# DS 593: Privacy in Practice

Systems for Privacy Cont’d

Instructor: Alishah Chator

# News?

# Password reuse is rampant: nearly half of observed user logins are compromised

2025-03-17


Radwa Radwan


Sabina Zejnilovic

5 min read

https://blog.cloudflare.com/password-reuse-rampant-half-user-logins-compromised/

# Last time

- How do we use these building blocks in real systems?

# Today

- How do we use these building blocks for privacy online?

# WiFi Security

- Wireless communications need encryption to have a meaningful notion of access
  - Goal: need to be on the network to see other traffic on the network
- No real protections against other people on the network or whoever is running the network
  - At least without using other tools

# Public Key Infrastructure (PKI)

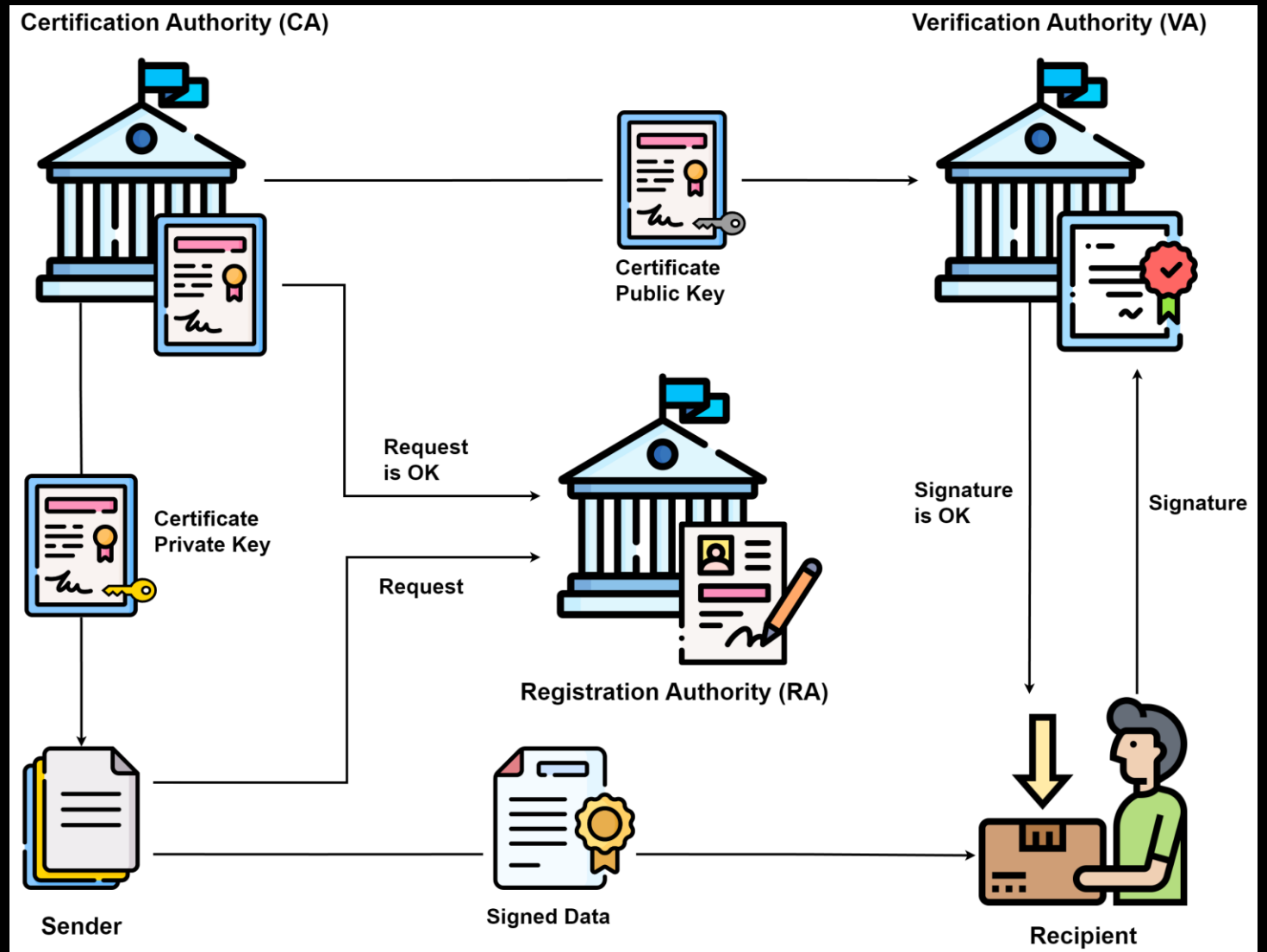
- A core challenge of cryptography is sharing keys
- We learned how public key cryptography helps by using both a sharable public key and a secret private key
- Problem: How do we associate a real world identity with a public key?

# Public Key Infrastructure (PKI)

- A PKI is a system that handles the credential management necessary for robust public key cryptography

- It provides a mapping of keys to identities by leveraging some trusted authority
  - Finding a suitable trusted authority is the key challenge

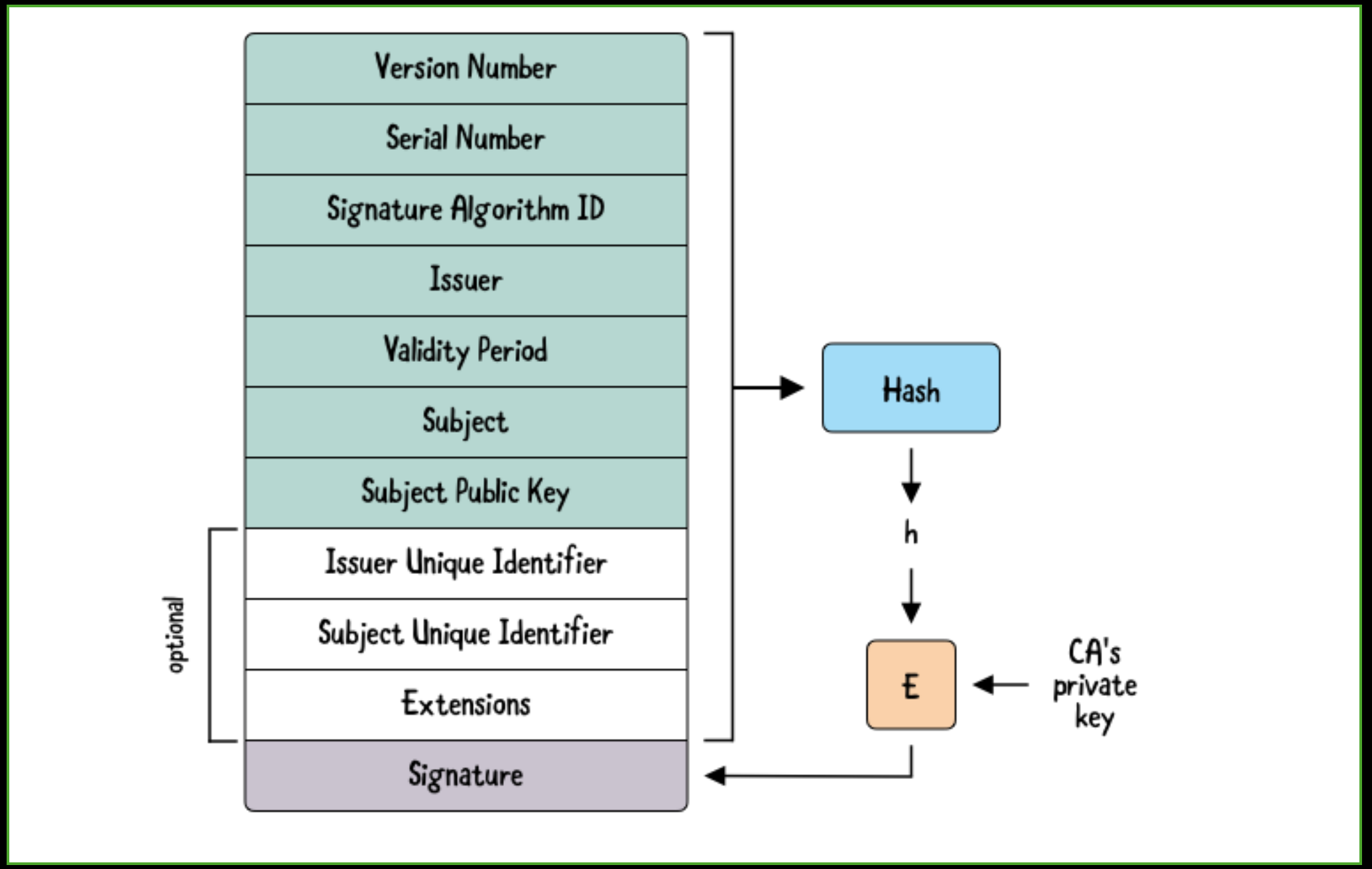
# Public Key Infrastructure (PKI)



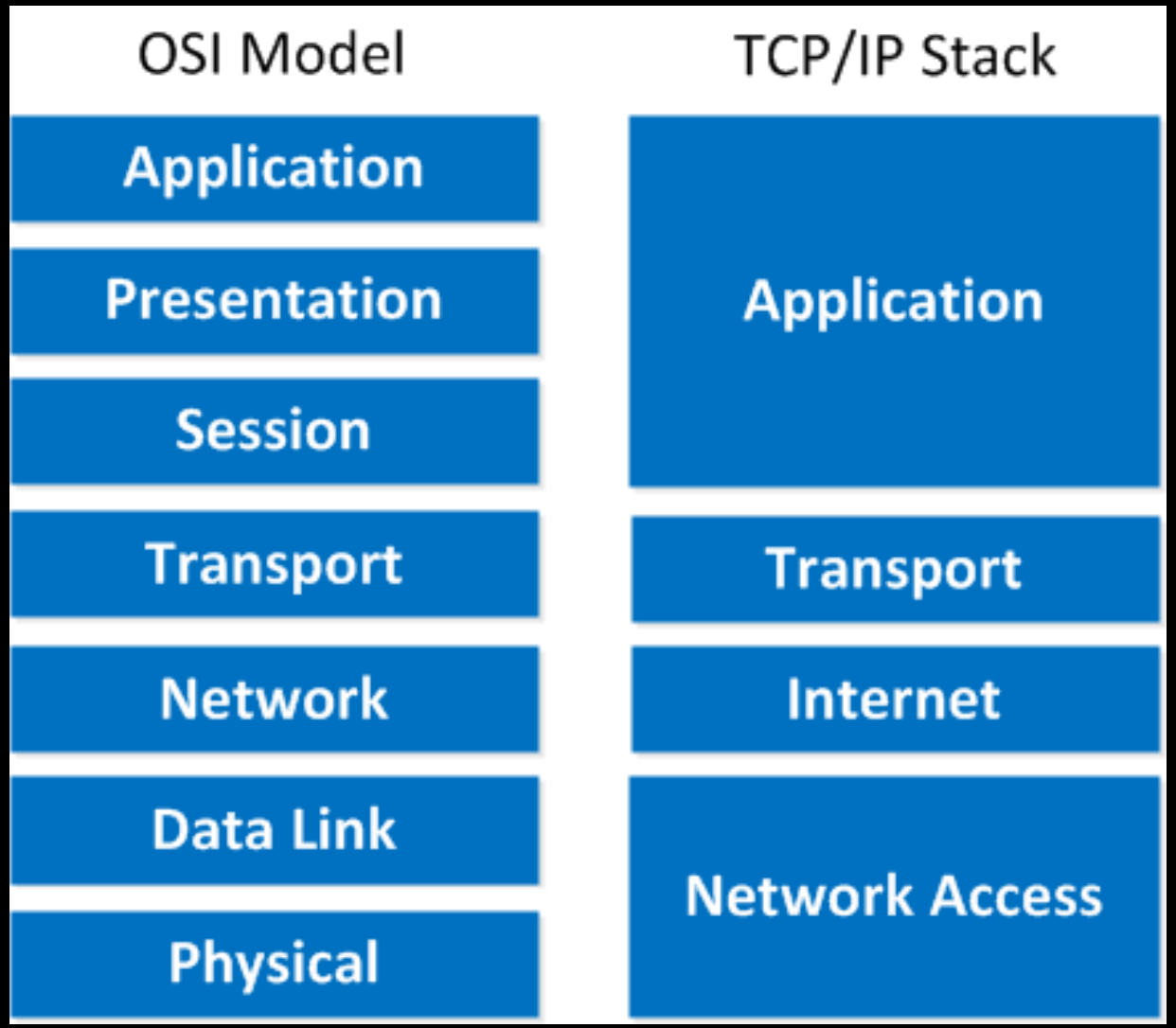
# Certificates

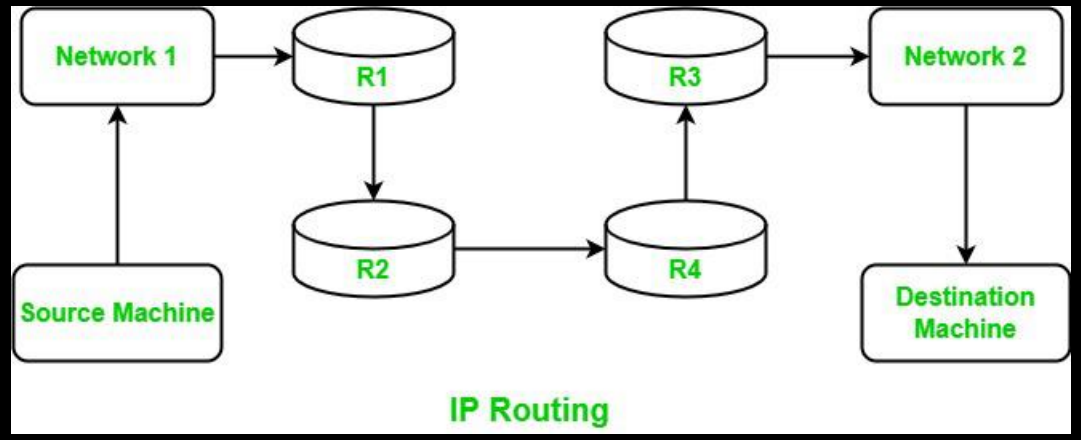
- A certificate is the main output of a PKI

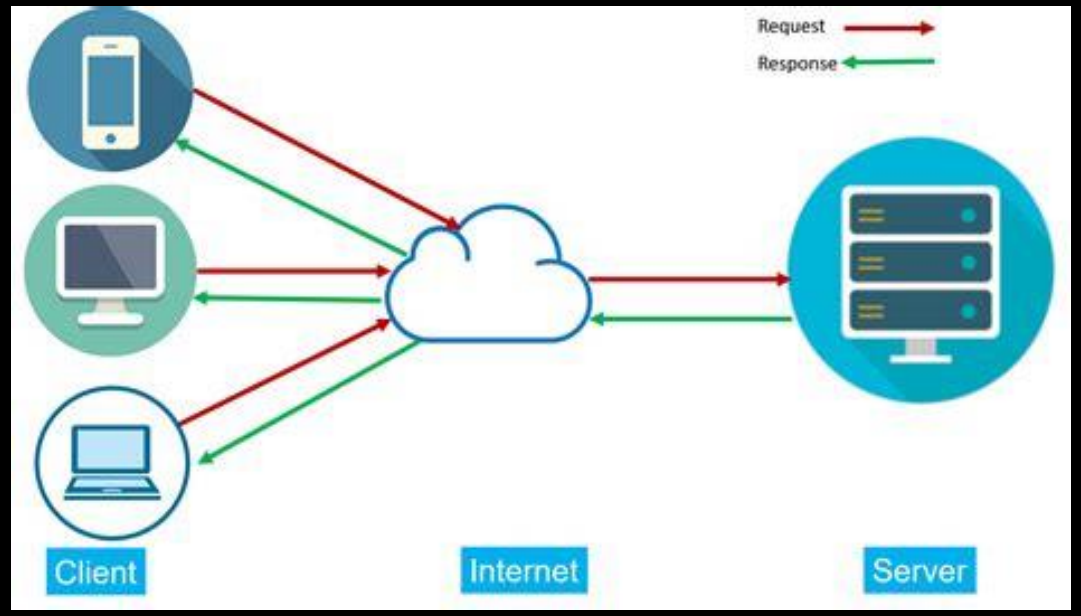
- No fixed format but generally it consists of:
  - Your identity
  - Your Public Key
  - A digital signature by a trusted authority


Version Number
Serial Number
Signature Algorithm ID
Issuer
Validity Period
Subject
Subject Public Key
Issuer Unique Identifier
Subject Unique Identifier
Extensions
Signature
optional
Hash
h
E
CA's private key

# Network Security



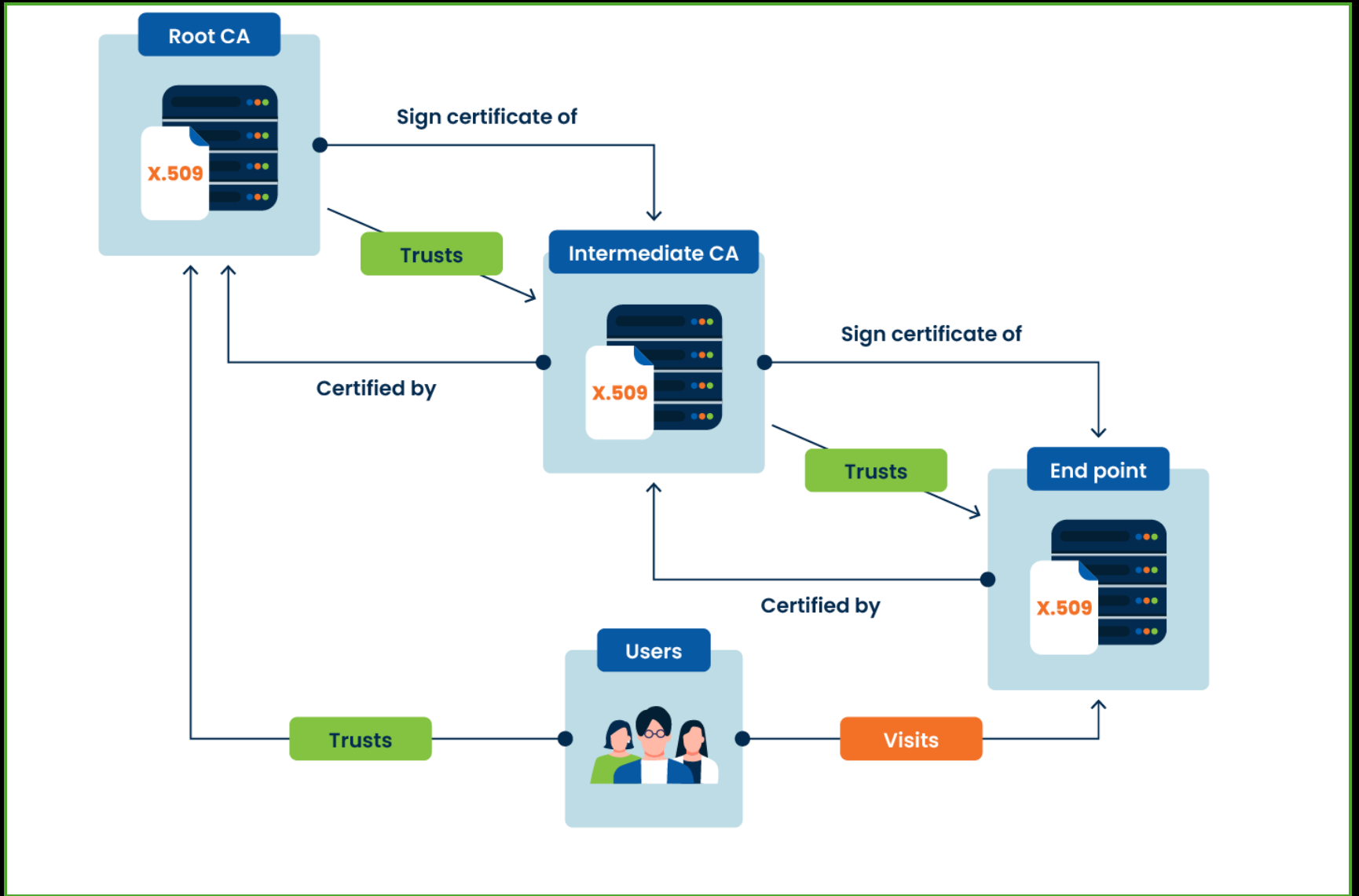
# Transport Layer Security (TLS)

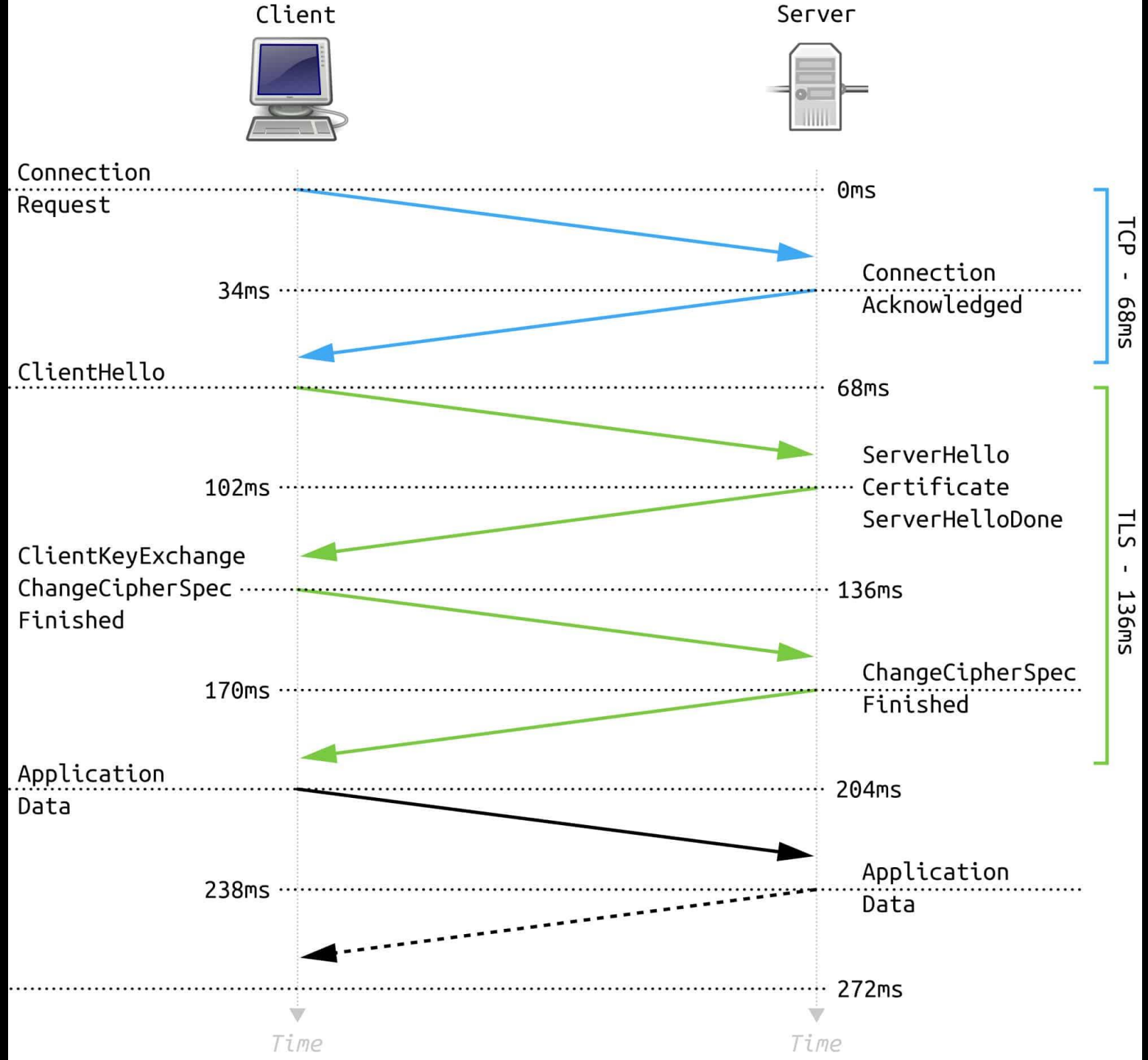
- Replaced the Secure Sockets Layer (SSL) protocol
- Secures HTTP connections – HTTPS
- One of the primary deployments of cryptography
- Ensures information is only shared between client and server

# Chain of Trust

- TLS leverages a PKI to provide robust authenticity to its certificates
- Certain trusted real world entities are designated as Root Certificate Authorities (CAs)
  - I.E. DigiCert, Verisign, Lets Encrypt
- These Root CAs sign the certificates of intermediate CAs who then sign the certificates of servers or end-users

# Chain of Trust




Client
Server
Connection
Request
0ms
Connection
Acknowledged
34ms
ClientHello
68ms
ServerHello
Certificate
ServerHelloDone
102ms
ClientKeyExchange
ChangeCipherSpec
Finished
136ms
ChangeCipherSpec
Finished
170ms
Application
Data
204ms
Application
Data
238ms
272ms
Time
Time
TCP - 68ms
TLS - 136ms

# HOW TLS WORKS?

Client

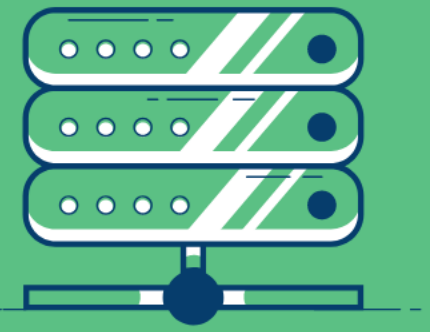
## 1. Handshake protocol

- Negotiate TLS protocol version
- Select cryptographic algorithms: cipher suites
- Authenticate by asymmetric cryptography
- Establish a secret key for symmetric encryption

## 2. Record protocol

- Encrypt outgoing messages with the secret key
- Transmit the encrypted messages
- Decrypt incoming messages with the secret key
- Verify that the messages are not modified

Symmetric bulk encryption



Server

https://medium.com/@techschool.guru/a-complete-overview-of-ssl-tls-and-its-cryptography-system-31336a85c278

# TLS Security Considerations

- You have to know about the Root CAs somehow
  - Certificate Pinning
- Only protects application layer data
  - Doesn’t hide IP address, URLs, or other metadata

# Other Network Security Protocols

- Many other protocols that protect other aspects of the network
  - IPSec
  - DNSSec
  - DoH

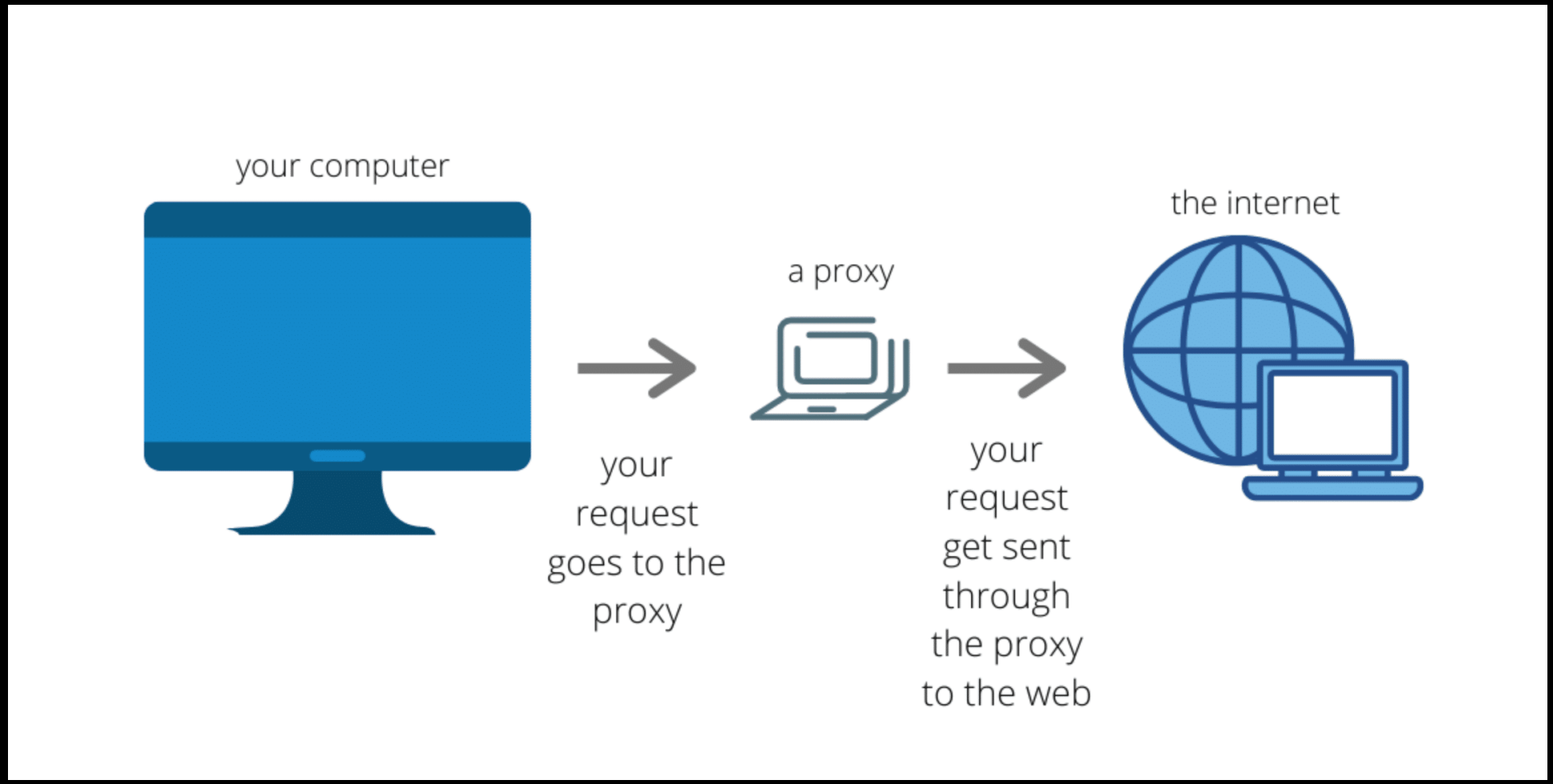
# Virtual Private Networks (VPN)

- The internet is large public system with many intermediaries
  - Can we pretend it is actually just a small closed network of only the computers I care about or trust?

- Problem: how to hide my browsing from intermediaries between me and the ”private network?”
  - Proxying

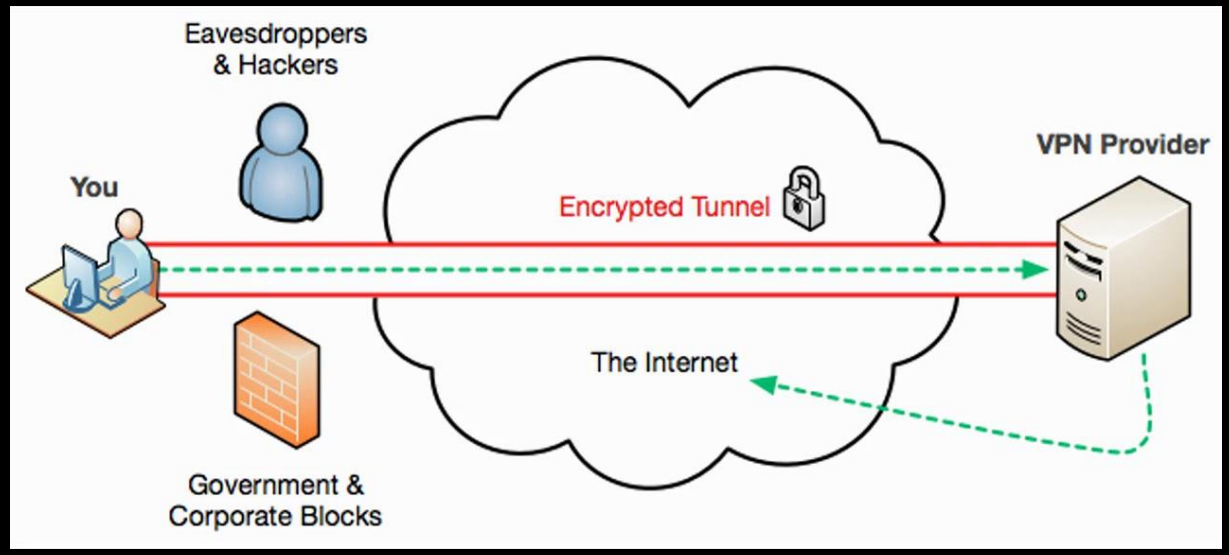
  - Tunneling

# Web Proxy


your computer
your request goes to the proxy
a proxy
your request get sent through the proxy to the web
the internet

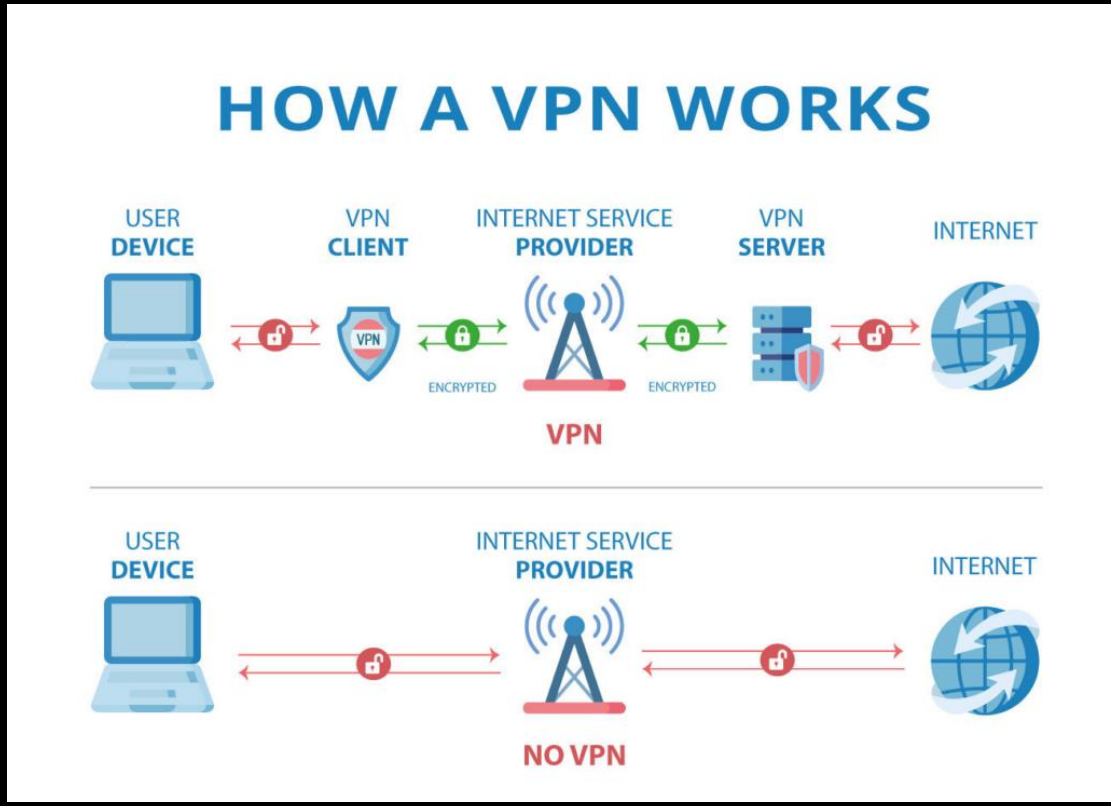
# Tunneling

- A way to build a secure connection to a particular server
- You encapsulate your actual network packet in a packet that is sent to the desired server



- A VPN sets up a secure tunnel to a VPN server which then acts as a proxy to forward your packets to the internet

- Depending on the location of the VPN server, this can allow for bypassing censorship or geolocation restrictions



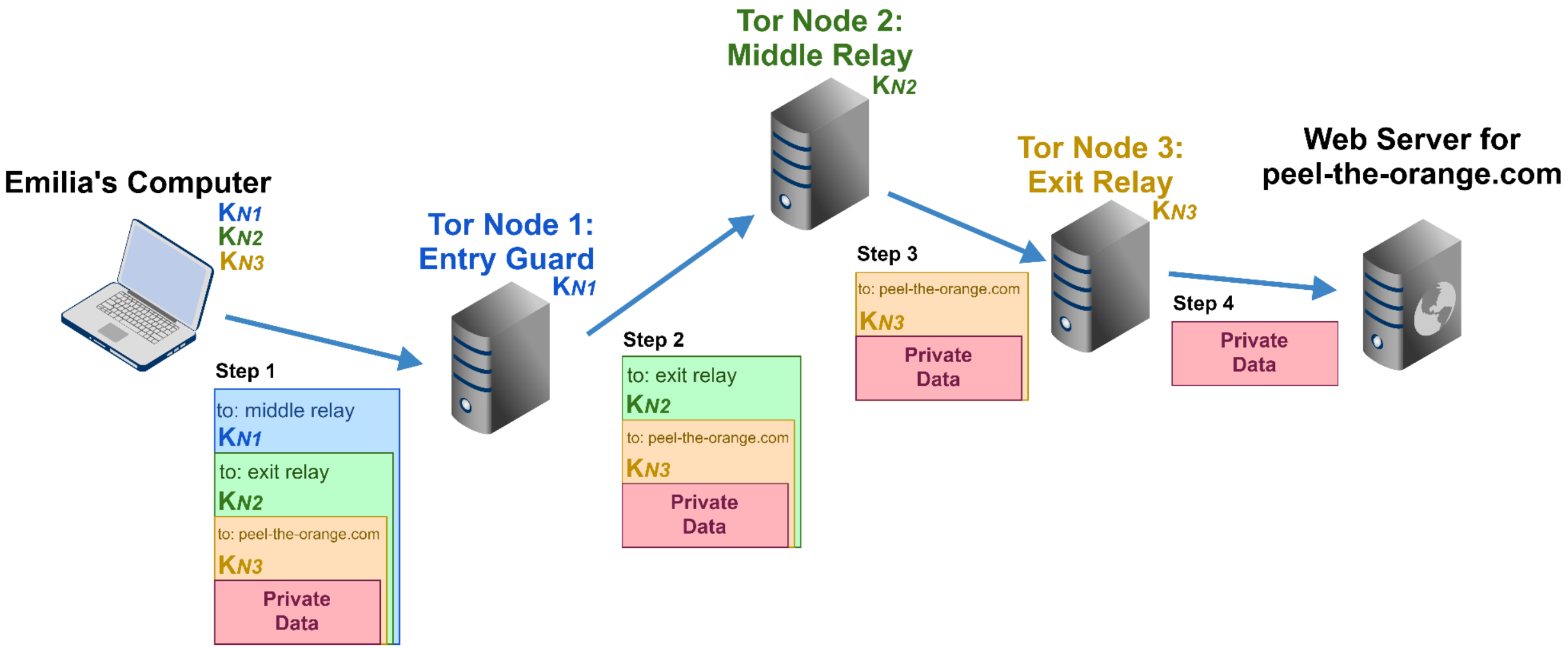
# VPN Considerations

- Primarily hides your data from ISP
- The VPN server needs to be trusted
  - Learns your IP, your browsing data, etc
  - Could log all this
- VPN Servers are often publicly known and easy to identify
- Important to make sure all traffic goes through the VPN

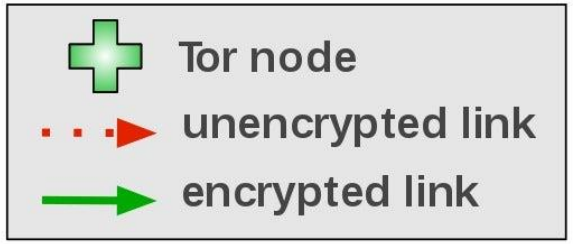
# The onion router (Tor)

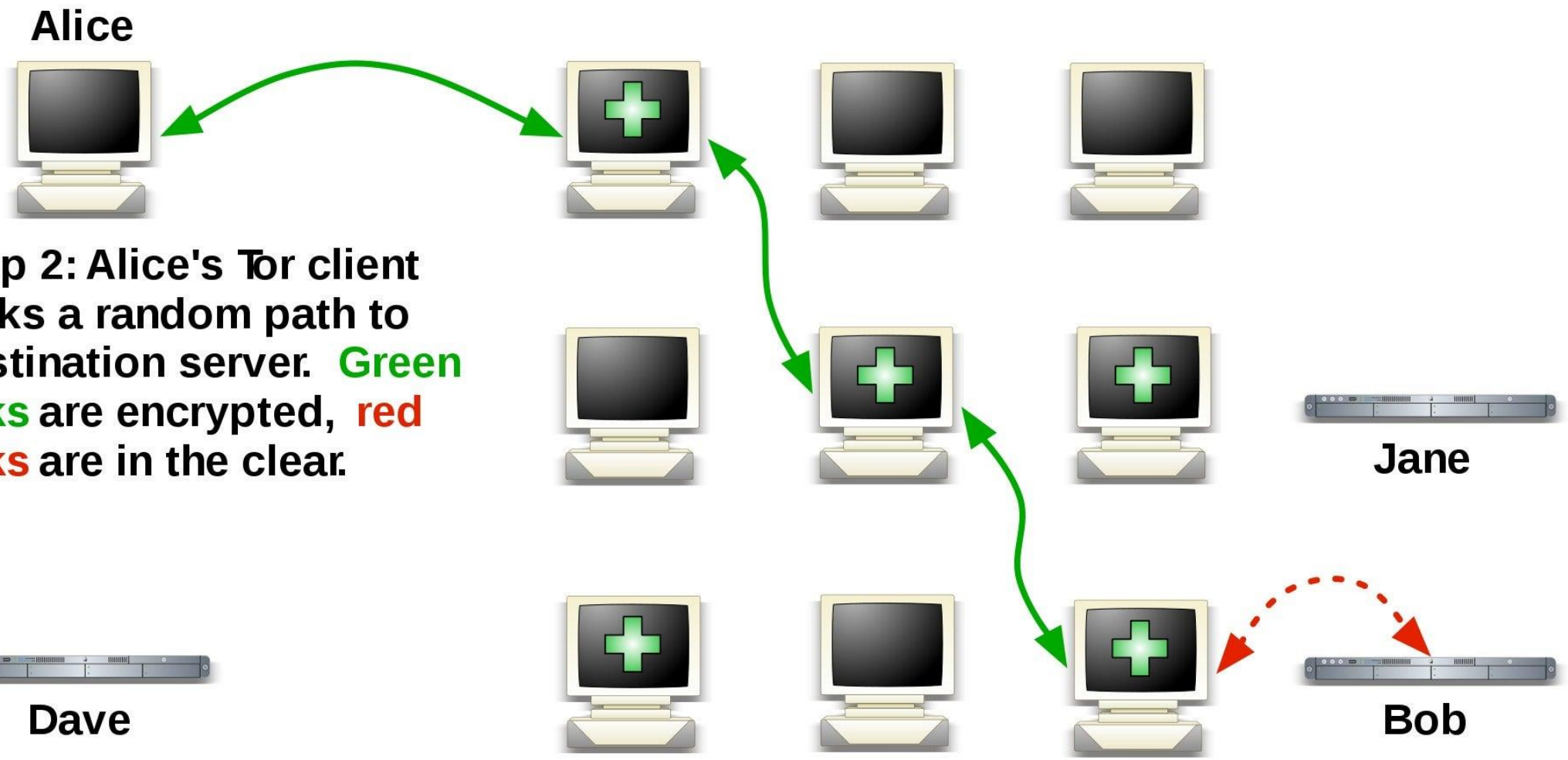
- Goal is to provide a strong guarantee of hiding your browsing data in a robust and decentralized way
  - Mitigating limitations of VPNs

- Tor does this by adding multiple layers of encapsulation to ensure no single entity learns everything about your browsing


Emilia's Computer
KN1
KN2
KN3
Tor Node 1:
Entry Guard
KN1
Tor Node 2:
Middle Relay
KN2
Tor Node 3:
Exit Relay
KN3
Web Server for
peel-the-orange.com
Step 1
to: middle relay
KN1
to: exit relay
KN2
to: peel-the-orange.com
KN3
Private
Data
Step 2
to: exit relay
KN2
to: peel-the-orange.com
KN3
Private
Data
Step 3
to: peel-the-orange.com
KN3
Private
Data
Step 4
Private
Data

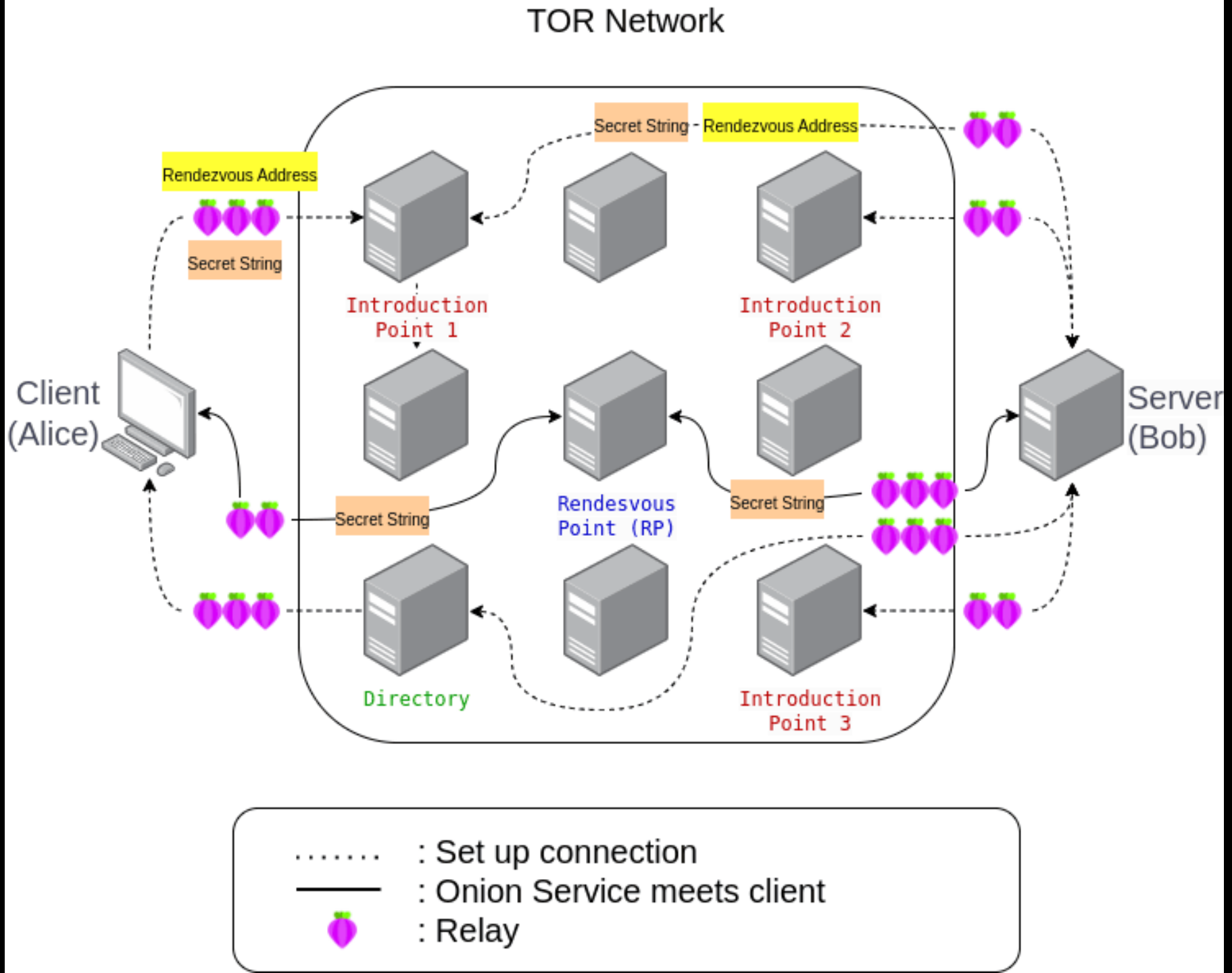
# How Tor Works: 2



# Other aspects of Tor

- Tor alone cannot bypass censorship as the relays can be blocked
  - Tor Bridges are a way to get around this
- Tor additionally has what are called hidden services, which can only be accessed within the Tor network


TOR Network
Secret String
Rendezvous Address
Rendezvous Address
Secret String
Introduction Point 1
Introduction Point 2
Client (Alice)
Server (Bob)
Secret String
Rendesvous Point (RP)
Secret String
Directory
Introduction Point 3
: Set up connection
: Onion Service meets client
: Relay

# Tor considerations

- Possibility of leakage if multiple relays on your circuit are compromised
  - How to avoid this?
- Need to make sure you are using secure application layer protocols as well (ex: HTTPS)
- Crucial to keep your Tor and non-Tor logins separate

# End to End Encryption (E2EE)

- So far we have focused on how to protect information as it passes through a network

- However, the end server may be able to see all of this
  - HTTPS only has data encrypted to and from the server

- E2EE is when the server only stores encrypted user data, *encrypted under a key that the server does not know*

Fig. 1a: Encryption in transit

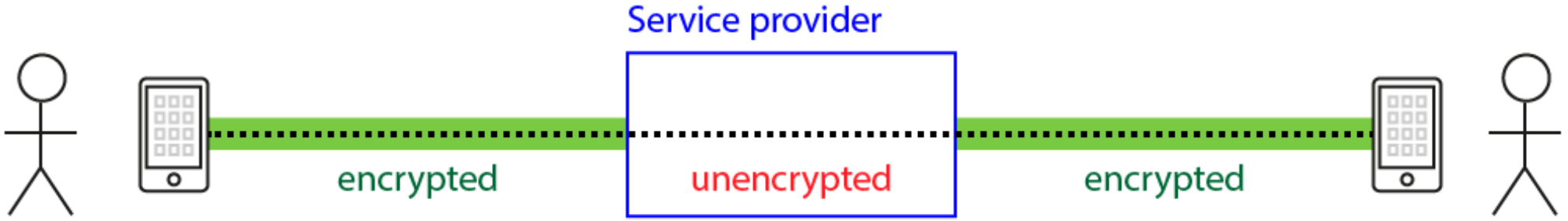



Fig. 1b: End-to-end encryption

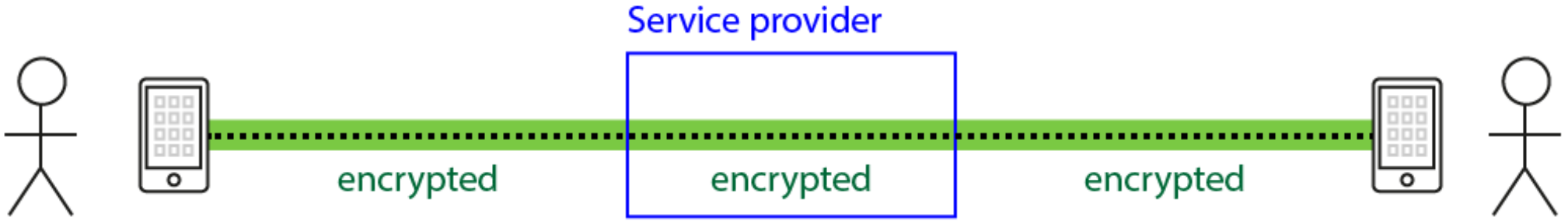



Fig. 1c: End-to-end encryption (no service provider)

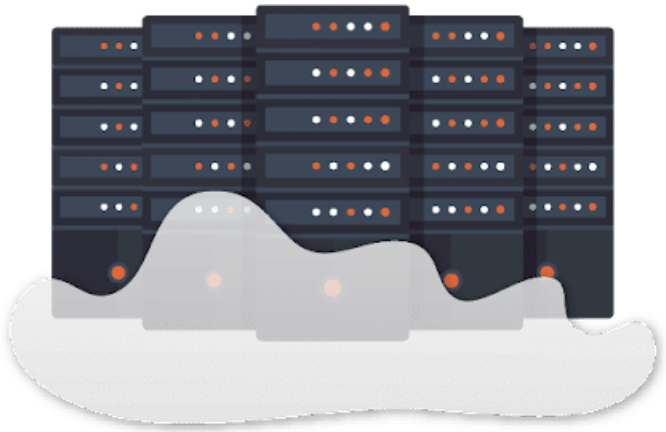
BOB

Send Me $

Bob uses Alice's public key to encrypt

6ekd890optak1

Servers



Encrypted e-mail to Alice

6ekd890optak1

Alice's public key

Bob's public key

ALICE

6ekd890optak1

Alice uses her private key to decrypt the message

Send Me $

# The Signal Protocol

- A protocol for E2EE messaging
  - Also an open-source app
- Messages are only readable by the users in a chat through their device specific keys
  - Lose access to device -> loss access to messages
- Many other platforms such as Whatsapp, Messenger, use an implementation of the protocol as well
- Provides *forward* and *post-compromise* security
  - The Double Ratchet regularly changes the public keys and per-message symmetric keys used

# Next Time

Defining Surveillance