Software Construction Laboratory

Week 7 Part 1

Lab 3

Mushi Zhou

Winter 2017

UCLA

Prof. Paul Eggert

Assignment 6 Help

- This is not a difficult assignment to do, but the concept behind are very important!
- For the lab, you want to time only the sorting time, not including the time to create test data
- To create test data, single-precision float points are 4 bytes, make sure you “od” correct number of values
- Use –A n to not print number byte offset
- Use “tr” to transform space to ‘\n’
- Use “sed” to delete all empty lines

Assignment 6 Help

For HW

- Do something useful using multiple threads
- You are doing computation for all pixels (width * height)
- The easiest would be running the entire pixel nested loop within each thread, and only execute the loop for selected pixels for that particular thread
- Define the entire nested to run in a new function with a pixel selecting variable
- Then you need to pass argument for each thread, what you can do is to define a struct that contains variables to each thread, because pthread only allow input function to take one argument (You need to malloc all arguments)
- Cast the argument to (void*) and cast back in each thread
- No need to pass enum values, which are global

Assignment 6 Help

- You need to modify makefile to compile the library using pthread
- (-lpthread) option
- Use diff to compare your output to the given output
- The are graphs, and you can view them using GIMP

Assignment 7 Announcement

- Ubuntu DVD Handout
- You are working with a team of 3
- Here is the assigned teams for you:

https://docs.google.com/a/g.ucla.edu/spreadsheets/d/14EWCZxL1x5By-Feub0b4G0tKC4jszQ-afDxuJa8a1o0/edit?usp=sharing

- If you can’t find your teammates, use this page to find his/her email: http://www.directory.ucla.edu/
- If you didn't get the DVD in class, come to the class next Monday
- Or your can drop by my office hours this week

What’s Going On with Assignment 7

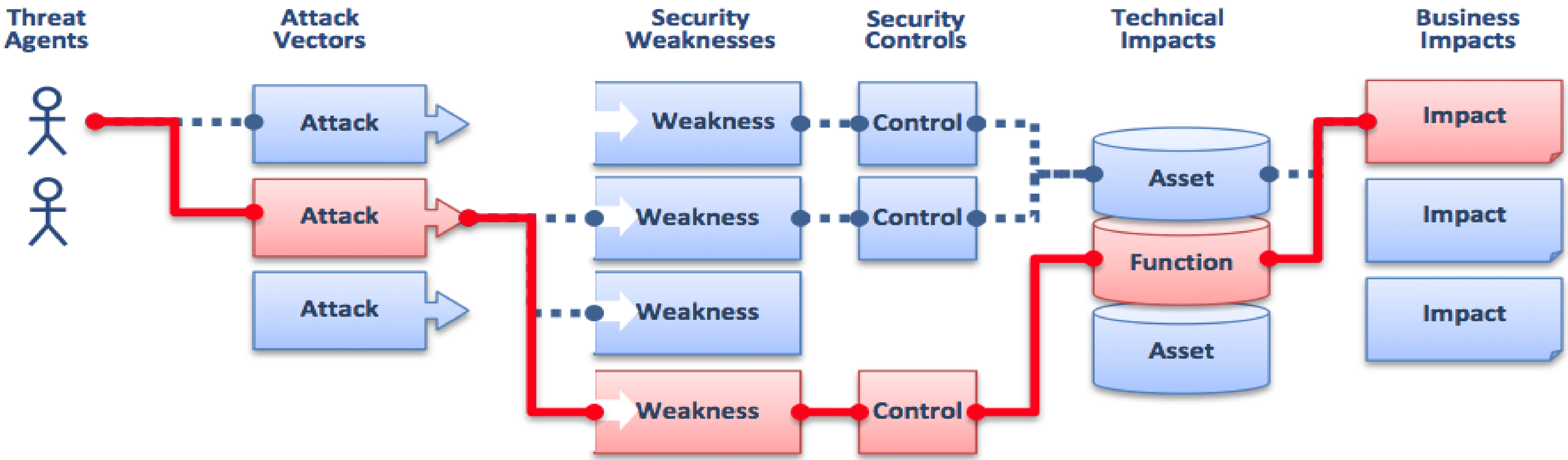
- DVDs handed out contain an image of Ubuntu 16.04.2 32bit
- If you insert the DVD into the lab machines, it will automatically boot into Ubuntu from DVD
- On your own computer, set boot order to start with DVD in your BIOS
- Choose “Try Out Ubuntu”, then you can use the OS. No installation needed
- Keep in mind nothing you do can be saved
- The assignment also serves as a goal for you to play around with Ubuntu
- You can start working on the assignment. I will be talking about some key knowledge you need next Monday to help you out

Outline

- Security Threats
- Authentication & authorization
- Chains of trust
- Firewalls & sandboxes
- Intrusion detection
- Backups
- Security policies
- OpenSSH (Part 2)
- ssh-agent (Part 2)
- GNU Privacy Guard (Part 2)

Security Threats

- In computer security, a threat is a possible danger that might exploit a vulnerability to breach security and therefore cause possible harms
- A threat can be either "intentional" or "accidental“, like electrical outrage



Common Types of Threats

Categorized by what they do:

- Spoofing of user identity (forgery)
- Tampering
- Repudiation (Author cannot authenticate itself)
- Information disclosure (privacy breach or Data leak or eavesdropping)
- Denial of Service (D.o.S.)
- Elevation of privilege

Spoofing

- Spoofing attack is a situation in which one person or program successfully masquerades as another by falsifying data, thereby gaining an illegitimate advantage

Examples:

- IP Address Spoofing Attacks
- GPS spoofing (militaries combats)
- E-mail address spoofing (fishing emails)

Denial of service (DDoS attack)

- A cyber-attack
- Temporarily or indefinitely disrupting services of a host connected to the Internet
- To make a machine or network resource unavailable to its intended users
- Typically accomplished by flooding the targeted machine or resource with superfluous requests in an attempt to overload systems and prevent some or all legitimate requests from being fulfilled
- The outage of web services such as Github, Tweet, towards the end of last year, was the result of such type of attack

Counter Measures to Threats

1. Establishing trust
2. Mechanism to block threats
3. Detecting threats
4. Backups
5. Policies

Establishing Trust

- Authentication vs Authorization
- What’s the difference?

- Two examples:
- 1. You type in your login and password to a website
- 2. You tires to run a file in /usr/bin on SEASNet Server and get: permission denied

- Which one is which?

Authentication VS Authorization

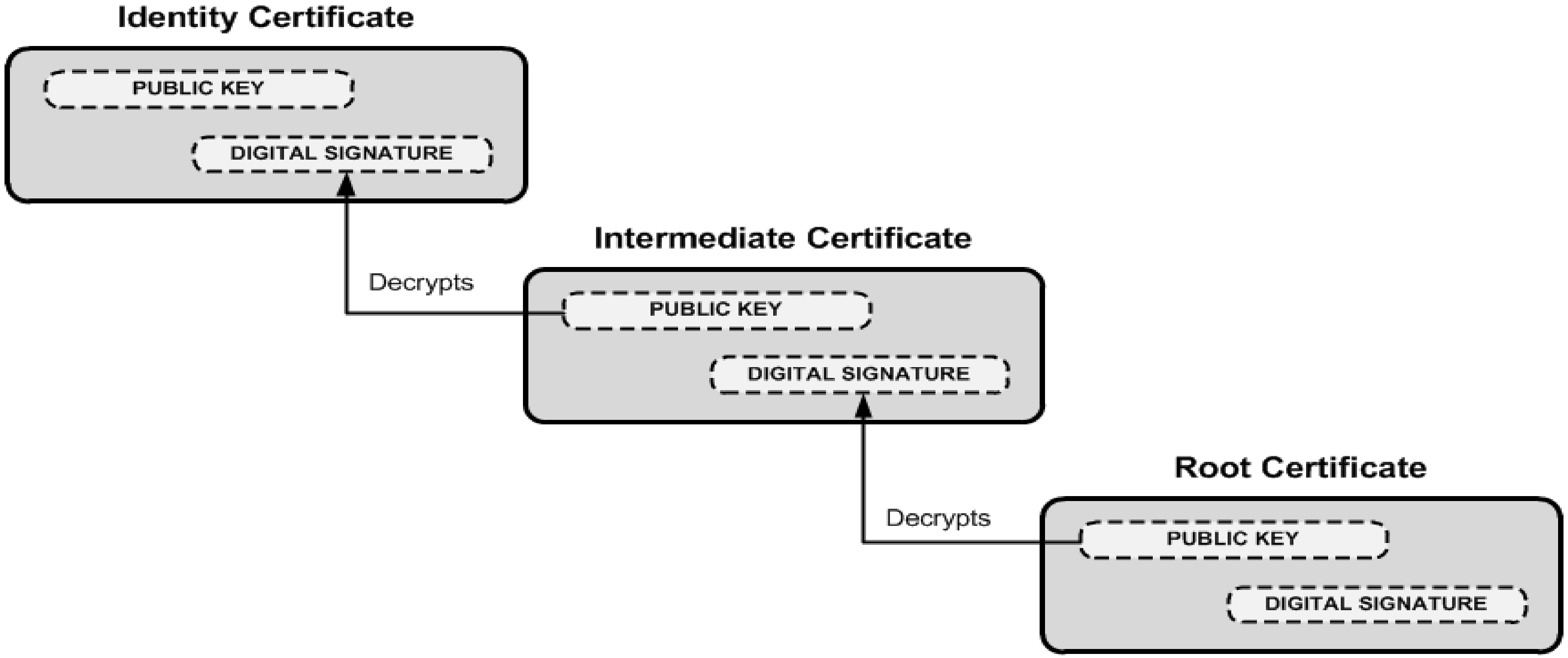
- Authentication is the process of ascertaining that somebody really is who he claims to be

- Authorization refers to rules that determine who is allowed to do what

- They are completely independent

Chain of Trust

- In order to create scalable & flexible services, there are often chain of trust
- This means there can be multiple layers of trusts or certificates in an application, so that not every stage needs to check with the trust anchor
- The top layer maintains certain properties that is propagated from the bottom layer



Example of Chain of Trust

1. Digital certificates are verified using a chain of trust

- You visit a website that is green on the left of its address, meaning it certificate if valid
- How do we know it if valid? Because it is signed by a authorized agent that resides higher in the trust chain. Eventually, the rust anchor is the root certificate authority, which is only a small number of authorities that is known by your browser

2. You are creating a account on a website. You are given the option to create using your Facebook account

- Since Facebook trust you, the website trusts Facebook. There is no need for the website to check with you, who is the trust anchor

Mechanism to Block Threats

Sandbox

- Used to execute untested or untrusted programs or code, possibly from unverified or untrusted third parties, without risking harm to the host machine or operating system
- By providing a tightly controlled set of resources for guest programs to run in (such as scratch space on disk and memory, network access, the ability to inspect the host system or read from input devices are usually disallowed or heavily restricted)

Example:

- Restrict HTML parser to not load any styles

Mechanism to Block Threats

Firewalls

- Network security system that monitors and controls the incoming and outgoing network traffic based on predetermined security rules
- By establishing a barrier between a trusted, secure internal network and another outside network, such as the Internet, that is assumed not to be secure or trusted

Example

- Firewalls on you OS

Detecting Threats

- Intrusion detection systems (IDS)

Two types:

- HIDS (host-based intrusion detection systems) -> monitoring operating system files
- NIDF (network intrusion detection systems) -> monitoring network traffics

Detection approach:

- signature-based detection (You will touch this in your HW)
- anomaly-based detection (Often requires machine learning)

Backups

- The copying and archiving of computer data so it may be used to restore the original after a data loss event
- Can be automatic or manually

- Examples:
- Your iCloud backups
- SEASNet snapshot feature

Security Policies

- The set of definition of what it means to be secure for a system
- Each unique system has its unique set of security policies
- It is important to be sure all of the security policy is enforced by mechanisms that are strong enough
- In complex systems, policies can be decomposed into sub-policies to facilitate the allocation of security mechanisms to enforce sub-policies
- A top-level security policy is essential to any serious security scheme and sub-policies and rules of operation are meaningless without it

Security Policies

- Enforcing security policies is like testing a program
- Enforcing sub policies is like unit testing
- Enforcing top policies is like system testing
- Complete correctness of either individual part alone does not guarantee the overall correctness.

Example:

- SEASnet server connection will drop any connection request if that incoming IP address is not within UCLA domain (This is why you need to use VPN at home)
- SSH protocol has a set of strongly enforced security policies