Introduction and Security Principles

CS 161 Spring 2022 - Lecture 1

First: Introduction

- Staff introductions: Nick and course staff
- Course overview: What will you learn in this class?
- Course logistics
  - Lectures, discussions, office hours, and exams
  - Resources and communication platforms
  - Collaboration and academic honesty
  - DSP and extenuating circumstances
  - Stress management and mental health
  - Ethics
  - Case studies and blue slides
- What is security? Why is it important?

Staff Introductions

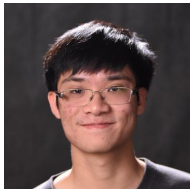
Who Am I? Nick Weaver

- A *lecturer* in the CS department
  - I am paid *exclusively* to care about my students & TA staff
- A researcher at the International Computer Science Institute
- Research focuses
  - Online criminality (including cryptocurrency)
    - Cryptocurrency is an amazing resource for comedy godl.
  - Online privacy
  - Public policy
  - Drones

Our team of talented TAs!

EvanBot
any/all

board games!



Nicholas Ngai
he/him

Shomil Jain
he/him

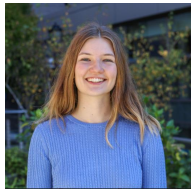
Learning to swim

Dev Bali

Siddharth Bansal
he/him

go Warriors

Grace Chen
she/her



Ana Cismaru
she/her

Prachi Deo

Jinan Jiang
he/him

Elizabeth John
she/her

procrastinates by playing piano

Solomon Joseph
he/him

Kenneth Lien
he/him

learning to surf!

Ayush Sehgal
he/him

Yogurt Park Sucks

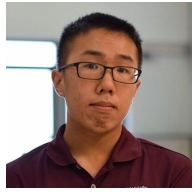
we're still not really sure what his title is…

Peyrin Kao
he/him

Fuzail Shakir
he/him

piazza and pizza addiction

Efsane Soyer
she/her



Simon Tsui
he/him

Sheqi Zhang
she/her

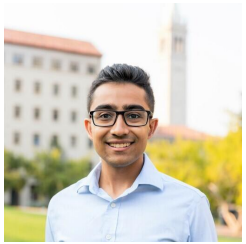
Boba runs in my blood

Abel Yagubyan
he/him

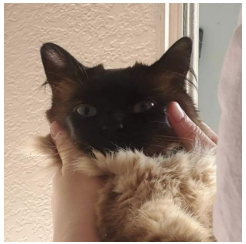
Our team of talented readers!

EvanBot
any/all

Jun Hee Han
he/him



Pranav Sukumar
he/him



is raising caterpillars

Vron Vance
they/them

Course Overview

Learning Objectives

- How to think adversarially about computer systems
- How to assess threats for their significance
- How to build computer systems with robust security properties
- How to gauge the protections and limitations provided by today's technology
- How attacks work in practice

- What mistakes *not to make!*

Course Outline

- Introduction to Security
  - What are some general philosophies when thinking about security?
- Memory Safety
  - How do attackers exploit insecure software? How do we defend against these attacks?
- Cryptography
  - How do we securely send information over an insecure channel?
- Web Security
  - What are some attacks on the web, and how do we defend against them?
- Network Security
  - What are some attacks on the Internet, and how do we defend against them?
- Miscellaneous Topics
  - Useful, interesting, or fun applications of topics

Extra Tools and Skills

- Some extra non-security-related skills you can take away from this class:
- Memory safety
  - x86 assembly: A commonly-used assembly language
  - Using GDB: Debugging C code
- Cryptography
  - Becoming a better consumer: Be able to analyze security products and pick the right security tools for your software
- Web Security
  - Software engineering: Understanding how websites are built and how your web browser interacts with remote web servers (CS 169 preview)
- Network Security
  - Networking: How the Internet works (CS 168 preview)

Prerequisites

- CS 61B: Ability to work with large and complex codebases, data structures
  - Relevant for Project 2 (500–1000 lines of Go code)
- CS 61C: Familiarity with low-level memory layouts and assembly
  - We'll have a lecture reviewing all the 61C material you need to succeed
  - Relevant for the memory safety unit (Project 1, first two weeks of class only)
- CS 70: Familiarity with basic mathematical notation and proof structures
  - Relevant for the cryptography unit
  - We'll review CS 70 material as we encounter it during the cryptography lectures
- An ability to pick up new programming languages quickly
  - Project 2 will be in Go

Course Logistics

Enrollment

- On January 14th, we expanded and pulled off students off the waitlist who indicated a special need to enroll in the course
  - We will *not* be expanding any further because there is no additional budget available
  - Current enrollment: 590 enrolled, ~100 waitlisted
  - Historically, about 10% of students enrolled drop the course

Hybrid Logistics

- We want to be as flexible and accommodating as possible
- The entire class can be completed in-person or remotely

Course Structure: Lectures

- You are here!
- Tuesday/Thursday, 12:30–2:00 PM PT
- First two weeks of lectures are online-only
  - In-person lectures start the week of Monday, February 1st
- Attendance is not taken

In-person	Synchronous online	Asynchronous online
• Live lectures in Dwinelle 155	• Live lectures over Zoom	• Lecture recordings posted on the website

Course Structure: Discussions

- Smaller sections led by a TA to practice with material
- Discussion schedule available at https://cs161.org/calendar
- All discussions start **next week** and are online-only for the first week
  - In-person discussions start the week of Monday, February 1st
- You can attend any discussion section you want
- Attendance is not taken

In-person	Synchronous online	Asynchronous online
• Discussion rooms posted on Piazza/calendar	• Discussion zoom links posted on Piazza	• Some sections will be recorded • Discussion walkthrough videos will be posted

Course Structure: Office Hours

- Space to ask questions about content, get help with projects, raise concerns with the course, etc. with a TA or instructor
- Office hours schedule available at https://cs161.org/calendar
- Office hours queue at https://oh.cs161.org/
- All office hours start **this week**
- All office hours will be available both in-person and online (after the first two weeks)

Course Structure: Exams

- Midterm: Friday, February 25th, 7:00–9:00 PM PT
- Final: Thursday, May 12th, 3:00–6:00 PM PT
- Everyone can choose between an in-person or online exam
  - Everyone is assigned to an in-person exam by default
  - To switch to a online exam, we need a reason, but *any* reason is acceptable
    - I'm not in Berkeley
    - I don't feel comfortable in a crowded room
    - I want to take the exam with my cat on my lap
- Alternate times are available for both in-person and online students
  - Primary time immediately after the scheduled exam
- Forms to request online/alternate exams will be released closer to the exam

Resources

- Textbook: https://textbook.cs161.org/
  - Free! There's no textbook you need to pay for.
  - Readings are optional, but past students have said the textbook is helpful
- Course website: https://cs161.org/
  - Course schedule, lecture slides, assigned readings, and other resources are all posted here

Platforms

- Piazza
  - Course-related communication should take place in Piazza or happen in office hours
  - For private matters, you can make a private post
  - Please don’t post publicly about project spoilers!
- Gradescope
  - All assignments are submitted and graded on Gradescope
- Email
  - cs161-staff@berkeley.edu for private matters
  - Piazza response is faster, but staff will monitor the email if you’re more comfortable with that

Grading Structure

- Homework: 10%
  - Completed individually
  - 7 homeworks in total, weighted equally
  - Gradescope with instant feedback: You can keep trying until you get the answer right
  - No credit for late homework, but we drop your lowest homework score
- Projects: 40%
  - P1 = 10%, P2 = 20%, P3 = 10%
  - Completed individually or in groups of 2
  - Reduced credit for submitting late (no credit after 3 days late)
- Midterm: 20%
- Final: 30%

Class Policies: Extensions

- We **do not** have slip days in this course!
- However, you can request extensions on any assignment for any reason:
  - https://cs161.org/extensions.html
  - Extensions ≤3 days will be automatically approved if submitted before the deadline
  - Extensions ≤7 days may be approved by a logistics TA
    - If not approved by the TA, the instructor may still approve the extension: Logistics TA can say yes but it takes the instructor to say no
    - You still have the automatic 3 day approval in any case
  - Extensions >7 days may be approved by the instructor
- It is **okay** to request an extension if things come up! We're here to support you!
  - Life happens.
  - You can always request an extension for any reason
  - We want to keep the forcing function effect of deadlines while reducing the associated stress

Class Policies: DSP

- Disabled Students' Program (DSP)
  - There's a variety of accommodations UC Berkeley can help us set up for you in this class
  - https://dsp.berkeley.edu/
- Are you facing barriers in school due to a disability?
  - Apply to DSP!
  - We maintain proper access controls on this information: Only instructors, course managers, head TAs, and logistics TAs can access any DSP-related info
- Our goal is to teach you the material in our course. The more accessible we can make it, the better.

Class Policies: Collaboration

- Asking questions and helping others is encouraged
  - Discussing course topics with other is welcome!
- Limits of collaboration
  - Don’t share solutions with each other (except project partners)
  - You should never see or have possession of anyone else’s solutions—including from past semesters

Class Policies: Academic Honesty

- We’re here to help! There are plenty of staff and resources available for you
  - You can always talk to a staff member if you’re feeling stressed or tempted to cheat
- Academic dishonesty policies
  - At minimum, the student will receive negative points on the assignment
    - Example: If the midterm is worth 150 points, the student will receive a score of -150 on the midterm.
  - The student will be referred to Nick Weaver and the Center for Student Conduct
    - CSC often doesn’t care that much about first-time cases! They are there to make sure a student doesn’t make the same mistake a second time.
  - If you take the class honestly, you don’t need to worry about these!

Class Policies: Academic Honesty

- As a computer security class, we view potential cheaters as “attackers.”
- Our threat model assumes “rational” attackers.
  - A rational attacker will only launch an attacker if (expected benefit) > (expected cost)
    - (expected cost) = (cost of launching attack) + *(cost of getting caught)* * (probability of getting caught)
- Two-fold approach to academic integrity:
  - Detection: Use our tools to analyze and detect instance of academic dishonesty.
    - You will learn that “security through obscurity” is bad, but *obscurity can help*. We have ways.
  - Response: *At minimum*, you will receive negative points on the assignment.
- “Nick doesn’t make threats. He keeps promises.”

Ethics

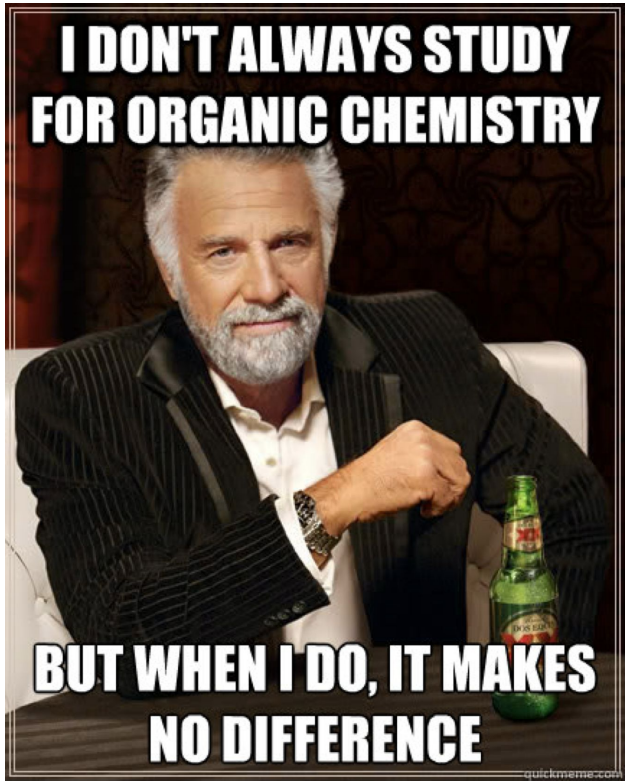
- In this class, you will learn a lot about attacks out of necessity
  - To be able to defend against the attacker, you must learn the techniques that attackers use
- It is usually okay to break into your own systems
  - This is a great way to evaluate your own systems
- It is usually okay to break into someone else's systems with their explicit permission
- It is **grossly unethical** and **exceedingly criminal** to break into someone else's systems without their permission

Stress Management and Mental Health

- We want to reduce your stress where we can
  - Project 2 (mid-semester) is going to be the most intensive part of this class, but we've made things lighter towards the end (when every other class has stuff due)
- **Your health is more important than this course**
- If you feel overwhelmed, there are options
  - Academically: Ask on Piazza, talk to staff in office hours, set up a meeting with staff to make a plan for your success this semester
  - Non-academic:
    - Counselling and Psychological Services (CAPS) has multiple free, confidential services
      - Casual consultations: https://uhs.berkeley.edu/counseling/lets-talk
      - Crisis management: https://uhs.berkeley.edu/counseling/urgent
    - Check out UHS's resources: https://uhs.berkeley.edu/health-topics/mental-health

Stress Management and Mental Health

- Failure is always an option
  - If something bad happens near the end of the semester, there are withdrawals and incompletes.
  - It is OK to fail or just barely pass… Nick's grades as a Berkeley Undergrad included:
    - B- in Physics 111BSC & Thermodynamics
    - C+ in Chem 112A (O-chem)
    - C in Physics 137A (Quantum)
  - Don't believe me? Stop by my office and see my transcript!



Case Studies and Blue Slides

- Security is often best taught through real-world case studies and stories
  - Lectures will sometimes use real-world examples to demonstrate concepts
  - Slides with a blue background are case study slides
- Content on blue slides are not tested on exams
  - You *do not* need to remember the exact details of the story
- Some blue slides will end in a **takeaway** that describes the moral of the story
  - You *do* need to understand the takeaway from the story

One Other Thing...

- There exists a classic game theory problem called the Prisoner's Dilemma.
  - For single-round Prisoner's Dilemma, the optimum strategy is "always-defect."
  - For multi-round Prisoner's Dilemma, the optimum strategy in practice is "tit-for-tat."
    - In other words, be nice unless someone isn't nice to you.
- **Takeaway**: Life is multi-round so be excellent to each other!
  - Making things hostile for others makes the world worse for all.
  - Stopping things from being hostile to others makes the world better for you.

What is security?

What is security?

Enforcing a desired property *in the presence of an attacker*

data confidentiality

user privacy

data and computation integrity

authentication

availability

…

Why is security important?

- It is important for our
  - physical safety
  - confidentiality/privacy
  - functionality
  - protecting our assets
  - successful business
  - a country's economy and safety
  - and so on…

Why is security important?

- Consider: Physical Safety

The Washington Post

Link

FBI probe of alleged plane hack sparks worries over flight safety

Drew Harwell *May 18, 2015*

PCWorld

Link

Pacemaker hack can kill via laptop

Jeremy Kirk *October 21, 2012*

Why is security important?

- Consider: Privacy/Confidentiality

Link

91 Percent of Healthcare Organizations Suffered Data Breaches in the Past Two Years

Jeff Goldman *May 12, 2015*

Link

Data Breach Tracker: All the Major Companies That Have Been Hacked

Karavbrandeisky *October 30, 2014*

In 2020, there were over 1001 breaches, affecting the data of 155,000,000 individuals

Why is security important?

- Consider: National security

THE WALL STREET JOURNAL.

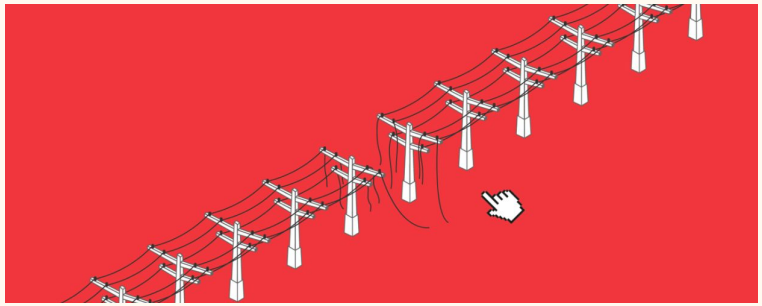
Link

America's Electric Grid Has a Vulnerable Back Door—and Russia Walked Through It

Rebecca Smith and Rob Barry

January 10, 2019

A Wall Street Journal reconstruction of the worst known hack into the nation's power system reveals attacks on hundreds of small contractors



What is hackable?

- Everything!
  - Especially things connected to the Internet
  - Assume that every system is a target
  - A casino was hacked because a fish-tank thermometer was hacked within the network

SLATE

Link

For the First Time, Hackers Have Used a Refrigerator to Attack Businesses

Julie Bort *January 17, 2014*

Next: Security Principles

- Security principles
  - Know your threat model
  - Consider human factors
  - Security is economics
  - Detect if you can’t prevent
  - Defense in depth
  - Least privilege
  - Separation of responsibility
  - Ensure complete mediation
  - Don’t rely on security through obscurity
  - Use fail-safe defaults
  - Design in security from the start

Know Your Threat Model

Textbook Chapter 1.1 & 1.12

The Parable of the Bear Race

Reminder: blue slides are case studies. Remember the takeaway, not the story!

"I don't have to outrun the bear. I just have to outrun *you*."

Takeaway: Even if a defense is not perfect, if it is more advantageous for attackers to attack somewhere else, it can be effective

Know Your Threat Model

- **Threat model**: A model of who your attacker is and what resources they have
- It all comes down to people: The attackers
  - No attackers = No problem!
  - One of the best ways to counter an attacker is to attack their reasons
- Why do people attack systems?

Money

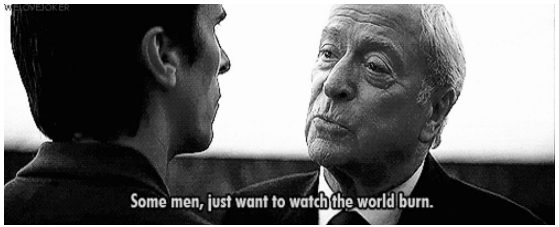
Advertisers

Politics

Anonymous

NSA

Fun



To watch the world burn

Security Principle: Know Your Threat Model

- Consider: Personal security
- Who and why might someone attack *you*?
  - Criminals might attack you for money
  - Teenagers might attack you for laughs or to win online games
  - Governments might spy on you to collect intelligence
  - Intimate partners might spy on you
    - This is a surprisingly dangerous threat model!

The National Security Agency (NSA)

- Stated purpose: To collect information to protect US national security
- Since its founding in 1952, the NSA has:
  - Decoded secret enemy communications in wars
  - Spied on people in the US and other countries (sometimes legally, sometimes illegally)
  - Participated in security research and helped develop security standards
  - Developed secret techniques for surveillance and cyberattacks

Threat Model: Common Assumptions for Attackers

- Assume the attacker…
  - Can interact with systems without notice
  - Knows general information about systems (operating systems, vulnerabilities in software, usually patterns of activity, etc.)
  - Can get lucky
    - If an attack only succeeds 1/1,000,000 times, the attacker will try 1,000,000 times!
  - May coordinate complex attacks across different systems
  - Has the resources required to mount the attack
    - This can be tricky depending on who your threat model is
  - Can and will obtain privileges if possible

Trusted Computing Base

- **Trusted computing base** (**TCB**): The components of a system that security relies upon
- Question: What would you want from a TCB?
- Properties of the TCB:
  - Correctness
  - Completeness (can't be bypassed)
  - Security (can't be tampered with)
- Generally made to be as small as possible
  - A smaller, simpler TCB is easier to write and audit.
  - **KISS principle**: Keep It Simple, Stupid

Consider Human Factors

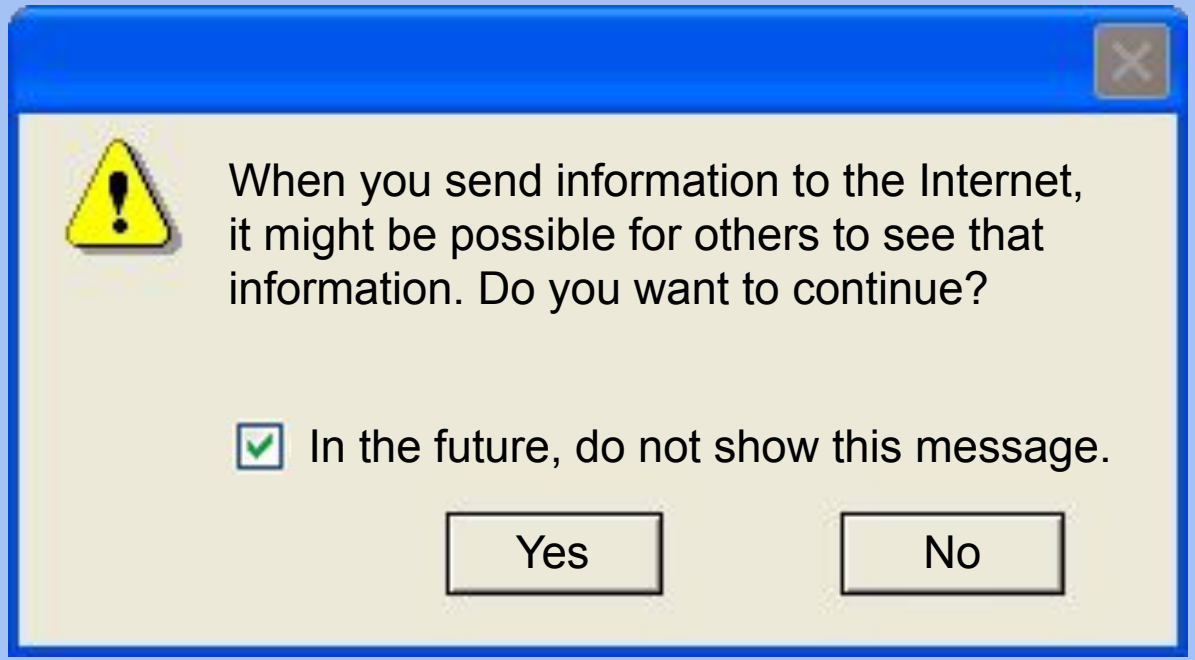
Textbook Chapter 1.2

It All Comes Down To People

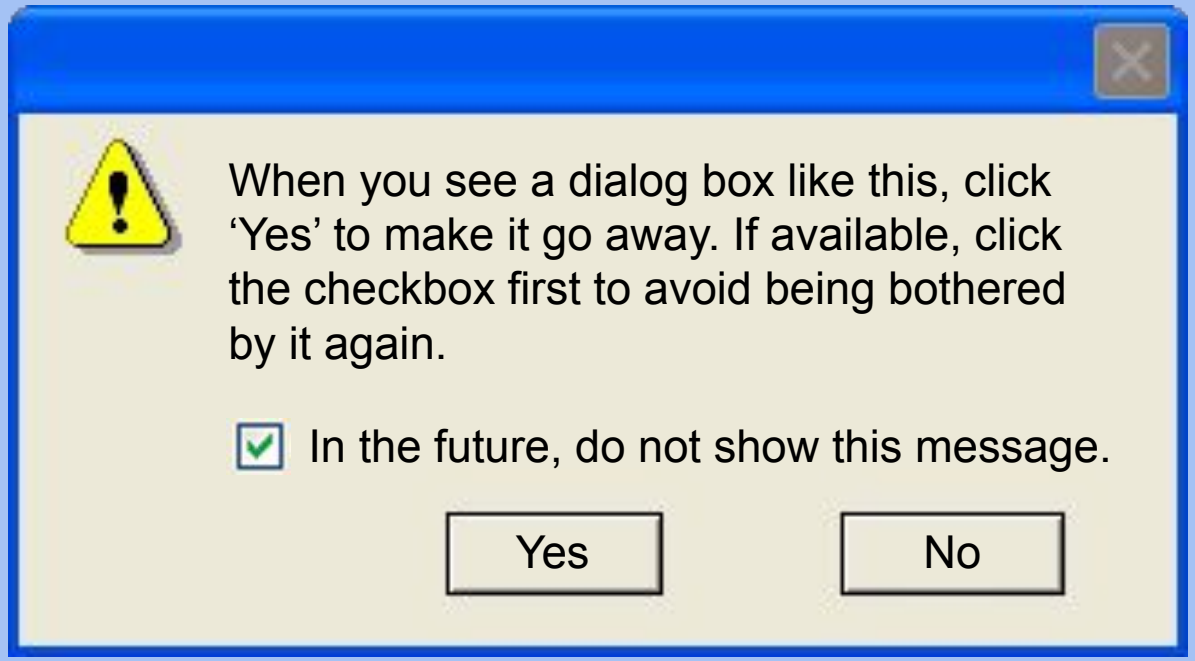
- The users
  - Users like convenience (ease of use)
  - If a security system is unusable, it will be unused
  - Users will find way to subvert security systems if it makes their lives easier
- The programmers
  - Programmers make mistakes
  - Programmers use tools that allow them to make mistakes (e.g. C and C++)
- Everyone else
  - Social engineering attacks exploit other people's trust and access for personal gain
- Consider the tools presented to users, and make them *fool*-proof

Physical security keys are designed to look like keys because humans are trained to protect keys

Warning Dialogs

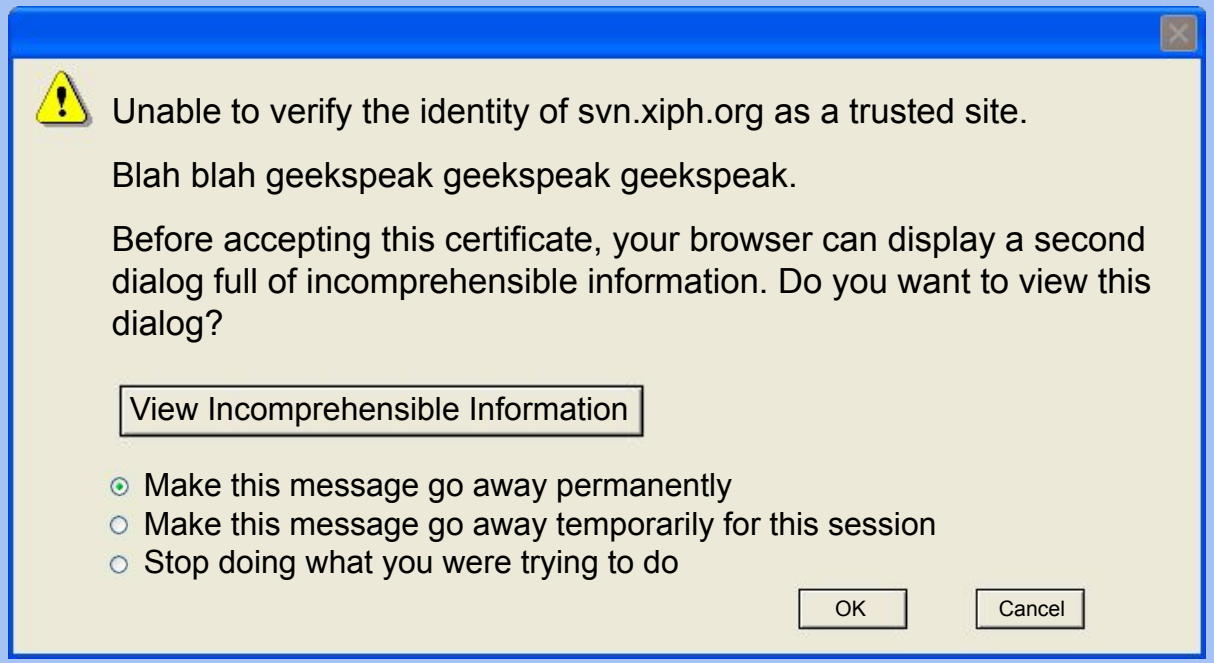


Warning Dialogs



Warning Dialogs

Warning Dialogs



The presence of warning dialogs often represent a failure: How is the user supposed to know what to do?

Takeaway: Consider human factors

Security is Economics

Textbook Chapter 1.3

Security is Economics

- Cost/benefit analyses often appear in security
  - The cost of your defense should be less than the cost of attacks happening
  - More security (usually) costs more
  - If the attack costs more than the reward, the attacker probably won't do it
- Example: You don't put a $10 lock on a $1 rock…
  - … unless a $1 rock can be used to attack something even more valuable
- Example: You have a brand-new, undiscovered attack that will work on anybody's computer. You wouldn't expose it on a random civilian.
  - iPhone security vulnerabilities are often sold for ~$1M on the market

Physical Safes

- We want our safes to stop people from breaking in, so let's measure them by how long it takes an expert to break into one:

TL-15 ($3,000)
15 minutes with common tools

TL-30 ($4,500)
30 minutes with common tools

TRTL-30 ($10,000)
30 minutes with common tools and a cutting torch

TXTL-60 (>$50,000)
60 minutes with common tools, a cutting torch, and up to 4 oz of explosives

Takeaway: Security is economics

Burglar Alarms

- Security companies are supposed to detect home break-ins
  - Problem: Too many false alarms. Many alarms go unanswered
  - Why is it useful to place a sign?
  - Placing a sign helps deter burglars from entering at risk of being caught…
    - … even if you don't have an alarm installed!
  - An attacker might prefer the neighbor without a sign

Detect If You Can’t Prevent

Textbook Chapter 1.4

Detect if You Can’t Prevent

- **Deterrence**: Stop the attack before it happens by making the attacker prefer to do something else
- **Prevention**: Stop the attack before it happens by making the attack fail
- **Detection**: Learn that there was an attack (after it happened)
  - If you can’t stop the attack from happening, you should at least be able to know that the attack has happened.
- **Response**: Do something about the attack (after it happened)
  - Once you know the attack happened, you should respond
  - Detection without response is pointless!

Response: Mitigation and Recovery

- Assume that bad things will happen! You should plan security in way that lets you to get back to a working state.
- Example: Earthquakes
  - Have resources for 1 week of staying put
  - Have resources to travel 50 miles from my current location
- Example: Ransomware
  - Ransomware: An attacker steals your data and demands payment in exchange for recovering your data
  - Keep offsite backups!
  - If your computer and house catch on fire, it should be no big deal.

Detection but no Response

- Cryptocurrency transactions are irreversible. If you are hacked, you can never recover your Bitcoins.
  - $68M stolen from NiceHash exchange in December 2017
  - Four multi-million-dollar attacks on Ethereum in July 2018
  - Coinbase: One *detected* theft per day
  - Keep track of the fun at web3isgoinggreat.com
- **Takeaway:** Prevention is great, but depending only on prevention can be *brittle*: When prevention fails, the system fails catastrophically

Bloomberg Link

Hacked Bitcoin Exchange Says Users May Share $68 Million Loss

Lulu Yilun Chen and Yuji Nakamura *August 5, 2016*

January 14, 2022

An attacker pulls about 350 ETH from Float Protocol's Rari Capital pool

Lack of liquidity in the Uniswap V3 FLOAT/USDC oracle allowed an attacker to manipulate the prices within the pool, then deposit it at a much higher rate. The hacker pulled about 350 ETH (equivalent to $1.1 million) out of the pool, though according to PeckShield they later returned around $250,000 for some reason.

Float PROTOCOL

Float Protocol logo
(attribution)

- Tweet thread by FloatProtocol
- Tweet by PeckShield

Hack or scam web3isgoinggreat.com *cryptocurrency*