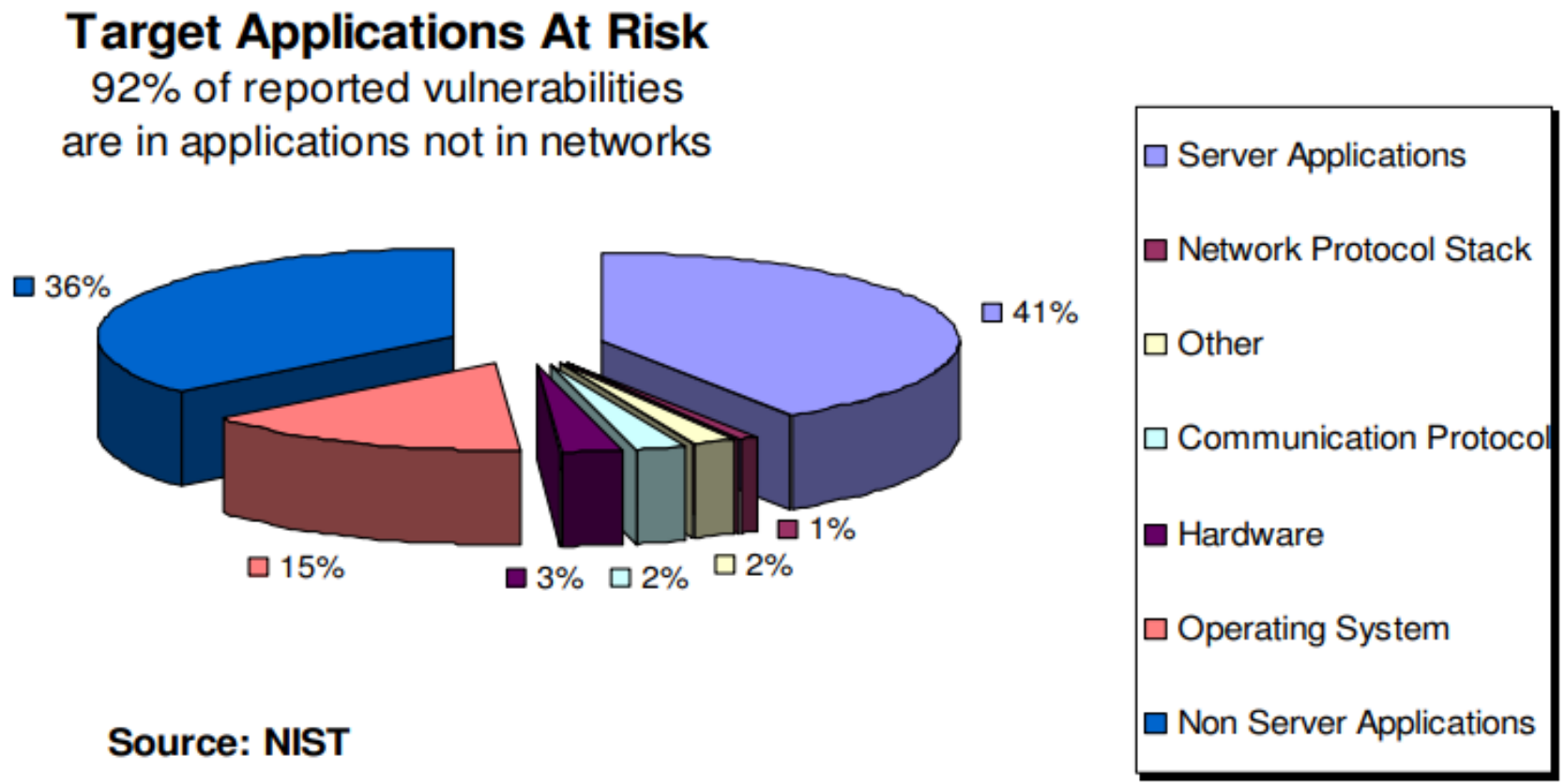
Dat klopt voor jou!

Secure coding principles – OWASP

Workshop Content

- Introduction round
- OWASP Top 10
- Hands-on with secure coding

What is at risk?



Open Web Application Security Project (OWASP)

- OWASP is an international non-profit organization
- Aims to provide developers with the resources in the form of technology and education to secure the web
- Offers a broad consensus on the most common security flaws/exploits
- Designed to raise awareness and the stress the need for security in web-based applications

OWASP Methodologies and Tools (1)

- **OWASP Top 10 Web Security Risks**
  - A standard awareness document for developers and security teams
  - Represents a broad consensus on the top security risks to web applications
- **OWASP Proactive Control**
  - Describes the most important controls every architect and developer must implement
  - Helps secure web apps by addressing common vulnerabilities early in development
- **OWASP Dependency-Track**
  - A platform for component analysis in the software supply chain
  - Identifies and reduces risks by leveraging Software Bill of Materials (SBOM)

https://owasp.org/

OWASP Methodologies and Tools (2)

- **OWASP Web Security Testing Guide (WSTG)**
  - A premier resource for testing web application security
  - A comprehensive guide for both developers and security professionals
- **OWASP SAMM**
  - Provides a measurable way to assess and improve the secure development lifecycle
  - Helps organizations build secure software by following a structured approach

https://owasp.org/

TOP 10

vijfhart
IT-OPLEIDINGEN

OWASP Top 10 Web Security Risks

- ✓ A01 Broken Access Control
- ✓ A02 Cryptographic Failures
- ✓ A03 Injection
- ✓ A04 Insecure Design
- ✓ A05 Security Misconfiguration
- ✓ A06 Vulnerable and Outdated Components
- ✓ A07 Identification and Authentication Failures
- ✓ A08 Software and Data Integrity Failures
- ✓ A09 Security Logging and Monitoring Failures
- ✓ A10 Server-Side Request Forgery (SSRF)

A01: Broken Access Control

- **Incorrect implementation of authentication & session management**
  - Enabling attackers to compromise passwords, keys, session tokes, or assume user identities

- **CWE-200:** Exposure of Sensitive Information to an Unauthorized Actor
- **CWE-201:** Exposure of Sensitive Information Through Sent Data
- **CWE-352:** Cross-Site Request Forgery

API and Authentication Risks

- Unprotected APIs that are considered “internal”
- Weak authentication that does not follow industry best practices
- Weak API keys that are not rotated
- Passwords that are weak, plain text, encrypted, poorly hashed, shared, or default passwords
- Authentication susceptible to brute force attacks and credential stuffing
- Credentials and keys included in URLs
- Lack of access token validation (including JWT validation)
- Unsigned or weakly signed non-expiring JWTs

https://apisecurity.io/encyclopedia/content/owasp/api2-broken-authentication

Qantas confirms personal data of over a million customers leaked in breach

By **Reuters**

July 10, 2025 12:06 AM GMT+2 · Updated 16 hours ago

Companies

Qantas Airways Ltd Follow

Medibank Private Ltd Follow

July 9 (Reuters) - Australia's Qantas Airways (QAN.AX) said on Wednesday more than a million customers had their phone number, birth date or home address accessed in one of the country's biggest cyber breaches in years.

BY ANDY GREENBERG SECURITY JUL 9, 2025 3:28 PM

McDonald's AI Hiring Bot Exposed Millions of Applicants' Data to Hackers Who Tried the Password '123456'

Basic security flaws left the personal info of tens of millions of McDonald's job-seekers vulnerable on the "McHire" site built by AI software firm Paradox.ai.

A02: Cryptographic Failure

- **Insufficient protection of sensitive data** (credit cards, tax IDs, authentication credentials, etc.)
- **Risks:**
  - Data theft or modification (leading to credit card fraud, identity theft, etc)
- **Main causes:**
  - Lack of protection for data-in-transit and data-at-rest
  - electronic social engineering attacks
- **CWE-259:** Use of Hard-coded Password
- **CWE-327:** Broken or Risky Crypto Algorithm
- **CWE-331:** Insufficient Entropy

BUSINESS INSIDER

TECH

A massive trove of 16 billion stolen passwords was discovered — here's what to do

By Jordan Hart

A03: Injection

- **Injection attacks (SQL, OS, LDAP injection)** occur when untrusted data is sent to an interpreter as part of a command or query
  - Attackers can trick the interpreter into executing unintended commands or accessing unauthorized data
- **Cross-Site Scripting (XSS)** occurs when an application takes untrusted data and sends it to a web browser without proper validation or escaping
  - Allows attackers to execute scripts in the victim's browser: hijacking sessions, defacing websites, or redirecting the user to malicious sites
- **CWE-79:** Cross-site Scripting
- **CWE-89:** SQL Injection
- **CWE-73:** External Control of File Name or Path

Example: Bulgarian NRA Hack (2019)

- **Attack type:** SQL injection
- **Target:** Bulgaria's National Revenue Agency (NRA)
- **Data exposed:** Personal data of 5 million citizens (PINs, tax returns)
- **Cause:** Poor input validation
- **Impact:** Attackers exfiltrated sensitive data, risking identity theft and public trust

A04: Insecure Design

- **An insecure design cannot be fixed by a perfect implementation**
  - A secure design can still have implementation flaws that lead to exploitable vulnerabilities
  - Secure design is crucial from the start, as vulnerabilities can stem from the design phase
- **Common contributing factor:**
  - Lack of business risk profiling in the software or system design
- **CWE-209:** Generation of Error Message Containing Sensitive Information
- **CWE-256:** Unprotected Storage of Credentials
- **CWE-501:** Trust Boundary Violation,
- **CWE-522:** Insufficiently Protected Credentials.

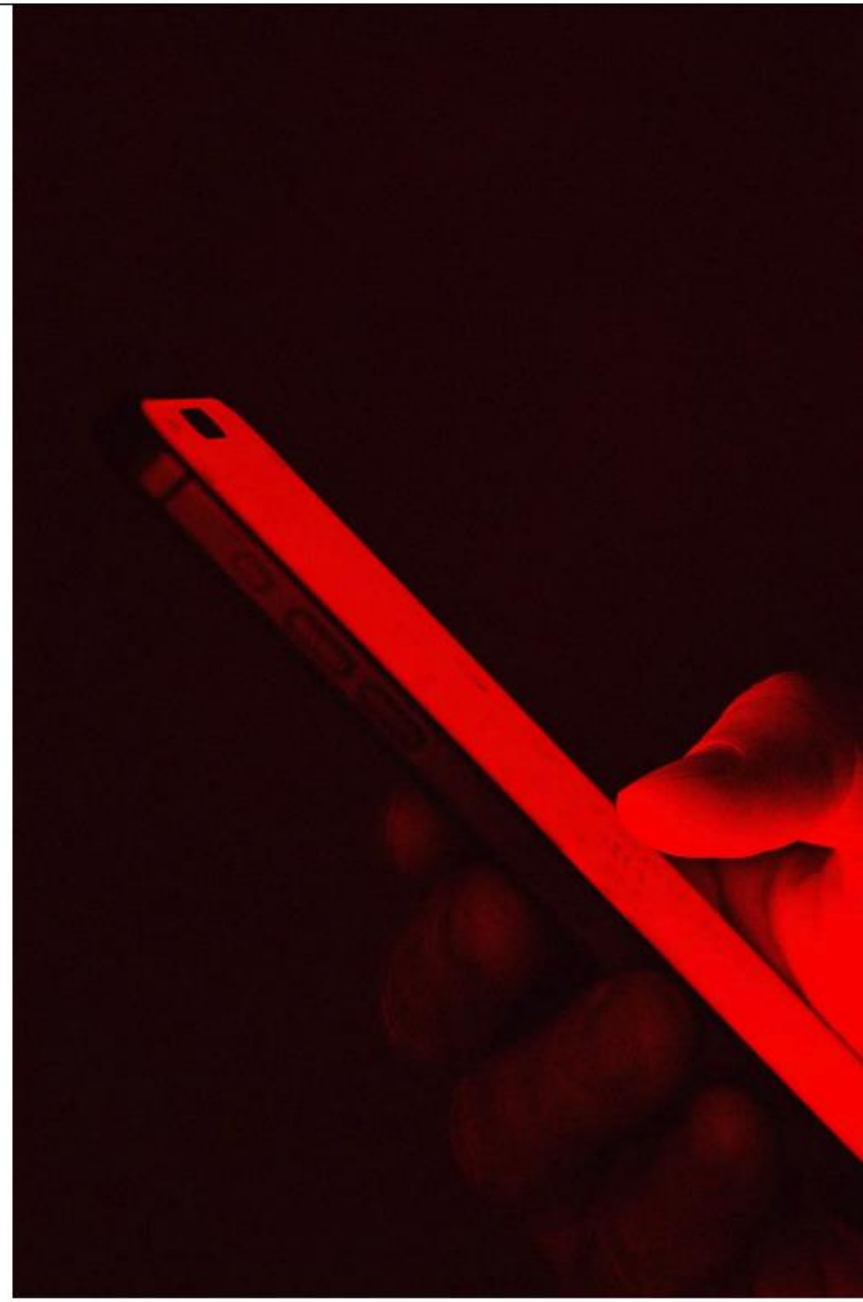
A05: Security Misconfiguration

- **Security misconfiguration** occurs when security settings are not properly configured, potentially exposing sensitive data or enabling unauthorized access
- **XML External Entities (XXE) vulnerability**
  - Older or poorly configured XML processors evaluate external entity references, leading to serious security risks
  - XXE attacks can disclose internal files, access internal file shares, perform port scanning, trigger remote code execution, and cause denial of service
- **CWE-16:** Configuration
- **CWE-611:** Improper Restriction of XML External Entity Reference

MICAH LEE SECURITY MAY 18, 2025 7:00 AM

How the Signal Knockoff App TeleMessage Got Hacked in 20 Minutes

The company behind the Signal clone used by at least one Trump administration official was breached earlier this month. The hacker says they got in thanks to a basic misconfiguration.



A06: Vulnerable and outdated Components

- **Use of outdated or unknown versions**
  - Includes both client-side and server-side components, as well as nested dependencies
- **Risks of using vulnerable, unsupported, or outdated software components:**
  - Web/application servers
  - Database management systems (DBMS)
  - APIs
  - Libraries
  - Runtime environments
  - And more
- **CWE-1104:** Use of Unmaintained Third-Party Components

Example: Log4j Vulnerability (2021)

- **Vulnerability:** Remote code execution (Log4Shell) in Log4j
- **Cause:** Unpatched vulnerability in an outdated version of Log4j
- **Impact:** Millions of systems exposed, including cloud services and enterprise apps
- **Takeaway:** Outdated components can lead to serious security risks

A07: Identity and Authentication Failures

- **Restrictions on what authenticated users can do are often not properly enforced.**
- **Attackers can exploit these flaws to:**
  - Access other users' accounts
  - View sensitive files or modify data
  - Change accesss rights
  - And more
- **CWE-287:** Improper Authentication
- **CWE-384:** Session Fixation
- **Improper Validation of Certificate with Host Mismatch**

A08: Software and Data Integrity Failures

- **Software and data integrity failures**
  - Code and infrastructure that does not protect against integrity violations
  - Example: Relying on plugins, libraries, or modules from untrusted sources, repositories, or CDNs
- **Risks with insecure CI/CD pipelines:**
  - Potential for unauthorized access, malicious code injection, or system compromise
- **CWE-829:** Inclusion of Functionality from Untrusted Control Sphere
- **CWE-494:** Download of Code Without Integrity Check,
- **CWE-502:** Deserialization of Untrusted Data

A09: Security Logging & Monitoring Failures

- **Issue:** Insufficient logging and monitoring, with missing or ineffective integration with incident response, make it easier for attackers to maintain persistence and cause damage
- **Detection delay**
  - Breaches often go undetected for over 200 days, typically noticed by external parties
- **CWE-778:** Insufficient Logging
- **CWE-117:** Improper Output Neutralization for Logs
- **CWE-223:** Omission of Security-relevant Information
- **CWE-532:** Insertion of Sensitive Information into Log File

A10: Server-Side Request Forgery (SSRF)

- **SSRF flaws** occur whenever a web application is fetching a remote resource without validating the user-supplied URL
- **Risk:**
  - Attackers can exploit SSRF to send crafted requests to unexpected destinations, bypassing firewalls, VPNs, or ACLs
- **Why it's increasing:**
  - Modern web apps often fetch URLs for features, making SSRF more common
  - The rise of cloud services and complex architectures has increased the severity of SSRF attacks

Exercises

Detecting Risks

1. Use of Static Application Security Testing (SAST, such as SonarQube)
2. Dynamic Application Security Testing (DAST, such as OWASP ZAP)
3. Code Reviews with Security Focus
4. Software Composition Analysis (SCA, such as Snyk)
5. Security Configuration Scanning

Quiz

Vijfhart

Rokus Janssen, adviseur en accountmanager bij Vijfhart voor KPN

Heb je vragen of opmerkingen, neem dan vooral contact op met mij voor passend advies. Je kan mij bereiken via:

E: r.janssen@5hart.nl

T: 088 542 78 88

Graag tot snel!