Literature reviews

Two assignments, one due this Friday and another due next Friday...

Assignment #2 (due Sep. 14th)

- Have your subject area advisor send me an email, and cc you, saying that they agree to be your subject area advisor and they understand what that entails
  - The email I sent to csfaculty and forwarded to you spells out what it entails
    - Can also ask me questions about it
  - They'll need to read drafts and give you feedback, but English grammar editing you and I'll take care of

Assignment #3 (due Sep. 21st)

- Write roughly a page of literature review for your chosen research area, and give a 5-minute presentation about your research area and literature review
- Please use LaTeX for all writing in this class, for presentations you can use whatever
- You must cite at least one reference that you needed to use the library or a resource other than the web to obtain

Related works section

- Convinces reviewers that you're not doing something that's already been done
- Gives readers some idea of how your work fits into the body of work in the area, and where there are still gaps
- Helps people in the future who are doing literature surveys to understand and locate additional works
- Summarizes the body of work in an area
  - “Systemization of Knowledge”

My examples of (1) learning from the systemization of knowledge in another paper's related works section; and, (2) getting something from the library that I couldn't find on the web; ...both relate to Fenton's Data Mark Machine.

Fenton's Data Mark Machine

- Goal is a “memoryless subsystem”
- Based on adding tags to a Minsky machine
- Journal paper and a dissertation
- Information can flow based on code that *doesn't* execute (called an implicit flow)

```
y = x;
    ...vs...
y = 0; if (x == 1) y=1;
```

Volume 17, Issue 2
1974

Article Contents

Abstract

< Previous Next >

Memoryless subsystems FREE

J. S. Fenton

The Computer Journal, Volume 17, Issue 2, 1 January 1974, Pages 143–147, https://doi.org/10.1093/comjnl/17.2.143

Published: 01 January 1974

Abstract

A memoryless subsystem is incapable of communicating unauthorised information about data input to the outside world. Such systems are important in the study of protection systems, but are difficult to implement. This paper derives a model of such a system and further gives a proof of its correctness

Review PDFs

- Journal paper
- Dissertation

Fenton's Dissertation

- Contacted the library at the Univ. of Cambridge
- Paid them some money (charged it to my advisor's research grant)
- They sent me a printed copy
  - Should have gotten microfiche (these days PDF might also be an option)

Years later, reading the following dissertation I gained an important insight into Fenton's work that I had missed despite reading his paper many times and Fenton's full dissertation once...

Confidentiality Enforcement Using Dynamic Information Flow Analyses

Gurvan Le Guernic

string vulnerabilities as those used by some worms or Trojan horses. Even if those analyses have some practical use, they are not fit to deal with confidentiality seen as secure information flow.

Yet, some binary code level dynamic analyses take into consideration implicit indirect flows (Brown and Knight, 2001; Fenton, 1974b; Saal and Gat, 1978; Vachharajani et al., 2004). Fenton (1974a,b) describes the addition of data marks to the abstract computer model of Minsky (1967). Data marks are fixed, except for the program counter's data mark which is computed dynamically. This means that storage locations can either contain only secret information or only public information. This feature increases the requirements on the size of the machine because, for example, it must contain a set of registers specifically dedicated to contain secret information and another distinct set of registers for public information. It also requires the existence of a distinct program (using different storage locations) for every possible configuration of security levels of inputs. Finally, it also requires the programmer or compiler to be aware of the security level of data manipulated at any time in order to be able to use storage locations with compatible data marks. Fenton (1974b) shows that, assuming public registers are accessible only at the end of the computation, its abstract machine is secure with the fixed data marks, but would not be with variable ones. Saal and Gat (1978) and Brown and Knight (2001) describe, without formally proving something similar to noninterference, an abstract machine in which some storage locations have a fixed data mark and others have a dynamically computed one.

The basic ideas behind the three works (Brown and Knight, 2001; Fenton, 1974b; Saal and Gat, 1978) are the same. When storing a value v to a fixed data mark storage location fl, the machine checks that the

Assorted tips

- Can access lots of papers from on-campus that you can't off-campus
  - SSH tunnels, Tor, VPNs, *etc.*
- Read a paper to see what papers it cites, put its title into Google Scholar to see which papers cite it
- Cluster as you go
- Don't cite things you didn't read
  - Skimming is sometimes okay

Related works section

- Convinces reviewers that you're not doing something that's already been done
- Gives readers some idea of how your work fits into the body of work in the area, and where there are still gaps
- Helps people in the future who are doing literature surveys to understand and locate additional works
- Summarizes the body of work in an area
  - “Systemization of Knowledge”

All four are important!