LECTURE 1: INTRODUCTION

Introduction

Why is Software Security Important?

What Is the Difference Between Software Security and Cybersecurity?

What are the three types of software security?

Major Concerns with Software Security

Software Security Tools and Responsibilities

INTRODUCTION

Software security refers to the practices and measures taken to protect software systems from vulnerabilities, threats, and attacks that could compromise their confidentiality, integrity, and availability.

Software security is critical because software vulnerabilities can lead to cyber-attacks, data breaches, and major disruptions of computer systems.

Software security aims to reduce risks by identifying threats early, designing secure architecture, following best coding practices, and testing rigorously. Implementing software security measures has become essential for organizations to protect their assets and customers in an increasingly interconnected digital world.

INTRODUCTION

Software security provides assurance that software systems function reliably and securely as wanted, despite malicious attempts to compromise them, careful attention to software security helps build resilient systems and foster trust in the growing role of software.

Software security requires a combination of secure design, developer training, automated analysis, testing rigor, and ongoing vulnerability monitoring , With increasing connectivity and complexity, the risks multiply each day a software security gap persists unaddressed.

WHY IS SOFTWARE SECURITY IMPORTANT?

Software security solutions help ensure data is protected while in transit and at rest. They also help protect against system vulnerabilities like malware and ransomware attacks. According to Veracode's State of Software Security 2024, as many as 94% of all the apps they tested had at least one security flaw.

Secure software development is incredibly important because there are always people out there who seek to exploit business data. As businesses become more reliant on software, these programs must remain safe and secure. With strong software security protocols in place, we can prevent attackers from stealing potentially sensitive information such as credit card numbers and trade secrets, and build trust among users. The theft of critical data can be catastrophic for customers and businesses alike. Malicious actors can abuse sensitive information and even steal users' identities. Additionally, companies can face legal penalties in the event of a data breach and suffer reputational harm.

Businesses can work to protect critical data by implementing software security techniques into their development life cycles. Applying security techniques enables organizations to proactively identify system vulnerabilities and better protect their software.

WHAT IS THE DIFFERENCE BETWEEN SOFTWARE SECURITY AND CYBERSECURITY?

While the terms “software security” and “cybersecurity” may sound interchangeable, they actually refer to two different concepts.

Software security focuses specifically on safeguarding software applications from threats, and attacks. This involves ensuring that the application code, data associated with the application, and the infrastructure supporting the application are secure.

Software security practices are integrated into the software development lifecycle (SDLC) to identify and mitigate risks within the software itself, making sure the application behaves securely in various operating environments.

By contrast, cybersecurity is a broader field that focuses on securing all types of systems. It protects networks, systems and programs. It’s usually applied at the level of an entire organization and is responsible for securing all internal and external company systems through technological or physical means.

TYPES OF SOFTWARE SECURITY?

There are three types of software security:

1. Application security

Application security involves ensuring that the code is secure by identifying and fixing vulnerabilities within the software itself. This includes practices such as code reviews, automated security scanning, secure coding practices, input validation, and penetration testing to ensure that the application is resilient against attacks.

2. Data security

It’s vital to protect the data that your application processes from unauthorized access. You can do this by encrypting data while it’s in storage or transit, using data masking techniques for sensitive data, and adding monitoring processes to any data movement. Additionally, applying role-based access controls can ensure the integrity of your data.

3. Infrastructure security in software applications

Infrastructure security involves securing the environment where the software operates, ensuring that the underlying systems and networks that support the application are protected. This includes monitoring network traffic for suspicious activities, setting up firewalls, ensuring secure configurations of servers and databases, and regularly updating and patching the infrastructure components.

MAJOR CONCERNS WITH SOFTWARE SECURITY

A security vulnerability can have major implications for healthcare organizations, financial institutions, homeland security agencies and more. It is important to identify these concerns quickly and proactively to avoid malicious attacks. Below are some of the top software security issues businesses are facing:

1.Phishing: Phishing happens when an attacker poses as someone else in an attempt to gain personal information, such as software credentials.

2.Distributed denial of service (DDoS) Attacks: A DDoS attack happens when an attacker overloads servers with packets, causing the software to crash.

3.Cloud service attacks: Companies are increasingly relying on cloud-based services to support remote workers. Some cloud infrastructure has vulnerabilities hackers can exploit.

SOFTWARE SECURITY TOOLS AND RESPONSIBILITIES

Building secure software is a group effort. All stakeholders in software development, from developers to executives, need to understand how software security practices benefit them. They must also understand the risks of not implementing them and allocate proper resources to security tasks.

There are several tools that an organization can leverage for software security:

1.Static application security testing: This tool examines source code at rest and flags vulnerabilities for developers to fix.

2.Dynamic application security testing: This tool examines an application's code while it is running and detects weaknesses in the software.

3.Software composition analysis: This tool checks for vulnerabilities against a software's governance guidelines. Software composition analysis is especially valuable for open source software.

4.Mobile application security testing: This tool analyzes mobile code to identify specific vulnerabilities that could lead to unique security risks, such as improper platform usage and insecure data storage.