IP addresses, detects traffic spikes, and flags blacklisted IPs or any sources of high-volume traffic. These can all be useful in alerting users of unusual activity that could be a means for a security breach, DDoS attack, or compromised systems within a network. Accurate logging was a challenge, as we didn’t want to input too much information in a manner that could overwhelm the system. The integrated vulnerability-scan regularly scans the target system (IP) with nmap and logs the results with information about ports, protocols, etc., that can be exploited. Scanning successfully may be a problem, as network setups differ and high traffic volumes can drastically affect results.