CSE543 - Introduction to Computer and Network Security

Module: Security Research Methods

Professor Trent Jaeger
Reading papers ...

• What is the purpose of reading papers?
• How do you read papers?
Understanding what you read

- Things you should be getting out of a paper
  - What is the central idea proposed/explored in the paper?
    - Abstract
    - Introduction
    - Conclusions
    - **Motivation**: What is the problem being addressed?
  - How does this work fit into others in the area?
    - **Related work** - often a separate section, sometimes not, every paper should detail the relevant literature. Papers that do not do this or do a superficial job are almost sure to be bad ones.
    - An informed reader should be able to read the related work and understand the basic approaches in the area, and how they differ from the present work.

These are the best areas to find an overview of the contribution
Understanding what you read (cont.)

• What scientific devices are the authors using to communicate their point?

› Methodology - this is how they evaluate their solution.
• Theoretical papers typically validate a model using mathematical arguments (e.g., proofs)
• Experimental papers evaluate results based on test apparatus (e.g., measurements, data mining, synthetic workload simulation, trace-based simulation).
  ‣ Empirical research evaluates by measurement.
• Some papers have no evaluation at all, but argue the merits of the solution in prose (e.g., paper design papers)
Understanding what you read (cont.)

• What do the authors claim?
  ‣ **Results** - statement of new scientific discovery.
  • Typically some abbreviated form of the results will be present in the abstract, introduction, and/or conclusions.
  • **Note**: just because a result was accepted into a conference or journal does necessarily not mean that it is true. Always be circumspect.

• What should you remember about this paper?
  ‣ **Take away** - what general lesson or fact should you take away from the paper.
  ‣ Note that really good papers will have take-aways that are more general than the paper topic.
Summarize Thompson Article

• Contribution
• Motivation
• Related work
• Methodology
• Results
• Take away
A Sample Summary

• **Contribution:** Ken Thompson shows how hard it is to trust the security of software in this paper. He describes an approach whereby he can embed a Trojan horse in a compiler that can insert malicious code on a trigger (e.g., recognizing a login program).

• **Motivation:** People need to recognize the security limitations of programming.

• **Related Work:** This approach is an example of a Trojan horse program. A Trojan horse is a program that serves a legitimate purpose on the surface, but includes malicious code that will be executed with it. Examples include the Sony/BMG rootkit: the program provided music legitimately, but also installed spyware.

• **Methodology:** The approach works by generating a malicious binary that is used to compile compilers. Since the compiler code looks OK and the malice is in the binary compiler compiler, it is difficult to detect.

• **Results:** The system identifies construction of login programs and miscompiles the command to accept a particular password known to the attacker.

• **Take away:** *What is the transcendent truth?????* (see next slide)
**Turtles all the way down ...**

- **Take away:** Thompson states the “obvious” moral that “you cannot trust code that you did not totally create yourself.” We all depend on code, but constructing a basis for trusting it is very hard, even today.
- ... or “trust in security is an infinite regression ...”

“A well-known scientist (some say it was Bertrand Russell) once gave a public lecture on astronomy. He described how the earth orbits around the sun and how the sun, in turn, orbits around the center of a vast collection of stars called our galaxy. At the end of the lecture, a little old lady at the back of the room got up and said: "What you have told us is rubbish. The world is really a flat plate supported on the back of a giant tortoise." The scientist gave a superior smile before replying, "What is the tortoise standing on?" "You're very clever, young man, very clever", said the old lady. "But it's turtles all the way down!"

Reading a paper

• Everyone has a different way of reading a paper.
• Here are some guidelines I use:
  ‣ Always have a copy to mark-up. Your margin notes will serve as invaluable sign-posts when you come back to the paper (e.g., “here is the experimental setup” or “main result described here”)
  ‣ After reading, write a summary of the paper containing answers to the questions in the preceding slides. If you can’t answer (at least at a high level) these questions without referring to the paper, it may be worth scanning again.
• Over the semester, try different strategies for reading papers and see which one is the most effective for you.
Reading a systems security paper

• What is the security model?
  ‣ Who are the participants and adversaries
  ‣ What are the assumptions of trust (trust model)
  ‣ What are the relevant risks/threats

• What are the constraints?
  ‣ What are the practical limitations of the environment
  ‣ To what degree are the participants available

• What is the solution?
  ‣ How are the threats reasonably addressed
  ‣ How do they evaluate the solution

• What is the take away?
  ‣ key idea/design, e.g., generalization (not solely engineering)

• Hint: I will ask these questions when evaluating course project.
Why write a paper?

• There are many reasons to write a paper:
  ‣ Articulate a new idea, thought, or observation ...
  ‣ Document your research ...
  ‣ Talk about new (observed) phenomenon ....
  ‣ Advance your career ...
  ‣ Because you have to ...

• Reality: publication is the coin of the realm in science, failure to do this successfully will lead to failure. You have to be effective at this to be a good (a) graduate student, (b) faculty member, or [sometimes] (c) researcher in professional research laboratory (IBM/AT&T/MS)
Where to publish?

- Venues for publication:
  - Tech report
  - Workshop
  - Conference
  - Journal

- Often your work will work through these from *preliminary* to *archival* versions of the work, sometimes branching or joining.

- *Book*: less frequent, more work.
Publication Tiers

- Not all publication venues are valued the same. Publication “tiers” tell the story
- 1st tier - IEEE S&P, USENIX Sec, CCS, TISSEC, JCS
  - 1.5 NDSS
- 2nd tier - ACSAC, ACNS, ESORICS, CSF, RAID, TOIT
- 3rd tier - SecureComm, ICISS
- 4th tier - HICCS
  - SClgen (WMSCI 2005)
• The editor-in-chief (EIC) receives the papers as they are submitted.
• The papers are assigned to associate editors for handling.
• Anonymous reviewers rate the paper:
  ‣ Accept without changes
  ‣ Minor revision
  ‣ Major revision
  ‣ Reject
The **PC Chair** is the person who marshals the reviewing and decisions of a conference. This is different than the **general chair**.

- **PC members** review, rate and discuss, the paper, then vote on which ones are accepted.
- The **acceptance rate** is the ratio of accepted to submitted papers.
Threat Model

• What is a Threat Model?
Threat Model

• What is a Threat Model?

• Accessibility of Target to Attacker

• Operations that Attacker can perform on the Target
Basis for Threat Model

• Common Weakness Enumeration Document

• Enumerations hundreds of threats
  ‣ But arranged hierarchically
  ‣ Relevant classes include various types of programming and configuration errors
    • Make an argument for how much of CWE hierarchy you have covered and the impact

• We will discuss this further over the rest of the semester
Course Projects

• The course project requires the students execute some limited research in security.
  ‣ Demonstrate applied knowledge
  ‣ Don’t try to learn some new non-security field
  ‣ Be realistic about what is possible in a one semester.
  ‣ However, the work should reflect real thought and effort.

• The grade will be based on: novelty, depth, correctness, clarity of presentation, and effort.

• Structure
  ‣ 3-4 students per group
Deliverables

• The chief product of the project will be a 12 page conference style paper (single column, 11pt font, reasonable margins).

• There will be several milestones:
  ‣ Project Team (now)
  ‣ Background and Related Work (11/6)
  ‣ Project Status Review (12/4)
  ‣ Final Project Write-up (12/21)

• This is a very important factor in your grade (30%) so you better take it seriously
  ‣ E.g., an exceptionally good (or poor) project may help (kill) grade
Project Teams

- **Cloud**: Wang, Ren, Suresh, Fan
- **Network**: Li, Nath, Song, Manisha
- **Packages/Name Res**: Malak, Severn, Ge, Yibo
- **Browser**: Irani, Wu, Sedghi, Tseng
- **Passwords**: Meghan, Ference, Schmidt
- **Mobile/Android**: Heqing, Jones, Meiram
- **Hardware**: Zhang, Zhan, Chi, Guttman
Project Choices

• Find Vulnerabilities
  ‣ Apply known methods to find flaws in new code
  ‣ Apply methods to find different vulnerability
  ‣ Use threat model to find flaws

• Prevent/Detect Attacks
  ‣ Apply known methods to protect new code
  ‣ Apply known methods to detect/prevent different attacks
  ‣ Use threat model to prevent attacks

• Prove Code Is “Secure”
  ‣ Define security and prove under assumptions
Project Actions

• I will provide some research papers next week
  ‣ Meet the week of 10/8

• Ideas:
  ‣ **Cloud**: Attack services (storage) for OpenStack
  ‣ **Network**: Malware, bots, bad guys runs your net
  ‣ **Packages**: Find name resolution and other pkg bugs
  ‣ **Browser**: Attacks on browsers, apps; WebKit defenses
  ‣ **Passwords**: Crackers, Ensure users get a good passwd
  ‣ **Mobile**: Get Android threats, launch and detect attacks
  ‣ **Hardware**: Lots of new HW features, evaluate via threats
Paper Evaluation

• A paper is evaluated on
  ‣ Novelty
  ‣ Correctness
  ‣ Impact
  ‣ Presentation
  ‣ Relevance
  ‣ “hotness”
Parts of a paper

• Parts of paper (vast generalization)

1. Abstract
2. Introduction
3. Problem /Related Work/Background
4. Solution/Experiment
5. Evaluation/Analysis/Implementation
6. Discussion (often, but not always)
7. Conclusions
Abstract

• One sentence each for:
  ‣ Area
    • Topic of work
  ‣ Problem
    • What’s the issue?
  ‣ Solution
    • How do you propose to address the problem?
  ‣ Methodology
    • What’s the experiment?
  ‣ Results
    • What did you find?
  ‣ Take Away: Lesson
Introduction

• One paragraph each on:
  • Area
    ‣ More elaborate
  • Problem
    ‣ Scenario
  • Why is problem not solved
    ‣ Brief of related work or the challenge
  • Proposed insight ("In this paper, ...")
    ‣ What is the experiment?
• Contributions -- What will the reader learn?
• Boilerplate outline (?)
Related Work/Background

• This is a statement of the work that led to this one.
  ‣ example or scenario behind problem
  ‣ who has done work in the area
  ‣ areas that inspired this work (not just technology)

• There are several reasons for related work section:
  ‣ Motivate the current work
  ‣ Differentiate from past work
  ‣ Establish “bona fides”

• Background
  ‣ Outline the Problem
    • May use an example scenario
  ‣ Material Related to the Solution
    • Why hasn’t it been solved
Solution and Experiment

• Experiment
  ‣ Means of showing truth
  ‣ Big Insight -- Hypothesis -- Claim
    • Show why it is interesting
  ‣ Expected Results
    • Informal proof/argument that is true

• Experiment types
  ‣ *Empirical* - measure some aspect of the solution
  ‣ *Analytical* - prove something about solution
  ‣ *Observational* - show something about solution
Implementation and Results

• Implementation: Experimental Platform
  ‣ Exact specification of platform
  ‣ Design may have more than implementation -- what did you implement?
  ‣ How are key design features/mechanisms implemented?

• Results
  ‣ Summarize -- what do the results mean?
  ‣ Specific experiments
    • We did X, saw Y
  ‣ What do the experiments prove
  ‣ What other experiments would you want to do based on these results?
Conclusion

• Like the abstract in past tense

• Problem
  ‣ What was the problem?

• Solution
  ‣ What was the insight and why was it expected to work?

• Method and Results
  ‣ What did you find?

• Take away: Lesson

• Future work
Hint

• Intro: tell them what you are going to tell them
• Body: tell them
• Conclusion: tell them what you told them.