Trip report

Software Assurance Forum
Arlington, Virginia
2-6 November 2009

Day 1 – Monday, 2 November

Session 1 – Tutorial on Enhancements for CAPEC

- 2 problems to solve:
  - Inconsistent granularity in CAPEC mitigations
  - Inability to measure the effectiveness of chosen mitigation strategies

- 2 solutions:
  - Use a NIST standard to standardize CAPEC mitigation elements so that granularity is consistent (see slides for details).
  - Develop 2 metrics:
    - Knock-out Effect
      - How many Parent Mitigation strategies are needed to fully mitigate a detailed attack pattern?
    - Parent Mitigation Power
      - Two part metric: x.y
        - x: count of unique APs that the Parent Mitigation helped mitigate
        - y: count of child Mitigations that can be traced back to the Parent Mitigation

- Sean Barnum was there and those enhancements should be integrated into CAPEC.

Session 2 – Tutorial on Software Quality with Parfait

- Software analysis tools are not used at Sun because:
  - Not scalable to MLOC
  - Not precise enough
  - Too expensive

- Parfait:
  - Software analysis tool developed by Sun (Australia)
  - Scalable by layering analyses from cheapest (easy bugs) to most expensive (complex bugs)
  - Precise by using sound analyses

- The framework seems excellent, very well layered and cleverly designed. However, they do not know yet if it will be available, neither as a commercial product nor as an OSS.

Session 3 – Cyber and SwA Education Introduction

- Joe Jarzombek briefly presented many available resources for training and education in SwA. It is all on the slides but here are a few examples:
  - https://www.vte.cert.org/vteweb/
  - https://www.act-online.net/
  - UK Secure Software Development Panel, I found this: http://www.ktn.qinetiq-tim.net/content/files/groups/securesoft/SSDSIG_softwareSecurityFailures.pdf
  - Software Security Engineering, Sean Barnum et al.
  - BSI
  - SAFECode
  - SwA Training and Education Working Group

Session 4 – Tutorial on Secure Programs in C/C++

- Use coding rules, such as CERT C.
- CERT has tools:
  - ROSE (static analysis) + run-time pointer-checking library (dynamic analysis)
- Far from exhaustive.

Session 5 – Tutorial on Expression of Security Requirements

- I was expecting a lot from this tutorial since this is a tough problem that we have been facing for a long time in DRDC Valcartier and are still facing with LSEC.
- It happened to be a talk on the properties of requirements and what they should express instead of how we can express them.

Day 2 – Tuesday, 3 November

That day was very high level. The general message was that the United States are aware of a lot of security problems. They are now trying to find solutions. I felt that the SwA Forum is welcoming presentations on *results*.

One panelist, Lou Kratz, from Lockheed Martin, mentioned that the F-22 is 2,5M SLOC and the F-35 is 8,5M SLOC. It confirms similar figures I have been told.

Day 3 – Wednesday, 4 November

Session 1 – Evolution in SwA Processes

- There is a need *right now* to train software developers, engineers, and third party developers in software security. There is a need to develop a *culture* for security.

- One big challenge is to be able to measure software security, to assess mitigations over time.

- One buzzword is *resiliency*, such in the CERT Resiliency Management Model: http://www.cert.org/resiliency/rmm.html . Resiliency is the new buzzword after robustness, ductility, survivability, etc.

- We need liability in software contracts. Developers need to be held responsible for the software they produce.

Session 2 – Technology Stakeholders

- Cyberspace is now a warfare domain, and probably the number one domain for future conflicts.

- Government IT procurement in the US is getting its software specifically configured in a more secure way. Example: Federal Desktop Code Configuration http://nvd.nist.gov/fdcc/index.cfm .

- SCAP: Security Content Automation Protocol. A set of open standards to monitor and report the security state of network devices.

- KDM: end-to-end traceability, from code to models to evidence to arguments to security requirements.

Session 3 – Software Supply Chain Integrity

- Very interesting challenge: create a contest for developing secure code and then finding vulnerabilities in this code. Idea given by Alan Paller, SANS.
    - Personal notes: I can think of something like that happening in Black Hat and Defcon. It would be very interesting to develop that contest in order to spot new talents and find new techniques both in secure coding and vulnerability analysis. However, it would need to have strong incentives to attract the best coders and hackers. The best ones do not necessarily want to see their techniques revealed and do not want to go to the "wrong side" (public/government, etc).

- David Stender, from IRS, presented an overview of software security in IRS. They do very concrete testing for hiring secure coders, such as "write a secure function that does that" or "find the vulnerabilities in this code".

Session 4 – Measuring Software Supply Chain Risk

- Definition of SwA: the level of confidence that software is free from vulnerabilities, either intentionally designed into the software or accidentally

inserted at anytime during its lifecycle, and that the software functions in the intended manner.

  - Personal notes: This definition sounds bizarre to me. If the software functions in the intended manner (my interpretation of "intended manner" is "by specification"), should not it be free from vulnerabilities anyway? I wonder why the two propositions are necessary. A flawed definition of software assurance would be rather ironic.

- Software security must be considered throughout the whole supply chain. When you buy software, you often buy a product that is the integration of many smaller products that come from many different vendors all around the world.

Day 4 – Thursday, 5 November

Session 1 – Acquisition and Mitigating Software Supply Chain Risk (SSCR)

- Software counterfeiting is a big problem and is hard to stop.

- Security must be ensured throughout the whole supply chain.

- Michael Brown, from the FAA, said that a year ago (in 2008), the FAA leaked 48 000 personal profile records. They found that it was because of a vulnerability in software.
  - They found many thousands buffer overflows in software developed by contractors.
  - They do a lot of IV&V. It is now included in its policy.

Session 2 – Way Forward for Mitigating SSCR

- Collaboration is key, you are not unique. Alone, you will not solve the problem.

- Manage risks, prepare for "the crash" if you cannot pay for "the backups". Then you at least know how to react.

- The Teaching and Education Working Group has a curriculum for training but it is not yet implemented somewhere.

- In December 09, the Working Groups will develop a high level Implementation Road Map (prioritization of actions to take next).

Day 5 – Friday, 6 November

Static Analysis Tool Exposition

- CERT Secure C Coding Standard. They have a library for AIR Integers, As-if Infinitely Ranged Integers, that is, integers that do not overflow or truncate, or if so, result in a runtime constraint violation.

- The analysis of tools performed by NIST in SATE is in progress and needs direction. It seems that they will get it from vendors.

- Veracode does binary analysis. It is a service company so they do not sell their product, code producers agree to get their code verified.

- Coverity says it focuses on code quality as a whole instead of "only" exploitability. This is in the vein of good engineering and resiliency.

- Andrew White, from NSA, mentioned that from his experience, 15-20% of all bugs in real-life applications are findable by static analysis.

- Paul Caseley, from DSTL, mentioned that it costs 100$ per line of code for *safety-critical systems* for analysis *and* correction.