High Recovery with Fewer Injections: Practical Binary Volumetric Injection Attacks against Dynamic Searchable Encryption

Xianglong Zhang, Wei Wang, Peng Xu, Laurence T. Yang, Kaitai Liang*

Huazhong University of Science and Technology

*Delft University of Technology
Ⅰ. Motivations

Ⅱ. Our Attacks

Ⅲ. Conclusion
I. Motivations
Dynamic Searchable Encryption (DSE)
Threats faced by DSE

- **Query**
  - Client
  - Generate token
  - Encrypted file
  - Search the database
  - Server

- **Update**
  - Client
  - Encrypt and upload
  - Store the file
  - Server

**Inject files for query recovery!**
Injection attack model
Previous injection attacks

- Zhang et al. [ZKP16]: Binary search attack, but require to **identify the injected files**, i.e., injected files access pattern.

- Poddar et al. [PWL+20]: Relies on the response length pattern (rlp), i.e., the number of response files, but require to **inject massive files** (Exceeding the number of keywords). ----- Volumetric attack (with rlp).

- Blackstone et al. [BKM20]: Relies on the response size pattern (rsp), i.e., the word count of returned files, but **still inject linear number of files**. ----- Volumetric attack (with rsp).
Previous injection attacks

- Zhang et al. [ZKP16]: Binary search attack, but require to identify the injected files, i.e., injected files access pattern.

- Poddar et al. [PWL+20]: Relies on the response length pattern (rlp), i.e., the number of response files, but require to inject massive files (Exceeding the number of keywords). ----- Volumetric attack (with rlp).

- Blackstone et al. [BKM20]: Relies on the response size pattern (rsp), i.e., the word count of returned files, but still inject linear number of files. ----- Volumetric attack (with rlp).

- **Summary:** No practical volumetric attacks with fewer injection length (No. of injected files) and injection size (No. of injected words).
Ⅱ. Our attacks
Our contributions

● Binary variable-parameter attack (BVA) with logarithmic injection length by exploiting the rsp.

● Binary volumetric matching attack (BVMA) to further reduce the injection size by exploiting the rlp and rsp.

● Extensive analysis against padding and update.
Parameters range: \#W is the number of known keywords, \( m \geq 1, \) offset \( \gg \#W, \gamma \geq \#W/2. \)

Optimal injection length and injection size.

<table>
<thead>
<tr>
<th>Attack</th>
<th>Injection length</th>
<th>Injection size</th>
</tr>
</thead>
<tbody>
<tr>
<td>[ZKP16]</td>
<td>( O(\log #W) )</td>
<td>( O(#W \log #W) )</td>
</tr>
<tr>
<td>[PWL+20] (Multiple-round attack)</td>
<td>( O(#W \log #W) )</td>
<td>( O(#W^2) )</td>
</tr>
<tr>
<td>[PWL+20]* (Single-round attack)</td>
<td>( O(m#W) )</td>
<td>( O(m#W^2) )</td>
</tr>
<tr>
<td>[BKM20] (Decoding attack)</td>
<td>( O(#W) )</td>
<td>( O(\text{offset} \cdot #W^2) )</td>
</tr>
<tr>
<td>[BKM20]* (Search attack)</td>
<td>( O(#W \log #W) )</td>
<td>( O(#W^2) )</td>
</tr>
<tr>
<td>Ours (BVA)</td>
<td>( O(\log #W) )</td>
<td>( O(\gamma #W) )</td>
</tr>
<tr>
<td>Ours (BVMA)</td>
<td>( O(\log #W) )</td>
<td>( O(#W \log #W) )</td>
</tr>
</tbody>
</table>
**Observe** `rsp` of unknown queries before injection

**Inject** logarithmic files with different size.

**Recover** the query `q` with `rsp_q`

If `rsp_q -rsp_l = k \cdot \gamma`, recover `q` as `w_k`.

- **Logarithmic** injected files, e.g., only 20 files for $10^6$ keywords.
- **\( \gamma \cdot \#W \)** injected words.
- Adjust `\gamma` to balance the injection size and recovery rate.
Experiments on BVA

- Set $\gamma = O(#W)$ is enough to achieve practical recovery, e.g., exceed 60% recovery in three datasets.
- Less injection size than decoding attack of [BKM20].
Similar to the process of BVA, but exploiting the difference of rsp and rlp before and after injection for query recovery.

Achieve the optimal injection size, i.e., $O(#W \log #W)$. 
- Similar high recovery rate (around 80%).
- Less injection length and injection size (save >99% injection costs).
Attacks against static padding (SEAL, [DPP+20])

Optimized attack against dynamic padding (ShieldDB, [VYS+21])

- Effectively bypass these paddings.
Here, we set the upper bound of $\gamma$.
A small $\gamma$ is actually enough.
Evaluations against update

- $\gamma = 32W$ can help us to achieve $>50\%$ recovery.
Ⅲ. Conclusion
Conclusion

- Two volumetric attacks with small injections and high recovery.

- Effectively against some paddings.

- An effective countermeasure to our attacks should be *hybrid* and *probabilistic*, i.e., being able to hide both file size and response length by random (or differentially private) noisy padding.
Thank you for listening!

Code available: https://github.com/Kskfte/BVA-BVMA

Contact information:
→ vrwudi@gmail.com
→ viviawangwei@hust.edu.cn
→ xupeng@hust.edu.cn