



University of Colorado **Boulder**



CSCI 3403

INTRO TO

CYBERSECURITY

Lecture: 1-2

Topic:
Fundamentals

Presenter: Matt
Niemiec

Announcements

- We have a course email now: cyber@colorado.edu
 - Use for requests, accommodations, etc.
- TAs office hours are on Moodle, not Piazza
- Lecture capture now exists
- HW 1 and Project 1 will be up tomorrow



Design Principles



University of Colorado **Boulder**

Book's Principles

Economy of mechanism

Fail-safe defaults

Complete mediation

Open design

Separation of privilege

Least privilege

Least common mechanism

Psychological acceptability

Isolation

Encapsulation

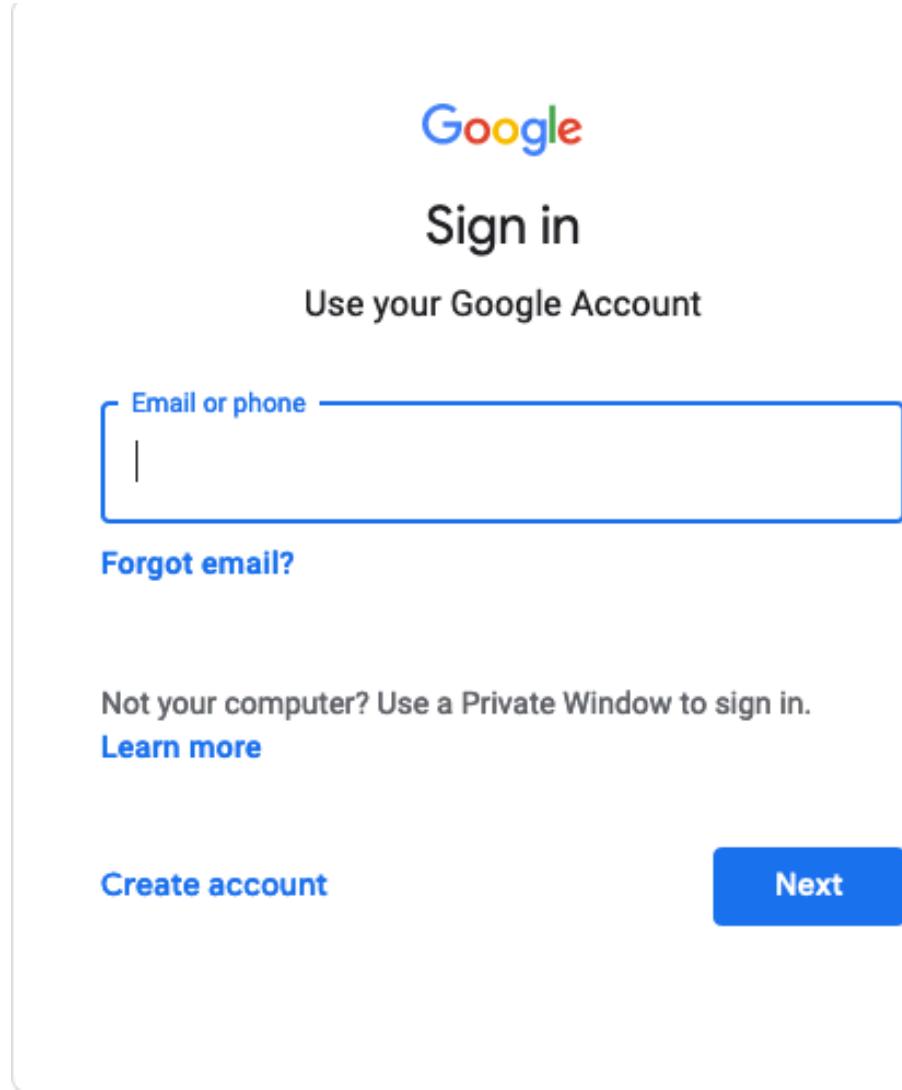
Modularity

Layering

Least astonishment



Example



The image shows a screenshot of the Google Sign-in page. At the top right, the "Google" logo is displayed in its signature blue, red, yellow, and green colors. Below it, the word "Sign in" is centered in a large, bold, black font. Underneath "Sign in", the text "Use your Google Account" is displayed in a smaller, gray font. A large input field is present, labeled "Email or phone" at its top left corner. To the right of the input field is a horizontal blue line. Below the input field, there is a small vertical line. At the bottom left of the input field area, the text "Forgot email?" is written in blue. Further down the page, the text "Not your computer? Use a Private Window to sign in." is displayed in gray, followed by a link "Learn more" in blue. At the bottom left of the page, there is a blue button with the text "Create account" in white. On the far right, there is another blue button with the text "Next" in white.

Google

Sign in

Use your Google Account

Email or phone

Forgot email?

Not your computer? Use a Private Window to sign in.
[Learn more](#)

Create account

Next



University of Colorado **Boulder**

Fail-Safe Defaults

- Permission should be given with explicit permission
- All systems fail
 - What will yours do when it does?
 - What are different ways it could fail?
- Fail-safe defaults in web login?



Least Privilege

- Every process and user shouldn't have more privileges than absolutely necessary
- If you don't need privileges, you shouldn't have them
- sudo chmod –R 777 /
- Uses in our example



Open Design

- One of the most commonly-violated principles
- Keep your algorithms known
 - E.g. encryption, secret bytes transfer
- You CANNOT design better than decades of review
- In our example



Minimize The Attack Surface Area

- Two things are twice as hard to defend as one thing
- Attack surfaces: hardware, software, data, networks
- Don't forget about old surfaces
 - Equifax
- Apply to our example?



Defense In Depth

- Don't rely on just one defense mechanism
- Redundancy is better
 - Redundant servers, authenticators, data
- In our example



Keep It Simple...

- If it's difficult to reason about, you probably made a mistake
- Assume people will take the time to break your system
- In our example



And Many Others!

- These are just a few
- Avoid getting caught in the details
- Know what each is protecting against



University of Colorado **Boulder**

Attack Trees



University of Colorado **Boulder**

Attack Surface Categories

Network Attack Surface

Vulnerabilities over an enterprise network, wide-area network, or the Internet

Included in this category are network protocol vulnerabilities, such as those used for a denial-of-service attack, disruption of communications links, and various forms of intruder attacks

Software Attack Surface

Vulnerabilities in application, utility, or operating system code

Particular focus is Web server software

Human Attack Surface

Vulnerabilities created by personnel or outsiders, such as social engineering, human error, and trusted insiders



Attack Surfaces

Consist of the reachable and exploitable vulnerabilities in a system

Examples:

Open ports on outward facing Web and other servers, and code listening on those ports

Services available on the inside of a firewall

Code that processes incoming data, email, XML, office documents, and industry-specific custom data exchange formats

Interfaces, SQL, and Web forms

An employee with access to sensitive information vulnerable to a social engineering attack



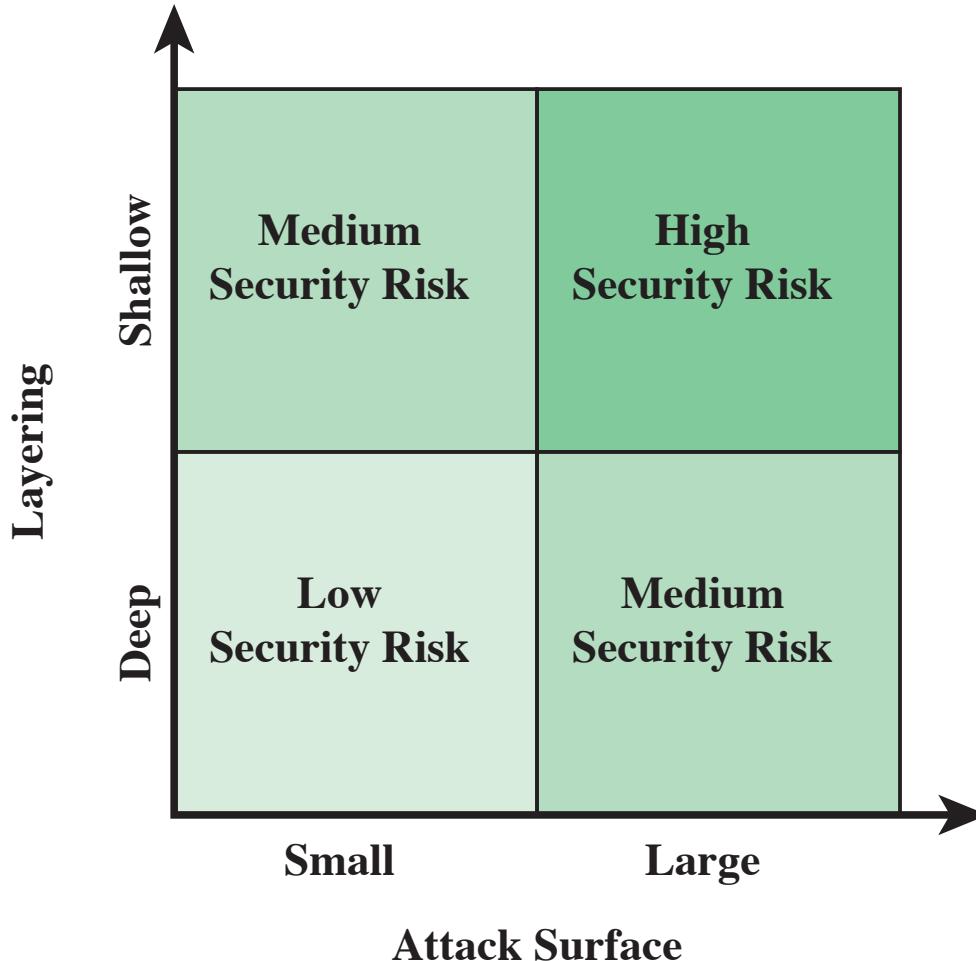


Figure 1.4 Defense in Depth and Attack Surface



Attack Trees

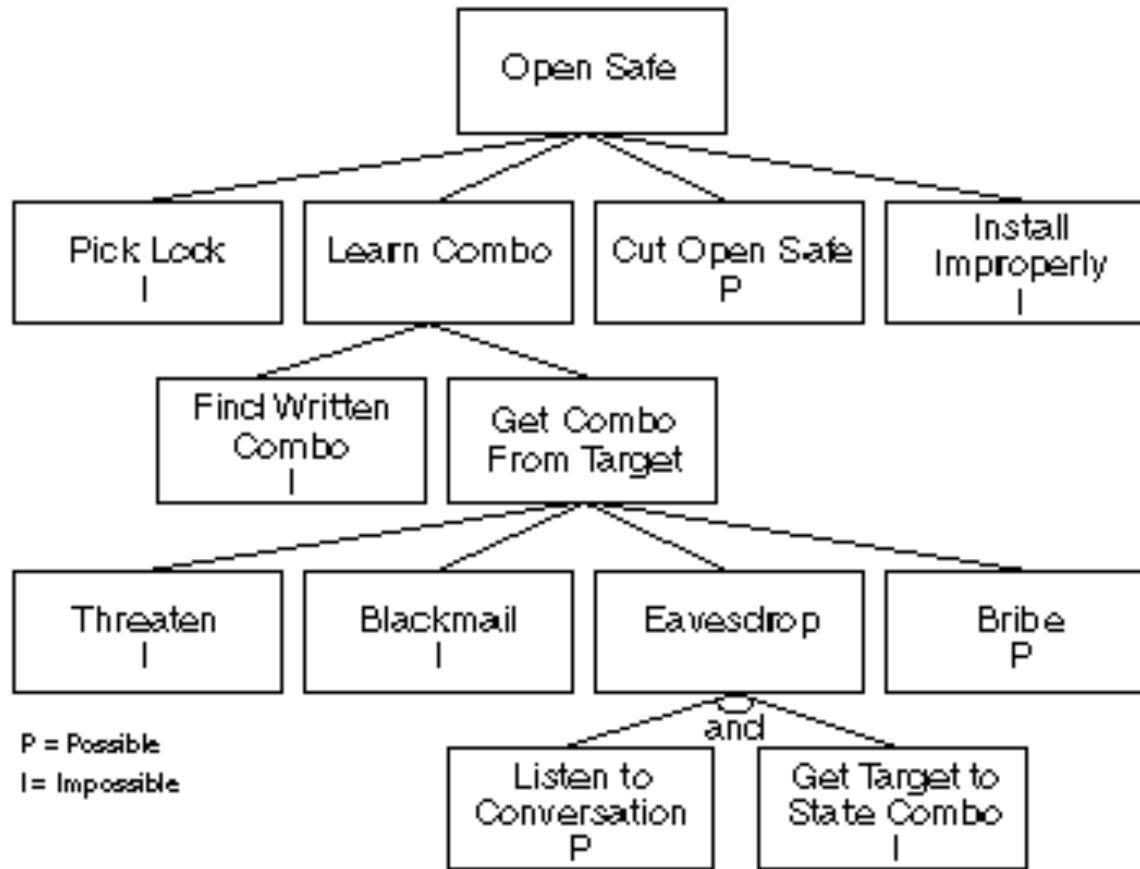
- Useful for thinking about vulnerabilities
- Demonstrates that nothing is “totally secure”
 - Secure for how long?
 - Secure from whom?
- Practice adversarial thinking

*Material borrowed from: https://www.schneier.com/academic/archives/1999/12/attack_trees.html



University of Colorado **Boulder**

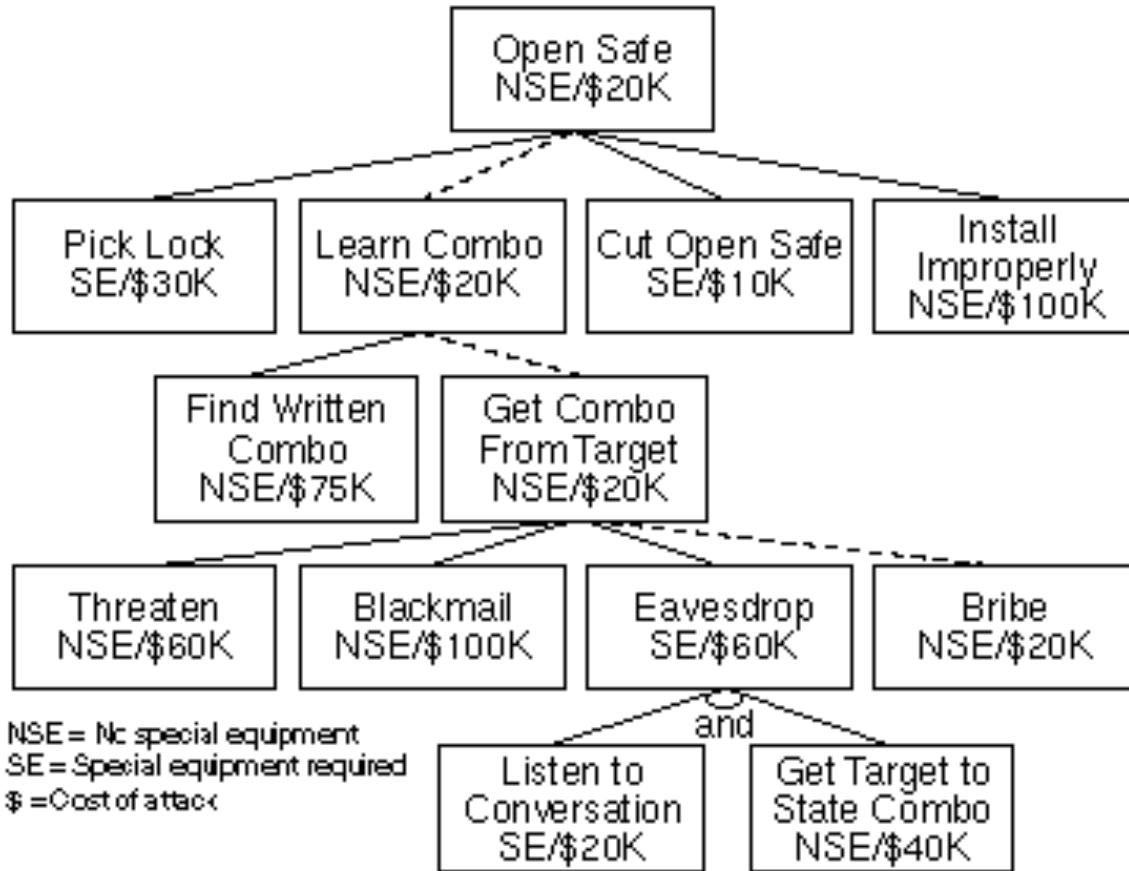
Sample Attack Tree



And vs. Or Nodes

- Do both things need to happen?
- Or just one?
- Helps us understand how difficult attack will be

Pricing Attack Tree



Try Another Example

- Card skimmers
- What's the root node?

Attack Trees Summary

- We want to understand the adversary
- Map out all possible attacks
- See where we're vulnerable

Conclusion

- Defense takes creativity
- We have some common principles to help
- Attack trees help us find where to apply principles

