WABBI

DEVSECOPS FUNDAMENTALS LEARNING MODULE

This module is designed to introduce the fundamental concepts of DevSecOps, aimed at helping individuals understand the importance of building secure code as part of an organization’s overall operational resilience.

WABBI

INTRODUCTION TO DEVSECOPS

Objectives:

- Understand what DevSecOps is beyond the buzzwords
- Recognize why DevSecOps and Operational Resilience go hand in hand

WABBI

What is DevSecOps: DevSecOps stands for Development, Security, and Operations, and it integrates security into the DevOps process. In contrast to traditional development methods where security is introduced late in the development lifecycle, DevSecOps ensures security is integrated from the outset.

Not implementing DevSecOps can negatively impact both development and security:

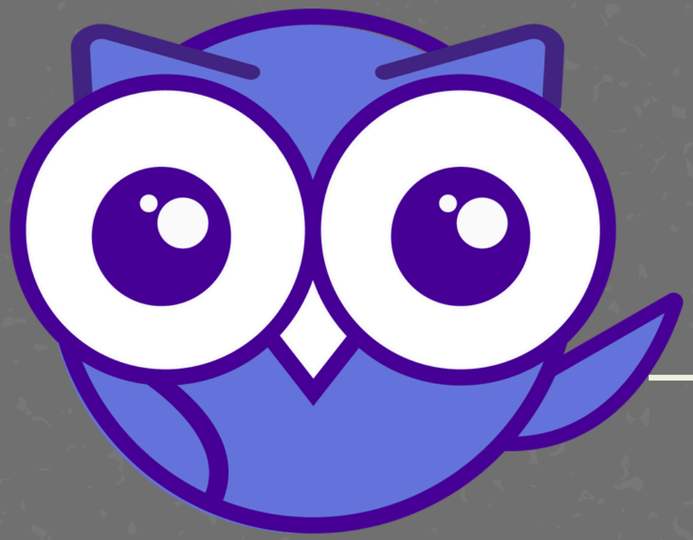
1. **Delayed Security Fixes:** Without security integrated into the development pipeline, vulnerabilities are often discovered late, leading to costly last-minute fixes and increased time to market.
2. **Increased Risk of Breaches:** Failing to incorporate continuous security practices can leave code exposed to undetected vulnerabilities, heightening the risk of cyberattacks and data breaches**.**
3. **Siloed Teams and Inefficiency:** Without DevSecOps, development, security, and operations teams may work in isolation, causing miscommunication, slower response times, and inefficient handling of security issues.

breaches begin due to defects in code

DHS

WABBI

UNDERSTANDING SECURITY DEFECTS



Objectives:

- Understand what makes up vulnerabilities
- Learn about real life translations

WABBI

WHAT MAKES UP A DEFECT?

Weakness

A condition in a software, firmware, hardware, or service component that could become a vulnerability

Vulnerability

A weakness that can be exploited by a threat source

CWE

Common Weakness Enumeration is a community-developed list of common software and hardware weaknesses

CVE

Common Vulnerabilities & Exposures is a publicly released list of known computer security threats

CVSS

Common Vulnerability Scoring System is a standardized ranking system for reported vulnerabilities

WABBI

VULNERABILITY EXPLOITS

REAL-WORLD EXAMPLES

Equifax (2017)

Impacting the PII of >40% of the U.S. population, this breach started with a known vulnerability in Apache Struts (CVE-2017-5638) which Equifax failed to patch due to a failure in internal processes, despite internal adminstrators being instructed to to apply the patch 2 days after its release. It was months later that the attackers began to exfiltrate data.

Additionally, while data governance practices should have protected against this, the attackers were able to remove the data undetected because Equifax had failed to renew a crucial public-key certificate 10 months earlier, preventing their security tools from inspecting encrypted network traffic. It was only upon the discovery of non-renewed certificate that the breach was discovered.

Read more about the story here.

Log4j (2021)

The Log4j vulnerability, known as Log4Shell (CVE-2021-44228), was discovered in December 2021 in Minecraft. A Remote code execution (RCE) weakness, it allowed attackers to take control of vulnerable systems remotely by sending a specially crafted log message that did not require any authentication to perform it.

Exploit code quickly appeared on GitHub. Attackers exploited the Log4j vulnerability globally, deploying cryptomining malware to hijack systems, using ransomware groups like Conti for attacks, and expanding botnets such as Mirai and Tsunami. State-sponsored actors from China, North Korea, and Iran also leveraged the flaw for espionage, while cloud services and enterprise platforms were targeted for sensitive data theft and infrastructure compromise.

Read the whole story here, here and here.

MISCONFIGURATION EXPLOITS

REAL-WORLD EXAMPLES

Target (2013)

The Target breach of 2013 was one of the largest retail data breaches in history, affecting over 40 million credit and debit card accounts and personal information of 70 million customers. The breach began when hackers infiltrated Target's network through a third-party vendor, a HVAC company, by stealing their credentials. Once inside, the attackers moved laterally through Target's internal systems, eventually gaining access to sensitive customer data stored in the company's point-of-sale (POS) systems.

A critical factor in the breach was a firewall misconfiguration. Target had firewalls in place, but these were not properly segmented to isolate sensitive parts of the network from less secure sections. As a result, once the attackers gained access, they were able to move across Target’s network without sufficient restrictions. This misconfiguration, coupled with missed alerts from security systems, made the breach both more severe and longer-lasting.

T-Mobile (2021)

In August 2021, T-Mobile experienced a major data breach caused by an API misconfiguration, leading to the exposure of personal data of over 40 million customers. The attackers exploited an improperly secured API, which allowed unauthorized access to sensitive customer information, including names, Social Security numbers, birth dates, and driver’s license details. The breach affected both current and former customers, as well as potential customers who had applied for credit with T-Mobile.

The API misconfiguration essentially allowed attackers to bypass authentication mechanisms and access the data directly. Once inside, the hackers harvested the information and later offered it for sale on dark web forums. This breach highlighted the importance of secure API design and proper configuration to prevent unauthorized access.

Read more here and here.

WABBI

GETTING STARTED WITH SECURE CODING

Objectives:

- Adopting a threat-modeling mindset
- Learn about the kinds of vulnerabilities

WABBI

THREAT MODELLING

Threat modeling is a process used to identify, assess, and address potential security risks in a system. It helps organizations understand vulnerabilities and design safeguards to protect against potential attacks.

Adopt a Threat Model Mindset:

- **Integrate Security Early**: Start thinking about security at the design phase by considering potential vulnerabilities and attack vectors before writing any code.
- **Collaborate Across Teams:** Regularly communicate with cross-functional teams (developers, security, and ops) to identify and address threats from multiple perspectives.
- **Use Threat Models for Code Reviews:** Apply threat modeling principles during code reviews to spot security flaws and address potential risks before deployment.

WABBI

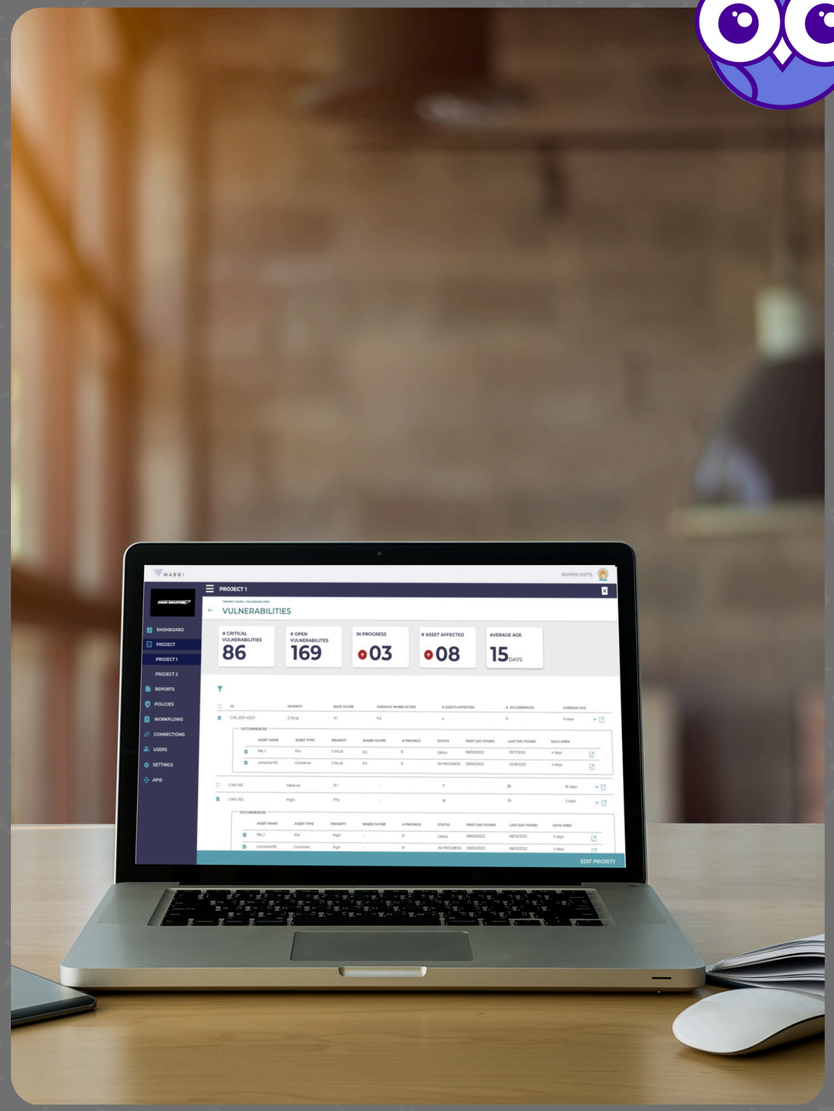
UNDERSTANDING VULNERABILITIES IN THE DEVELOPMENT WORKFLOW

Objectives:

Recognize the different types of vulnerabilities and how they impact software security during development.

Understand the role of DevSecOps in early detection and mitigation of vulnerabilities during the coding phase.

WABBI



What is a Vulnerability?

A vulnerability is a flaw or weakness in an application that can be exploited to compromise systems or data. Vulnerabilities can emerge during any stage of development and can affect the performance, security, and reliability of applications.

Common Types of Vulnerabilities:

- Injection Flaws (e.g., SQL injection)
- Cross-Site Scripting (XSS)
- Broken Authentication
- Misconfigured Security Settings

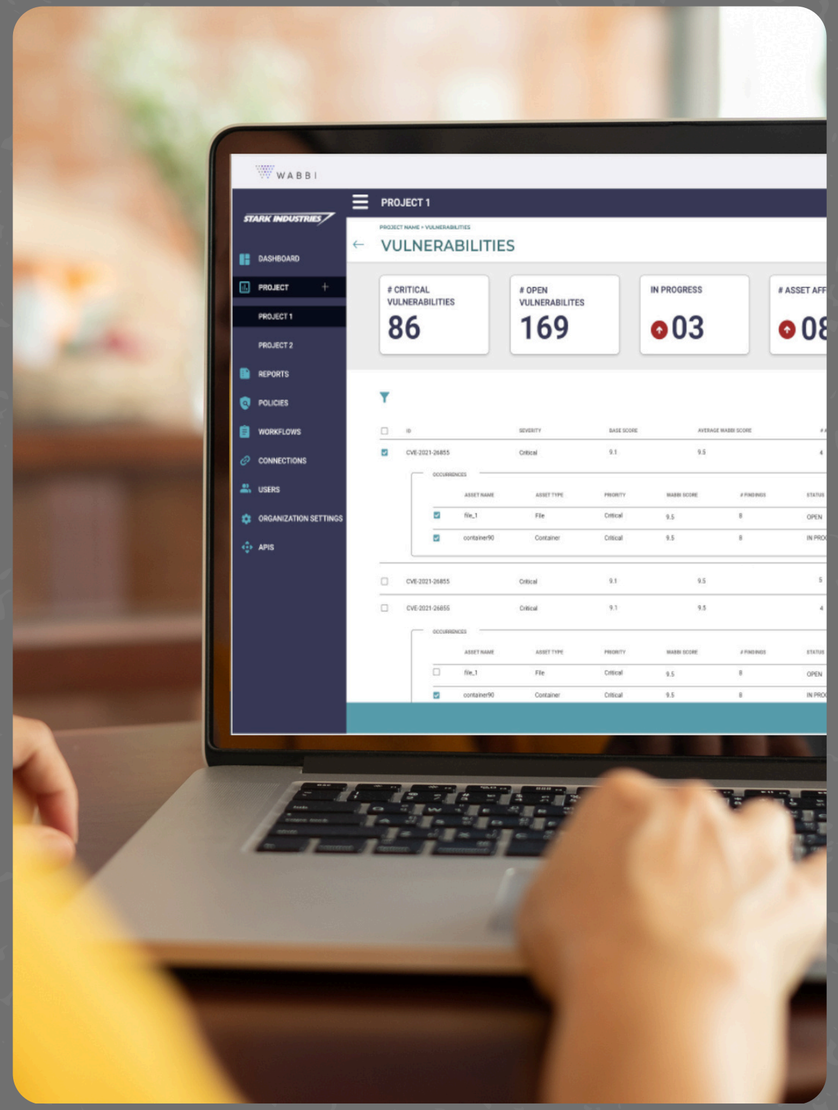
WABBI

Context in DevOps:

In a continuous development environment, such as those used in DevOps, vulnerabilities can quickly proliferate. Integrating automated tools to scan for these issues during both coding and deployment helps in identifying and addressing vulnerabilities before they become critical.

Tools and Techniques:

- **SAST (Static Application Security Testing):** Scans the codebase for vulnerabilities as developers write it.
- **DAST (Dynamic Application Security Testing):** Simulates attacks on the application while it's running, testing for exploitable vulnerabilities.



WABBI

INJECTION VULNERABILITIES

- **Definition:** Injection vulnerabilities occur when untrusted data is sent to an interpreter as part of a command or query. The most common form is SQL injection, but other types include command injection and LDAP injection.
- **Impact:** This can lead to data leakage, corruption, or unauthorized access to sensitive information. Attackers may execute arbitrary commands or queries, potentially controlling the system.
- **DevSecOps Role:** By integrating security checks early, DevSecOps can help prevent injection attacks by incorporating secure coding practices and regular code scanning for injection flaws.

WABBI

CROSS-SITE SCRIPTING (XSS)

- **Definition:** XSS occurs when an attacker injects malicious scripts into content that is then executed by another user's browser. It exploits vulnerabilities in web applications that don't properly validate user input.
- **Impact:** Attackers can steal session tokens, deface websites, or redirect users to malicious sites.
- **DevSecOps Role:** Automating input sanitization and output encoding during the development process helps minimize the risks of XSS vulnerabilities.

WABBI

BROKEN AUTHENTICATION AND SESSION MANAGEMENT

- **Definition:** This vulnerability arises when applications do not correctly handle authentication tokens, session identifiers, or passwords. Attackers exploit flaws in the authentication mechanisms to impersonate legitimate users.
- **Impact:** Attackers can gain unauthorized access to sensitive data or functions by hijacking user sessions or bypassing authentication altogether.
- **DevSecOps Role:** Continuous monitoring and testing of authentication protocols through tools in the CI/CD pipeline ensure robust protection against such vulnerabilities.

WABBI

SECURITY MISCONFIGURATION

- **Definition:** Misconfigurations happen when security settings in an application, server, or database are not implemented correctly or remain at default values.
- **Impact:** This can open doors for attackers to exploit weak passwords, outdated software, and unpatched systems.
- **DevSecOps Role:** Automating configuration management and continuous scanning for outdated components help mitigate misconfigurations.

WABBI

ACTIVITY

Visit the National Vulnerability Database and read some of the latest vulnerability reports.

Read how to use CWEs in the primer and find how you can use them in your job in their user stories.

RESOURCES

WABBI

- Read about CWE
- What is DevSecOps
- The Future of DevSecOps with AppSec
- Understand CVSS
- Visit the National Vulnerability Database and read some of the latest vulnerabilities
- Why DevSecOps Has Failed to Fulfill its Promise
- Decoding the AppSec Alphabet Soup
- Understand CVE