Privacy at the communication layer

Salsa: A Structured Approach to Large-Scale Anonymity.
Arjun Nambiar and Matthew Wright 2006

CS-721

Carmela Troncoso
http://carmelatroncoso.com/
Finding nodes is a hard task...

Most of the papers in the previous classes concentrated in how to send messages, and what happened once routes are chosen.

Yet in Tor / Crowds plenty of our discussion went into node lookup!

Also we talked about

“Bridging and Fingerprinting: Epistemic Attacks on Route Selection”
Danezis & Syverson
That showed how important node knowledge is.

Salsa is more about how to find nodes than about how to use them for anonymity.
What we have seen so far

**Tor**: the directory authority makes available the list of all nodes in the system

**Crowds**: the directory authority makes available the list of all nodes in the system

**DC networks**: not even mentioned... everybody knows everybody

**Others**

**Tarzan**: 
Core principle: **Me relay, you relay** (by M. Freedman)
- difficult to block everyone (censorship resistance)
- cover traffic for all
- no edges of the network, no first node

Layered encryption (not exactly onion routing)
Final NAT (“Pseudonymous NAT”) to connect to exterior
Cover traffic – “mimic” nodes exchange traffic
Source-based routing through mimics

**Node discovery**
- “Gossiping”: nodes ask neighbours for nodes
  from weakly connected to fully connected

**Node selection**
- limited to domains: avoid easy control of paths

Need full connection to
Avoid biases and leaks
What we have seen so far

**Tor**: the directory authority makes available the list of all nodes in the system

**Crowds**: the directory authority makes available the list of all nodes in the system

**DC networks**: not even mentioned... everybody knows everybody

**Others**

**Tarzan**: peers discover all peers via gossiping

**MorphMix**
Introducing MorphMix: Peer-to-Peer based Anonymous Internet Usage with Collusion Detection
Marc Rennhard and Bernhard Plattner, 2002

Non-source routed (Crowds-like)
- No lookup needed (only neighbours) → scalable
- What if first is an adversary?
  - Attacker nodes appear more often: Witness to collect offered nodes

Layered encryption (not exactly onion routing)

**Node discovery**
- Memory: try nodes you knew last time
- Server nodes
distribute “some” nodes (always random)
use several servers to avoid attacks
- Also from offered extensions
What we have seen so far

**Tor**: the directory authority makes available the list of all nodes in the system

**Crowds**: the directory authority makes available the list of all nodes in the system

**DC networks**: not even mentioned... everybody knows everybody

**Others**

**Tarzan**: peers discover all peers via gossiping

**MorphMix**: peer needs to know few nodes
What we have seen so far

**Tor**: the directory authority makes available the list of all nodes in the system

**Crowds**: the directory authority makes available the list of all nodes in the system

**DC networks**: not even mentioned... everybody knows everybody

**Others**

**Tarzan**: peers discover all peers via gossiping

**MorphMix**: peer needs to know few nodes

**Salsa**: know few nodes and redundancy anti-sybil

**Scalability**

**Intersection attacks**

**Sybil attacks**
Redundancy → more information

Redundant lookups: protect against active adversaries

But leave more information to passive adversaries

The more redundancy, more likely one of searches is run by a malicious node

No anonymity for the lookup!

Redundancy $\rightarrow$ More Information

Anonymity of communication

Nodes in the second level do not know initiator

One node per level compromised (redundancy $\uparrow$ probability)

If the malicious nodes de-anonymize the lookup...

(a) Bridging an honest first stage

Redundancy → more information

Anonymity of communication

Nodes in the second level do not know initiator

One node per level compromised (redundancy ↑ probability)

**AP3 = Crowds + secure lookup**

A. Mislove, G. Oberoi, A. Post, C. Reis, P. Druschel, and D. S. Wallach, 2004

**Crowds analysis = Pr[next to initiator]**

Non-anonymous Lookups

Information about lookups help identifying the predecessor!
The problem with non anonymous lookups is that the adversary links Sender – Node sought

Let's hide the node sought!!

Ask nodes for their finger list instead of the target

1. Ask fingers from top list of candidates
2. If a finger is nearer, put in top list
3. Repeat until list does not change
Let's hide the node sought!!

Ask nodes for their finger list instead of the target

1. Ask fingers from top list of finger candidates
2. If a finger is nearer, put in top list
3. Repeat until list does not change

But the lists reveal information!

If you are asked: you are a predecessor of target
If your fingers are not asked: they are successors of the target
Takeaways

- **Node discovery is difficult**
  - Scalability
  - Sybil prevention

- **Fixing problems... May add more problems**
  - More interactions result in more information

- **Solutions:**
  - Anonymity in lookup (TORSK, take a look)
  - Private information retrieval (Tor-alike)
Anonymous communications to protect anonymity

Can do more: censorship resistance

Next week

Telex: Anticensorship in the Network Infrastructure