Implementing the Infosec Color Wheel into an Urban Air Mobility Software and System Development Lifecycle

Kenneth Freeman[1] and Steve W. Garcia, Jr.[2]
NASA Ames Research Center, Moffett Field, CA, 94035, United States

This research will provide information on the importance of collaboration between software development teams and cybersecurity teams. As Urban Air Mobility software services are developed, a collaborative approach between software development and software assurance teams will limit the exposed vulnerabilities and weaknesses.

I. Introduction

A significant aim of cybersecurity attackers is to steal valuable information from target organizations. For future airspace systems, this could create the risk of unauthorized access to restricted, confidential information. One approach to reducing the risk of unauthorized access to confidential information is to leverage security principles during the software development life cycle. At the 2017 Black Hat conference, April Wright gave the presentation, "Orange Is the New Purple." In this presentation, April Wright notes that initially, cybersecurity teams consisted of a Red and a Blue Security Team. The Blue Team performed defensive security consisting of incident response activities, operational security, threat hunting, and digital forensic activities. The Red Security Teams performed offensive penetration tests using ethical hacking techniques, including web application scanning and social engineering attacks. April Wright explains this traditional cybersecurity team approach is exercised late in the software development life cycle and lacks interaction between the development and security teams. Wright introduces the concept of three new teams that would improve cybersecurity throughout the software and system development lifecycles.

II. Objectives

This paper will identify a secure software development methodology and a security testing methodology implemented in the Urban Air Mobility (UAM) development of software services to ensure that the software is resistant from threats and free from potential vulnerabilities and weaknesses. The researchers will review the Open Web Application Security Project (OWASP) security model to identify secure coding requirements. The researchers will study the Penetration Testing Execution Standard (PTES) to determine requirements to perform security testing on developed software and systems. The identified requirements will be used by the Blue, Red, Purple, and Yellow teams to make cybersecurity a collaborative effort and not an afterthought for UAM software development.

III. Environment

Future UAM environments will leverage a service-based airspace system. Services, such as micro-weather services, will be provided to UAM vehicle operators. These types of services will operate, utilizing a wide range of information technology services and technologies. As shown in Figure 1, in addition to the aerial vehicles, the UAM environment will consist of a wide range of diverse systems supporting flight missions. The UAM comprises four

[1] Aerospace Engineer, NASA Ames Research Center.
[2] Senior Security Architect, Intrinsyx Technologies Corporation.

significant high-level components: end-users, cyber-physical systems, cloud services, and on-premise computing services.

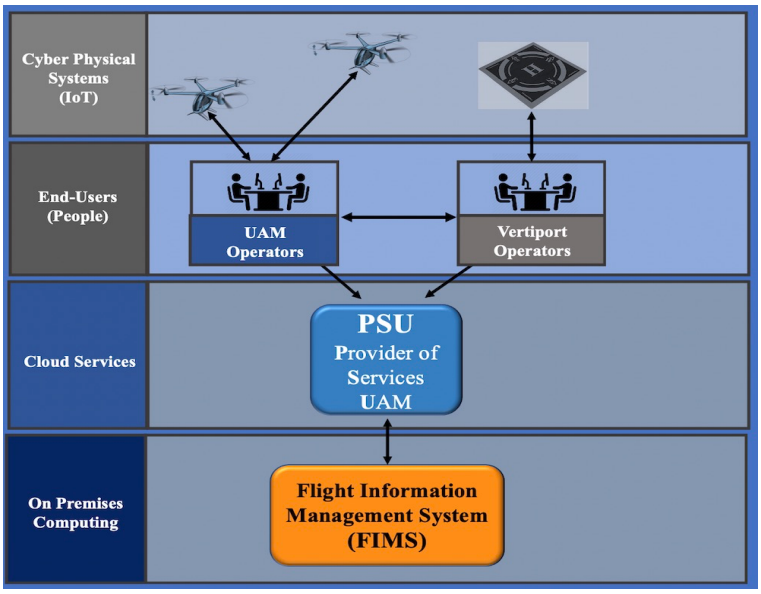

Figure 1 - UAM High-Level Components

Each of these components will incorporate software to provide UAM services. These UAM services' security posture will be enhanced by incorporating security principles during the software development life cycle. The Blue, Red, Purple, and Yellow teams will support the development of UAM services in a secure manner.

IV. Cybersecurity Collaboration

Cybersecurity collaboration includes building relationships and trust with other groups within the organization. Data protection should be a priority of all people in the organization. The confidentiality, integrity, and availability of the organization's data need to be a priority for every person in an organization.

To provide security principles to the software development lifecycle for the target UAM services, the four teams below will need to work together.

A. Blue Team

The Blue Team takes a defensive approach to cybersecurity. The Blue Team is responsible for vulnerability scanning, verification and validation, and monitoring of identified vulnerabilities. The Blue Team is also responsible for Incident Response (IR), as well as Digital Forensics (DF).

B. Red Team

The Red Team is responsible for performing penetration testing of the organization. The Red Team is authorized to perform black-box testing, social engineering, and external web application scanning.

C. Purple Team

The Purple Team is a combination of the Red and Blue Teams. The teams will work together to improve the cybersecurity posture of the organization.

D. Yellow Team

The Yellow Team consists of software developers, programmers, application developers, and software architects.

V. Open Web Application Security Project (OWASP) Software Assurance Maturity Model (SAMM)

The four teams will leverage the OWASP Software Assurance Maturity Model to assess and provide security during the UAM software development lifecycle. The OWASP Software Assurance Maturity Model (SAMM) v2 is a prime maturity model for software assurance that provides a practical and measurable way for organizations to analyze and improve their software security posture. OWASP SAMM supports the complete software lifecycle, including development and acquisition, and is technology and process agnostic. [11] The business functions of the OWASP SAMM v2 are:

A. **Governance**

Governance focuses on the processes and activities related to how an organization manages overall software development activities. This includes concerns that impact cross-functional groups involved in the development and business processes established at the organization level.

B. Design

Design concerns the processes and activities related to how an organization defines goals and creates software within development projects. In general, this will include requirements gathering, high-level architecture specifications, and detailed design.

C. Implementation

Implementation is focused on the processes and activities related to how an organization builds and deploys software components and their related defects. Activities within the Implementation function have the most impact on the daily life of developers. The joint goal is to ship reliably working software with minimum defects.

D. Verification

Verification focuses on the processes and activities related to how an organization checks and tests artifacts produced throughout software development. This typically includes quality assurance work such as testing, but it can also include other review and evaluation activities.

E. Operations

The Operations Business Function encompasses those activities necessary to ensure confidentiality, integrity, and availability throughout the operational lifetime of an application and its associated data. Increased maturity with regard to this Business Function provides greater assurance that the organization is resilient in the face of operational disruptions and responsive to changes in the operational landscape. [11]

VI. Penetration Testing Execution Standard (PTES)

The Blue and Purple teams will leverage the PTES standard to test the UAM software for vulnerabilities and weaknesses. The penetration testing execution standard consists of seven (7) main sections. These sections cover everything related to a penetration test - from the initial communication and reasoning behind a penetration test through the intelligence gathering and threat modeling phases. Testers are working behind the scenes to understand the tested organization better.

A. Pre-engagement Interactions

Defining scope is arguably one of the most critical components of a penetration test

B. Intelligence Gathering

Intelligence Gathering is the phase where data or "intelligence" is gathered to guide the assessment actions.

C. Threat Modeling

The standard focuses are on two key elements of traditional threat modeling - assets and attackers (threat community/agent).

D. Vulnerability Analysis

Vulnerability testing is the process of discovering flaws in systems and applications which can be leveraged by an attacker.

E. Exploitation

The exploitation phase of a penetration test focuses solely on establishing access to a system or resource by bypassing security restrictions.

F. Post Exploitation

The purpose of the Post-Exploitation phase is to determine the machine's value and maintain control of the device for later use.

G. Reporting

A final report should be provided as a presentation to the penetration test outcome's leadership and staff. Another copy of the report needs to be emailed to the penetration test's requestor to ensure the organization has a formal copy of the final report. [9]

VII. Conclusion

This paper will identify methodologies for secure development and security testing of UAM software and systems. The Blue, Red, Purple, and Yellow teams will continually improve the UAM cybersecurity posture. As cybersecurity threats evolve, the UAM cybersecurity team's thinking must adapt and try to get ahead. The scan, find, and fix mentality is no longer acceptable.

References

Periodicals

[1] Colbert, E. J., Kott, A., & Knachel, L. P. (2020). The game-theoretic model and experimental investigation of cyber wargaming. *Journal of Defense Modeling and Simulation, 17*(1), 21-38. doi:10.1177/1548512918795061

[2] Granåsen, M., Granåsen, M., Andersson, D., & Andersson, D. (2016). Measuring team effectiveness in Cyber-defense exercises: A cross-disciplinary case study. *Cognition, Technology & Work, 18*(1), 121-143. doi:10.1007/s10111-015-0350-2

[3] Jiangjun Tang; George Leu; Hussein A. Abbass, "Computational Red Teaming for Air Traffic Management," in Simulation and Computational Red Teaming for Problem Solving, *IEEE*, 2020, pp.263-272, doi: 10.1002/9781119527183.ch15.

[4] Karim, N. S. A., Albuolayan, A., Saba, T., & Rehman, A. (2016). The practice of secure software development in SDLC: An investigation through existing model and a case study. *Security and Communication Networks, 9*(18), 5333-5345.

[5] Md Tarique, J. A., & Pandey, D. (2017). An integration of threat modeling with attack pattern and misuse case for effective security requirement elicitation. *International Journal of Advanced Research in Computer Science, 8*(3)

[6] N. Teodoro and C. Serrão, "Web application security: Improving critical web-based applications quality through in-depth security analysis," *International Conference on Information Society (i-Society 2011)*, London, 2011, pp. 457-462, doi: 10.1109/i-Society18435.2011.5978496

[7] Priya, S. S., & Arya, S. S. (2016). Threat modeling for a secured software development. *International Journal of Advanced Research in Computer Science*, 7(1)

[8] Raghavan, V., & Zhang, X. (2017). An integrative model of managing software security during information systems development. *Journal of International Technology and Information Management, 26*(4), 83-109.

[9] Wagner, N., Şahin, C. Ş., Winterrose, M., Riordan, J., Hanson, D., Peña, J., & Streilein, W. W. (2017). Quantifying the mission impact of network-level cyber defensive mitigations. *Journal of Defense Modeling and Simulation, 14*(3), 201-216. doi:10.1177/1548512916662924

Books

[10] Hickey, M. (09/09/2020). Hacking ethically and legally. (2020). Indianapolis, Indiana: John Wiley & Sons, Inc. doi:10.1002/9781119561507.ch2

[11] Hsu, T. H., & ProQuest Ebooks. (2018). *Hands-on security in DevOps: Ensure continuous security, deployment, and delivery with DevSecOps*. Birmingham: Packt Publishing, Limited. Chaps. 2, 3, 5, 20.

[12] Sheward, Mike. *Security Operations in Practice*, BCS. Learning & Development Limited, 2020, Chaps. 2, 9.