CSC 574
Computer and Network Security

Anonymity and Privacy

Alexandros Kapravelos
kapravelos@ncsu.edu

(Derived from slides by William Robertson)

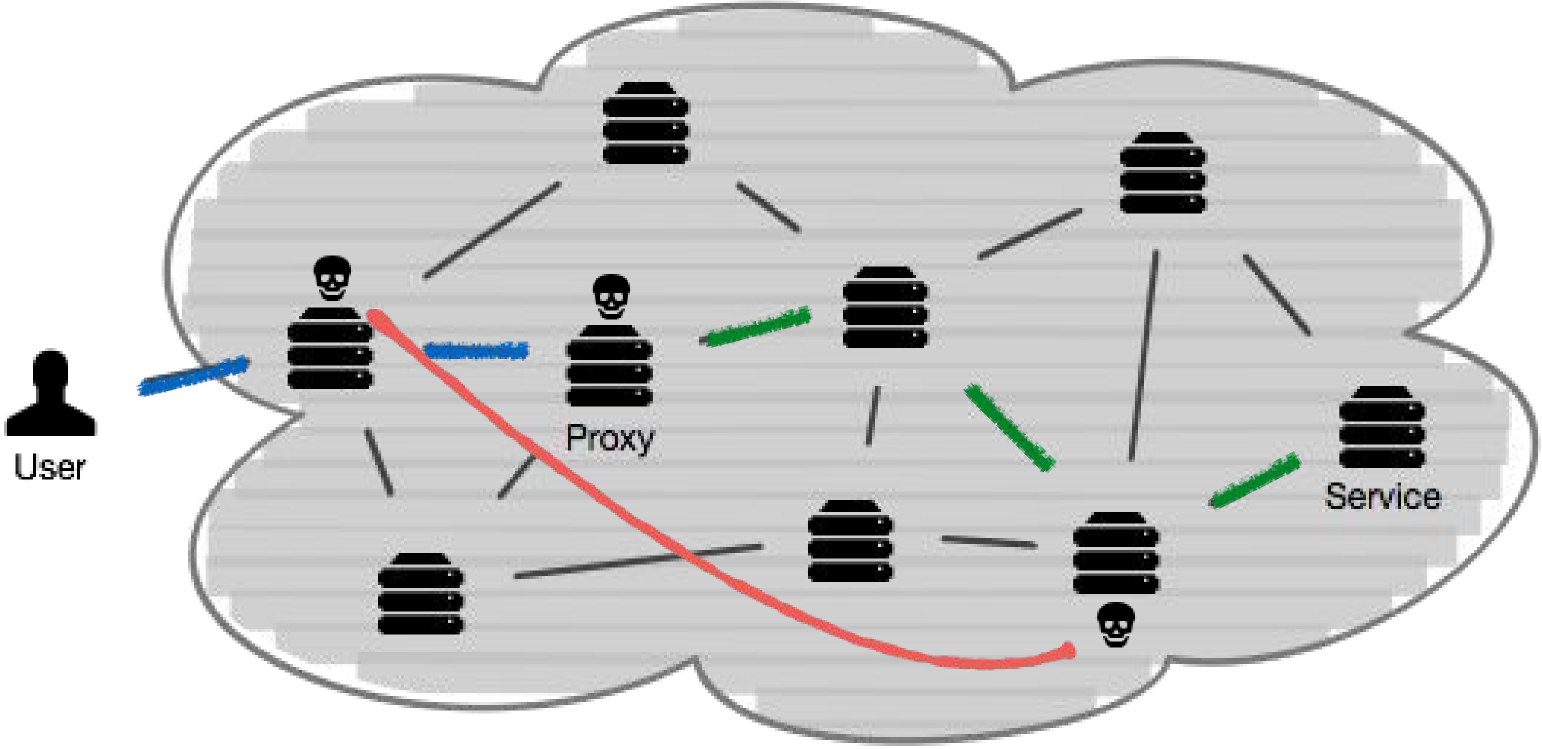
Anonymity

- Users often want some measure of anonymity or privacy in the network
    - As opposed to confidentiality, anonymity focuses on concealing *identity*
    - Threat model usually considers a powerful adversary – i.e., nation-state, network operator
- Anonymity can be abused, but there are also many good reasons to support it
    - e.g., whistleblowing, political protest

Network Anonymity

- Anonymity in the network focuses on concealing who is communicating with whom
    - i.e., defeating traffic analysis
- Adversary controls network, or can observe network at many different points
- Proxies or VPNs are one mechanism for disassociating sources and destinations
    - But, one malicious actor can defeat the security of a proxy-based approach

Proxies



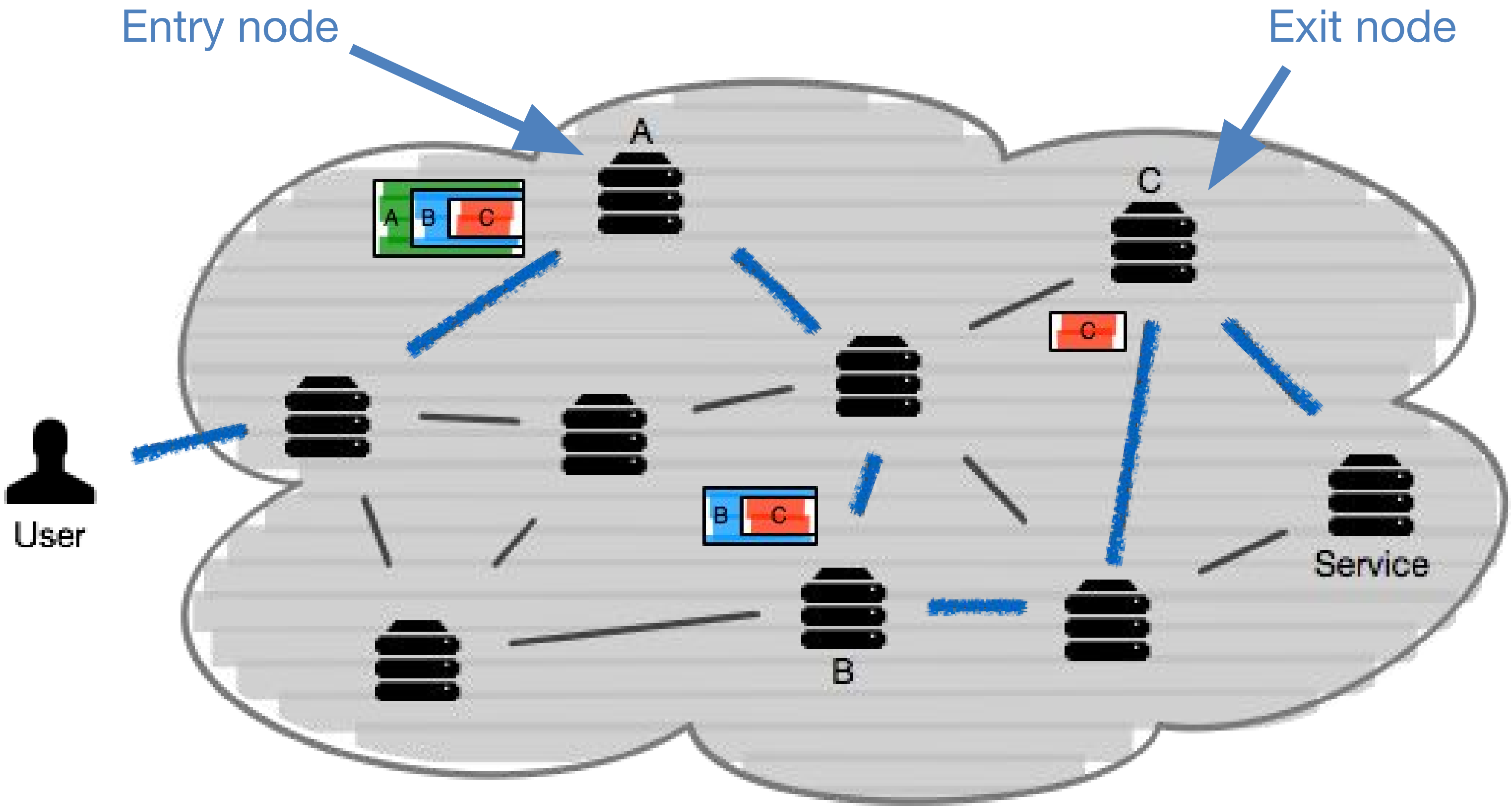
Proxies

- Proxies are easy to overcome if you're powerful enough
  - Compromise a proxy
  - Run malicious proxies
  - Correlate between different network vantage points
- Can this scheme be improved to defend against such an adversary?

TOR

- The Onion Router (TOR) improves on the basic proxy scheme
  - Instead of relying upon one router, Tor nodes form an overlay network of proxies on the Internet
  - Users randomly select a path – i.e., virtual circuit – using Tor nodes
- Uses *onion routing* to partially conceal routing information in the overlay
  - Overlay path is wrapped in layers of encryption, like an onion

TOR



TOR

- Users can conceal whom they're communicating with
  - Each layer of encryption hides successive hops in the overlay
- No single point of failure; why?
  - Nodes don't know path beyond immediate neighbors
  - Entry node doesn't know exit node, and vice-versa
- TOR also allows services to conceal their identity (hidden services)

TOR Threat Model

- What is TOR’s threat model?
  - (Semi-) global adversary
  - Can observe (a fraction of) network traffic
  - Can generate, modify, delete, or delay traffic
  - Can operate their own onion routers
  - Can compromise some fraction of onion routers

TOR Questions

- TOR provides perfect forward secrecy; how?
  - Initiator negotiates session keys with each circuit hop
  - Session keys destroyed after circuit torn down

Perfect Forward Secrecy

- A typical usage of public key cryptography is to generate a random symmetric session key (why?)
- If an attacker compromises a server's private key, he can decrypt future sessions (of course)
  - But, he can also decrypt *past sessions* as long as they were recorded
- PFS avoids this by using Diffie-Hellman key exchange to create a session key
  - Both sides create a fresh *ephemeral* DH keypair to negotiate a session key
  - DH keypairs immediately destroyed afterwards
  - Thus, they session key is *never sent over the connection*

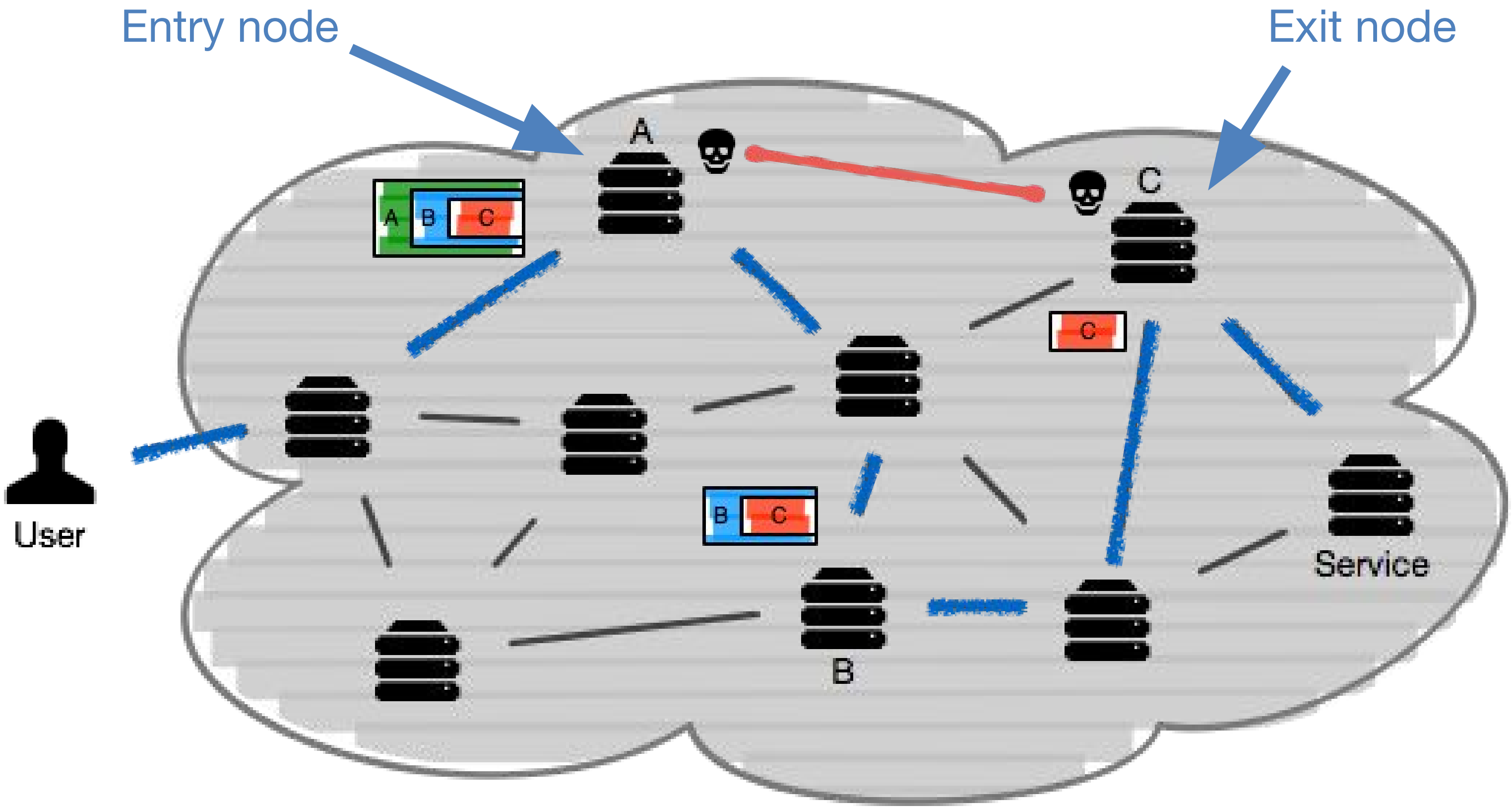
TOR Questions

- TOR provides perfect forward secrecy; how?
  - Initiator negotiates session keys with each circuit hop
  - Session keys destroyed after circuit torn down
- TCP streams are multiplexed over circuits; why?
  - Efficiency; setting up a circuit is not cheap
  - Improves anonymity; why?
- TOR does not attempt to provide steganographic protection; what does this mean?

Correlation Attacks

- TOR, by design, is a low-latency anonymity network
  - Trades off security against communication latency
- How can timing be used to deanonymize TOR users?
  - What if the adversary is controls some number of TOR nodes?
  - By correlating network timing, malicious TOR nodes can identify network flows belonging to a TOR user
  - If the entry and exit nodes are malicious, the adversary can identify the user and the service

Correlation Attacks



Information Leakage

- Even if TOR is used, applications can leak data that deanonymize the user
  - DNS queries
  - BitTorrent (control messages, DHT entries)
- Exit nodes are particularly powerful; why?
  - At the exit node, all encryption layers have been stripped, revealing the message destined for the service
  - Messages often contain identifying information – e.g., web browser cookies
  - TOR should be used in conjunction with application-level encryption!

TOR Node Blocking

- Users need to be able to discover TOR nodes to build overlay circuits
    - This means adversaries can also do so
- Exit nodes are often blocked or filtered by network services
    - TOR is abused, so this is reasonable
- Entry nodes are also blocked by repressive regimes
    - Denies access to TOR network
    - TOR rate-limits node discovery by network prefix, but this is bypassable (how?)
- Less a vulnerability per se, more of a denial of service

TOR and Heartbleed

- TOR uses the OpenSSL library for cryptography
  - In April 2014, the Heartbleed vulnerability was discovered
  - Exploitation revealed TOR node secret keys
  - Possession of secret keys would allow adversaries to strip away onion layers, revealing more of the overlay path