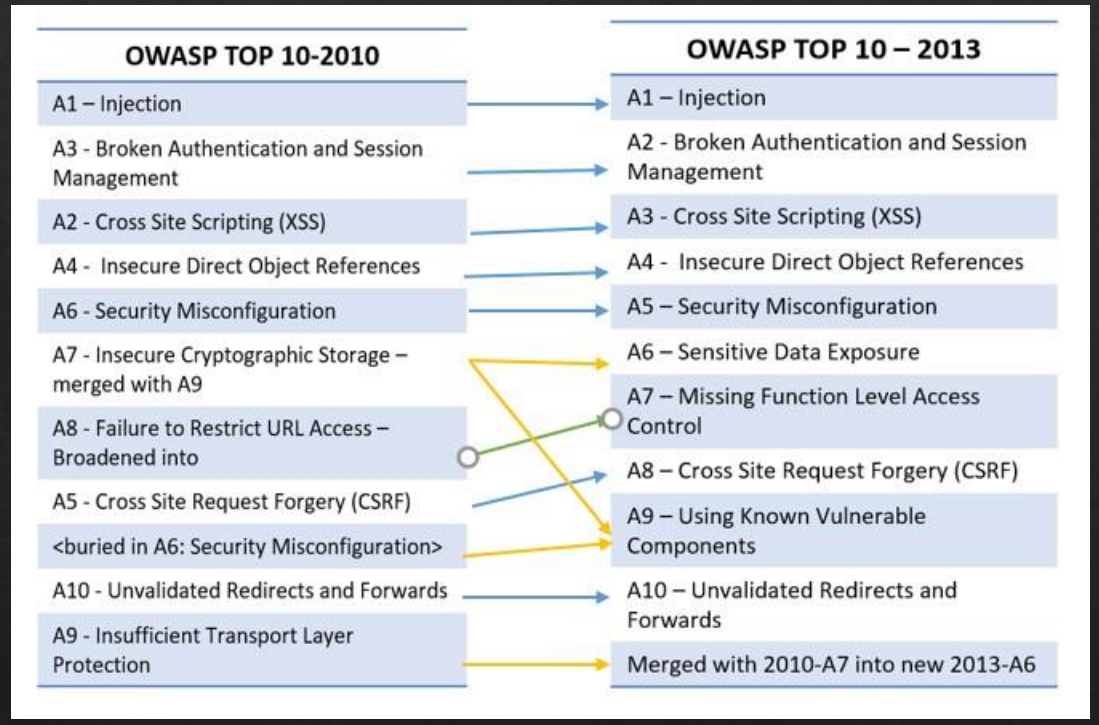
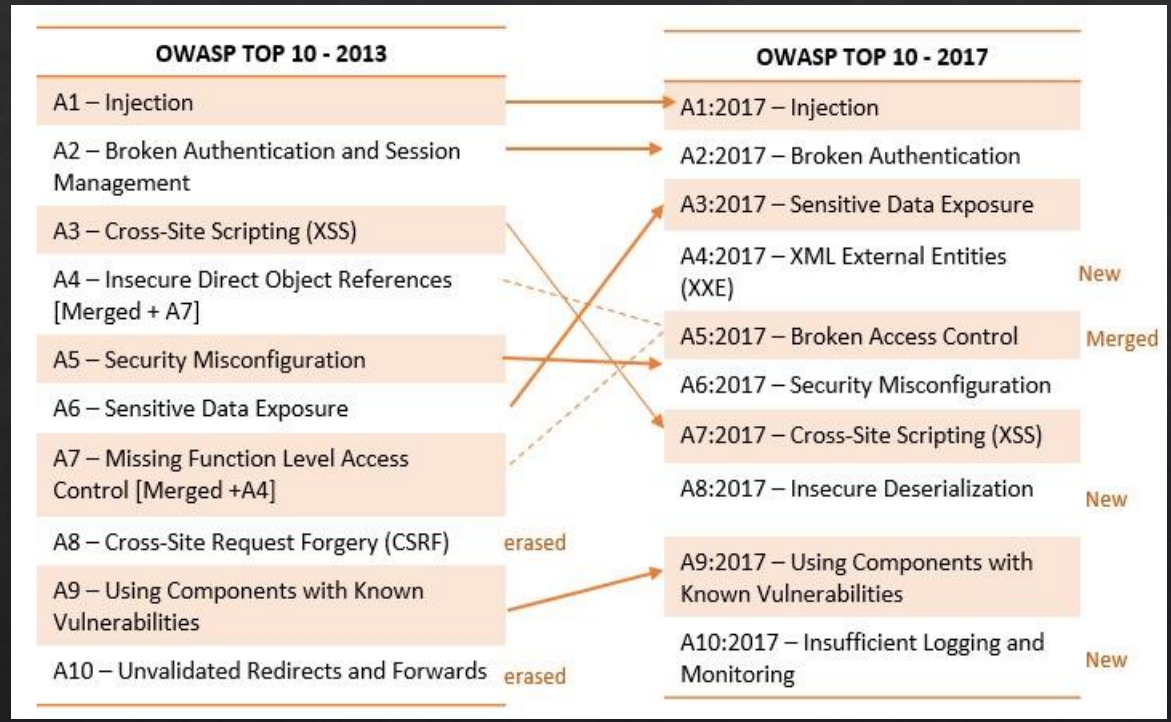

OWASP
TOP 10
Security Risks &
Vulnerabilities
ΟΙΚΟΝΟΜΙΚΟ
ΠΑΝΕΠΙΣΤΗΜΙΟ
ΑΘΗΝΩΝ
ATHENS UNIVERSITY
OF ECONOMICS
AND BUSINESS
MSc: Information Systems Development and Security
Μάθημα: Διοίκηση & Τεχνολογίες Κυβερνοασφάλειας
Καθηγητής: Δρ. Γρίτζαλης
Φοιτήτριες: Αθανασίου Λυδία f3312102, Δρούγκα Σοφία f3312105
2021 CWE Most Important Hardware Weaknesses
MOST IMPORTANT
2021
CWE
HARDWARE WEAKNESSES
New OWASP Top 10 2021 » Hacking Lethani

OWASP Top Ten Changes Between 2010 and 2017

OWASP maintains the Top 10 list and has done so since 2003. Every 2-3 years the list is updated in accordance with advancements and changes in the AppSec market. The OWASP top ten is one of the most widely used lists in the IT security world.



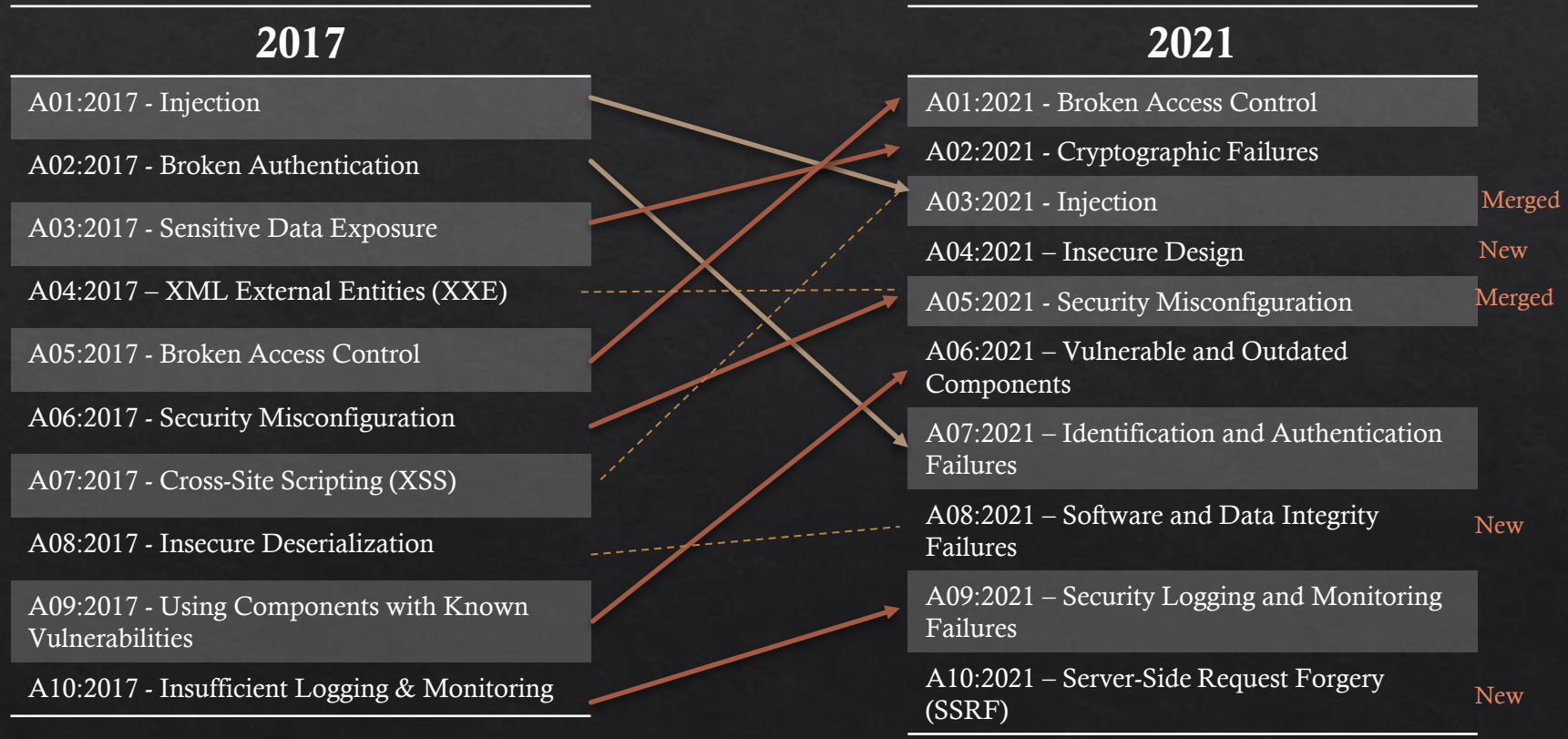
As we can see in tables above, the vast majority of top ten 2010 rested the same but they changed level of importance. The highlights are the A7:2010-Insecure Cryptographic Storage that merged with A9:2010 – Insufficient Transport Layer Protection, thus creating a new category named Sensitive Data Exposure in risk list for 2013. Moreover, the A7:2010 became part of A9:2013 – Using Known Vulnerable Components. Last but not least, A8:2010 was broadened into A7:2013-Missing Function Level Access Control and moved up to OWASP 2013 list.



As we can see in tables, the changes between top ten vulnerabilities for 2013 and 2017 are important. Some of them remain the same such as Injection and Broken authentication but we observe that most of them change level of importance e.g. Sensitive Data Exposure moved from number 6 to number 3 in 2017 list. Last but not least, A4 and A7 of 2013 list are merged to A5:2017 – Broken Access Control. We should also need to highlight the addition of two new categories at list 2017 the A8 and the A10.

OWASP Top Ten Changes Between 2017 and 2021

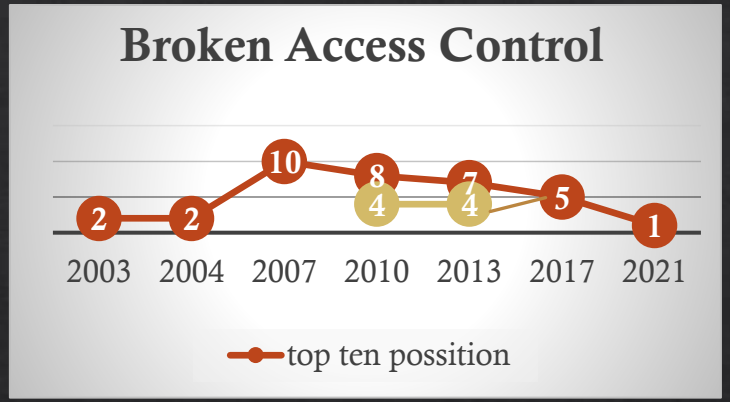
As we can see in tables, there are important changes between top ten vulnerabilities for 2017 and 2021. We can observe renames, merges and additions. In the rest of this presentation we will analyse them one by one.



A1:2021 – Broken Access Control ←A5:2017- Broken Access Control

Moves up from the fifth position in the first category with the most serious web application security risk.

- Another two risks from the 2013 version were merged in the 2017 OWASP Top 10: Insecure direct object references(4) and missing function level access control(7) were merged into broken access control 2017.
- Access control enforces policy such that users cannot act outside of their intended permissions. Failures typically lead to unauthorized information disclosure, modification, or destruction of all data or performing a business function outside the user's limits.

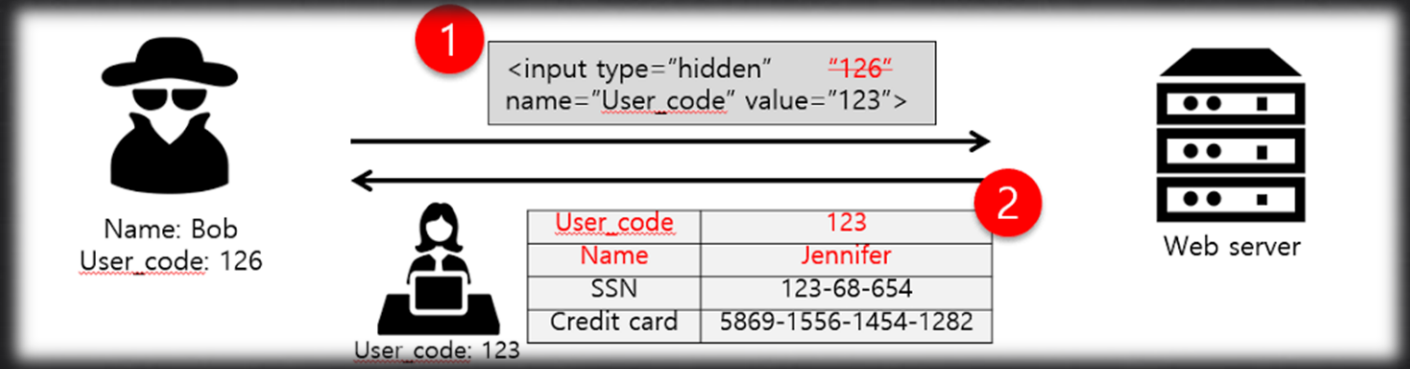


- **Prevention:**
  Use penetration testing in order to detect unintended access-control, Deny by default, Log access control failures etc.
- **Example:** The application uses unverified data in a SQL call that is accessing account information:

```
pstmt.setString(1, request.getParameter("acct"));
ResultSet results = pstmt.executeQuery( );
```

An attacker simply modifies the browser's 'acct' parameter to send whatever account number they want. If not correctly verified, the attacker can access any user's account.

```
https://example.com/app/accountInfo?acct=notmyacct
```



Bug Bounty for Beginners (part 2) broken access control | by Sayin0 | Medium

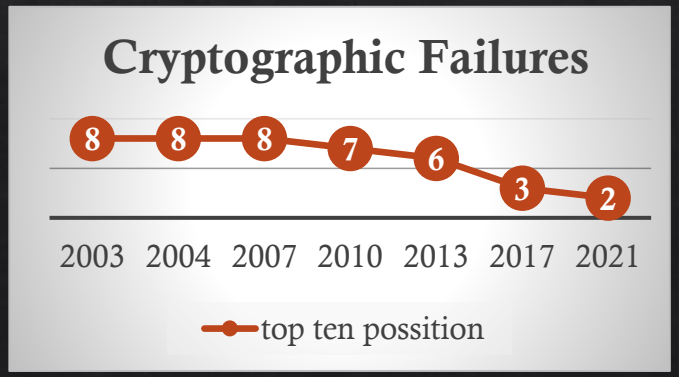
A2:2021 –Cryptographic Failures ← A3:2017-Sensitive Data Exposure

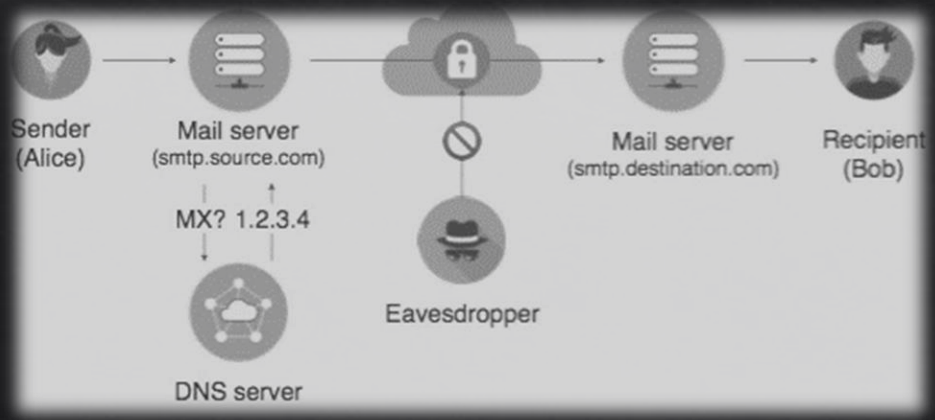
Moves up from third position in the second. The renewed name focuses on failures related to cryptography as it has been implicitly before.

- Sensitive data exposure is when important stored or transmitted data (such as social security numbers) is compromised.
- **Example:** Financial institutions that fail to adequately protect their sensitive data can be easy targets for credit card fraud and identity theft.
- Some sensitive data that require protection are: Credentials, Credit card numbers, Social Security Numbers, Medical information, Personally identifiable information (PII) etc.
- Responsible sensitive data collection and handling have become more noticeable especially with the General Data Protection Regulation (GDPR). This is a new data privacy law that came into effect May 2018. It mandates how companies collect, modify, process, store, and delete personal data.
- **Types of attack**: SQL Injection, Phishing Attack, Man in the middle attack.

- **Detection:** Data leak detection using scanning tools.

- **Prevention:**
  - Enforce Strict Data Encryption.
  - Have a well-defined password policy.
  - Give access to data if required.
  - Encrypt backups as well.





Cryptographic Failures in Practice | by Frank Wang | MIT Security Seminar | Medium

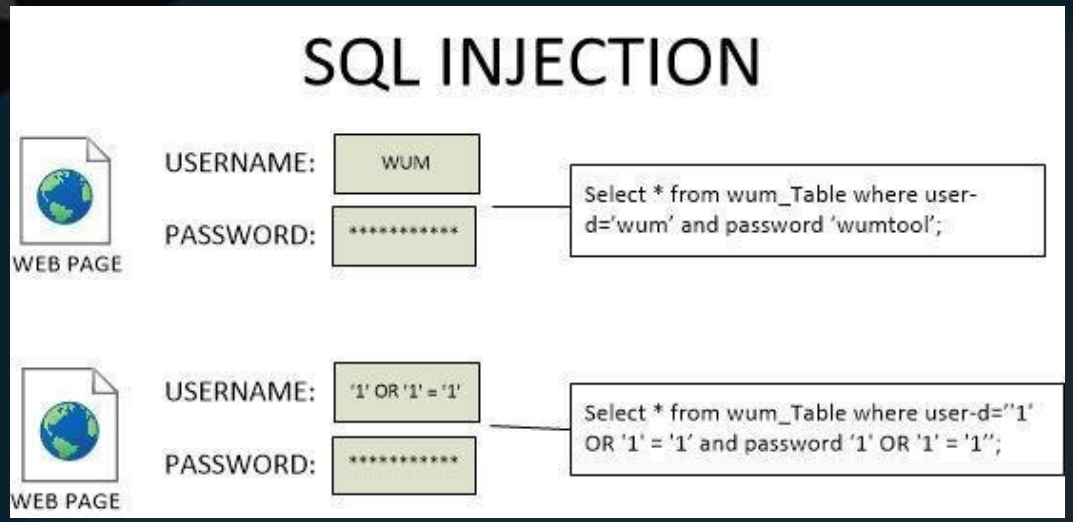
A3:2021 – Injection ← A1:2017-Injection & A7:2017-XSS

It slides down from the first (2017) to the third position and Cross-site Scripting is now part of this category in this edition.

- Injection occurs when malicious data is sent by an attacker into a web application. The attacker's intent is to make the application do something it was not designed to do.
- SQL injection is one of the most common injection flaws found in applications (33.3%) and it can be caused by use of untrusted data by an application when constructing a vulnerable SQL call.
- Most common example is the SQL query consuming untrusted data:

```
String query = "SELECT * FROM accounts WHERE custID = '" +
               request.getParameter("id") + "'";
```

This query can be exploited by calling up the web page executing it with the following URL: http://example.com/app/accountView?id=' or '1'='1 causing the return of all the rows stored on the database table.



A SQL injection attack. | Download Scientific Diagram (researchgate.net)

- The SQL injection causes a leak of sensitive data and compromises an entire WordPress installation.
- Anything that accepts parameters as input can potentially be vulnerable to a code injection attack.

A04:2021 – Insecure Design

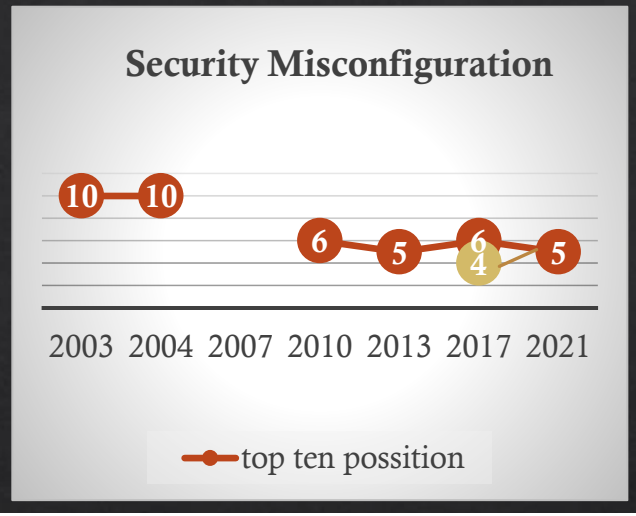
Is a **new** category for 2021, with a focus on risks related to design flaws. It calls for more use of threat modelling, secure design patterns and principles, and reference architectures.

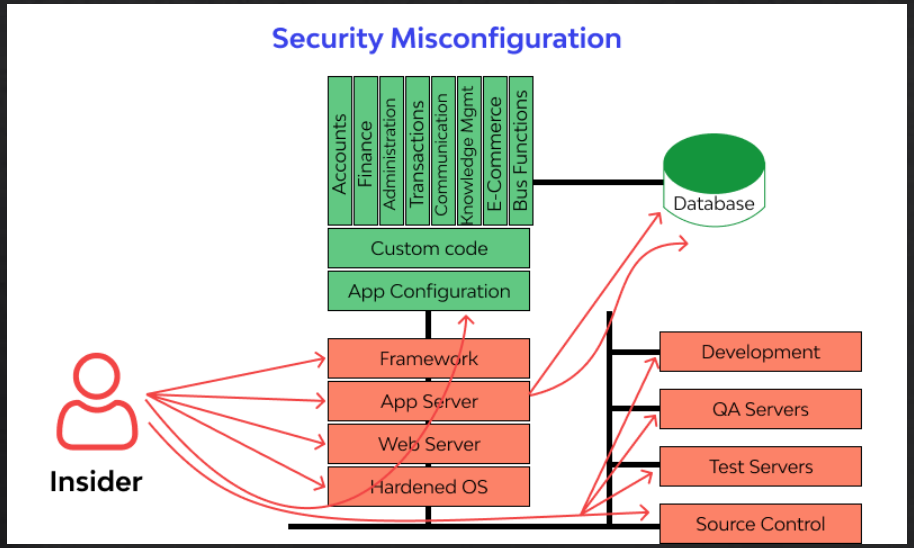
- Insecure design is a broad category representing different weaknesses, expressed as "missing or ineffective control design." It cannot be fixed by a perfect implementation as by definition, needed security controls were never created to defend against specific attacks.
- One of the factors that contribute to insecure design is the lack of business risk profiling inherent in the software or system being developed, and thus the failure to determine what level of security design is required.
- **Prevention**
  - Establish and use a secure development lifecycle with AppSec professionals to help evaluate and design security and privacy-related controls.
  - Establish and use a library of secure design patterns or paved road ready to use components.
  - Use threat modelling for critical authentication, access control, business logic, and key flow.
  - Write unit and integration tests to validate that all critical flows are resistant to the threat model.
  - Limit resource consumption by user or service.

A5:2021 –Security Misconfiguration←A4:2017-XXE & A6:2017-Security Misconfiguration

It moves up from #6 position in the previous edition to #5 (2017) position in OWASP 2021 list.
The former category for A4:2017-XML External Entities (XXE) is now part of this risk category.

- Security Misconfiguration is defined as failing to implement all the security controls for a server or web application, or implementing the security controls, but doing so with errors.
- **Types of Attack**
  - Brute force/credential stuffing.
  - Code & command injection.
  - Buffer overflow.
  - Forceful browsing.
- **Detection**
  - Continuous identification of application security posture with web security firewall through automated security scans and manual Pen-Testing (SonarQube, AppTrana).
- **Prevention**
  - Change credentials (usernames, passwords) for default accounts regularly.
  - Upgrade or remove out-of-date software versions.
  - Close all unnecessary services and ports.
  - Use logs for system monitoring.



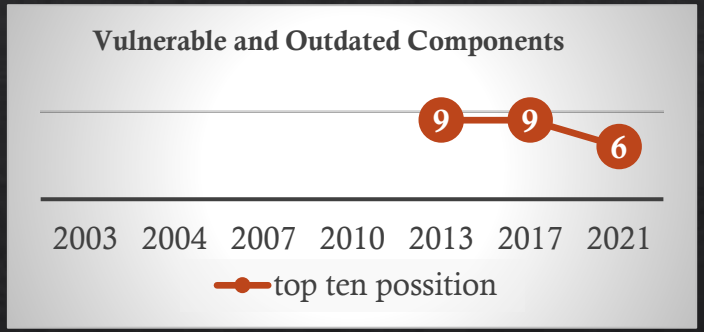


https://www.wallarm.com/what/a6-security-misconfiguration-2017-owasp

A06:2021 – Vulnerable and Outdated Components ← A09:2017 - Using Components with Known Vulnerabilities

It moves up to the sixth position in OWASP 2021 list from the ninth in OWASP 2017 list. The former category A09:2017- Using Components with Known Vulnerabilities is now renamed A06: Vulnerable and Outdated Components.

- The process of using software components with known gaps in security which can be identified and exploited by hackers or intruders.
- Almost all applications contain vulnerabilities, due to weaknesses of their components or dependency libraries. Most vulnerabilities are unintentional, but some are deliberately left by developers.
- **Types of Attack**
  - Code injection on web or database server.
  - Cross site scripting.
- **Detection**
  - Use specialized vulnerability detection tools such as HDIV, Crashtest Security Suite.
- **Prevention**
  - Identify all components, versions, dependencies and libraries used by the application.
  - Establish security policies and best practices for component use, including acceptable licenses.
  - Monitor known security vulnerabilities in databases, project newsletters, mailing lists etc.
  - Add security wrappers to components, including the disablement of unnecessary functionalities.



A7:2021 – Identification and Authentication Failures ← A2:2017- Broken Authentication

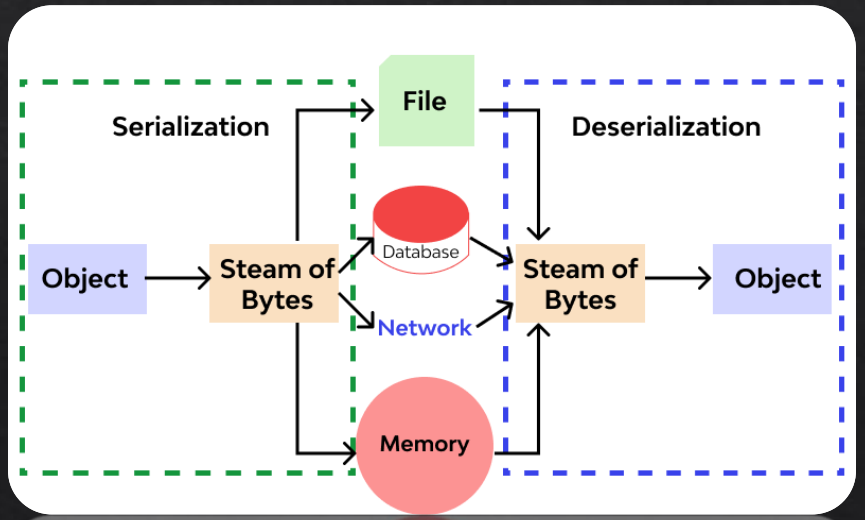
It was previously Broken Authentication and is sliding down from the second (2017) to seventh position in OWASP 2021 list.

- Certain applications are often improperly implemented. Specifically, functions related to authentication and session management, when implemented incorrectly, allow attackers to compromise passwords, keywords, and sessions.
- **A broken authentication** vulnerability can allow an attacker to use manual and/or automatic methods to try to gain control over any account they want in a system or to gain complete control over the system.
- **Types of Attack:** Phishing attack, Brute force attack, Man in the middle attack.
- **Detection:** Black-list/White-list based tools.
- **Example:** A web application allows the use of weak or well-known passwords (e.g. "password1") → Solution: Multi-factor authentication can help reduce the risk of compromised accounts.
- **Prevention**
  - Using a trusted third party.
  - Awareness of users.
  - Block the phishing e-mails by various spam filters.
  - Using strong encryption – Hashing.
  - Using multi factor authentication.
  - Do not ship or deploy with any default credentials.

A8:2021- Software and Data Integrity Failures ← A8:2017-Insecure Deserialization

Is a **new category** in OWASP 2021 list. A08: 2017 – Insecure Deserialization is now part of A8: 2021 Software and Data Integrity Failures.

- Software and data integrity failures relate to code and infrastructure that does not protect against integrity violations. Integrity failures affect application software which relies upon plugins, libraries, or modules from untrusted sources, repositories, and content delivery networks (CDNs)
- This category stands by itself for 2 reasons:
  1. The A06:2021 category "Vulnerable and Outdated Components" groups vulnerabilities derived from individual software components while A08 groups vulnerabilities injected into the payloads of otherwise benign artifacts before delivery.
  2. It remains the best category for deserializing untrusted data.
- **Types of attack**
  - Update without signing. For example, home routers, set-top boxes and device firmware. do not verify updates via signed firmware.
  - SolarWinds malicious update.
  - Insecure Deserialization.
- **Prevention**
  - Ensure libraries and dependencies consume trusted repositories.
  - Ensure CI/CD pipeline has adequate security configuration.
  - Ensure unsigned or unencrypted serialized data is not sent to untrusted clients without integrity check or digital signature.



A8: Insecure Deserialization ‖ - Top 10 OWASP 2017 (wallarm.com)

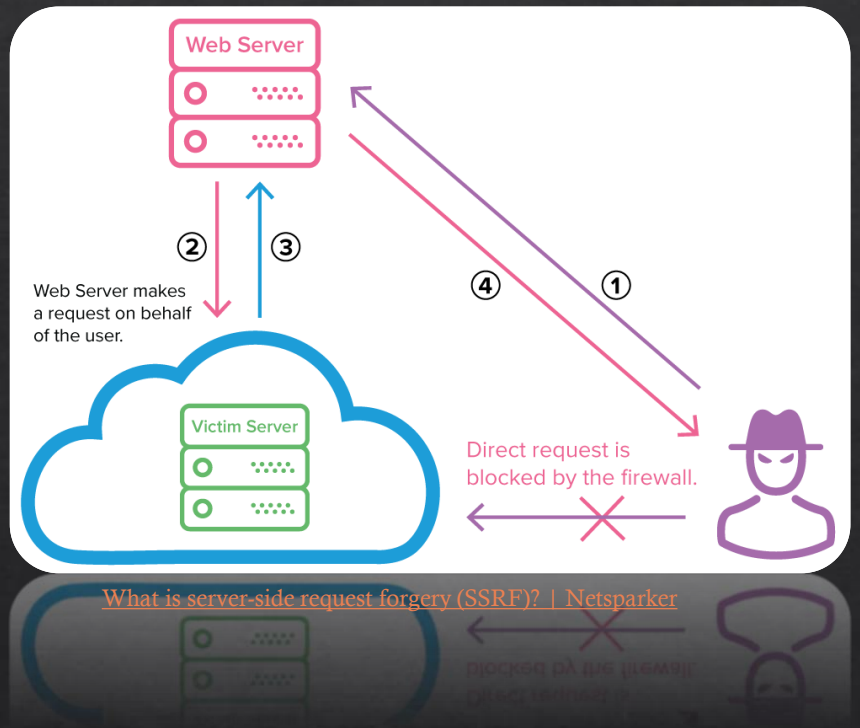
A09:2021 – Security Logging and Monitoring Failures ← A10:2017 - Insufficient Logging & Monitoring

It moves up from #10 to # 9 position in OWASP 2021. The former category A10:2017 Insufficient Logging & Monitoring is now renamed A9:2021 Security Logging and Monitoring Failures.

- Insufficient logging and monitoring is missing security critical information logs or lack of proper log format, context, storage, security and timely response to detect an incident or breach.
- **Types of attack:** Botnet Attacks, DNS Attacks, Insider Threats.
- Fundamental reasons causing insufficient logging & monitoring systems:
  - Unlogged events and transactions.
  - Missing log backups & breach escalation plans.
  - Poor authentication management.
  - Ineffective training on logging and monitoring.
- Prevention
  - Ensure login, access control failures, and server-side input validation failures are logged with sufficient user context for suspicious accounts identification.
  - Ensure high-value transactions have an audit trail with integrity controls to prevent tampering or deletion, such as append-only database tables.
  - Establish effective monitoring and alerting to detect and respond to suspicious activities in a timely fashion.
  - Use application protection framework such as AppSensor.

Your Guide To Insufficient Logging & Monitoring & Prevention (crashtest-security.com)

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



What is server-side request forgery (SSRF)? | Netsparker

Is a new category in OWASP 2021 risk list.

- In a Server-Side Request Forgery (SSRF) attack, the attacker can abuse functionality on the server to read or update internal resources. This kind of attacks allow an attacker to make requests to any domains through a vulnerable server by making the server connect back to itself, to an internal service or resource, or to its own cloud provider.
- SSRF vulnerabilities occur when a server does not validate user-submitted URLs when they fetch remote resources.
- **Types of attack**
  - Attack Against the Server—Injecting SSRF Payloads.
  - XSPA—Port Scanning on the Server.
  - Obtaining Access to Cloud Provider Metadata.
- **Detection**
  - Dynamic application security testing (DAST) for vulnerability scanning such as Netsparker.
- Prevention
  - implement firewall policies.
  - Implement HTTP CONNECT proxy (Smokescreen).

Bibliography:

- OWASP (2021). *OWASP Top Ten*. [online] Owasp.org Available at: https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&cad=rja&uact=8&ved=2ahUKEwiOtrSZ1Z_0AhV4gP0HHbgcDooQFnoECAkQAQ&url=https%3A%2F%2Fowasp.org%2Fwww-project-top-ten%2F&usg=AOvVaw3q5_mdZZBOIUp7TKNCuUgl [Accessed 17 Nov. 2021].
- www.synopsys.com. (n.d.). *What Is the OWASP Top 10 and How Does It Work? | Synopsys*. [online] Available at: https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&cad=rja&uact=8&ved=2ahUKEwin1bXm1Z_0AhU7hP0HHaRpAeoQFnoECAYQAw&url=https%3A%2F%2Fwww.synopsys.com%2Fglossary%2Fwhat-is-owasp-top-10.html&usg=AOvVaw2Hp-GFRbz56F5eMuksYHk_ [Accessed 17 Nov. 2021].
- owasp.org. (n.d.). *Table of Contents | OWASP*. [online] Available at: https://owasp.org/www-project-top-ten/2017/ . [Accessed 10 Nov. 2021].
- Google.com. (2017). [online] Available at: https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&cad=rja&uact=8&ved=2ahUKEwjV4vmf15_0AhXAiv0HHfeTCuoQFnoECAUQAQ&url=https%3A%2F%2Fowasp.org%2Fwww-project-top-ten%2F2017%2F&usg=AOvVaw1J-pla3xU8D4H9zN6Mt_qx [Accessed 17 Nov. 2021].
- Sucuri. (n.d.). *OWASP Top 10 Security Vulnerabilities 2021*. [online] Available at: https://sucuri.net/guides/owasp-top-10-security-vulnerabilities-2021/ [Accessed 10 Oct. 2021].
- Google.com. (2021). [online] Available at: https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&cad=rja&uact=8&ved=2ahUKEwjPh4mH2J_0AhW4_rsIHbtzBUoQFnoECAQQAQ&url=https%3A%2F%2Fowasp.org%2FTop10%2FA01_2021-Broken_Access_Control%2F&usg=AOvVaw30ih7WgWsjvkKuzhEFQKhn [Accessed 12 Oct. 2021].
- Google.com. (2021). [online] Available at: https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&cad=rja&uact=8&ved=2ahUKEwjroqO12J_0AhVi8LsIHec7B8wQFnoECAQQAQ&url=https%3A%2F%2Fowasp.org%2FTop10%2FA02_2021-Cryptographic_Failures%2F&usg=AOvVaw0yfmj22h8FGd52MZksBIsV [Accessed 27 Oct. 2021].
- Mateo Tudela, F., Bermejo Higuera, J.-R., Bermejo Higuera, J., Sicilia Montalvo, J.-A. and Argyros, M.I. (2020). On Combining Static, Dynamic and Interactive Analysis Security Testing Tools to Improve OWASP Top Ten Security Vulnerability Detection in Web Applications. *Applied Sciences*, 10(24), p.9119.
- Hacking Lethani. (2021). *New OWASP Top 10 2021» Hacking Lethani*. [online] Available at: https://hackinglethani.com/owasp-top-10-2021-en/ [Accessed 18 Nov. 2021]