Introduction to Web Vulnerabilities

Welcome to the World of Web Security

Understanding and Hunting the OWASP Top 10

A01:2021 – Broken Access Control

What is it?

Broken Access Control refers to inadequate access control mechanisms that allow unauthorized users to access restricted functionalities or data. Failures may lead to unauthorized data exposure, modification, or destruction.

Common Vulnerabilities

- Bypassing access control checks.
- Insecure direct object references (IDORs)
- Missing access controls for API.
- Elevation of privilege.
- CORS misconfiguration.
- Force browsing.

Hunting Tips

- Test for URL parameter tampering, such as changing a user ID.
- Attempt to access privileged pages as a standard user.
- Look for exposed API endpoints that accept requests without authentication.

Example Attack Scenarios

1. Modifying the 'acct' parameter in an SQL call allows you to read data about a different user account.
2. Unauthorized access to admin pages by using a match and replace rule of “admin=true”.
3. Forging requests to gain unauthorized privileges.

A02:2021 – Cryptographic Failures

What is it?

Cryptographic Failures are issues stemming from the improper use of cryptographic functions, often leading to sensitive data exposure or system compromise.

Common Vulnerabilities

- Storing passwords in plaintext.
- Clear text data transmission.
- Weak encryption algorithms.
- Lack of secure key management.
- Unvalidated server certificates.

Hunting Tips

- Check for clear text data transmission.
- Review cryptographic algorithms used.
- Validate server certificates.
- Investigate insecure key management practices.

Example Attack Scenarios

1. Automatic decryption of credit card numbers in a database allows data exfiltration.
2. Downgrading HTTPS to HTTP and stealing session cookies.
3. Password databases with unsalted hashes lead to cracked passwords.

A03:2021 – Injection

What is it?

Injection vulnerabilities occur when untrusted data is executed as code within an application, potentially leading to malicious actions, such as remote code execution and sensitive data exfiltration.

Common Vulnerabilities

- SQL Injection (SQLi).
- Cross-Site Scripting (XSS).
- Command Injection.

Hunting Tips

- Test for untrusted data usage in queries.
- Look for dynamic queries without proper parameterization.
- Inject malicious payloads to trigger errors.
- Test input fields for proper sanitization.

Example Attack Scenarios

1. Vulnerable SQL calls using untrusted data leads to attackers querying the database at will.
2. Remote code execution via command injection.
3. Client-side javascript code execution in the victim's browser via cross-site scripting.

A04:2021 – Insecure Design

What is it?

Insecure Design focuses on design flaws that can introduce security risks, emphasizing the importance of threat modeling and secure design principles. Impact often leads to security controls being bypassed, leading to account takeovers or undesired actions.

Common Vulnerabilities

- Business logic flaws.
- Lack of security controls in architecture.
- Poorly designed authentication flows.

Hunting Tips

- Analyze the application's design and architecture.
- Understand the logic behind the application, and attempt to skip steps; such as triggering an order confirmation before the checkout process.

Example Attack Scenarios

1. Insecure credential recovery workflow relying on "questions and answers".
2. Exploiting group booking discounts to book an excessive number of seats without making a deposit.
3. Lack of protection against bots and domain logic rules to protect against inauthentic purchases.

A05:2021 – Security Misconfiguration

What is it?

Security Misconfiguration arises from improperly configured settings. This often includes use of unnecessary features and default accounts that lead to increased attack surface, data exposure, and system compromise.

Common Vulnerabilities

- Default credentials left unchanged.
- Improperly configured permissions.
- Enabled unnecessary features.
- Exposing sensitive files or directories.

Hunting Tips

- Search for exposed configuration files.
- Check for misconfigured permissions.
- Test default credentials.
- Identify unnecessary features and default accounts.

Example Attack Scenarios

1. Compromise servers via default admin credentials.
2. Directory listing exposes sensitive data or code.
3. Detailed error messages reveal vulnerabilities or sensitive information.

A06:2021 – Vulnerable and Outdated Components

What is it?

This category highlights the risks associated with using outdated or vulnerable third-party components in applications, causing increased exposure to fixed vulnerabilities and system compromise.

Common Vulnerabilities

- Using libraries with known vulnerabilities.
- Using unmaintained third-party components or outdated frameworks.
- Unpatched software dependencies.

Hunting Tips

- Identify outdated components.
- Check for known vulnerabilities in libraries.
- Leverage Shodan, Nuclei, or other tools to discover specific vulnerable services.

Example Attack Scenarios

1. Attackers targeting unpatched or misconfigured systems.
2. Using public POCs against outdated components to compromise systems.

A07:2021 – Identification and Authentication Failures

What is it?

Identification and Authentication Failures focus on issues related to identification processes, often leading to unauthorized access and account takeover.

Common Vulnerabilities

- Lack of anti-automation controls or multi-factor authentication.
- Weak or default passwords.
- Session hijacking, or session fixation.
- Weak or ineffective credential recovery.

Hunting Tips

- Brute force usernames/passwords, perform credential stuffing.
- Check for hard-coded passwords.
- Check for session-related vulnerabilities to attempt account takeovers.

Example Attack Scenarios

1. Credential stuffing attack using a list of known passwords.
2. Use of default credentials to access admin accounts.
3. Incorrect session timeouts allow attackers to access old sessions.

A08:2021 – Software and Data Integrity Failures

What is it?

Software and Data Integrity Failures emphasize making assumptions related to software updates, critical data, and CI/CD pipelines without verifying integrity. Often these failures lead to “supply-chain” attacks, and remote code execution.

Common Vulnerabilities

- Inclusion of functionality or plugins from untrusted sources.
- Lack of integrity verification for updates.
- Untrusted search path.

Hunting Tips

- Test the application's functionality to see if it includes untrusted code or web functionality from external sources
- Verify if downloaded code updates contain integrity checks.
- Examine the application's configuration files and settings to check for insecure search paths, improper cookie handling, or inclusion of untrusted functionality.

Example Attack Scenarios

1. Attacks on systems with unsigned firmware updates.
2. The SolarWinds malicious update incident.
3. Insecure deserialization leading to remote code execution.

A09:2021 – Security Logging and Monitoring Failures

What is it?

Security Logging and Monitoring Failures involve inadequate logging and monitoring practices, impacting incident detection and response.

Common Vulnerabilities

- Insufficient event logging.
- Failure to detect and alert on anomalies.
- Failure to log auditable events.
- Lack of forensic capabilities.

Hunting Tips

- Review logging practices and alerting thresholds.
- Look for missing log events.
- Look for blind spots in monitoring.

Example Attack Scenarios

1. Data breach goes undetected due to lack of monitoring.
2. Delayed notification of a data breach.
3. Inadequate alerting for active attacks.

A10:2021 – Server-Side Request Forgery (SSRF)

What is it?

Server-Side Request Forgery (SSRF) occurs when an attacker can manipulate server requests, potentially leading to data exposure or attacks on internal resources behind firewalls and ACLs.

Common Vulnerabilities

- Fetching remote resources without validation.
- Exploiting trust in internal networks.
- Manipulating server requests.

Hunting Tips

- Check for unvalidated user-supplied URLs.
- Manipulate requests to access internal resources.

Example Attack Scenarios

1. Port scanning internal servers.
2. Accessing sensitive data, such as local files or internal services.
3. Compromising internal services, such as metadata storage of cloud services.

Stay Safe, Keep Learning!

Explore each category, test your skills, and help secure the web! Remember, as a security researcher, continuous learning and practice are key. Stay curious, keep honing your skills, and explore the ever-evolving world of web application security. Happy hunting!