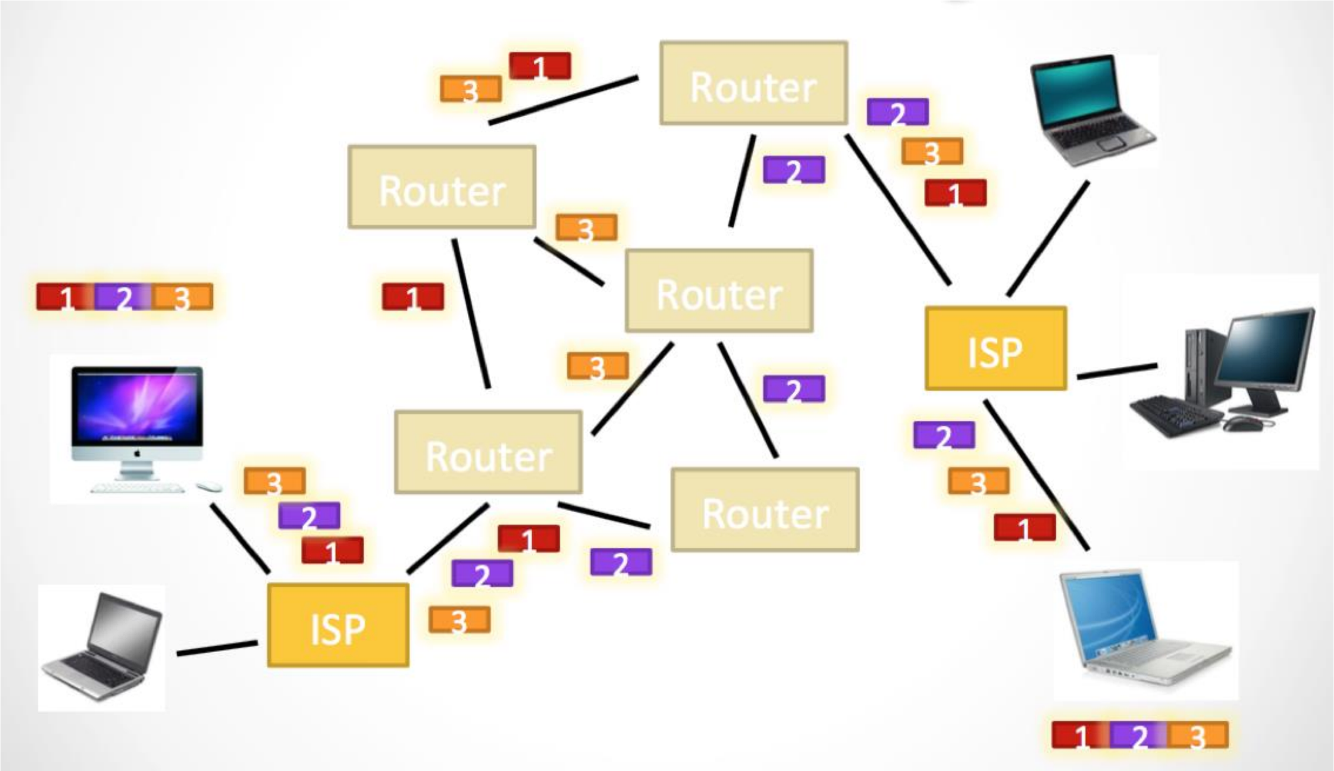
Internet Security: Authentication and Encryption

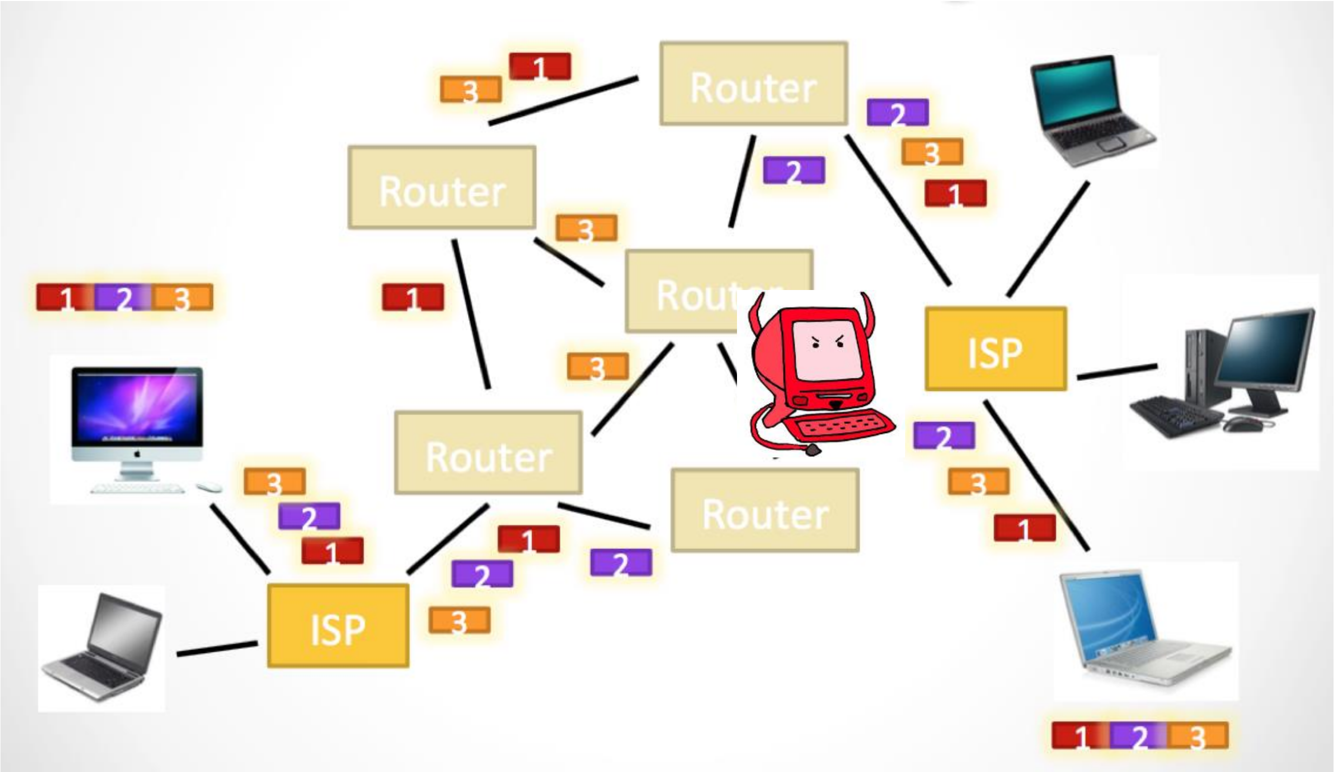
Kelly Rivers and Stephanie Rosenthal

15-110 Fall 2019

The Internet: A Utopian Vision



The Internet: Reality



Main Questions

- First – **who are the bad actors or adversaries?**
  - What do they want?
  - What are their resources?
- Second – **what are our security needs?**
  - Do we need a secure approach as a sender or receiver?
  - Do we care about data privacy?
  - Do we care about data not changing?
- Third – **how do we stop bad actors?**
  - Use authentication to verify identity.
  - Use encryption to protect data.

Who are the adversaries?

- People with varying objectives
  - To damage a person or group, to get money, to find confidential information
- People with varying resources
  - May have a great deal of money to buy servers and equipment, or may have very little money but a great deal of time to do social engineering
- People with varying experience
  - National intelligence agencies are small in number, with great levels of knowledge and ability. ‘Script kiddies’ may use scripts they don’t understand, but have power in numbers

What do adversaries do?

- Some adversaries perpetuate criminal attacks
  - Scamming other users, destroying systems, identity theft, credit card fraud
- Other adversaries engage in privacy violations
  - Stealing a person’s data (SSN, nude pictures) to target them specifically
- Some adversaries want to take down systems
  - spamming a server with so much information that it gets overwhelmed
- Others want to gather large-scale data to use for profit
  - Usernames and passwords, traffic analysis to understand which servers communicate most frequently, requesting excess information from browsers to sell user databases to advertisers later

How do they do those things?

Why is the internet vulnerable?

The internet has three characteristics that make attacks more common and give attackers protection:

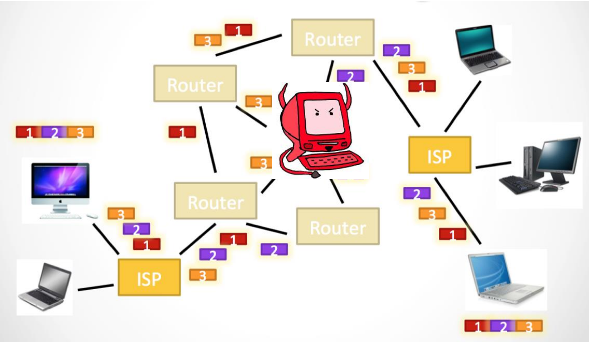
- **Automation** – you can write a program to repeat an action indefinitely
- **Action at a distance** – you do not need to be physically present to start a security attack
- **Technique propagation** – it’s easy to distribute security vulnerability code to other adversaries

Common Attacks: DDOS

DDOS: A Distributed Denial of Service Attack

- When the internet is working correctly, websites know how much traffic to expect at any given point, and can run multiple servers to handle the load.
- In a **DDOS attack**, adversaries intentionally send continuous requests to servers in order to overwhelm them. This makes it impossible for ordinary users to get responses, which makes it look like the site is down.
- DDOSing can happen with no malicious intent when a website is not prepared for sudden load

Common Attacks: Man-in-the-Middle



- Not all routers act like good routers! An adversary can set up a router that pretends to be someone else, in order to intercept packets on their way to the destination
- It can then use common protocols to put the packets back together and read their contents, and maybe even change them
- This is especially easy to do on public/unencrypted wifi (don't go on public wifi to check bank accounts, or potentially even type your password)
- It can also be done by organizations to check your internet activity or disallow particular websites (be careful in internships!)

Common Attacks: Malware

- Malware is software that written with the intent of damaging other people’s computers
  - May infect computer with viruses or delete data
  - May hold data hostage unless the user pays the adversary
  - May be installed silently to spy on the user
- Malware usually gets to a computer via the internet, as it must come from outside the machine
  - Can also come from USB sticks and other shared memory (for example, Stuxnet)
  - Often disguised as innocent files, like pdfs
  - Relies on security flaws in the operating system of the computer

Why are these attacks a problem? What do we want?

- Often, we want our data to remain private
  - For personal reasons, or because of cultural norms and biases
  - Especially true of current social hot topics: LGBTQ issues, health issues, economic beliefs
  - Also for practical reasons- we don’t want people to get our bank account information or credit card numbers!
- Often, we want to make sure data remains secure
  - When we submit official grade reports, we don’t want the grades to change mid-submission
  - We also don’t want other people to change our computers without our knowledge!
- Often, we want to authenticate that we are who we say we are
  - We don’t want any random person to be able to send emails from our own email addresses
  - We don’t want any random website to pretend that it’s google.com

Security: Defense against an Adversary

- When talking about the internet abstractly, we assume everyone acts for the common good
- In reality, there may be some **adversaries** who are trying to steal data or intercept communication for their own benefit.
- The field of **internet security** assumes that there is a specific adversary we need to secure our data from, and generates techniques to do just that.

How do we achieve security?

- To achieve security and privacy: cryptography and encryption
  - Analogy: put a digital lock on data before sending it out. Only people with the right key can then open it.
  - An adversary who doesn't have the key can't read or change the data. Privacy and security are achieved!
- To achieve authentication: passwords and certificates
  - I am who I say I am, because I know the secret code
  - You are who you say you are, because I trust that piece of paper you have
- How do we implement this?
  - There are many different methods encryption methods for sharing messages

Authentication

- We often need to prove that we are who we say we are
  - To send email, access bank accounts, submit assignments...
- But we also want other people to prove that they are who they say they are!
  - We want the REAL person we’re emailing, or our ACTUAL bank
- Authentication uses known information to prove that people are who they say they are

User Identification

- Common approach: have someone provide a **username** and **password** combination, where only the user knows the password
- Servers store passwords in an **encrypted format (hopefully)**, and only check password equality after encrypting
  - This keeps passwords safe from third parties
- Passwords can be:
  - Short or long pieces of text
  - Biometric data (fingerprint, voice recognition, handwriting)
  - A physical token

Password Stengths and Weaknesses

- Adversaries can technically guess passwords with brute-force attacks, but that's difficult in practice
- More common approaches:
  - Dictionary attacks – users often use real words as passwords. These are much easier to guess!
  - Fake login screens – pretending to be the actual service so that users enter the plaintext of the password
  - Social engineering – get the user to provide the password over email or the phone by pretending that it's needed
    - This is how much hacking actually takes place these days

Certificate Authorities

- *Certificate Authorities* sign digital certificates indicating the authenticity of a sender (typically businesses)
    - They authenticate the senders themselves by communicating with them in the real world!
- Senders provide copies of their certificates along with their message or software.
- But can we trust the certificate authorities?
    - Sometimes. Browsers often come with installed lists of trusted authorities.

Encryption

- We also encrypt (encode) our data so others can't understand it (easily) except for the person who is supposed to receive it.
- We call the data to encode plaintext and the encoded data the ciphertext.
- Encoding and decoding are *inverse functions* of each other.
- Basic assumption: the encryption/decryption *algorithm* is known; only the key is secret
  - The key is the password that helps someone decrypt a message
  - As long as the key is strong, it will be near-impossible for others to guess it

Common Encryption Encodings

Caesar Cipher

Key idea – shift the letters in the alphabet by a certain amount to encrypt the message. Shift it the same number of letters back in the other direction to decrypt.

Example: "Hi, my name is Stephanie" -> shifted 5 characters (and lowercase)

"mn, rd sfrj nx xyjumfsj"

If your message receiver knows 5, they can decode by shifting by -5 letters

Common Encryption Encodings

Substitution Cipher

Key idea – since there are only a finite (26) number of Caesar ciphers, instead mix up all the letters randomly and substitute the ith letter for the ith index in the substitution

Example: "Hi, my name is Stephanie" -> [qwertyuiopasdfghjklzxcvbnm]

h is the 7th letter (0 index), so use the 7th substitution i

i is the 8th letter, so use the 8th substitution o, …

Complete message: "io, dn fqdt ol lzthiqfot"

Common Encryption Encodings

Substitution Cipher

Key idea – since there are only a finite (26) number of Caesar ciphers, instead mix up all the letters randomly and substitute the ith letter for the ith index in the substitution

There are 26! 4x10^23 combinations of letters, so the likelihood of decoding a message is very low unless you have the key (the substitution list)

Many other encodings

- The most popular approach today is to just multiply the message by really big numbers to get different bit encodings
  - We'll talk more about this approach on Friday.
- If we could multiply numbers really quickly, we could try a lot of different encodings, but in general we cannot so this encoding scheme is pretty safe for now
  - If P = NP, then multiplication would be very fast and encryption would break!

HTTPS

- Security protocol for the Web, to encrypt web traffic
- Purpose:
  - Authentication (prevent “man in the middle” attacks that could alter the messages being sent)
  - Privacy (prevent eavesdropping)