Fulgor: Concurrent and Privacy Preserving Transactions with Payment-Channel Networks

Giulio Malavolta\(^1\), Pedro Moreno-Sanchez\(^1\), Aniket Kate\(^1\), Matteo Maffei\(^1\), Srivatsan Ravi\(^1\)
\(^1\)Purdue University \(^1\)CISPA, Saarland University \(^1\)TU Vienna

\(^*\) Both authors contributed equally and are considered to be co-first authors

1. **Bitcoin Blockchain**

Bitcoin today:
- Decentralized cryptocurrency
- Payments logged in blockchain
- Widely used in practice
  - \(\rightarrow\) > 200,000 daily payments
  - \(\rightarrow\) > 12M accounts

Scalability Issues:
- New block every 10 min on average
- Limited to \(~10\) payments per second

(3) Problem Definition (I): Privacy

Challenges:
- Find path capacity without revealing individual channels capacity
- Perform payment revealing payment value only to users in the path

Our privacy goals:
1. Test Privacy: Only reveal whether there is enough capacity in the path
2. Value Privacy: Reveal payment value only to users in the path
3. Anonymity: Sender and receiver remain unknown to other users

(6) Fulgor: Payment Operation

Decentralized payment operation in two phases:
1. Hold coins from sender to receiver
2. Release coins from receiver to sender

Concurrent payments are prioritized by their identifiers
- Payments with higher identifier are forwarded first
- If not enough capacity, payment with higher identifier is queued

(5) Fulgor: Test Operation

Ideas:
- Communication between neighbors. A user only knows her neighbors
- Test operation reveals whether path capacity is smaller than a value \(v\)

\[
\text{test}_{\text{Cap}}(\{i_1\} \cap U_1, R_{cv}, v) : \quad \forall i \in [n-1]
\]

\[
(1) \quad r_i \leftarrow (0, 1)^k
\]

\[
(2) \quad c_i \leftarrow \text{Enc} (\lambda_i, r_i) | v| (n+2)
\]

\[
(3) \quad r_0 \leftarrow (0, 1)^k
\]

\[
(4) \quad c_0 \leftarrow \text{Enc} (\lambda_0, r_0) | R_{cv}
\]

\[
(5) \quad r_0 \leftarrow (0, 1)^k
\]

\[
(6) \quad \pi \leftarrow \pi^m
\]

Send(Rcv, (r_0, ..., r_n)) → u

\[
(7) \quad \text{return} (c_{i-1}) \cap U_2, (r_i, 0, v) \text{ to } u
\]

(7) Implementation

Implementation details:
- Proof-of-concept in Python
- Path up to 20 users
- Results similar to non-private version

Evaluation:

<table>
<thead>
<tr>
<th></th>
<th>Computation</th>
<th>Communication</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sender</td>
<td>12 ± 1.4 ms</td>
<td>10244 B</td>
</tr>
<tr>
<td>Int. User</td>
<td>16 ± 0.44 ms</td>
<td>10244 B</td>
</tr>
<tr>
<td>Receiver</td>
<td>3 ± 1.2 ms</td>
<td>8 B</td>
</tr>
</tbody>
</table>

(8) Conclusions

- Fulgor is payment-channel network compatible with current Bitcoin
- Provides privacy properties and non-blocking concurrent payments
- Efficient test and payment operations

Acknowledgments: This project has been supported by the CERIAS/Intel Research Assistantship