Information Leak in the Chord Lookup Protocol

Charles W. O’Donnell
Vinod Vaikuntanathan
Massachusetts Institute of Technology

August 25, 2004
4th IEEE International Conference on Peer-to-Peer Computing
Peer-to-Peer Privacy

- P2P systems often designed to trust participants, privacy not a concern
  - Gnutella

- Systems cannot guarantee privacy by compromising efficiency
  - Freenet
Thoughts

- If existing systems offer a reasonable amount of security, why go for a perfectly secure, but less efficient solution?

- **Chord** is very efficient, but security is ad-hoc and unanalyzed. How much privacy are we losing for efficiency?

- **Anonymity** our main concern as data privacy better ensured by encryption, etc.
Anonymity

**Requester Anonymity**
- The origin of a request (for an item of data) is untraceable by any **passive observer** adversary.

**Per-Request Anonymity Set**
- The set of possible initiators of a single request $x$ for the data item $D$, as seen by the adversarial node $N$:

$$P_{x}^{N,D}$$

**Average Anonymity Set**
- The expected value of the per-request anonymity set size over a uniform distribution of the set of possible requests for the data item $D$.

$$A_{N,D}^{N,D} = \mathbb{E}(|P_{x}^{N,D}|)$$
Per-Request Anonymity Set

- Request $r$ for data $z$ on $E$

$P_r^{L,z} = \{J, K, N, O\}$
Per-Request Anonymity Set

Request $r$ for data $z$ on $E$

$I^L_r = \{All \ except \ L\}$
Average Anonymity Set

\[ A^{N,D} \text{ for Node } L, \text{ network of } n \text{ nodes} \]

- \[ \alpha_K^2 = 16 \]
- \[ \alpha_P^2 = 4 \]
- \[ \alpha_Q^2 = 4 \]

\[ A^{N,D} = \frac{1}{n} \left[ \sum \alpha_i^2 + (n - \sum \alpha_i^2 - 1)(n-1) \right] \]

- Reqs Seen
- Reqs NOT seen

Diagram showing network topology with nodes and connections.
Anonymity Metrics

- Anonymity sets used before to evaluate privacy goals of systems (Tarzan [FM02])

- Quantifies complexity adversary encounters to determine original sender.

- Independent measurement from protocol implementation
Chord Overview

Structured P2P protocol using **DHTs**

- Shared flat address space for *IDs* and *Data Keys*
- Identifier determined by hashing IP and Virtual Node Identifier
- Data Key determined by hashing data item name

\[
ID_N = H(\text{IP}_N \cdot VNI_N) \quad \quad D_{data} = H(data)
\]

- Data stored in node ID which is closest, but prior to data’s key
**Chord Overview - Lookups**

- Lookup table has \( j \) logarithmic entries

**Routing of Lookups** (basic recursive mode)

- If Data between you and **successor**, return IP via path
- Else forward to closest table entry to Data Key

---

<table>
<thead>
<tr>
<th>( j )</th>
<th>( 0x\text{value} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0x1</td>
</tr>
<tr>
<td>1</td>
<td>0x2</td>
</tr>
<tr>
<td>2</td>
<td>0x4</td>
</tr>
<tr>
<td>3</td>
<td>0x8</td>
</tr>
</tbody>
</table>

---

0x15EE wants 0xD029
Chord Nodes See Little

- Requests seen an inverse metric to anonymity sets

**Thm 1 [SMLK03]**: Given data key $D$, expected number of lookups which traverse a random node is:

$$\Theta(\log n)$$

Any given Chord node sees few requests for a given Data
Bounds on Chord Anonymity Sets

- Distance from data critical to anonymity set size

- **Thm 2**: The further the observer is from the data, the less he knows about who requests it.
  
  - If the distance of a node from the data is $d$, with $n$ nodes, the size of its anonymity set is greater than:

    $$A^{N,D} \geq \frac{n}{12d^2} + n\left(1 - \frac{1}{d}\right) - 2$$

- Adversary hindered greatly by this
  
  Given 10,000 node network
  
  - Node 1 prior to Data has $A^{N,D} \geq 832$
  - Node 2 prior to Data has $A^{N,D} \geq 5207$

Observers away from data have Avg. Anon Set sizes near $n$
Corollaries

- Anonymity set size a trade-off between
  - Number of requests seen
  - Size of Per-Request anonymity set

- **Cor 1:** Average size of anonymity set over all nodes is: $\Omega(n)$

- **Cor 2:** Average number of requests seen by any observer is: $O\left(\frac{n}{d}\right)$

<table>
<thead>
<tr>
<th>$P_x^{N,D}$</th>
<th>Alice</th>
<th>Bob</th>
</tr>
</thead>
<tbody>
<tr>
<td>1st Request</td>
<td>3</td>
<td>n-1</td>
</tr>
<tr>
<td>2nd Request</td>
<td>3</td>
<td>1</td>
</tr>
</tbody>
</table>

Alice $A_{N,D}^N = 3$

Bob $A_{N,D}^N = \frac{n-1+1}{2}$

Distance away from data

Experimentation Goals

- Demonstrate analytical results within simulation
- Analyze additional system-wide effects:
  - Data Caching
  - Routing Variations: finger-table stretch, successor list, location caching
- Determine real-world anonymity of Chord Lookups in a stable network
- Simulation used 10,000 nodes with address space of $2^{32}$
  - Average results for $P_{x}^{N,D}$ and $A_{N,D}$ using uniformly random lookups
Experimental results match theoretical bounds

“Steps” appear in experiments because network is not infinitely continuous space
Data Caching

- Nodes cache data they have previously requested
  - Initiator caching or Path caching

- Data caching spreading data around the ring
  - Should reduce requests seen by all observers nearest to data

One less request seen, so

\[ A_{new}^{N,D} = \frac{n \cdot A_{old}^{N,D} + (n-1) - 4}{n} \]
**Finger Table Stretch**

- Increase size of finger-table creating more “fingers”
  - Fingers can stretch further than ½ of circle

- Shifts system-wide number of requests seen in even manner
Successor List

- Chord nodes always know of one successor.
  Expand successor list to next immediate $n$ nodes.

- Observers closest to data see low anonymity.
  - Only effects anonymity of observers closest to data.
  - Does nothing for observers further away.

One less request seen, so

$$A_{new}^{N,D} = \frac{n \cdot A_{old}^{N,D} + (n-1) - 9}{n}$$
**Location Caching**

- Nodes cache locations of previously queried hosts
  - Acts like a **dynamic finger-table**
  - Uses Initiator caching or Path caching
  - Able to reach very far around circle

- Reduces number of hops a lot by bypassing most intermediate nodes

One less request seen, so

\[
A_{new}^{N,D} = \frac{n \cdot A_{old}^{N,D} + (n-1)-1}{n}
\]
Utility of Experimental properties

- **Data caching** and **location caching** best for $A^{N,D}$
- **Successor list** helps increase $A^{N,D}$ of observers closest to data
- **Variable finger table** has little effect on anonymity
Successor List Improvements

- The very closest observers do benefit greatly
- Improvements come in clear steps dependent on list size
Conclusions

- **Anonymity depends on distance** observer is from data:
  \[ A^{N,D} \geq \frac{n}{12d^2} + n \left( 1 - \frac{1}{d} \right) - 2 \]

- **Anonymity can be shown to vary in a predictable manner** given the mode of deployment of Chord (e.g. data caching, location caching, successor lists, finger table sizing)

- Under considerations **Chord meets certain anonymity concerns**, while maintaining fast lookup times

- Future considerations might include multiple observers, effects of network churn, network topology discovery
Danke
Merci
Grazie
Grazcha