Date:23/02/2023

GROUP F ASSIGNMENT 1

Group F Members:

23862 SHEMA David
23702 NZITATIRA Hirwa Kevin
22064 SHEMA Vierry
23169 NIYONZIMA Nshuti Fabrice
24298 MANDALI Innocent
23071 MUNYANEZA Jean Francois
23200 UMUKUNDWA Emmanuella
22956 NSENGIYUMVA Alain Richard

SOFTWARE SECURITY ASSIGNMENT 1

1. Threat actors are individuals, groups, or entities that pose a potential risk to the security of an organization or system. They are typically individuals or groups that have the capability to exploit vulnerabilities in a system or network, and use this access to achieve their own goals. Threat actors can be categorized into several types, including:

Nation-state actors: Nation-state actors are a type of threat actor that are sponsored or directed by a nation-state or government entity. They are often motivated by political or strategic objectives, such as gathering intelligence, conducting espionage, or disrupting the activities of other nations. Nation-state actors have a wide range of resources and capabilities at their disposal, including sophisticated tools and techniques, advanced technologies, and highly skilled personnel.

Cybercriminals: are individuals or groups who use technology and the internet to carry out criminal activities such as theft, fraud, extortion, and other forms of illegal activity. They often use sophisticated techniques and tools to exploit vulnerabilities in computer systems, networks, and software to gain unauthorized access and steal information or money.

Hacktivists: Hacktivists are individuals or groups who use their hacking skills to promote a political or social cause. They may carry out cyber-attacks, deface websites, or steal and leak sensitive information in order to draw attention to their cause or to damage the reputation of a particular organization or government.

Hacktivism can take various forms, including distributed denial-of-service (DDoS) attacks, website defacements, and data breaches. Hacktivists may target government agencies, corporations, or other entities that they perceive as being in opposition to their cause.

Some of the most well-known hacktivist groups include Anonymous, LulzSec, and AntiSec, who have carried out a number of high-profile attacks over the years. Hacktivists may also use social media platforms to spread their message and coordinate their activities.

Insiders: These are employees, contractors, or other individuals with authorized access to an organization's systems or data who use that access to carry out malicious activities. Insider threats can be intentional, such as when an employee steals data for financial gain or to sell to a competitor, or unintentional, such as when an employee accidentally exposes sensitive information due to carelessness or lack of training.

Insiders can pose a significant risk to an organization's security because they often have knowledge of the organization's systems and processes and can use that knowledge to carry out attacks that are difficult to detect. Insider attacks can take various forms, including data theft, sabotage, and fraud.

In conclusion, nation-state actors, cybercriminals, hacktivists, and insiders make up the four main threat actor groups. Each of these organizations has different reasons and aims, and they each employ various strategies, methods, and practices to get there. Creating effective cybersecurity tactics and reducing the risk of cyberattacks require an understanding of these groups' features.

2. The Software Security Knowledge Catalog (SSKC) is a collection of seven knowledge catalogs developed by the Software Assurance Forum for Excellence in Code (SAFECode). These catalogs are designed to provide software developers, architects, and security professionals with a comprehensive understanding of software security and the skills needed to build secure software. Here are the seven catalogs:
Fundamentals: The Fundamentals catalog provides an overview of software security concepts, principles, and best practices. It covers topics such as threat modeling, secure coding, security testing, and vulnerability management.

Secure Software Development: During the whole software development life cycle, the Secure Software Development catalog focuses on the abilities and knowledge required to create secure software (SDLC). It addresses issues including secure coding standards, secure testing procedures, and security needs.

Secure Software Architecture and Design: The concepts and methods of secure software architecture and design are covered in the Secure Software Architecture and Design library. Threat modeling, security patterns, secure communication protocols, and security in distributed systems are some of the subjects covered.

Secure Software Testing: The expertise required to test software for security flaws is covered in the Secure Software Testing catalog. It covers subjects including static code analysis, fuzz testing, vulnerability scanning, and penetration testing.

Secure Operations: The concepts and procedures of secure software operations are covered in the Secure Operations catalog. It covers subjects including secure configuration management, secure deployment, incident response, and security monitoring.

Supply Chain: The supply chain management practices and principles are included in the supply chain catalog. It covers subjects including vendor risk management, third-party software evaluation, and software bill of materials (SBOM).

Legal and Compliance: The legal and regulatory requirements for software security are covered in the Legal and Compliance category. It covers issues including data security, privacy, intellectual property, and export restrictions.

3. a**. Cross-Site Scripting Vulnerability (XSS Cross-Site Scripting (XSS)** is a type of vulnerability that allows attackers to inject malicious scripts into web pages viewed by other users. XSS attacks occur when an application fails to properly validate or sanitize user input, allowing an attacker to inject code that is executed in the victim's browser.

b. **SQL Injection Vulnerability:** This is a type of security vulnerability that allows attackers to execute malicious SQL statements on a web application's backend database. SQL injection occurs when an application fails to properly sanitize user input before passing it to an SQL database query. This allows an attacker to insert SQL commands into the input, which are then executed by the database.

The consequences of SQL injection attacks can be severe, including data theft, data destruction, and unauthorized access to sensitive data. SQL injection attacks can be carried out in a number of ways, including inserting SQL commands into input fields on a web form, modifying HTTP request headers, and tampering with cookies.

c. **Broken Authentication and Session Management Vulnerability:** Broken Authentication and Session Management is a type of vulnerability that occurs when web applications fail to properly manage user authentication and session management. This can allow attackers to gain unauthorized access to user accounts and sensitive data.

The Broken Authentication and Session Management vulnerability can occur due to a range of factors, such as weak or easily guessable passwords, insecure session management practices, and session fixation attacks.

d. **Insecure Direct Object Reference Vulnerability**: Insecure Direct Object Reference (IDOR) is a type of vulnerability that occurs when an application allows a user to directly access and manipulate an object, such as a file, record, or resource, without proper authorization checks. This can lead to unauthorized access and modification of sensitive data or functionality.

An IDOR vulnerability arises when an application relies on user-supplied input, such as a parameter or identifier, to access an object directly, without validating whether the user is authorized to access that object. For example, an application might use a user ID as an input parameter to access a user's profile page, without checking if the user is authenticated and authorized to access that page.

e. **Cross-Site Request Forgery (CSRF**): Vulnerability Cross-Site Request Forgery (CSRF) is a type of security vulnerability that occurs when an attacker tricks a user into performing an unwanted action on a web application, using the user's authenticated session. CSRF attacks can allow an attacker to perform actions such as transferring funds, changing account details, or submitting malicious content, by using the victim's authenticated session to submit forged requests to the application.

CSRF attacks can occur when an application does not implement sufficient CSRF protections, such as using unique tokens to verify that a request is legitimate. Attackers can exploit this vulnerability by tricking a user into clicking on a specially crafted link, visiting a malicious website, or opening a malicious email attachment, that contains a forged request to the vulnerable application.