

Data collection on vulnerabilities caused by design errors

Harald Terkelsen

2006-10-05

- Explore vulnerability databases with a focus on design flaws

Proposed and supervised by Hanno Langweg

What is special about design flaws?

- They cost more to fix.
- They are not easily detected by testing or static analysis tools.
- Two different implementations of a flawed design will both be vulnerable

- 1 What properties do vulnerabilities caused by security related design flaws have?
- 2 What can a classification of security related design flaws look like?
- 3 What are the most typical design flaws?

- 1 What properties do vulnerabilities caused by security related design flaws have?
 - literature review
 - qualitative content analysis.
- 2 What can a classification of security related design flaws look like?
 - qualitative content analysis.
- 3 What are the most typical design flaws?
 - quantitative observation study

Criteria to decide which vulnerability databases to use:

- Common Vulnerabilities and Exposure list (CVE) compatible
- Freely available
- Downloadable
- Language used must be English.

Vulnerability Databases

Name and URL	Content	Downloadable
Computer Associates Vulnerability Encyclopedia http://www3.ca.com/securityadvisor/vulninfo/browse.aspx	9979	No
Dragonsoft vulnerability database http://vdb.dragonsoft.com/	2242	No
ISS X-Force http://xforce.iss.net/xforce/search.php	21000	No
National Vulnerability Database http://nvd.nist.gov/	15494	Yes
Open source vulnerability database http://www.osvdb.org/	10767	Yes
Public Cooperative vulnerability database https://cirdb.cerias.purdue.edu/coopvdb/public/	10573	No
Secunia Advisories http://secunia.com/advisories/	11300	No
Security Focus http://www.securityfocus.com/vulnerabilities/	15570	No
Security Tracker http://www.securitytracker.com/	10000	No
US-CERT vulnerability notes database http://www.kb.cert.org/vuls/	1591	No

Vulnerability Databases

- National Vulnerability Database chosen
- Includes references to other databases
- Mirrors CVE
- Can be downloaded in XML format
- Wrote two small programs
 - One to parse the XML and create a local SQL database
 - One to extract samples of vulnerabilities from the SQL database to examine
- Marked some vulnerabilities as not design flaw based on keywords in description: *xss, cross-site, sql injection, buffer overflow, heap overflow, format string, sanit, symlink, input validat*
 - Marked 5426 vulnerabilities out of 15491

What is a Design Flaw?

Definition

Design flaw: An error in a software's design that exists independently of a concrete implementation. This behavior typically manifests itself in the algorithms, datastructures, or interfaces used.

Examples of design flaws from the literature

- Weak encryption
- Trusting data and systems
- Bad error handling
- Incorrect or missing access control mechanism
- IEEE 1016-1998 Software Design Descriptions
 - Data: Initial value, acceptable values
 - Processing: Loop termination criteria, path conditions, handling of contingencies

Exploring vulnerability databases

Experience

- Time consuming
- Deciding if a given flaw is a design flaw is often very hard
- Descriptions in the databases are not good enough.
Several sources must be checked
- Our definition of a design flaw is not precise.
- Some flaws can have been introduced under both design or implementation
- Abstraction level
- Classification of a flaw in the database does not always match our classification or the classification in other databases

Exploring vulnerability databases

- 640 vulnerabilities examined
- 122 identified as design flaws
- 80 are maybe design flaws
- 339 other type of flaws
- 41 unknown

Software flaw taxonomies and classifications

<i>Author</i>	<i>Year</i>	<i>Name</i>
Abbot	1976	Security analysis and enhancements of computer operating systems
Bisbey II	1978	Protection analysis: Final report
Landwehr	1994	A Taxonomy of computer program security flaws, with examples
Aslam	1996	Use of a taxonomy of security faults
Lindqvist	1997	How to systematically classify computer security intrusions
Du & Mathur	1998	Categorization of software errors that led to security breaches
Krsul	1998	Software vulnerability analysis
Piessens	2002	A Taxonomy of software vulnerabilities in internet software
Jiwani	2002	Maintaining software with a security perspective
Langweg	2004	A classification of malicious software attacks
Weber	2005	A software flaw taxonomy: aiming tools at security
Tsipenyuk	2005	Seven Pernicious Kingdoms: A taxonomy of software security errors

Software flaw taxonomies and classifications

Taxonomic characteristics^a

^aIvan Krsul PhD thesis

- Objectivity:** The features must be identified from the object known and not from the subject knowing. The attribute being measured should be clearly observable
- Determinism:** There must be a clear procedure that can be followed to extract the feature
- Repeatability:** Several people independently extracting the same feature for the object must agree on the value observed
- Specificity:** The value for the feature must be unique and unambiguous

Software flaw taxonomies and classifications

- Weber et al, 2005, A software flaw taxonomy: aiming tools at security
 - not unambiguous, but the authors argue that if a flaw can be classified under several categories, it is a result of the characteristics of the flaw itself.
- Tsipenyuk, 2005, Seven Pernicious Kingdoms: A taxonomy of software security errors
 - The taxonomy aims to be more practical than theoretical complete

A proposed classification

Not finished yet!

Access control	Authentication	Missing
		Insufficient state/algorithm
		Static credentials
		Plaintext credentials
		Weak obfuscation/encryption of credentials
		Backdoor
		Account lockout
	Authorization	Missing
		Insufficient
		Object creation with insecure default rights
		Hook to execute programs or code
		Action without user confirmation
	Neutralized by other privileges	Missing access check
Insufficient access check		
Encipherment	Data storage	Missing
	Communication	Missing
		Using ECB mode
Miscellaneous	Logging	Incorrect/spoofed
		Exposure of secrets in logs
	Responses	Different responses depending on exists/no exists of data
	Calculation	Mirrored data
		Resource amplification

Another dimension based on IEEE-1016-1998?

- Processing
- Data
- Interface

- Finish the classification
- Classify the design flaws found
- Find more design flaws and classify
- Analyze the result

Questions?