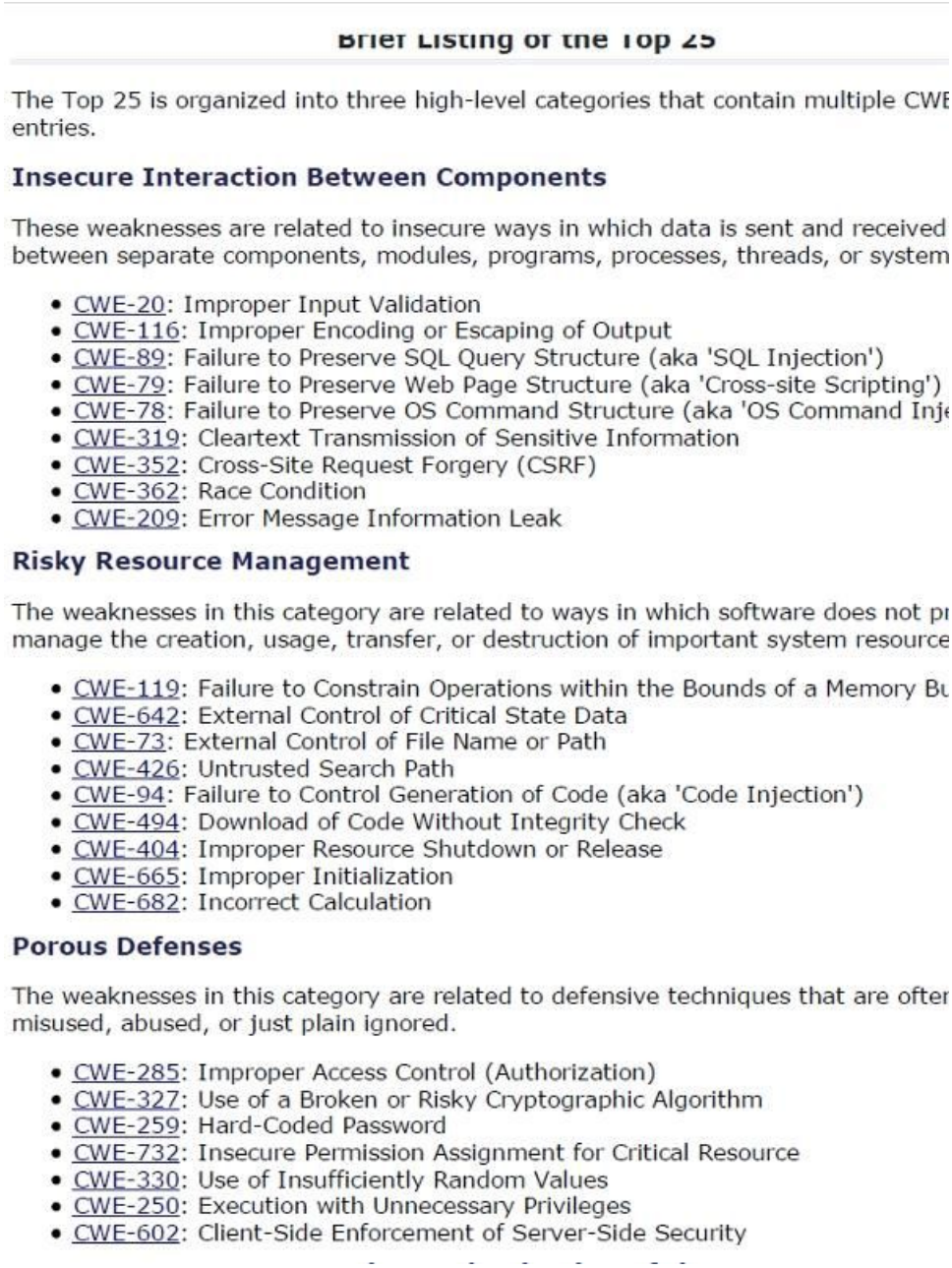
Link: https://www.computerwoche.de/a/top-25-der-gefaehrlichsten-programmierfehler-veroeffentlicht,1883837

Liste des Grauens

Top 25 der gefährlichsten Programmierfehler veröffentlicht

Datum: 13.01.2009
Autor(en):Uli Ries

Eigentlich alle IT-Sicherheitsplagen, von Online-Krimininalität über Datenklau bis hin zum Patchen von Software, lassen sich auf die verheerenden Folgen der 25 häufigsten Programmierfehler zurück führen. Das zumindest geht aus einer von Softwarefirmen und US-Regierungsbehörden veröffentlichten Liste hervor.



Brief Listing of the Top 25

The Top 25 is organized into three high-level categories that contain multiple CWE entries.

Insecure Interaction Between Components

These weaknesses are related to insecure ways in which data is sent and received between separate components, modules, programs, processes, threads, or system

- CWE-20: Improper Input Validation
- CWE-116: Improper Encoding or Escaping of Output
- CWE-89: Failure to Preserve SQL Query Structure (aka 'SQL Injection')
- CWE-79: Failure to Preserve Web Page Structure (aka 'Cross-site Scripting')
- CWE-78: Failure to Preserve OS Command Structure (aka 'OS Command Inje
- CWE-319: Cleartext Transmission of Sensitive Information
- CWE-352: Cross-Site Request Forgery (CSRF)
- CWE-362: Race Condition
- CWE-209: Error Message Information Leak

Risky Resource Management

The weaknesses in this category are related to ways in which software does not pr manage the creation, usage, transfer, or destruction of important system resource

- CWE-119: Failure to Constrain Operations within the Bounds of a Memory Bu
- CWE-642: External Control of Critical State Data
- CWE-73: External Control of File Name or Path
- CWE-426: Untrusted Search Path
- CWE-94: Failure to Control Generation of Code (aka 'Code Injection')
- CWE-494: Download of Code Without Integrity Check
- CWE-404: Improper Resource Shutdown or Release
- CWE-665: Improper Initialization
- CWE-682: Incorrect Calculation

Porous Defenses

The weaknesses in this category are related to defensive techniques that are often misused, abused, or just plain ignored.

- CWE-285: Improper Access Control (Authorization)
- CWE-327: Use of a Broken or Risky Cryptographic Algorithm
- CWE-259: Hard-Coded Password
- CWE-732: Insecure Permission Assignment for Critical Resource
- CWE-330: Use of Insufficiently Random Values
- CWE-250: Execution with Unnecessary Privileges
- CWE-602: Client-Side Enforcement of Server-Side Security

Top 25: MITRE/SANS listen die 25 gefährlichsten Fehler, die von Programmieren begangen wurden.
Foto: MITRE/SANS

Die **2009-CWE/SANS-Top-25-Liste**[1] der gefährlichsten Programmierfehler führt die schwerwiegendsten Fehler auf, die für ernst zu nehmende Schwächen verantwortlich sein können. Zu den Programmierfehlern gehören schlampige Absicherungen von Benutzereingaben (input validation), mangelndes Einhalten der **SQL**[2]-Anfrage-Struktur (**SQL injection**[3]) und fehlerhafter Aufbau von Internetseiten (cross-site scripting). Die aufgelisteten Fehler tauchen laut den Listenverfassern häufig auf, sind einfach zu entdecken und leicht auszunutzen.

Gefährlich sind die Schlampereien, weil Angreifer sie oft ausnutzen können, um die Software unter ihre Kontrolle zu bringen, Daten zu klauen oder die Programme komplett unbrauchbar zu machen. Die negativen Folgen der erfassten Bugs sind den Verfassern zufolge überaus weitreichend. So sind alleine zwei der Programmierfehler für Sicherheitslücken in mehr als 1,5 Millionen Webseiten verantwortlich. Die Lücken wurden von Cyber-Kriminellen dazu ausgenutzt, um **Trojaner**[4] auf die PCs der Websitebesucher zu schleusen.

Erstellt wurde die Liste des Grauens von **MITRE**[5] und dem **SANS Institute**[6] über einen Zeitraum von drei Jahren. Beide Organisationen sind Teil des Projekts Common Weakness Enumeration (**CWE**[7]), das von der US-Heimatschutzbehörde organisiert wird. Nachdem im vergangenen Jahr über 700 Programmierfehler ins CWE-Listing aufgenommen wurden, baten MITRE und SANS Experten von **Software- und IT-Sicherheitsfirmen**[8], die Liste auf 25 Bugs einzudampfen.

Dadurch sollten die Softwarehersteller nicht nur auf die übelsten Programmierfehler aufmerksam gemacht werden. Das Ziel war es darüber hinaus, gemeinsame Tools und Schulungsmaßnahmen zu entwickeln, um die Probleme anzugehen. Software-Managern und **CIOs**[9] kann die Top 25 als Messlatte dienen, um Fortschritte bei der Absicherung der eigenen Software festzuhalten.

Zu den knapp 30 Firmen und Organisationen, die an der Bewertung der Bugs mitgearbeitet haben, gehören die Softwarehersteller **Apple**[10], **Microsoft**[11], **Oracle**[12] und **Red Hat**[13]. Außerdem die IT-Sicherheitshersteller **EMC**[14], **McAfee**[15] und **Symantec**[16], die National Security Agency (NSA) und das amerikanische Energieministerium. Ebenfals beteiligt waren das Computer Emergency Response Team (CERT) and das the Open Web Application Security Project (**OWASP**[17]).

Links im Artikel:

[1] **http://cwe.mitre.org/top25/**
[2] **https://www.computerwoche.de/schwerpunkt/s/SQL.html**
[3] **http://de.wikipedia.org/wiki/SQL_Injection**
[4] **https://www.computerwoche.de/schwerpunkt/t/Trojaner.html**
[5] **http://www.mitre.org/**
[6] **http://www.sans.org/**
[7] **http://cwe.mitre.org/**
[8] **http://cwe.mitre.org/compatible/organizations.html**
[9] **https://www.computerwoche.de/cio-des-jahres/2008/**
[10] **https://www.computerwoche.de/schwerpunkt/a/Apple.html**
[11] **https://www.computerwoche.de/schwerpunkt/m/Microsoft.html**
[12] **https://www.computerwoche.de/schwerpunkt/o/Oracle.html**
[13] **https://www.computerwoche.de/schwerpunkt/r/Red-Hat.html**
[14] **http://germany.emc.com/**
[15] **http://de.mcafee.com/**
[16] **http://www.symantec.com/de/de/index.jsp**
[17] **http://www.owasp.org/**