Tarzan:

A Peer-to-Peer Anonymizing Network Layer

Michael J. Freedman, NYU

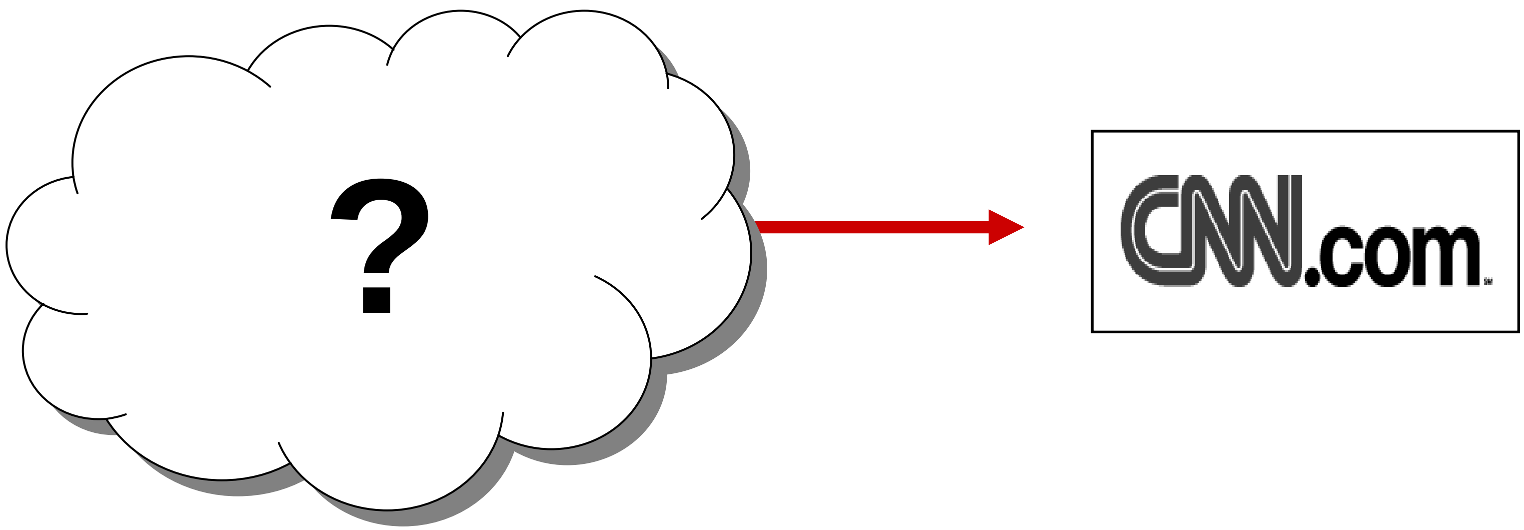
Robert Morris, MIT

ACM CCS 2002

http://pdos.lcs.mit.edu/tarzan/

The Grail of Anonymization

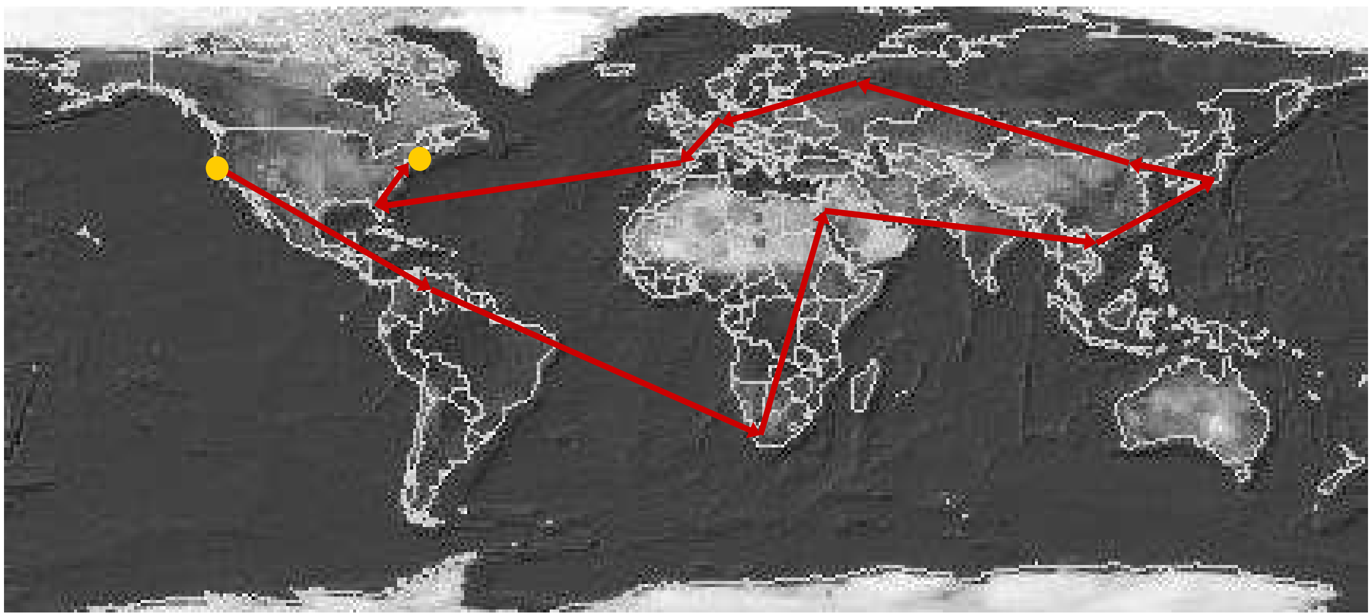
- Participant can communicate anonymously with non-participant



- User can talk to CNN.com
- Nobody knows who user is

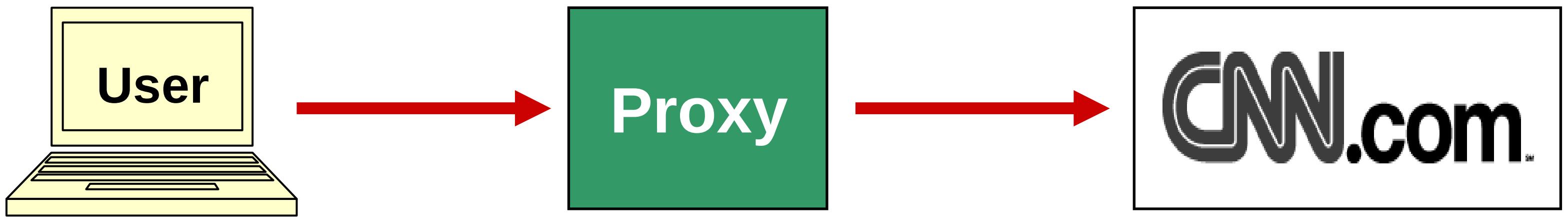
Our Vision for Anonymization

- Thousands of nodes participate
- Bounce traffic off one another



- Mechanism to organize nodes: peer-to-peer
- All applications can use: IP layer

Alternative 1: Proxy Approach



- Intermediate node to proxy traffic
- Completely trust the proxy

Anonymizer.com

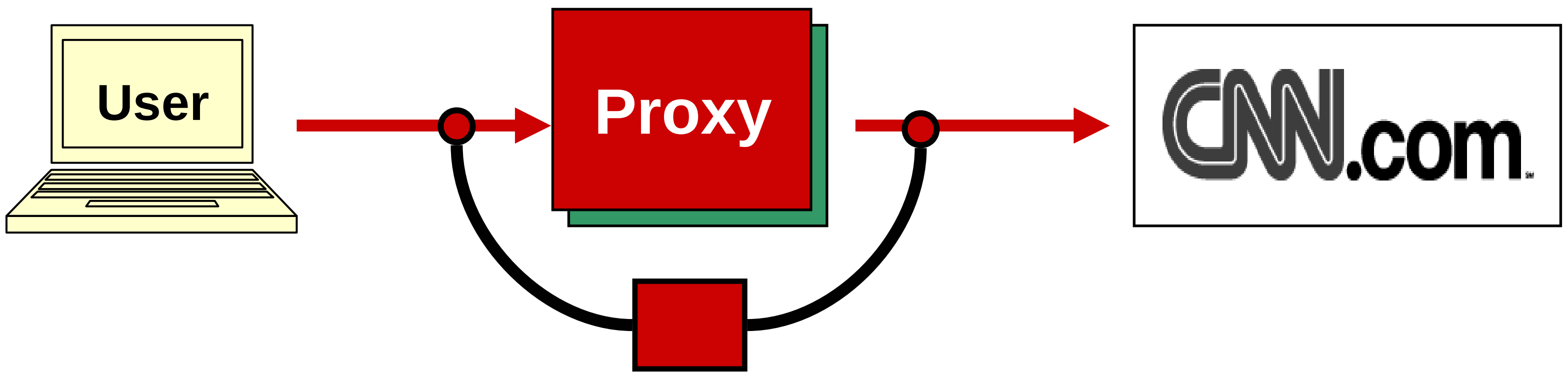
Threat model

- Corrupt proxy(s)
  - Adversary runs proxy(s)
  - Adversary targets proxy(s) and compromises, possibly adaptively

- Network links observed
  - Limited, localized network sniffing
  - Wide-spread (even global) eavesdropping

  e.g., Carnivore, Chinese firewall, ISP search warrants

Failures of Proxy Approach

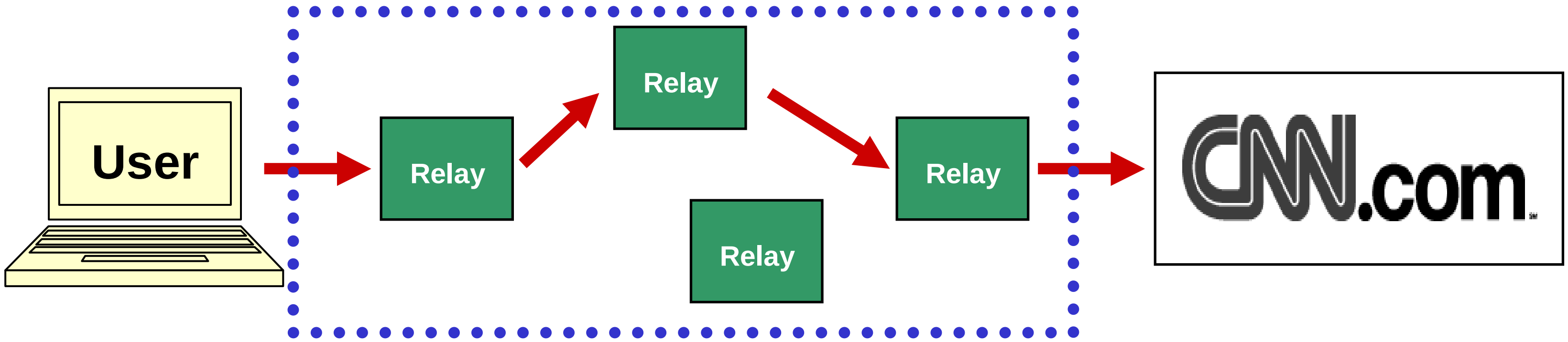


- Proxy reveals identity
- Traffic analysis is easy

Failures of Proxy Approach

- Proxy reveals identity
- Traffic analysis is easy
- CNN blocks connections from proxy
- Adversary blocks access to proxy (DoS)

Alternative 2: Centralized Mixnet

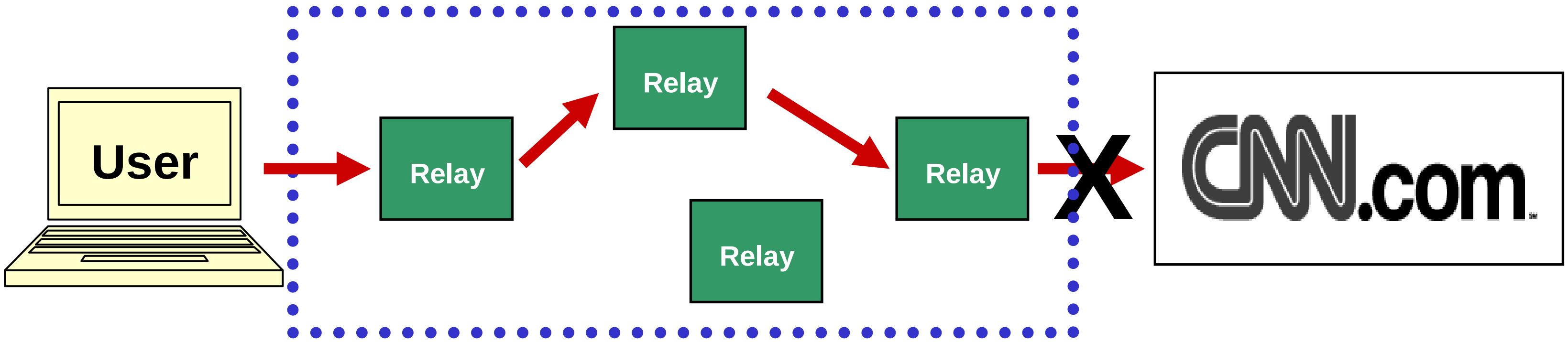


- MIX encoding creates encrypted tunnel of relays
  - Individual malicious relays cannot reveal identity
- Packet forwarding through tunnel

Onion Routing, Freedom

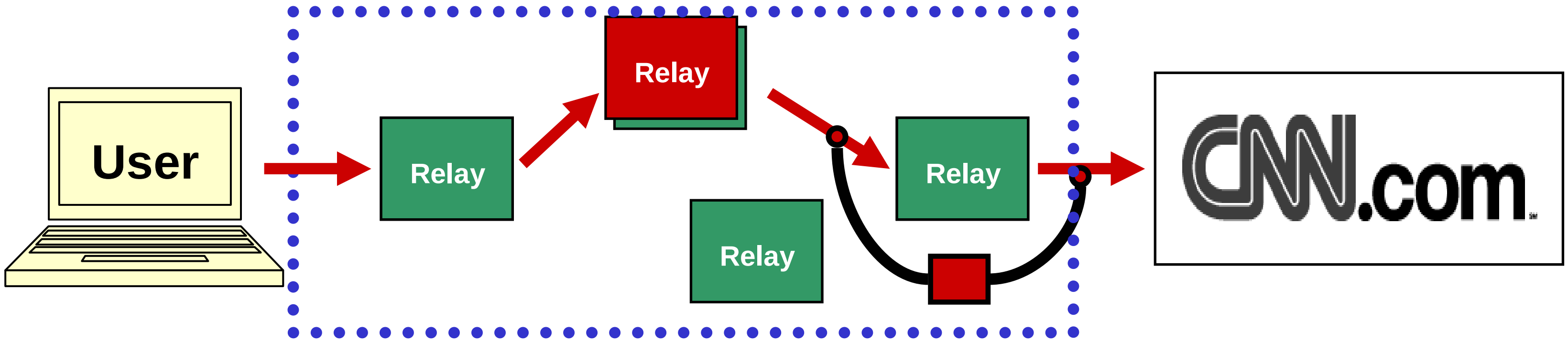
Small-scale, static network

Failures of Centralized Mixnet



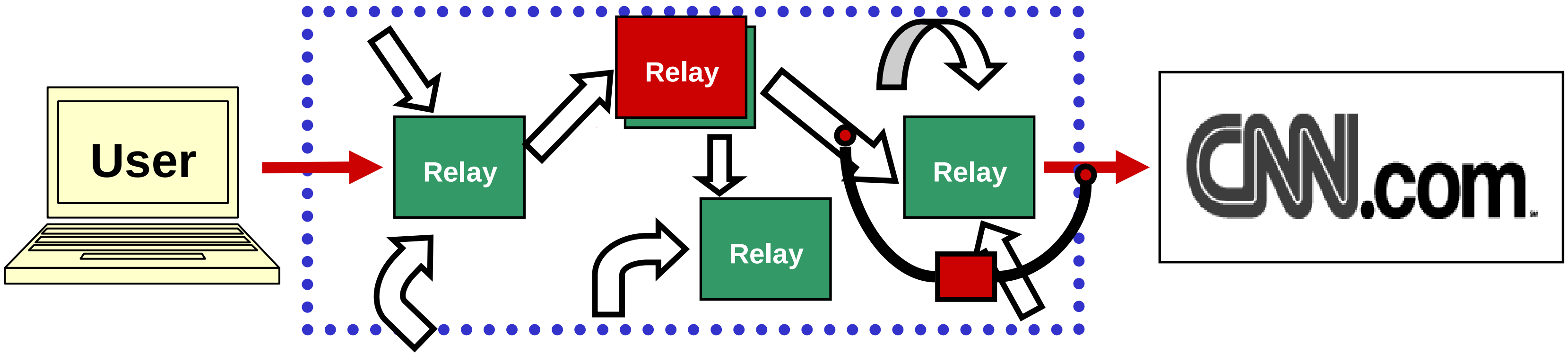
- CNN blocks core routers

Failures of Centralized Mixnet



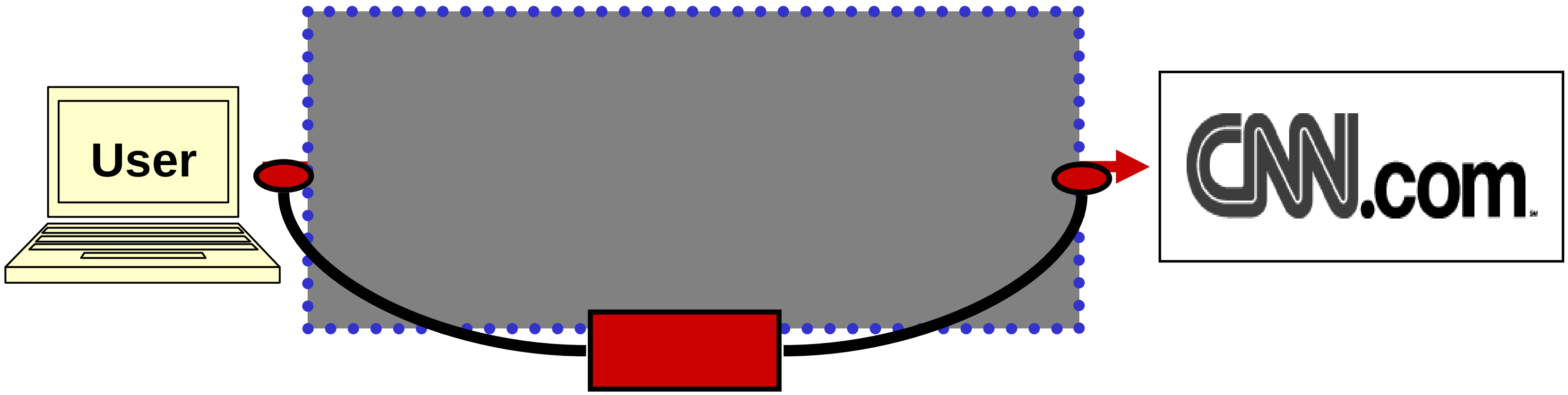
- CNN blocks core routers
- Adversary targets core routers

Alternative 2: Centralized Mixnet



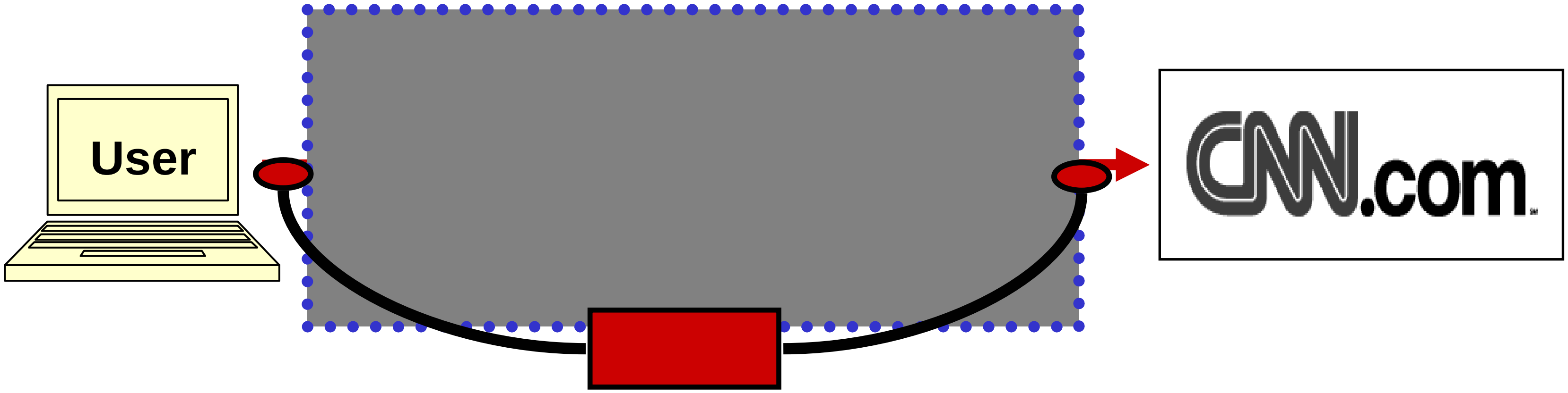
- CNN blocks core routers
- Adversary targets core routers
- So, add cover traffic between relays

Failures of Centralized Mixnet



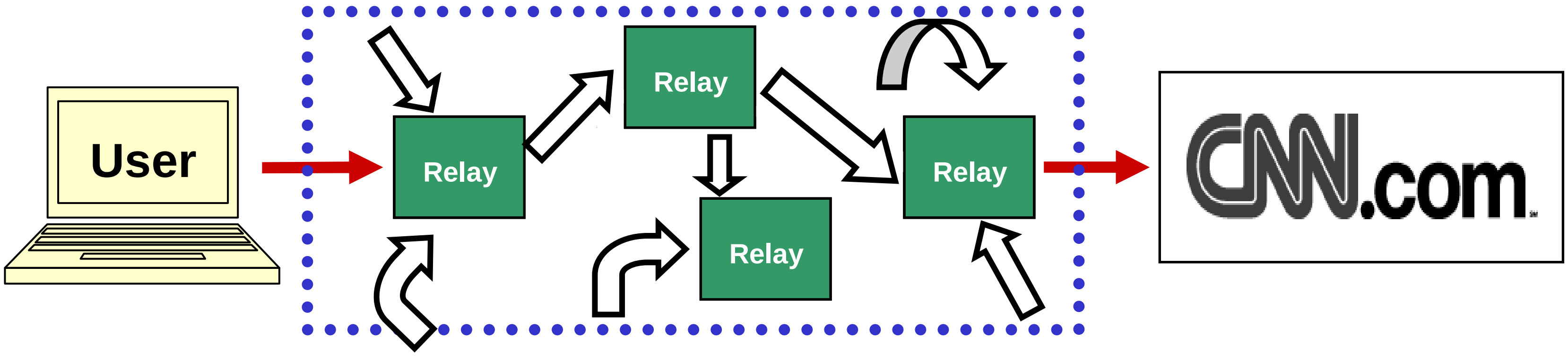
- CNN blocks core routers
- Adversary targets core routers

Failures of Centralized Mixnet



- CNN blocks core routers
- Adversary targets core routers
- Still allows network-edge analysis

Failures of Centralized Mixnet

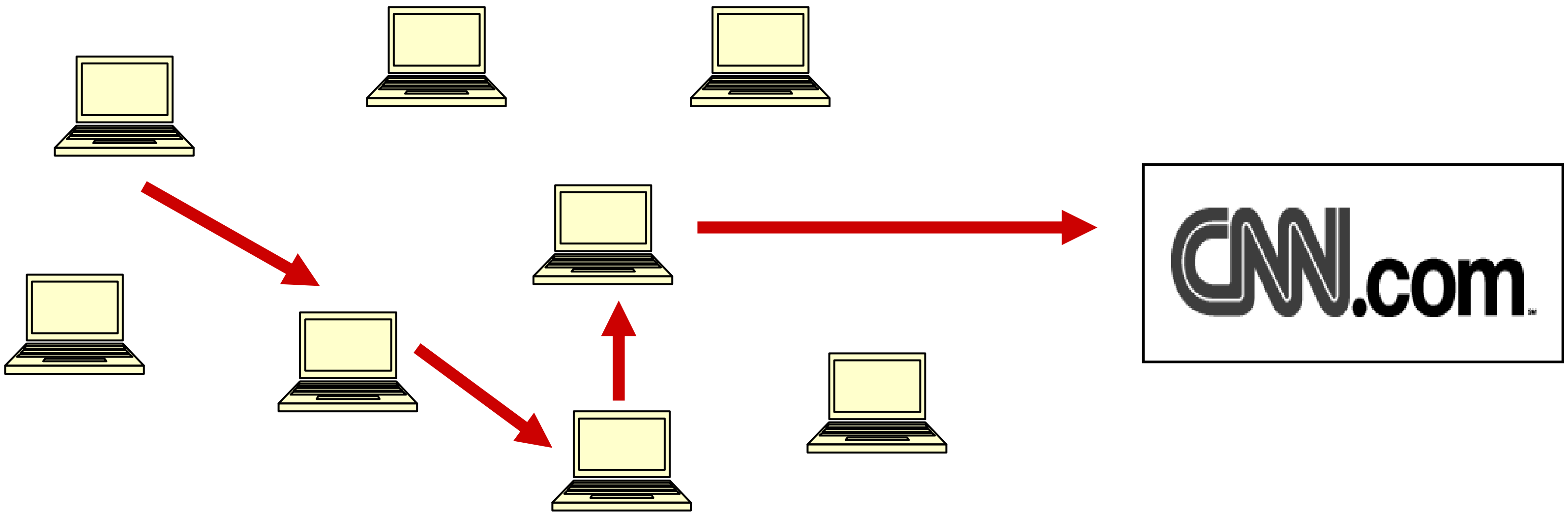


- Internal cover traffic does not protect edges
- External cover traffic prohibitively expensive?
  - n^2 communication complexity

Tarzan goals

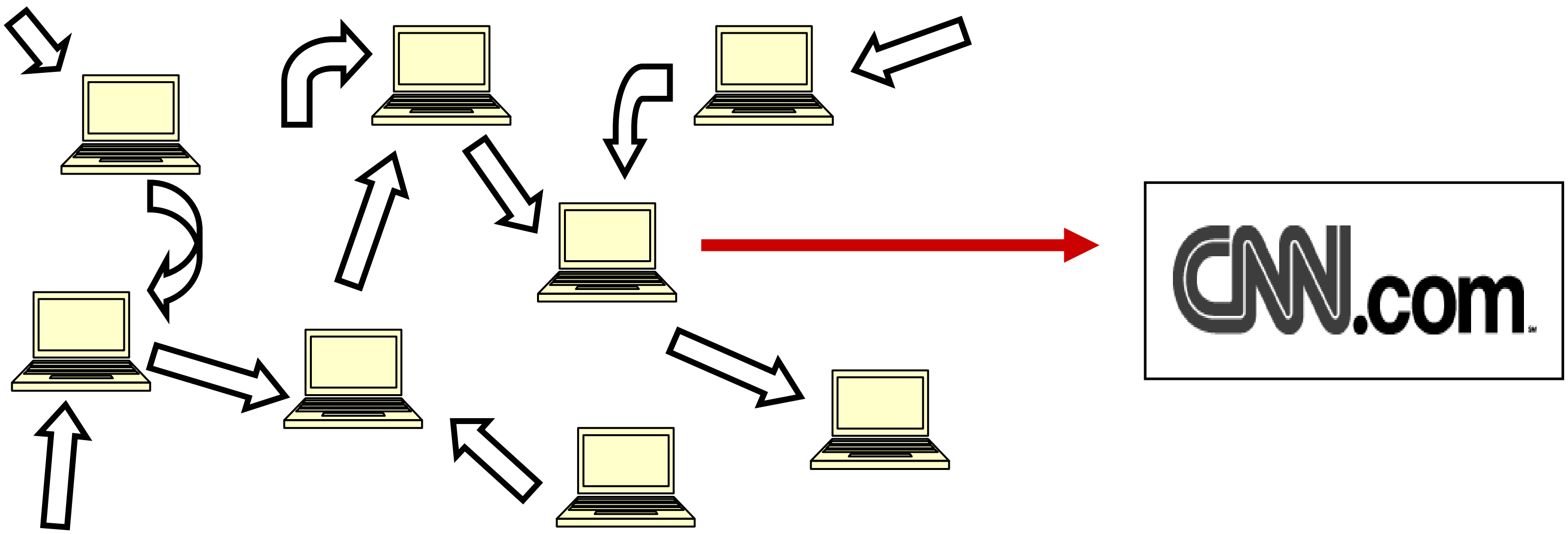
- No distinction between anon proxies and clients
- Anonymity against corrupt relays
- Anonymity against global eavesdropping
- Application-independence

Tarzan: Me Relay, You Relay



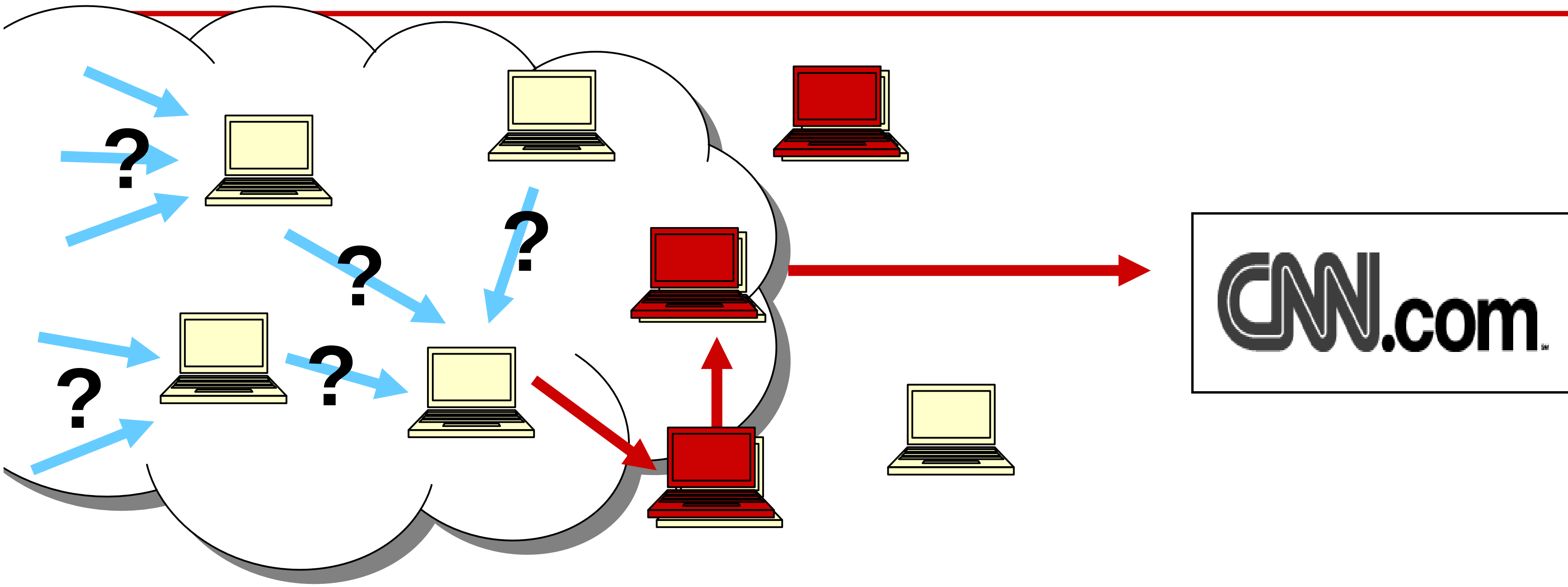
- Thousands of nodes participate
  - CNN cannot block everybody
  - Adversary cannot target everybody

Tarzan: Me Relay, You Relay



- Thousands of nodes participate
- Cover traffic protects all nodes
  - Global eavesdropping gains little info

Benefits of Peer-to-Peer Design

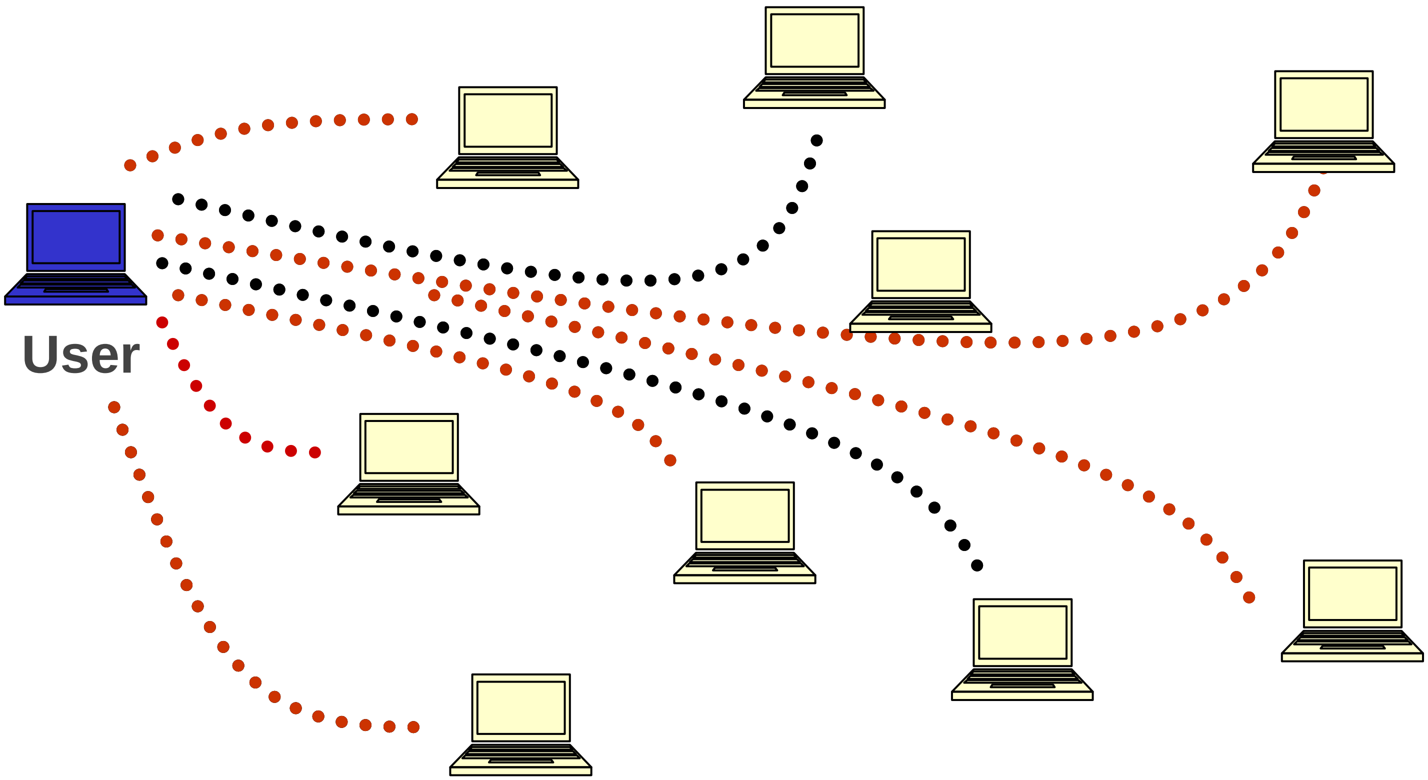


- Thousands of nodes participate
- Cover traffic protects all nodes
- All nodes also act as relays
  - No network edge to analyze
  - First hop does not know he's first

Tarzan goals

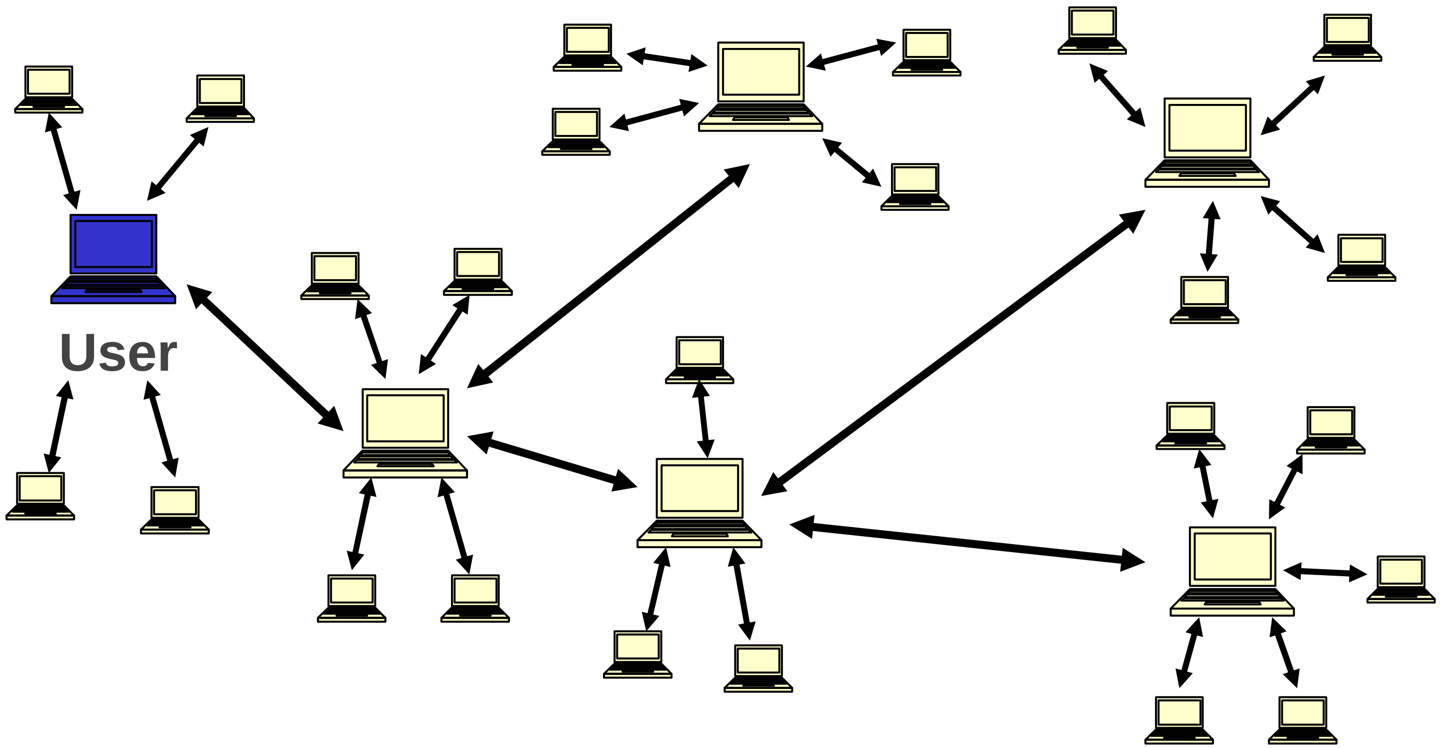
- No distinction between anon proxies and clients
- Anonymity against corrupt relays
- Anonymity against global eavesdropping
- Application-independence

Tarzan: Joining the System



1. Contacts known peers to learn neighbor lists
2. Validates each peer by directly pinging

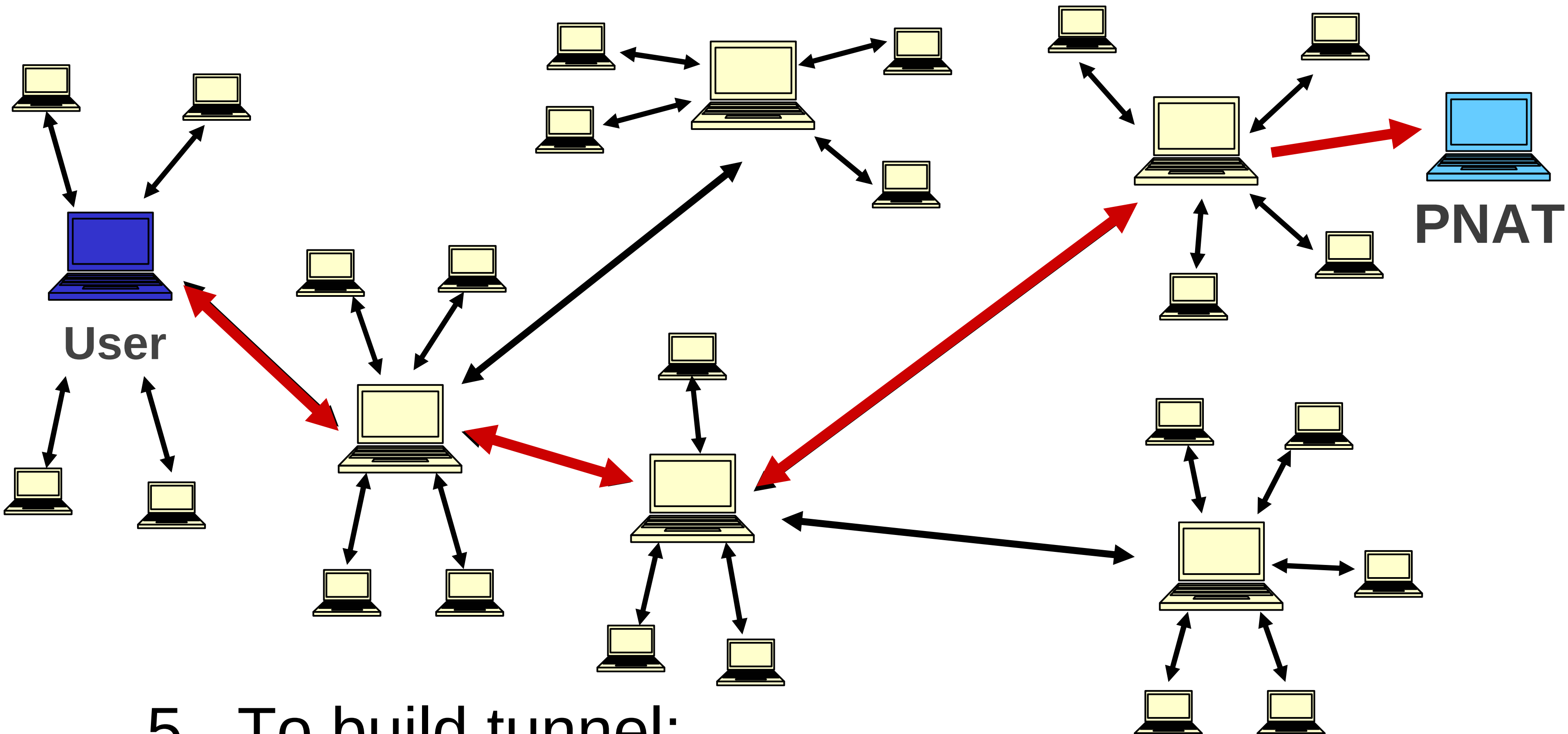
Tarzan: Generating Cover Traffic



4. Nodes begin passing cover traffic with mimics:

- Nodes send at some traffic rate per time period
- Traffic rate independent of actual demand
- All packets are same length and link encrypted

Tarzan: Selecting tunnel nodes



5. To build tunnel:

   Iteratively selects peers and builds tunnel from among last-hop's mimics

But, Adversaries Can Join System

But, Adversaries Can Join System

- Adversary can join more than once by spoofing addresses outside its control

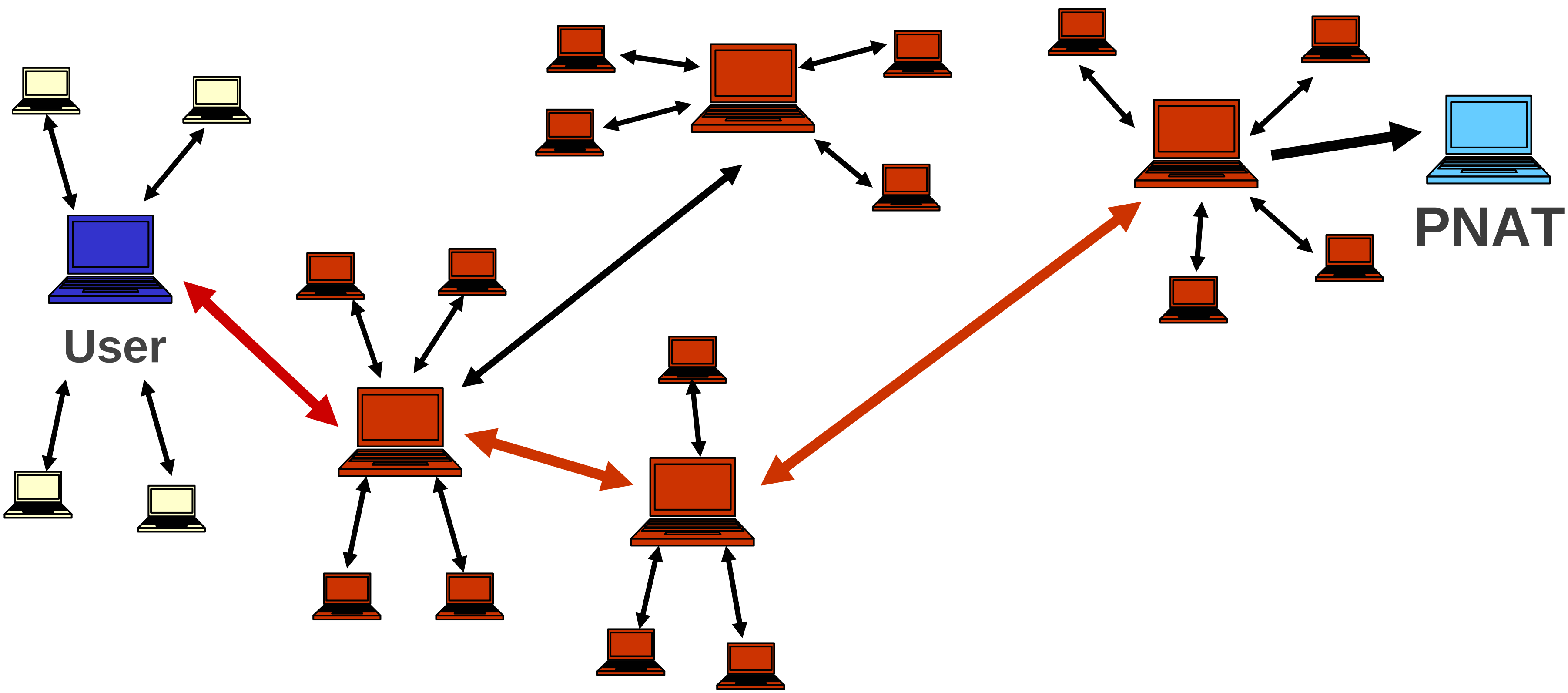
✓ Contact peers directly to validate IP addr and learn PK

But, Adversaries Can Join System

- Adversary can join more than once by running many nodes on each machine it controls

✓ Randomly select by subnet "domain" (/16 prefix, not IP)

But, Adversaries Can Join System



- Adversary can join more than once by running many nodes on each machine it controls

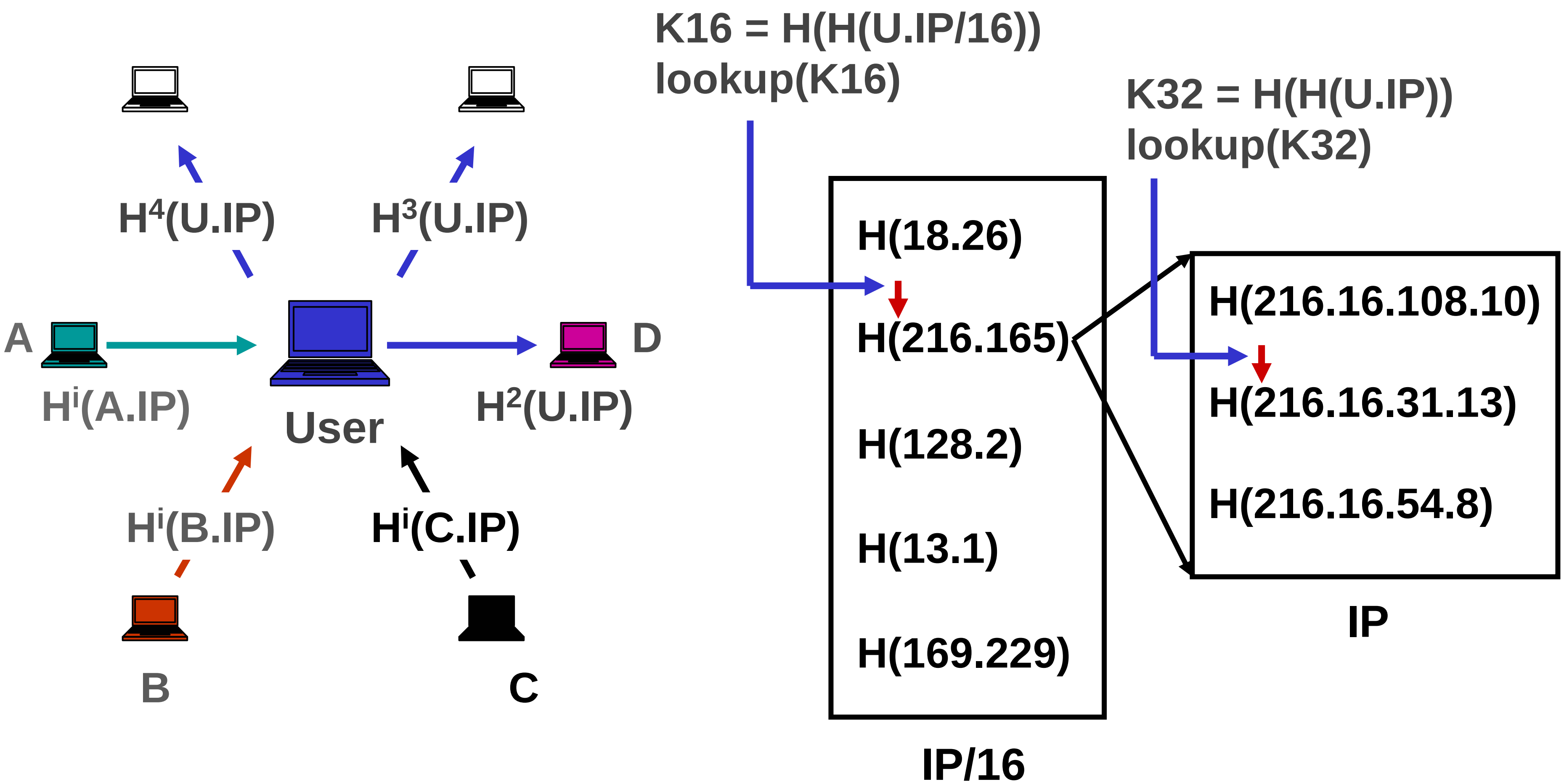
✓ Randomly select by subnet “domain” (/16 prefix, not IP)

But, Adversaries Can Join System

- Colluding adversary can only select each other as neighbors

✓ Choose mimics in universally-verifiable random manner

Tarzan: Selecting mimics

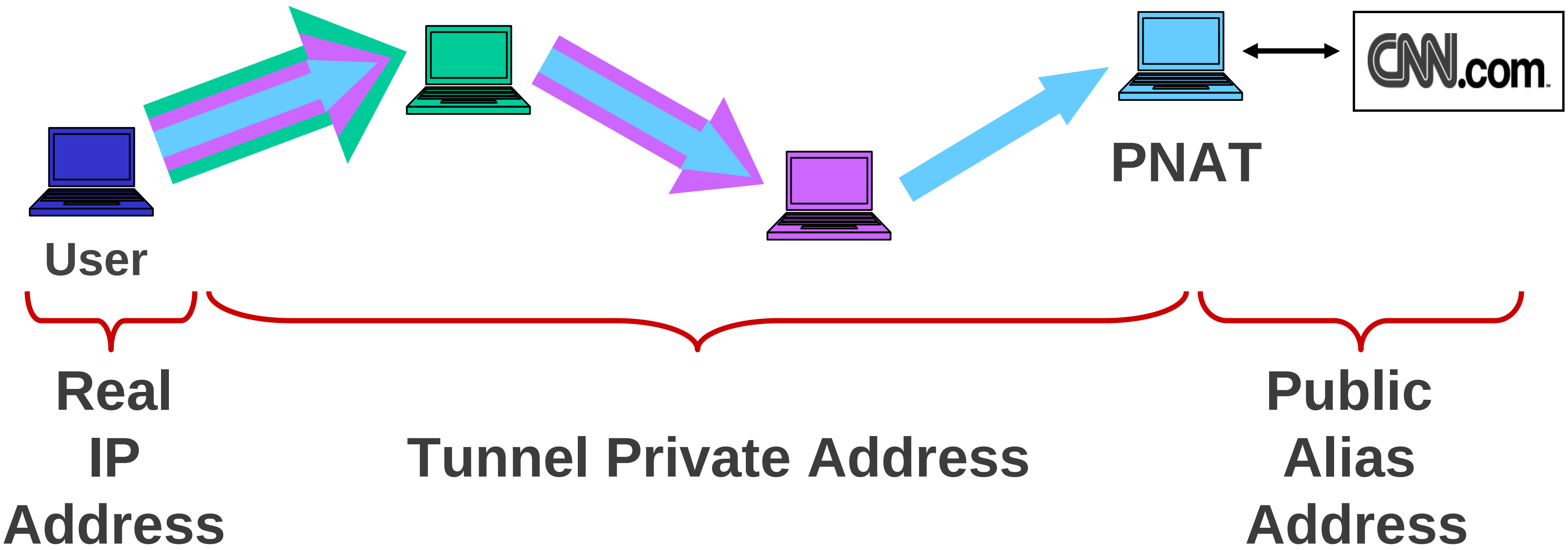


3. Nodes pair-wise choose (verifiable) *mimics*

Tarzan goals

- No distinction between anon proxies and clients
  - Peer-to-peer model
- Anonymity against corrupt relays
  - MIX-net encoding
  - Robust tunnel selection
  - Prevent adversary spoofing or running many nodes
- Anonymity against global eavesdropping
  - Cover traffic protects all nodes
  - Restrict topology to make cover practical
  - Choose neighbors in verifiably-random manner
- Application-independence
  - Low-latency IP-layer redirection

Tarzan: Building Tunnel

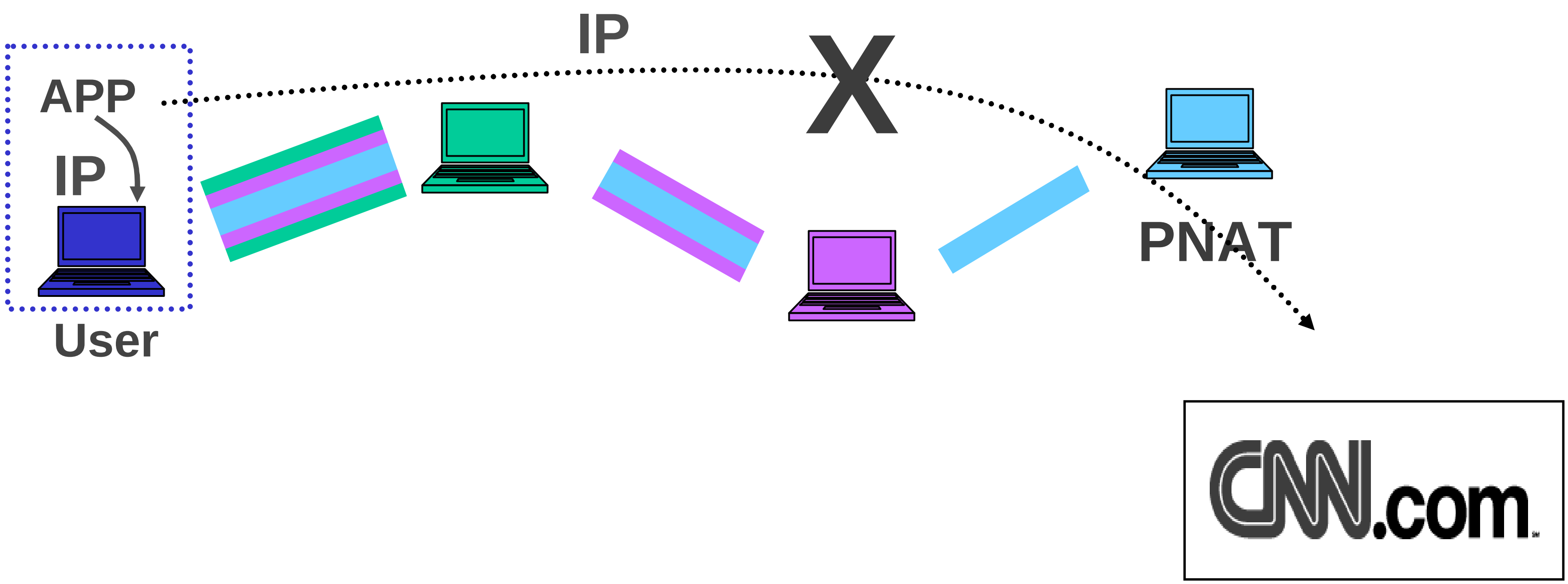


5. To build tunnel:

Public-key encrypts tunnel info during setup

Maps flowid → session key, next hop IP addr

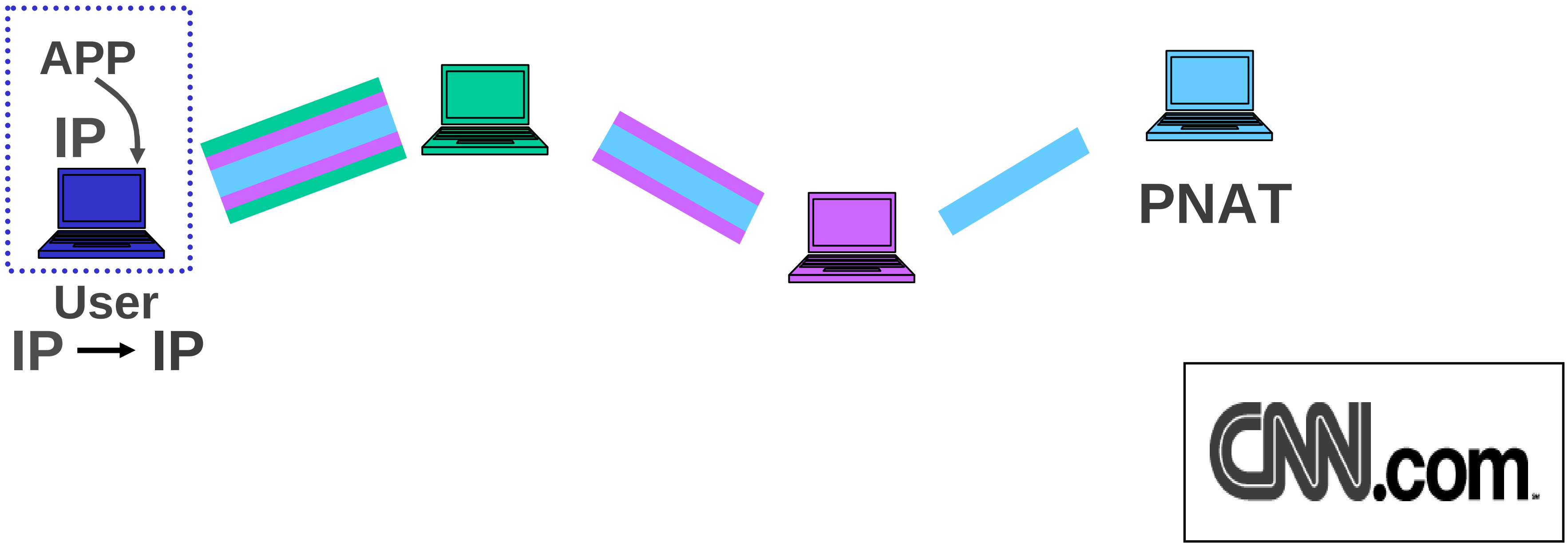
Tarzan: Tunneling Data Traffic



6. Reroutes packets over this tunnel

   Diverts packets to tunnel source router

Tarzan: Tunneling Data Traffic

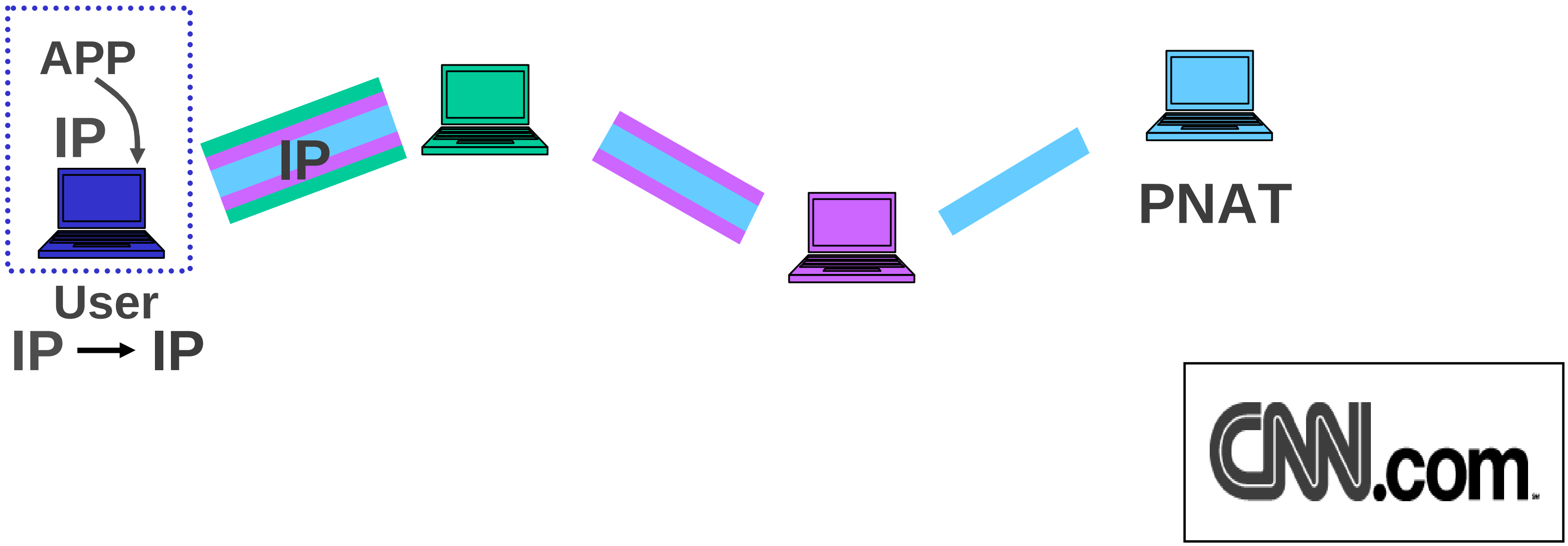


6. Reroutes packets over this tunnel

   NATs to private address 192.168.x.x

   Pads packet to fixed length

Tarzan: Tunneling Data Traffic

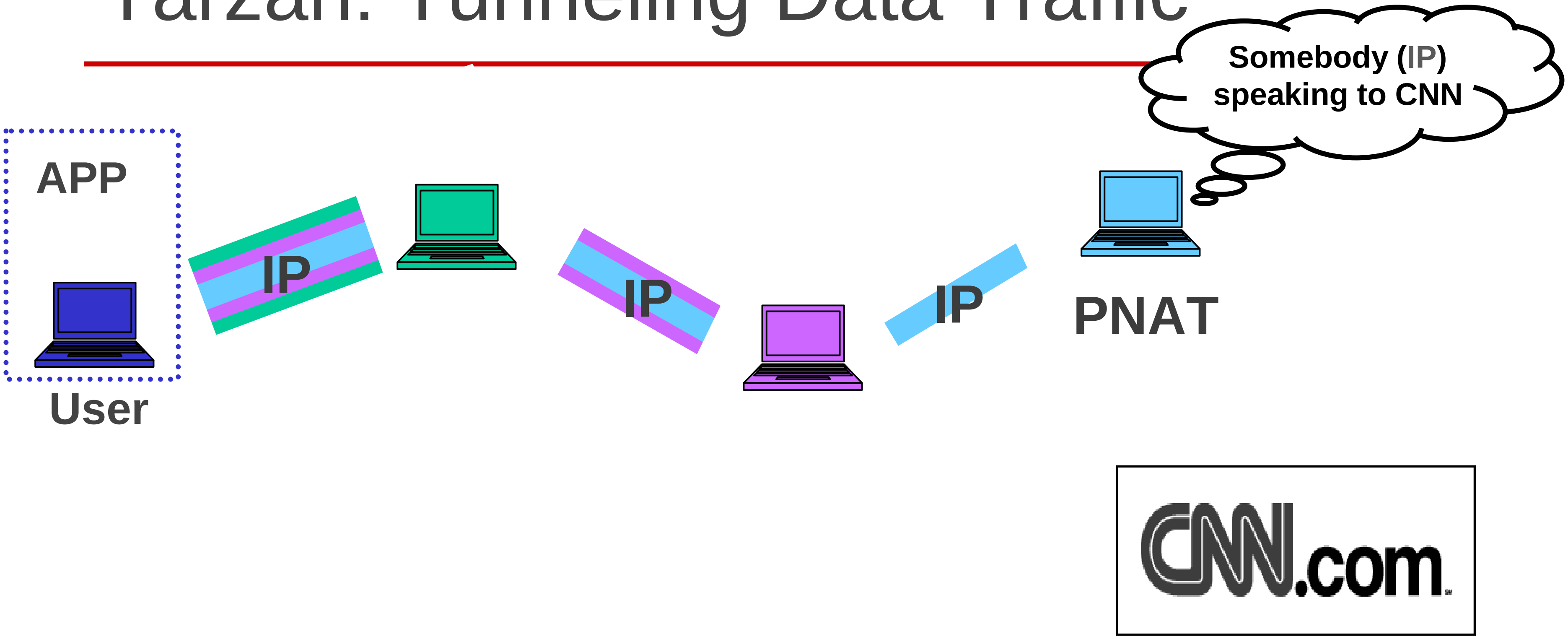


6. Reroutes packets over this tunnel

   Layer encrypts packet to each relay

   Encapsulates in UDP, forwards to first hop

Tarzan: Tunneling Data Traffic

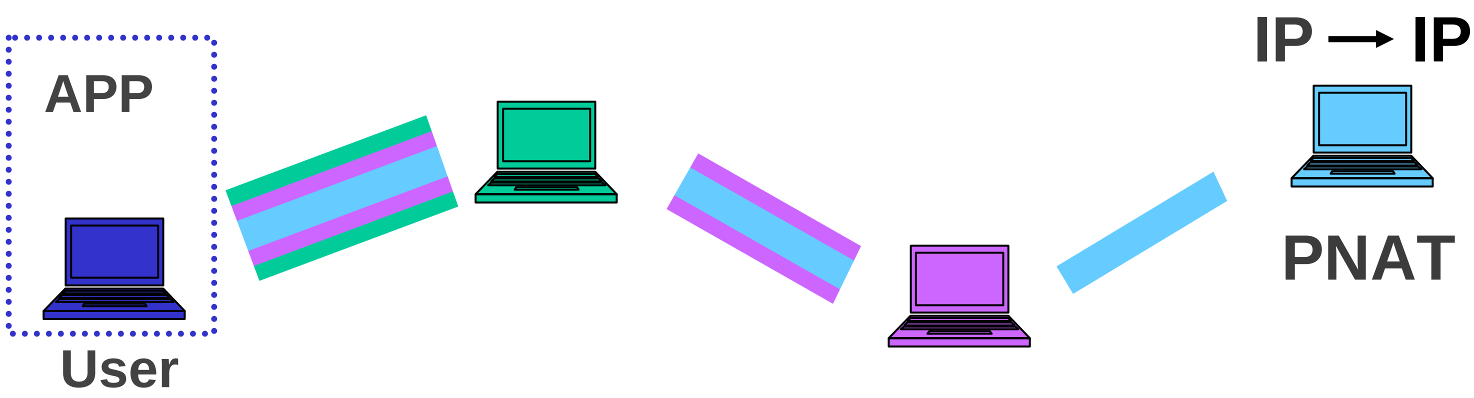


6. Reroutes packets over this tunnel

    Strips off encryption

    Forwards to next hop within cover traffic

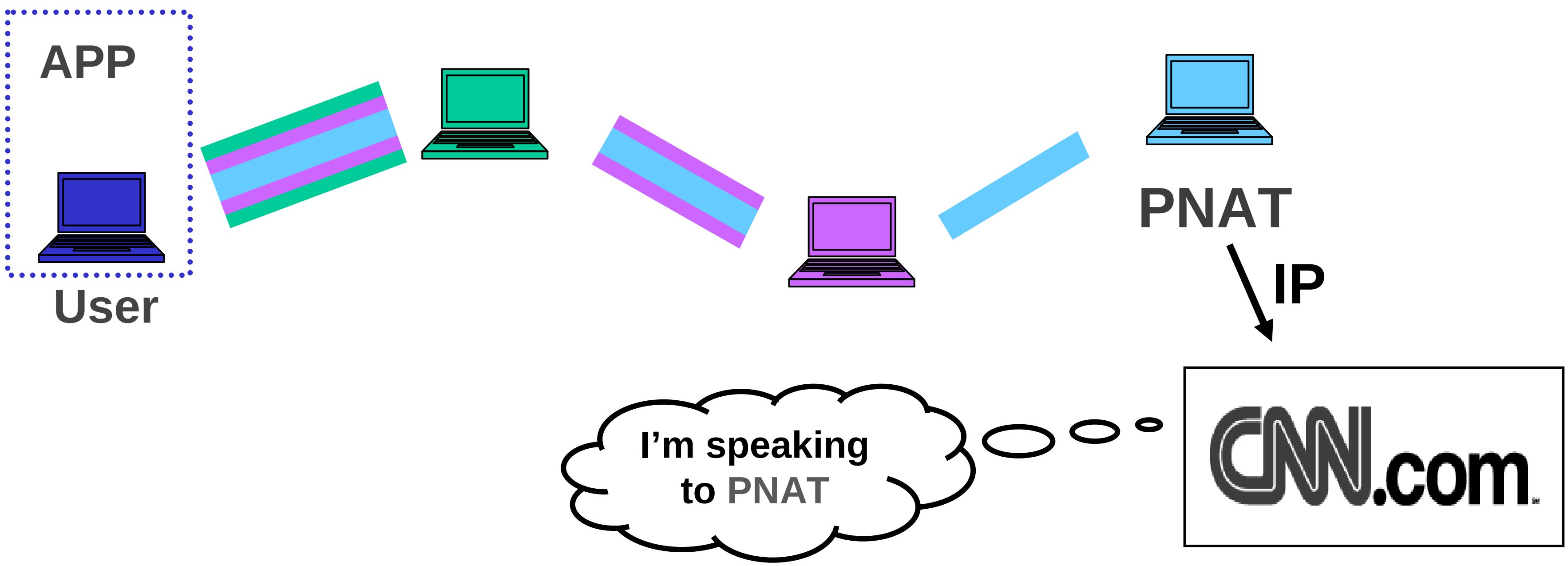
Tarzan: Tunneling Data Traffic



6. Reroutes packets over this tunnel

NATs again to public alias address

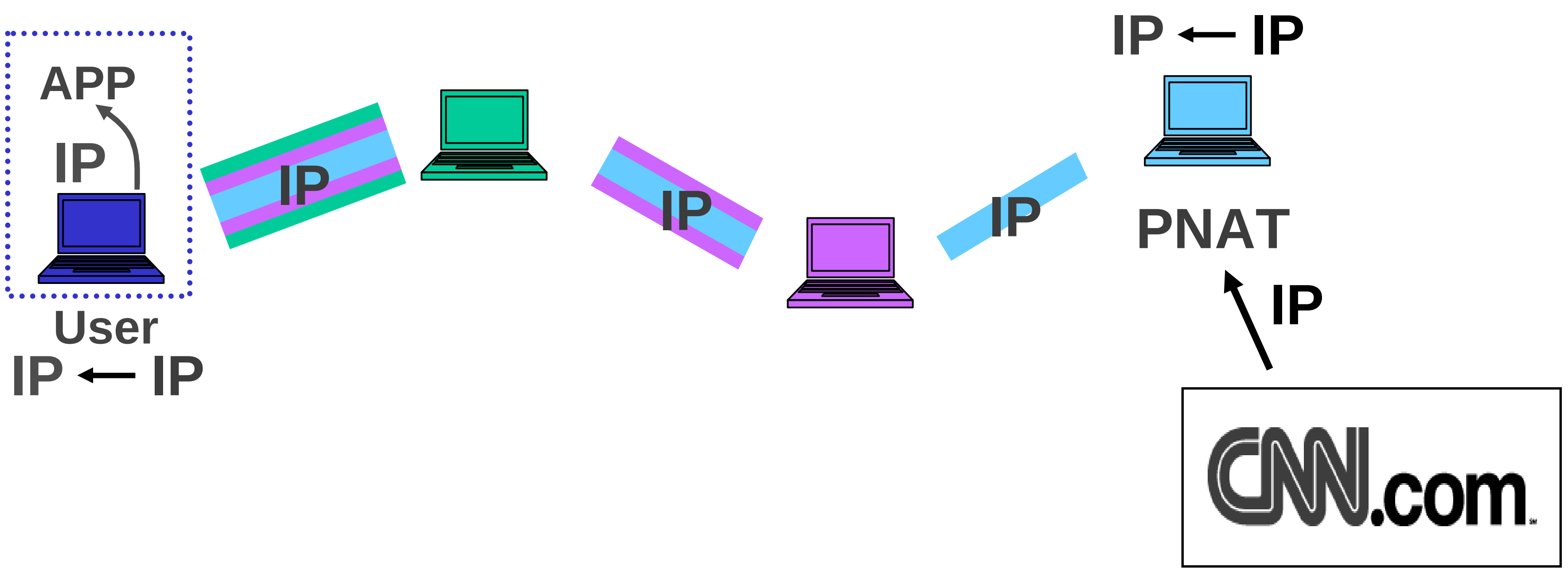
Tarzan: Tunneling Data Traffic



6. Reroutes packets over this tunnel

Reads IP headers and sends accordingly

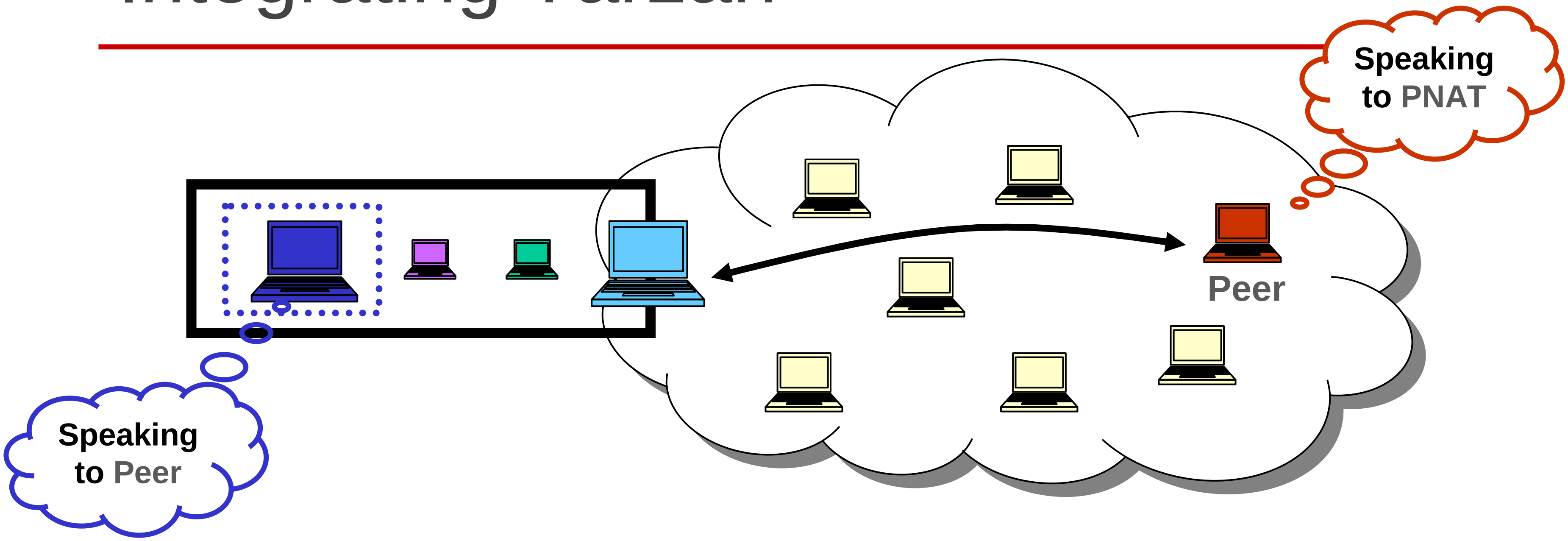
Tarzan: Tunneling Data Traffic



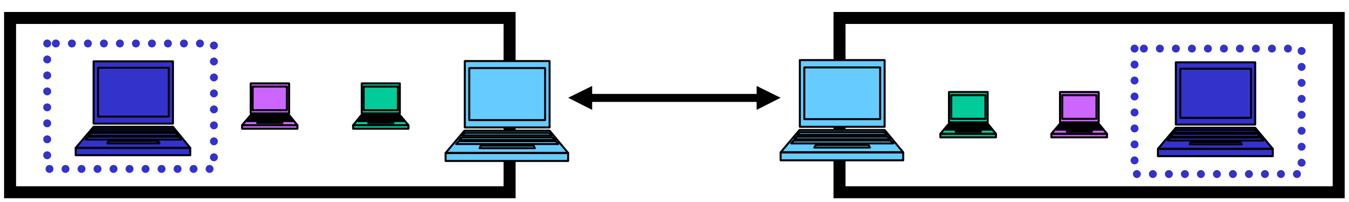
6. Reroutes packets over this tunnel

Response repeats process in reverse

Integrating Tarzan



Use transparently with existing systems



Can build double-blinded channels

Packet forwarding and tunnel setup

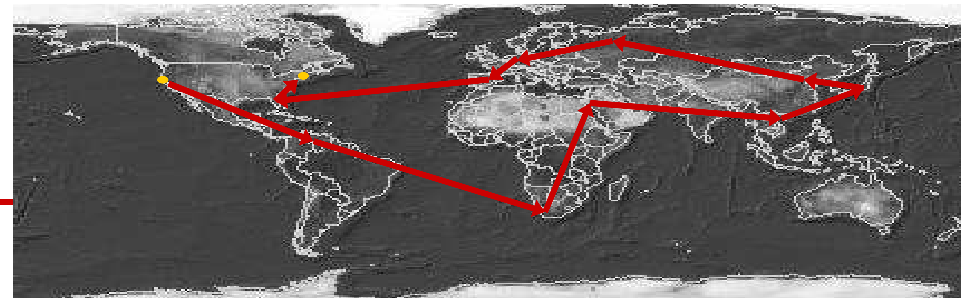
- Tunnel Setup (public key ops)

  ~30 msec / hop latency + network delay

- Packet forwarding (without cover traffic)

pkt size	latency	throughput
64 bytes	250 µsec	7 Mbits/s
1024 bytes	600 µsec	60 MBits/s

Summary



- Application-independence at IP layer
  - Previous systems for email, web, file-sharing, etc.
- No network edge through peer-to-peer design
  - Core routers can be blocked, targetted, or black-box analyzed
- Anonymity against corrupt relays and global eavesdropping
  - Cover traffic within restricted topology
  - MIX-net tunneling through verified mimics
- Scale to thousands
  - Towards a critical mass of users

http://pdos.lcs.mit.edu/tarzan/

Packet forwarding and tunnel setup

Pkt size (bytes)	**Latency** (μ-sec)	**Throughput** (pkts/s)	**Throughput** (Mbits/s)
64	244	14000	7.2
512	376	8550	35.0
1024	601	7325	60.0

Tunnel length	**Setup latency**	**Variance (1 StD)**
1	30.19	1.38
2	46.54	0.53
3	68.37	0.73
4	91.55	1.20

(msec)