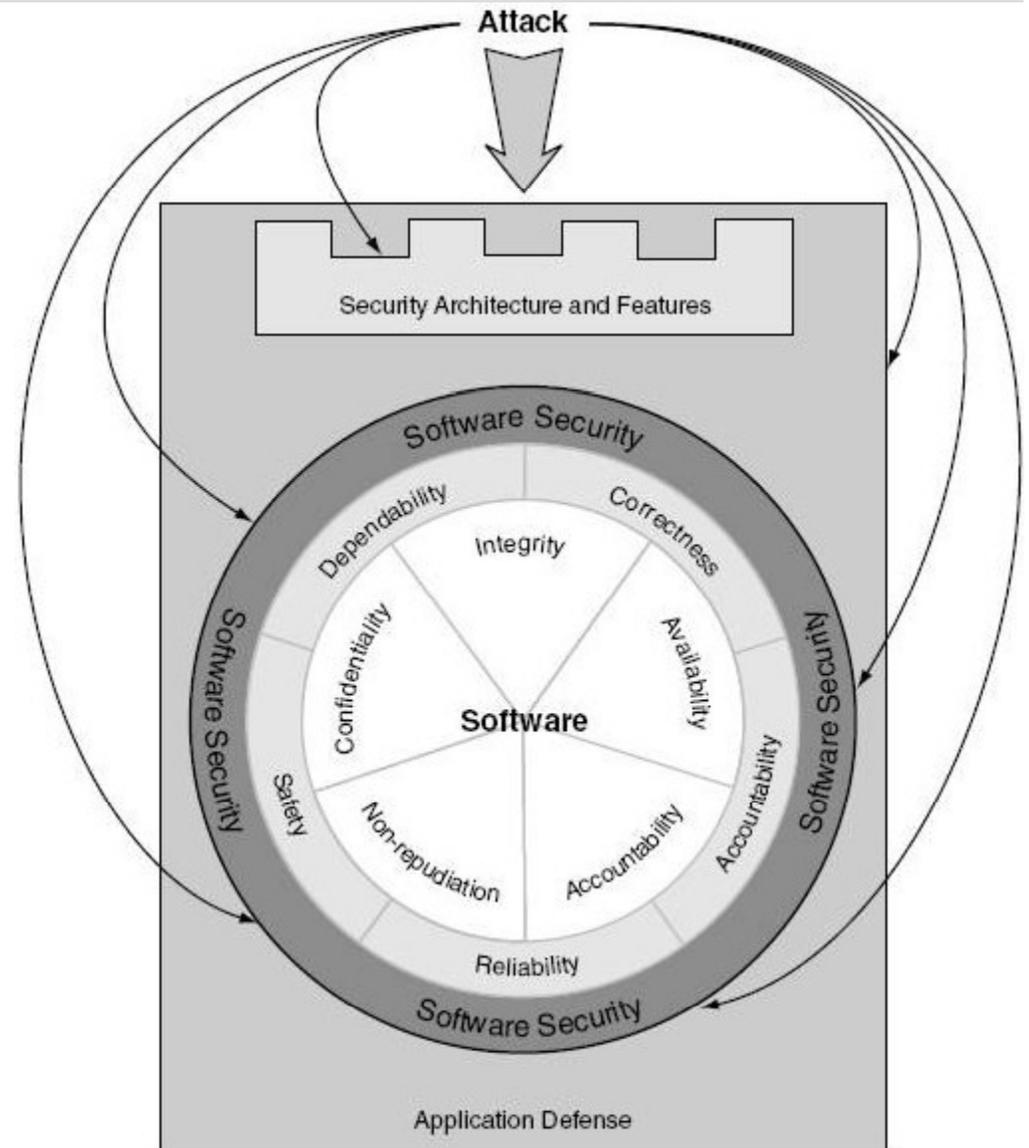


Foundational Elements of Software Assurance Measurement

- Core of measuring software assurance is capturing:
 - How the software may be vulnerable to attack
 - How the software behaves when under attack
- Common Weakness Enumeration (CWE)
- Common Attack Pattern Enumeration and Classification (CAPEC)



Common Weakness Enumeration (CWE)



- Weaknesses are characteristics of software that may lead to vulnerability
- Existence of weaknesses in software can be objectively measured through the use of various techniques and tools
- Software assurance is determined by the absence of weaknesses identified as relevant for a given context and assurance level

- CWE is an effort targeted at standardizing the capture and description of weaknesses and providing a useful collection to be leveraged by the community
- Community effort developed from dozens of sources
- CWE version 1.0 was released last month
- Already being used in education, tools, software risk assessment, policy, etc.
- Currently being integrated into assurance cases, international standards and other domain efforts

- <http://cwe.mitre.org>



Common Weakness Scoring System (CWSS)

- All weaknesses are not of equal importance in every context
- Method is needed to determine which weaknesses are most critical for a given situation and to prioritize them for mitigation
- Such a mechanism exists for Vulnerabilities (CVSS)

- A new effort is beginning to create a similar scoring system for weaknesses
- Such a scoring system will need to be broader than CVSS, using heuristics to encompass factors that make the weakness more or less likely to exist, to be exploitable and to have greater or lesser impacts such as:
 - Technical context of the weakness
 - Technical context of the software
 - Business/mission context of the software
 - Risk thresholds for the owners/users of the software
 - Temporal context of weakness notoriety
 - Relevant mitigating controls

Common Attack Pattern Enumeration and Classification (CAPEC)



- Attack Patterns (APs) formally capture common approaches to attacking software
- APs can be used to create objectively measurable test cases which simulate various attacks on the software
- Attack resistance can be measured by determining if the objective of the attack (captured in the AP) was successful or not

- CAPEC is an effort targeted at standardizing the capture and description of APs and providing a useful collection to be leveraged by the community

- Further info:
 - <http://capec.mitre.org>
 - Attack Pattern article series on Build Security in website
(<https://buildsecurityin.us-cert.gov/daisy/bsi/articles/knowledge/attack.html>)

