P⁴PCN: Privacy-Preserving Path Probing for Payment Channel Networks

Ruozhou Yu, Assistant Professor, Department of Computer Science
North Carolina State University

with Yinxin Wan, Vishnu Teja Kilari, Guoliang Xue (Arizona State University),
Jian Tang (Syracuse University), Dejun Yang (Colorado School of Mines)
Blockchain Basics

Blockchain is a distributed sequential / transactional data store (a ledger) whose security (non-manipulability) is guaranteed via distributed consensus.

The biggest challenge of blockchain right now is its scalability issue due to global consensus.

Payment channels were invented to enable instant payment settlement, high transaction throughput.

Bound by crypto protocols, a payment channel is able to ensure blockchain-level security with an assumption on blockchain availability (connectivity).

Channels are more importantly used to construct multi-hop networks (PCN).

Blockchain Scalability
Example: Bitcoin

1. Tx Throughput < 7 transactions per second (tps)
2. Tx Confirmation Time ~1 hour (6-block conf.)

Off-chain Payment Channel | Instant Transactions via Local Consensus

1. On-chain (global) transaction
2. On-chain deposit for off-channel opening
3. Instant transactions via local consensus
4. On-chain arbitration when someone cheats

Do we really need global consensus?

Global Consensus
Every user validates all transactions to accept.

(Chained) Hash Pointers
Efficient data storage, dissemination and validation.

Incentive
Incentivized participation and honest validation.

Off-chain Channel
Smart Contract-based On-chain Arbitration
Local Consensus
Disagree

Ruozhou Yu, Yinxin Wan, Vishnu Teja Kilari, Guoliang Xue, Jian Tang, Dejun Yang

This research was supported in part by NSF grants 1704092, 1717197, 1717315, and 1525920.
Ruozhou Yu, Yinxin Wan, Vishnu Teja Kilari, Guoliang Xue, Jian Tang, Dejun Yang

PCN Basics

A well-connected PCN enables instant payment to arbitrary parties in the network with blockchain-level security.

Nevertheless, routing is a big problem, because the network is:
1. Fully distributed
2. Highly dynamic

Many algorithms employ path probing to find payment paths with enough capacity (balance).

Probing is used to gather current path information for dynamic routing.

However, probing commonly reveals sender & recipient information for a payment, leading to privacy concerns!

Payment through Channel
Balances BEFORE payment
$\mu 30 \rightarrow$ $\mu 25 \rightarrow$ $\mu 5$
Balances AFTER payment
$\mu 80 \leftarrow$ $\mu 105 \leftarrow$

Multi-hop Payment in PCN

Quest: Find a set of paths that satisfy a payment
Given: Only local balance information for each node

A Typical Dynamic PCN Routing Algorithm

1. Sender sends out probes.
2. Each intermediary updates balance.
3. Recipient selects path, and confirms back.
4. Each intermediary reserves and forwards.
5. Sender repeats until enough paths.

Privacy Concerns

#1 Sender / Recipient Privacy
Adversary may infer sender & recipient location & identity from probes.

#2 Cross-link Inference
Adversary may infer sender/recipient location by seeing a probe on two links.

#3 Path Confidentiality
Adversary may extract the probed paths either to locate sender/recipient or “steal” the paths (denial-of-service).


IEEE GLOBECOM 2019
CISs6: Security & Privacy 1
This research was supported in part by NSF grants 1704092, 1717197, 1717315, and 1525920.
P4PCN: Privacy-Preserving Path Probing for Payment Channel Networks

Ruozhou Yu, Yinxin Wan, Vishnu Teja Kilari, Guoliang Xue, Jian Tang, Dejun Yang

This research was supported in part by NSF grants 1704092, 1717197, 1717315, and 1525920.

Existing Anonymous Communication Protocols

Example: Onion Routing
1. Obtain all intermediate pub keys.
2. Wrap message & forwarding info with each key.
3. Each intermediary peels off one layer and forwards.

Problems
1. Before a probe is sent, sender does not know which paths it will take, hence public keys are not available.
2. There is no way to modify payload to append/update probed information.

Our Idea (based on Sphinx[2] and Universal Re-Encryption (URE)[3])

In-Path ElGamal Key Exchange[2]
Each intermediary establishes a symmetric key using a sender-supplied ElGamal component.

Reversed Onion
Established symmetric key is used to attach probed path in a reversed onion manner:
- Adds path info and onion-encrypts
- Re-encrypts with obfuscation key

Universal Re-Encryption[3]
Each intermediary further re-encrypts the entire probe (header + payload) to avoid inter-link inference.

Example: Onion Routing
1. Obtain all intermediate pub keys.
2. Wrap message & forwarding info with each key.
3. Each intermediary peels off one layer and forwards.

Privacy-preserving path probing has a main challenge:

The paths to be probed are not known in advance!

This prevents us from using existing anonymous communication protocols, all requiring knowing the intermediate public keys.

Thus, we define a new secure protocol for probing and information collection.

Anonymous Probing

Our construction novelly combines Sphinx [2] and URE [3], enabling in-path information appending with full anonymity guarantee.

We address additional challenges:
• Reversed onion for appending
• URE-aware ElGamal key exchange
• ElGamal component hiding

Our protocol enables efficient creation and processing of probes, as well as having a smaller probe size, compared to another construction (also our new contribution based on URE).

We believe the protocol can also find applications in many other scenarios, such as sensor or trust networks.

**Our Results**

**Evaluation Results (with HUM[3])**

**Probe Processing Time**

- \( P^4PCN \)
- \( HUM \)

**Probe Size**

- \( P^4PCN \)
- \( HUM \)

**Discussions**

- **Flooding**: opportunistic probing and other methods will be explored.
- **Other applications**:
  - Wireless sensor networks
  - Vehicular networks
  - Anonymous trust network

---