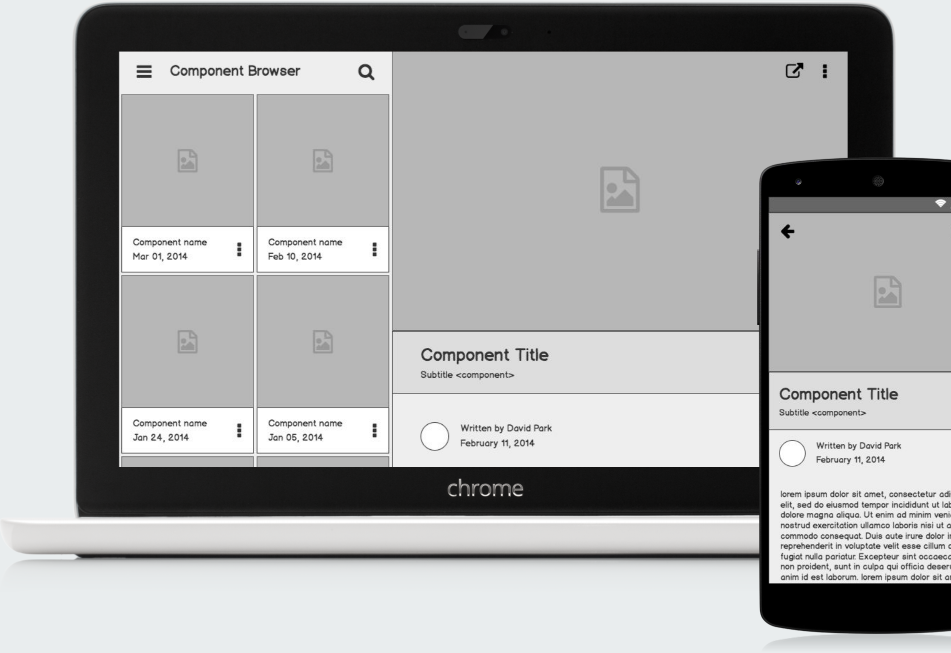
OWASP Security Risks

Injection Attacks

Insufficient Logging & Monitoring



Injection Attack

Injection Attack

What is this?

Definition: attacker injects malicious input that becomes executable commands

Common types:

- SQL Injection
- OS Command Injection
- LDAP Injection
- XPath Injection

Core idea: **turning data into unexpected instructions**

Injection Attack

Impact of it?

Data Exposure: attackers can read sensitive information

Data Manipulation: modification or deletion of records

Authentication Bypass: login checks can be tricked

Remote System Compromise: OS commands may be executed

Injection Attack

How to Prevent it?

Use **Parameterized Queries / Prepared Statements**

Strong **Input Validation** (whitelisting formats)

Encoding and Sanitization of user input

Apply **Principle of Least Privilege** to database accounts

Deploy **Web Application Firewall (WAF)** for detection

Insufficient Logging & Monitoring

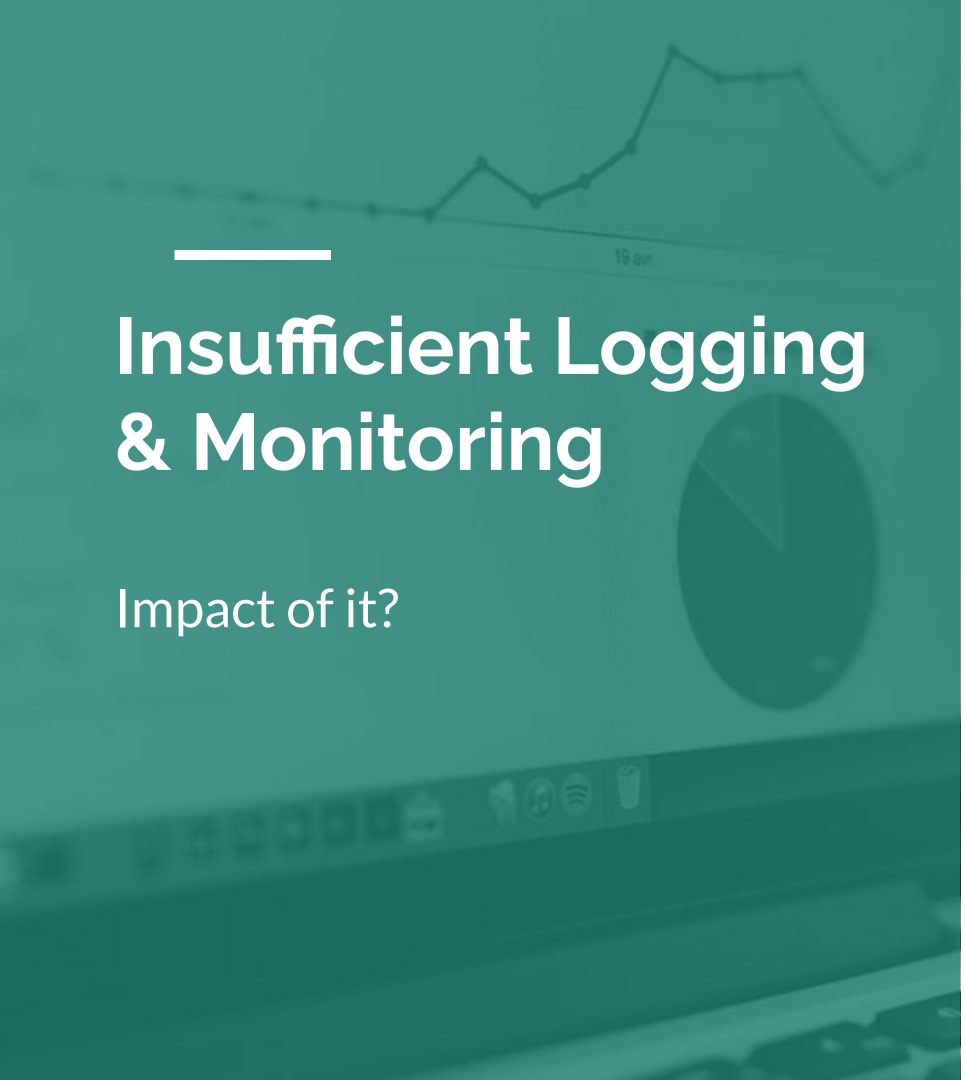
Insufficient Logging & Monitoring

What is it?

Missing or incomplete audit logs of critical actions

Lack of real-time monitoring and alerting

Consequence: attacks executed without being detected or investigated



Insufficient Logging & Monitoring

Impact of it?

Undetected Attacks: brute-force, injection, privilege abuse

No Incident Traceability: difficult to analyze or respond

Longer Attacker Dwell Time: attackers remain in the system unnoticed

Insufficient Logging & Monitoring

How to Prevent it?

Log all critical events:

- Login attempts
- Privilege changes
- Administrative actions
- Failed authentication

Use **Centralized Logging / SIEM** (Splunk, ELK, Azure Sentinel)

Configure **Real-Time Alerts** for anomalies

Protect logs from tampering; apply access controls

Define log retention policies (90–180 days or based on compliance needs)

Perform **regular log audits**

Conclusion

Injection attacks remain one of the **most dangerous** OWASP vulnerabilities.

Insufficient logging & monitoring allows attackers to operate undetected.

Effective security requires both:

- **Preventive Controls** (secure coding, input validation)
- **Detective Controls** (logging, monitoring, alerting)

Security Principle: **“Prevent early, detect quickly.”**

Questions?