PERFORCE

WHITEPAPER

How to Improve Cybersecurity for Tomorrow's Aerospace and Defense Needs

Introduction

Without exception, today's military and aerospace programs are assessing and revising their cybersecurity strategies. Evolutions in technology and doctrine are calling for systems that push data, connectivity, and intelligence to new heights, leading to an increasing number of potential attack surfaces and vectors. As Motherboard explains about a recent presentation from the Pacific Northwest National Laboratory:

"[The] potential of catastrophic disaster is inherently greater in an airborne vehicle... [and it's only] a matter of time before a cyber security breach on an airline occurs."

As vulnerabilities grow more frequent and complex, development teams must mitigate their risks by understanding and implementing the secure development practices explained in this white paper.

Contents

The Biggest Forces Driving Change

The biggest forces driving change in military and aerospace cybersecurity include:

THE DATA-DRIVEN BATTLEFIELD

Information is as important as ordnance in today's mission operations. The efficient exchange of mission-critical data is the difference between winning and losing.

From battlefield vehicles to network operations at the edge, every node presents an attractive attack vector for malicious entities. Reflected in newer military doctrines and guidance, the data-driven battlefield must be protected like a global enterprise, with strategies covering threat prevention, protection, detection, attribution, and reaction.

For example, the Air Force EMSO doctrine states:

"Electromagnetic spectrum operations (EMSO) comprise all coordinated military actions to exploit, attack, protect, and manage the electromagnetic environment (EME) to achieve the commander's objectives."

SYSTEM MODERNIZATION

Maintaining persistent awareness, understanding, and responsiveness on the digital battlefield requires advanced systems to match. To keep pace with tight budget constraints and increasing global threats, many agencies have made the shift from legacy acquisition programs to modernization projects, laying the foundation for next-generation capabilities in unmanned systems, artificial intelligence (AI), and robotics.

As outlined in the United Kingdom Ministry of Defense's Defense in a Competitive Age:

"Our response will be founded on modernization and integration. We will play to our traditional strengths whilst acknowledging that we will need to adapt to be ready for the threats of the future."

The relationship between modernization and cybersecurity is a double-edged sword. Without upgrades, military systems are vulnerable to legacy vulnerabilities and increasing amounts of cyberattacks. Yet upgrades may also increase the likelihood of a newly exposed attack surface.

INTEROPERABILITY

While coordination between systems, operations, and nations isn't a new concept, today's interoperability needs are inextricably linked with technology. Devices must share signals with each other, protocols must be compatible, and the chain of data from source to analysis must be unbroken.

In the rush to integrate systems and be first to field, it's hard to assess the complex web of software and services that make up the total threat landscape.

RAPID ADAPTABILITY

Given the fluid nature and complex characteristics of the future digital landscape, the ability to rapidly update systems safely and securely will be a key differentiator. As the Department of Defense Software Modernization Strategy explains:

"The vision for software modernization is simple - deliver resilient software capability at the speed of relevance. Resilience implies software that is high-quality and secure, able to withstand and recover in the face of challenging conditions. Speed of relevance implies the accelerated delivery needed to maintain a competitive advantage."

For military and aerospace in particular, rapid adaptability presents a unique challenge, as some embedded systems are in the field for years – often with no over-the-air (OTA) access. With longer service life comes the need to harden security upfront, before deployment, to reduce attack risks down the line.

COMPLIANCE WITH REGULATORY STANDARDS

Attempting to address all these forces together are government and industry standards that guide, and often enforce, secure development and delivery of best practices. These guidelines cover everything from requirements definition to secure testing during development. Without compliance, the customer will often not accept the software.

Standards are especially important for securing the software supply chain, as adversaries tend to look at the lower tiers to exploit vulnerable links.

These market forces are why an effective cybersecurity strategy must connect today's development tools with tomorrow's needs.

Today's Multi-Dimensional Cybersecurity Risk Landscape

Malicious actors leverage cybersecurity holes to steal data, cause damage, and disrupt computing systems. For military and aerospace, these threats also impact mission success and personnel safety. Vulnerabilities arise through programming and process deficiencies – at any stage of the development lifecycle – that could lead to an adversary gaining access to the system.

For military and aerospace software, these deficiencies are introduced through:

- **Legacy Software:** Older code comprises the majority of systems today and pose unique cybersecurity challenges, like the absence of maintainer support and the lack of testing against the latest vulnerabilities.
- **New Development:** While the latest code should be subject to the latest security testing techniques and standards, the perception that security is expensive and the desire to reduce project timelines usually leaves minimal resources for security testing and remediation.
- **Supply Chain/Commercial Off-the-Shelf (COTS):** Like OSS, incorporating commercial third-party software into other applications may also incorporate vulnerabilities. The Log4j vulnerability is an example of both an OSS and supply chain exposure and while not an intentional attack, it caused millions of devices to be vulnerable.

No matter the origin, code introduced into the system may have malware, bugs, or other vulnerabilities that are unknown to the developer. Without a strategic approach to analyzing and vetting all software, teams will remain unaware of threat vectors embedded in their products.

A Matter of Standards

The military and aerospace sector has many standards and certifications for software intended to be deployed in live environments. These standards range from coding best practices designed to prevent vulnerabilities from entering the code base (e.g., DISA STIG, CWE Top 25) to guidance on the top vulnerabilities, data protection, and recommended security controls that apply to organizations as a whole (e.g., OWASP, NIST).

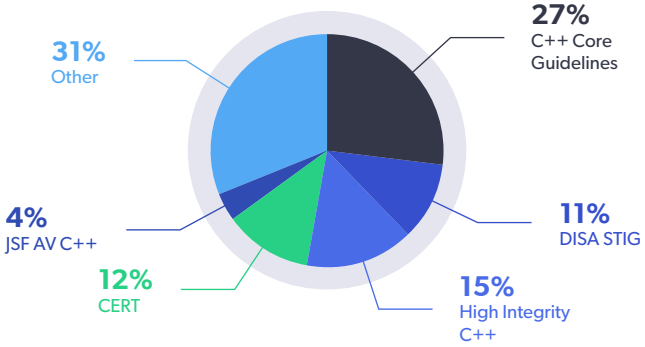

Figure 1: Top coding standards used by software development professionals in aerospace and defense (Source: State of Aerospace & Defense Software Development Survey Results)

By adopting processes and tools that adhere to industry standards, development teams don't have to reinvent the security wheel. For military and aerospace, the following secure coding standards are directly applicable and often mandated for suppliers.

COMMON WEAKNESS ENUMERATION (CWE)

The Common Weakness Enumeration is a community-developed list of cybersecurity weaknesses in software and hardware. The CWE Top 25 lists the most widespread and critical weaknesses that could lead to severe software vulnerabilities.

CERT

CERT Coding Standards are language-specific guidelines developed by a community of software development and software security professionals. Supporting C, C++, and Java, each guideline includes a risk assessment to help determine the possible consequences of violating that specific rule or recommendation.

DISA STIG

The Defense Information Systems Agency (DISA) Security Technical Implementation Guide (STIG) provides guidance on how an organization should handle and manage security software and systems. This includes three compliance levels, called categories, that indicate the severity of the risk of failing to address a particular security weakness.

OWASP

The Open Web Application Security Project (OWASP) is a non-profit foundation focused on improving software security through tools and knowledge. The OWASP Top 10 specifies 10 critical security risks for applications, based on analyzing the exploits most often used by hackers and causing the most damage.

MISRA AND MISRA C:2012

Developed and maintained by manufacturers, component suppliers, and engineering consultancies, the MISRA standard provides coding guidelines for C and C++ that ensure code is safe and secure. The MISRA C:2012 addenda include guidelines to strengthen security by identifying how each MISRA rule maps to the secure coding rules in ISO/IEC TS 17961:2013 and CERT C.

By demonstrating adherence to these standards, manufacturers meet compliance requirements and boost their credibility to stakeholders, customers, and potential partners.

Modernizing Software Development

Protecting against cybersecurity attacks requires elevating their importance within the culture and infrastructure of software development.

"Recognition and acceptance for the need to change from legacy ways of conducting development, procurement, and execution of operations in this highly dynamic environment has spread across the Department of Defense (DoD) and within the militaries of our allies. This is especially true regarding how software development, procurement, delivery, and execution connects all actions across the battlespace."

– Joint All-Domain Operations: Redesigning the Ecosystems of Engagement, Professional Services Council

The following are recommendations for an effective application security improvement program.

MAKE SECURITY A DEVELOPMENT PRIORITY

Improving security comes with deliberate investments into training, tools, and processes to detect and remediate vulnerabilities before they're released. These efforts must be led from the top of the development organization and start with understanding what various teams are doing now and defining a roadmap to overcome any gaps.

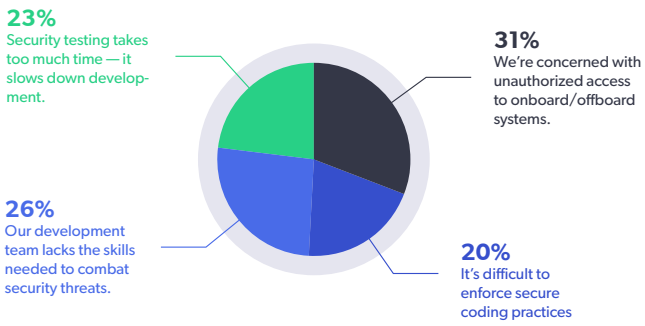

Figure 2: Top security concerns for software development professionals in aerospace and defense (Source: State of Aerospace & Defense Software Development Survey Results)

The goals to consider when planning an application security improvement program include:

- Analyzing and reporting on the security health of applications and code bases.
- Improving compliance with industry standards and secure coding practices.
- Reducing the cost of finding and remediating security vulnerabilities.
- Creating an incident response plan that defines actions to take in the event of a security breach.
- Ensuring security training, processes, and best practices are standardized across teams.

TRAIN DEVELOPERS AND QA

Adopting a cybersecurity training program ensures the team is familiar with the policies and tools that better protect the organization. The lack of training was a top concern in our recent State of Aerospace & Defense Software Development Survey, as shown in Figure 2.

Beyond the education and tactical skills development, ensuring everyone is aware of cybersecurity risks and remediation efforts helps shift individual thinking from "security as a destination" to security as an ongoing discipline.

UNDERSTAND DEVSECOPS

The software landscape will only increase in complexity, as will the frequency of cyber threats, necessitating the need to include rapid adaptability into development pipelines. DevSecOps gets developers thinking about security first, streamlining the integration of secure coding and testing techniques into workflows without impacting release timelines.

DevSecOps augments traditional DevOps phases with security-specific actions. Based on threat modelling and scanning, DevSecOps pipelines analyze data and metrics to remediate vulnerabilities. Static application security testing (SAST) tools, for example, are a key enabling technology for DevSecOps, helping identify issues early and recommending solutions for vulnerabilities, errors, and bugs.

SECURE YOUR SUPPLY CHAIN

"The Federal Government must take action to rapidly improve the security and integrity of the software supply chain, with a priority on addressing critical software."

– Executive Order on Improving the Nation's Cybersecurity

Supply chain security isn't a new concept for military and aerospace developers, but its importance has never been higher. Malicious attackers are savvy to the exploitability of code at lower levels of the supply chain and teams must strongly consider the influence of procured software, open source code, and SaaS tools on deployed products.

Formulating and implementing a secure supply chain strategy includes:

- Creating, communicating, and enforcing supply chain security policies to all development teams, IT, partners, and vendors.
- Adopting tools and processes to validate newly procured code and audit existing ones.
- Maintaining a database of all third-party software packages and versions in use by development teams.

USE AUTOMATED TOOLS

Automated tools are an essential component of an application security improvement program because they reduce the need for human input and intervention. With the lack of resources and potential for human error, automated security testing can provide comprehensive code coverage and check against the latest vulnerabilities and exploits. They can also overcome threats that are hard to detect, or even designed to be undetectable by humans.

Conclusion

Cyberattacks on military and aerospace applications will evolve over the lifespan of systems in ways that cannot be foreseen by the original developers. As vulnerabilities grow more frequent and complex, development teams must mitigate their risks with secure development practices and automated testing tools.

The business and reputational realities have never been clearer, as are stronger customer and government mandates to improve cybersecurity. Security leaders and software developers must enable their organizations to embrace the fact that securing their code doesn't mean slowing the team down; rather, it's about implementing automated tools that make it easier to prioritize security.

Why SAST Is a Key Enabling Technology for Aerospace and Defense Cybersecurity

Automated security testing of code comes in two types: dynamic application security testing (DAST) and static application security testing (SAST). SAST tools, like Klocwork, analyze source code to find vulnerabilities and identify gaps in compliance across the entire code base. These technologies are easy to integrate into existing workflows and infrastructure and can scale to enterprise-size code coverage.

Beyond structural code analysis, Klocwork provides a deep-dive, data-flow analysis that extends all the way out to the leaf nodes of a call chain, as well as across procedures, translation units, and dependent components that encompass a project's software bill of materials (SBOMs). This deep analysis automatically discovers defects and vulnerabilities that are otherwise extremely difficult for human code reviewers to identify.

Klocwork analyzes source code as it's being written, helping developers detect and address security vulnerabilities at the earliest stage of the lifecycle.

Combined with built-in capabilities that support the speed of DevSecOps, Klocwork supports the earliest and fastest detection of code vulnerabilities, compliance issues, and rule violations in the development pipeline.

HOW KLOCWORK SUPPORTS APPLICATION SECURITY IMPROVEMENT FOR AEROSPACE AND DEFENSE

- Detects code vulnerabilities, compliance issues, and rule violations earlier in application development.
- Enforces industry and coding standards, including CWE, CERT, and OWASP.
- Supports the NIST risk management framework (RMF) through discovery of DISA ASD STIG rule violations with corresponding compliance reporting for certified information security managers (CISMs).
- Reports on compliance across product versions.
- Provides a proprietary integration of the Secure Code Warrior training modules in the context of a detected defect with a clear explanation of a specific vulnerability, empowering developers to be more security aware coders.
- Supports a shift-left approach — analysis available everywhere, including developer desktop and DevSecOps pipelines.
- Delivers fast feedback and provides the exact location of vulnerabilities and their cause.

See for yourself how Klocwork helps ensure that your embedded software is secure, compliant, and reliable. Request your free 7-day trial today.

REGISTER NOW