Developing Building Codes for Building Code

IFIP WG 10.4 70th Meeting

Carl Landwehr

GWU / CSPRI

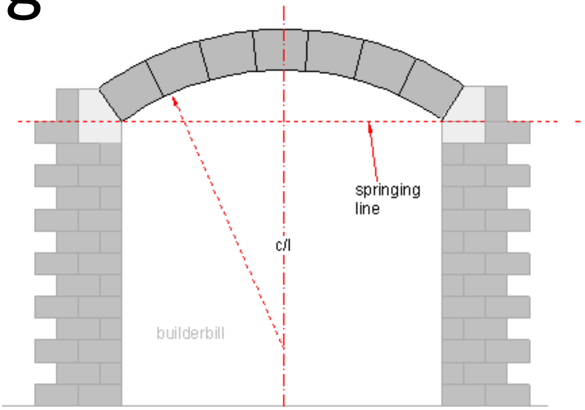
Carl.Landwehr@gmail.com

Firefighters vs. Fire Prevention

- There is now a large market of “Cyber Security Jobs” – but it’s mostly a market for firefighters, not engineers

- Security vulnerabilities are engineering defects, not an incurable disease demanding weekly flu shots

- Borrowing the mechanism of building codes could help us reduce these vulnerabilities significantly



A segmental arch

The talk on one slide

- We know how to build much better security into systems than we have done
- Reasons for this are many, including
  - Lack of consensus on basic requirements for software construction
  - Difficulty for consumers to specify such requirements or to recognize that a product meets them
- Creating a "building code" for software with security requirements might help establish consensus and enable consumer specification
  - Historical development of building codes
  - Where a building code for software might focus
- Such a code has been drafted for medical device software more than a year ago and is available;
- A coming workshop (Nov. 2016) aims to do the same for power system software. Please help! http://cybersecurity.ieee.org/building-code/

Categories of Code Elements

A. Avoid/detect/remove vulnerabilities at the implementation stage (8 elements)

B. Assure proper use of cryptography (2 elements)

C. Assure Software/Firmware Provenance and Integrity (3 elements)

D. Impede attacker analysis/exploitation (4 elements)

E. Enable detection/attribution of attack (1 element)

Categories not populated:

F. Assist in safe degradation of function during an attack

G. Assist in restoration of function after an attack

H. Support maintenance of operational software without loss of integrity

I. Support privacy requirements

From “Sufficient Evidence”*

[About dependable software generally]:

“As is well known to software engineers…, by far the largest class of problems arises from errors made in the eliciting, recording, and analysis of requirements. A second large class arises from poor human factors design…”

[About security vulnerabilities]:

“Security vulnerabilities are to some extent an exception; **the overwhelming majority of security vulnerabilities reported in software products – and exploited to attack [them] – are at the implementation level.** The prevalence of code-related problems, however, is a direct consequence of higher-level decisions to use programming languages, design methods, and libraries that admit these problems. In principle, it is relatively easy to prevent implementation-level attacks but hard to retrofit existing programs.”

***Software for Dependable Systems: Sufficient Evidence?* National Research Council, 20007

References

ACSAC paper motivating the approach:

- Landwehr, C.E. A Building Code for Building Code: Putting What We Know Works to Work. In Proc. 29th Annual Computer Security Applications Conference (ACSAC), New Orleans, Dec 2013. http://www.landwehr.org/2013-12-cl-acsac-essay-bc.pdf

CACM Viewpoint

- We Need a Building Code for Building Code. Viewpoint column, Comm. ACM 58, 2 (Feb. 2015), pp. 24-26 .http://www.landwehr.org/2015-02-cacm-viewpoint-bldg.pdf

IEEE Cybersecurity Initiative Report: Medical Device Software Building Code

- Building Code for Medical Device Software Security. (with Thomas Haigh). IEEE Computer Society, March, 2015:
- http://cybersecurity.ieee.org/images/files/images/pdf/building-code-for-medica-device-software-security.pdf

And website

- https://sites.google.com/site/bcformdss/

Call for Participation in Power Software Security Building Code Workshop at UIUC Nov 16-18 2016

http://cybersecurity.ieee.org/building-code/