Cyber Security/IT Security Analysis

Kennedy Space Center

Spaceport Command and Control System

Student Name: Ryan Alexander Gunawan
Academic Level: Undergraduate Senior
Academic Major: B.S. Aerospace Engineering, Computer Science minor
Academic Institution: California State Polytechnic University, Pomona

Mentor Name: Robert G. Van Arsdalen
Mentor Job Title: Information System Security Officer
Org Code/Branch: (NE-EC) / Electrical Branch
Division: Computer Systems
Directorate: Engineering and Technology

Cyber Security for the Spaceport Command and Control System *Vulnerability Management and Compliance Analysis*

Ryan A. Gunawan[1]
California State Polytechnic University - Pomona, Pomona, California, 91768

Nomenclature

CVE	= Common Vulnerabilities and Exposures
DEE	= Development External Enclave
KSC	= Kennedy Space Center
NASA	= National Aeronautics and Space Administration
NIST	= National Institute of Standards and Technology
NVD	= National Vulnerability Database
NGVM	= Next Generation Vulnerability Management
OEE	= Operational External Enclave
SCCS	= Spaceport Command and Control System

I. Introduction

With the rapid development of the Internet, the number of malicious threats to organizations is continually increasing. In June of 2015, the United States Office of Personnel Management (OPM) had a data breach resulting in the compromise of millions of government employee records. The National Aeronautics and Space Administration (NASA) is not exempt from these attacks. Cyber security is becoming a critical facet to the discussion of moving forward with projects. The Spaceport Command and Control System (SCCS) project at the Kennedy Space Center (KSC) aims to develop the launch control system for the next generation launch vehicle in the coming decades. There are many ways to increase the security of the network it uses, from vulnerability management to ensuring operating system images are compliant with securely configured baselines recommended by the United States Government.

II. Objectives

The IT Security division is responsible for monitoring security protocols, vulnerabilities, malicious code, system configurations, and system events to correct known vulnerabilities and provide cost-effective IT security implementation and continuous monitoring. The confidentiality, integrity and availability of resources for the SCCS project are a crucial aspect especially because the project is starting to accelerate in development. As the project approaches operational status, the resources needed for the operational phase also need to be deployed to meet the needs of the project.

"Vulnerability" is a broad term in the information assurance industry. It can be better defined by limiting the scope to instances of susceptible software or applications that an attacker may attempt to utilize to gain access to resources. Configuration management is another aspect of managing and limiting vulnerabilities. Having updated software is a good attempt at securing your system, but without having the proper and secure settings configured and enabled, a system is still vulnerable.

Vulnerabilities are found on a regular basis and compiled in the National Vulnerability Database, managed by the National Institute of Standards and Technology (NIST). NIST has also produced Special Publication 500-83, which identifies Security and Privacy Controls for Federal Information Systems and Organizations. This document outlines the recommended settings as well as configurations to ensure that government information systems are secure.

In order to effectively support the Cyber Security and IT Security teams, becoming familiarized with the project's assets was key to understanding how hardware was connected. There are various networks within the Launch Control

[1] Cyber Security / IT Security Intern, Computer Systems Branch (NE-EC), NASA Kennedy Space Center

Center (LCC). The Operational External Enclave (OEE) and the Development External Enclave (DEE) provide resources for the operational and development software developers respectively. These networks contain various components and network assets such as master consoles, remote displays, and their supporting equipment.

III. Cyber Security Internship - Spaceport Command and Control System

A. Vulnerabilities Targeting Development Sets with Remote Access to Operational Sets

The first aspect of my project was to take a closer look at vulnerabilities within the LCC, specifically with development computers that have remote access with permissions to access the operational firing rooms. While this may not be as critical of an issue because the project is not near the operational phase, it is important to understand the current status of vulnerabilities and the potential effects in the case of a data breach.

To accomplish this, next generation vulnerability management software was utilized. This software scans a given list of computers to identify possible vulnerabilities as well as to recommend what courses of action were available to mitigate the risk of a data breach. The recommendation is usually in the form of a patch or a recommended configuration setting. From gathering the information from the Next Generation Vulnerability Management (NGVM) software, an extensive list of vulnerabilities was generated. In an ideal environment, all those vulnerabilities would be addressed, but because resources are limited, there are only so many actions that can be resolved. As such, it is important to prioritize the resolution of these vulnerabilities. Utilizing the National Vulnerability Database (NVD), NIST has provided a severity level for each vulnerability. The scale uses “Low”, “Medium”, and “High” risk ratings, based off of the Common Vulnerability Scoring System (CVSS), which utilizes the vulnerability characteristics to determine how likely an attack would utilize that vulnerability as well as how devastating the potential damage would result in order to generate the risk rating. A vulnerability analysis was then conducted to determine the extent of the threat that these vulnerabilities posed based on the computers’ configuration settings and network connections.

Taking the vulnerabilities generated by the NGVM software as well as the results of the vulnerability analysis, a recommendation was compiled to provide to system administrators who could apply necessary changes to ensure the risks were mitigated. However there are instances where certain vulnerabilities could not be mitigated because of a requirement for certain versions of software or configuration settings for the set to operate as desired. If this was the case, additional documentation is required to trace the accepted risks.

Reports were generated and presented to the SCCS IT Security group regarding the overall status of the sets. Issues such as certain machines being unavailable for scans or intriguing vulnerabilities were brought up so that key personnel within the IT Security group were made aware of the current situation.

B. Automating the Accountability and Reporting Processes of Vulnerability Management

The current process for discovering, analyzing and mitigating new vulnerabilities in the system begins with sifting through generated reports from vulnerability scanners. Through these reports, security analysts determine whether or not certain vulnerabilities do exist within the system or if they are false positives. However, the reports generated are lengthy and time consuming to process. One of the ways to expedite the accountability and reporting process of newly found vulnerabilities is to automate the prioritization of vulnerabilities to address. Using Visual Basic, scripts were developed to facilitate automating data processing. New fields were inserted to determine whether certain machines belonged to the operational set or the development set. Because the development set operates on a quicker patch cycle than the operational set, this results in the need for the development set to be prioritized when addressing vulnerabilities. A “critical patch” field was also created to provide the user with the flexibility and convenience of entering in a list of CVE IDs and to identify certain vulnerabilities with the specific CVE ID. The user could then filter the vulnerability report to display only pertinent or desired information. Finally the last feature added to the automation script was to determine the top ten vulnerabilities in the operational as well as the development set.

The script then generates interactive charts for this data to give the user a visual representation of the information generated. This also allows the user to eliminate or include variables that may be pertinent to a specific query. These charts are then used in vulnerability management meetings for discussion.

C. Non-Secure Configuration Management of Firewalls

Another aspect of my project was to determine whether the firewalls protecting the SCCS network were properly configured per the United States Government Configuration Baseline (USGCB) recommendations. Using NGVM software, non-compliances with the USGCB recommendations were generated. Taking the trifecta of confidentiality, integrity and availability, a prioritization order was determined to ensure the high risk non-compliances were addressed first. A collaboration was then formed with network engineers to determine which secure configurations to implement based on protocols and services needed by developers. These controls were then implemented on a test bed firewall to ensure that the implementation of these secure configurations would not interfere with necessary services. Following a successful implementation of the secure configurations on the test bed firewall, documentation was written and provided to network engineers to implement.

D. Developing Scripts for Interfacing with NGVM Software

The NGVM software uses a custom language to discover vulnerabilities or non-compliances with the security policies for the specific system. The software also allows security engineers to upload custom scripts to detect and determine user defined conditions. My task was to develop a script to detect the specific version of the operating system for reporting. This required an understanding of the process how computers are updated as well as how the NGVM software operates.

E. Establishing a Cyber Security Lab

The establishment of a cyber security lab for the SCCS project provides several benefits for security analysts within the project. This cyber-security lab will provide a resource for other analysts to learn about the tools that exist and how certain configurations affect an attacker's ability to compromise a system. The lab consists of two servers with multiple virtual machines connected on a virtual network, which is separated from the KSC networks. This ensures that usage of the lab will not interfere with the daily operations or introduce external risks to the project. Establishing a Kali Linux distribution on a virtual machine as the attacker, we can launch exploits and attacks on virtual machines with varying operating systems. Future goals include expanding the server into additional computers, operating systems, firewalls and network switches.

IV. Conclusions

My role within the organization was to provide support for the IT Security group at the Kennedy Space Center within the Spaceport Command and Control Systems project. Most notably during this internship, I have researched vulnerabilities and their effect on mission critical assets, utilized Next Generation Vulnerability Management software to provide metadata to assist security analysts in understanding patterns within the vulnerabilities, and led an effort to tighten configurations into a more secure state for crucial firewalls. From supporting the vulnerability management cycle to developing a cyber security lab, I have had an amazing experience at the Kennedy Space Center and I hope to be working with this group again in the future.

V. Acknowledgements

I would like to thank Rob Van Arsdalen for his mentorship throughout my time at the Kennedy Space Center. His guidance and support in both my professional and personal life have enriched my experience greatly.

I would also like to thank Oscar Brooks, Caylyne Shelton, and Jamie Szafran for providing support with the multiple internship events that took us around center to learn more about the history that had taken place at Kennedy Space Center.

I would like to thank Rose Austin and Grace Johnson with the Education Office for managing the OSSI internship program and for their advice in making the most out of my time at Kennedy.

There were many people I have interacted with on a daily basis, and so I'd like to thank Ross Nordeen, Tom Henning, Aqil Assalil, Jerrace Mack, and Darrrell Thomas for their support with my various projects as well as making it exciting to come into work each day.