Differentially Private Oblivious RAM

Sameer Wagh*, Paul Cuff†, Prateek Mittal*

July 24, 2019

*Princeton University, †Renaissance Technologies
Access data **privately** from **private** database.
Introduction: Oblivious RAM

User receives record $R$
Introduction: Oblivious RAM

Obliviousness: Adversary should not know $R$
Client-server environments
Trusted Execution Environments such as SGX-based enclaves
The Problem?

User

Record R

Database
The Problem?
The Problem: Overhead

Overheads

- Logarithmic bandwidth overhead ($\geq 100\times$)
- Logarithmic storage overhead
Can we improve performance by relaxing privacy?
Key Insight: Improve Performance by Relaxing Privacy

- Statistically private ORAM
  - Better performance at the cost of privacy loss
  - Challenge: Can we provide rigorous guarantees?
Key Insight: Improve Performance by Relaxing Privacy

- **Statistically private ORAM**
  - Better performance at the cost of privacy loss
  - Challenge: Can we provide rigorous guarantees?

- **Efficiency**
  - Reduce performance overheads – bandwidth, local storage
  - Achieve privacy proportional to application resources
Key Insight: Improve Performance by Relaxing Privacy

Differential Privacy

- Formalize Differentially Private ORAM
- Introduce Root ORAM
Key Insight: Improve Performance by Relaxing Privacy

Differential Privacy
- Formalize Differentially Private ORAM
- Introduce Root ORAM

Root ORAM
- Theoretical Results
- Empirical Results
- Private Information Retrieval
Differentially Private Oblivious RAM
DP-ORAM Intuition

Program → a → ORAM Protocol → O → Server

Input access sequence

Output access sequence

Client
DP-ORAM Intuition

Program $\rightarrow$ a $\leftarrow$ ORAM Protocol $\rightarrow$ O $\leftarrow$ Server

Input access sequence

Output access sequence

Distribution
DP-ORAM Intuition

Neighboring access sequences

\[ |a_1 - a_2| = 1 \]
The statistical closeness for Differential Privacy is expressed as:

$$Pr[\text{ORAM}(a_1) \in S] \leq e^\epsilon Pr[\text{ORAM}(a_2) \in S] + \delta$$
Protocol Construction
Root ORAM: Storage

**Mapping**

1 --> Leaf₃
2 --> Leaf₁
3 --> Leaf₁
4 --> Leaf₇
5 --> Leaf₅
6 --> Leaf₅
7 --> Leaf₈
8 --> Leaf₆

**Client**

**Server**

N = 2^L Leaves

K = 1
Root ORAM: Invariant

**Mapping**

1 --> **Leaf**₃
2 --> **Leaf**₁
3 --> **Leaf**₁
4 --> **Leaf**₇
5 --> **Leaf**₅
6 --> **Leaf**₅
7 --> **Leaf**₈
8 --> **Leaf**₀

**Client**

**Server**

$K = 1$

$N = 2^l$ Leaves
Root ORAM: Updated mapping

**Mapping**

1 → LEAF$_3$
3 → LEAF$_1$
4 → LEAF$_7$
5 → LEAF$_5$
6 → LEAF$_5$
7 → LEAF$_8$
8 → LEAF$_8$

**Client**

**Server**

K = 1

N = 2$^l$ LEAVES
Root ORAM: Updated mapping

**Mapping**

1 $\rightarrow$ LEAF$_3$
2 $\rightarrow$ LEAF$_2$
3 $\rightarrow$ LEAF$_1$
4 $\rightarrow$ LEAF$_7$
5 $\rightarrow$ LEAF$_5$
6 $\rightarrow$ LEAF$_5$
7 $\rightarrow$ LEAF$_8$
8 $\rightarrow$ LEAF$_6$

**Client**

**Server**

$K = 1$

$N = 2^k$ LEAVES
Key Insight

- Uniform mapping $\Rightarrow$ Conventional Security
Key Insight

- Uniform mapping $\Rightarrow$ Conventional Security
- Non-uniform mapping $\Rightarrow$ DP-ORAM Security
Root ORAM: Updated mapping

**Mapping**

1 --> Leaf₃
2 --> Leaf₂
3 --> Leaf₁
4 --> Leaf₇
5 --> Leaf₅
6 --> Leaf₅
7 --> Leaf₈
8 --> Leaf₁₀

**Client**

**Server**

K = 1

\( N = 2^k \) Leaves
Impact

- Lower average placement $\Rightarrow$ Improved performance
- Privacy loss
Root ORAM: Write back

<table>
<thead>
<tr>
<th>MAPPING</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 → Leaf₃</td>
</tr>
<tr>
<td>2 → Leaf₂</td>
</tr>
<tr>
<td>3 → Leaf₁</td>
</tr>
<tr>
<td>4 → Leaf₇</td>
</tr>
<tr>
<td>5 → Leaf₅</td>
</tr>
<tr>
<td>6 → Leaf₅</td>
</tr>
<tr>
<td>7 → Leaf₈</td>
</tr>
<tr>
<td>8 → Leaf₆</td>
</tr>
</tbody>
</table>

\[ K = 1 \]

\[ N = 2^k \text{ LEAVES} \]
Root ORAM: Lowest Common Intersection

**Mapping**

- 1 → Leaf₃
- 2 → Leaf₂
- 3 → Leaf₁
- 4 → Leaf₇
- 5 → Leaf₅
- 6 → Leaf₅
- 7 → Leaf₈
- 8 → Leaf₈

**Client**

**Server**

- K = 1
- N = 2ᴺ⁻¹ Leaves
Root ORAM: Lowest Common Intersection

**Mapping**

1 --> Leaf\(_3\)
2 --> Leaf\(_2\)
3 --> Leaf\(_1\)
4 --> Leaf\(_7\)
5 --> Leaf\(_5\)
6 --> Leaf\(_5\)
7 --> Leaf\(_8\)
8 --> Leaf\(_8\)

**Client**

**Server**

N = 2\(^k\) Leaves

K = 1
Root ORAM: Lowest Common Intersection

MAPPING

1 -> LEAF₃
2 -> LEAF₂
3 -> LEAF₁
4 -> LEAF₇
5 -> LEAF₅
6 -> LEAF₅
7 -> LEAF₈
8 -> LEAF₀

K = 1

N = 2ᵏ LEAVES

CLIENT

SERVER
Database view before access

**Mapping**

1 → LEAF₃
2 → LEAF₂
3 → LEAF₁
4 → LEAF₇
5 → LEAF₅
6 → LEAF₅
7 → LEAF₈
8 → LEAF₀

**Client**

**Server**

K = 1

N = 2¹ leaves
Database view after access

Mapping

1 --> Leaf\textsubscript{3}
2 --> Leaf\textsubscript{2}
3 --> Leaf\textsubscript{1}
4 --> Leaf\textsubscript{7}
5 --> Leaf\textsubscript{5}
6 --> Leaf\textsubscript{5}
7 --> Leaf\textsubscript{8}
8 --> Leaf\textsubscript{6}

K = 1

N = 2\textsuperscript{k} leaves
Results
Differentially Private ORAM Protocol

The Root ORAM protocol with parameters $k, p$ is $(\epsilon, \delta)$-differentially private for the following choice of $\epsilon$ and $\delta$

$$\epsilon = 2 \log \left( \frac{1 + (2^k - 1) \cdot p}{1 - (1 - \delta_{k0})p} \right)$$

$$\delta = M \cdot \left( \frac{1 + (2^k - 1) \cdot p}{N} \right)^M$$

(1)

where $\delta_{k0}$ is the Kronecker delta, $M$ is the size of the access sequence and $M > \text{total stash size.}$
Performance Improvements

Improvement in stash usage for $(L, k, Z) = (15, 1, 4)$
Key takeaway

DP-ORAM can enhance performance at the cost of privacy
Application: Private Information Retrieval
Private Information Retrieval (PIR)

Access data **privately** from **public** database.

---

ORAM based PIR

- ORAM has been used previously for PIR [7, 59]


ORAM based PIR

- ORAM has been used previously for PIR [7, 59]


ORAM based PIR

- ORAM has been used previously for PIR [7, 59]


ORAM based PIR

- ORAM has been used previously for PIR [7, 59]


ORAM based PIR

- ORAM has been used previously for PIR [7, 59]


ORAM based PIR

- ORAM has been used previously for PIR [7, 59]


ORAM based PIR

- ORAM has been used previously for PIR [7, 59]


Security-Bandwidth trade-offs for DP-PIR protocols (Toledo et al. [54], Path-PIR [42], and Path ORAM [53]).
DP-ORAMs provide significant performance benefits for DP-PIR
Conclusion
Summary

- Formalized Differentially Private ORAMs
- Introduced a family of DP-ORAM protocols
- Analyzed security, performance
- Showcased utility for Private Information Retrieval

Source code is available at https://github.com/inspire-group/Root-ORAM
Summary

- Formalized Differentially Private ORAMs
- Introduced a family of DP-ORAM protocols
- Analyzed security, performance
- Showcased utility for Private Information Retrieval

- Possible to enhance performance by relaxing privacy

Source code is available at https://github.com/inspire-group/Root-ORAM
Summary

- Formalized Differentially Private ORAMs
- Introduced a family of DP-ORAM protocols
- Analyzed security, performance
- Showcased utility for Private Information Retrieval

- Possible to enhance performance by relaxing privacy

Source code is available at
https://github.com/inspire-group/Root-ORAM
Thank you!
Thank you!

Questions?