USENIX Security ‘23

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*

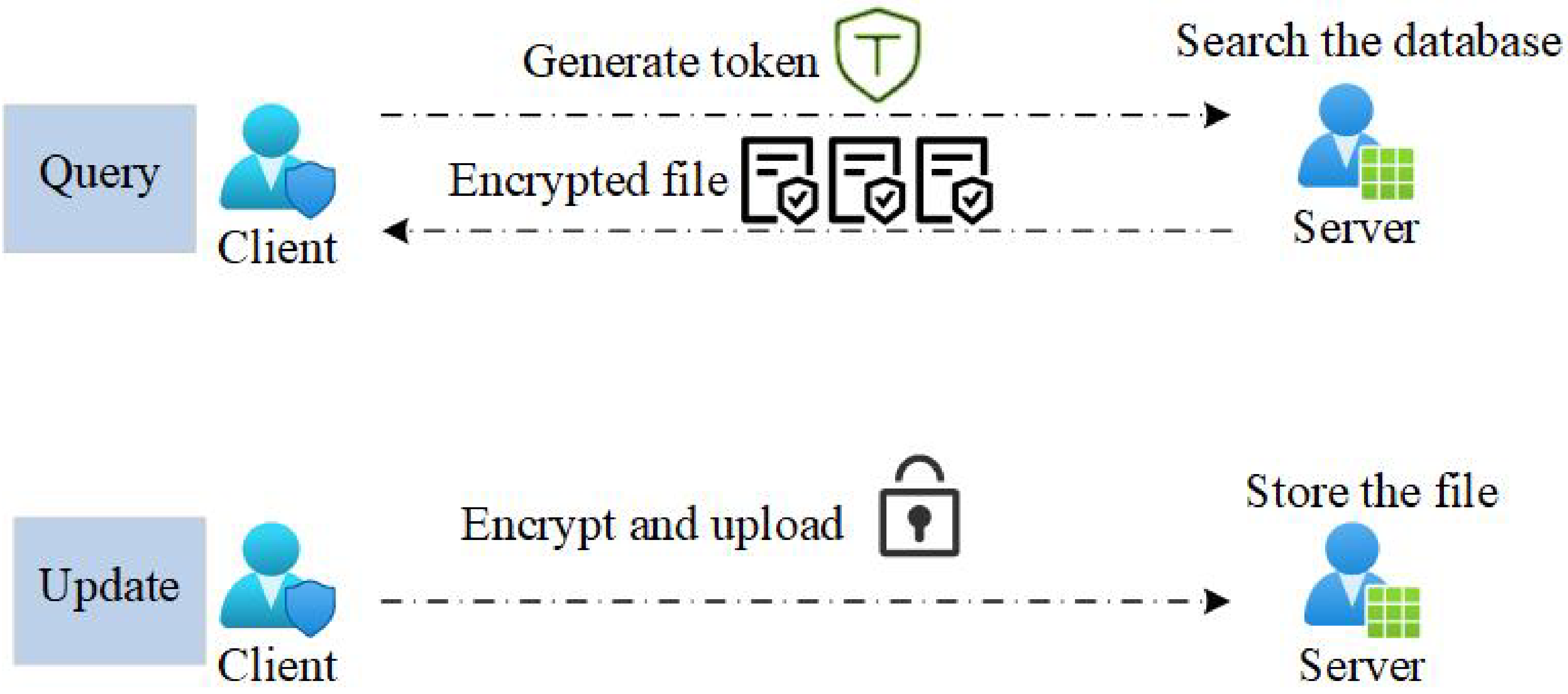
Huazhong University of Science and Technology

**Delft University of Technology*

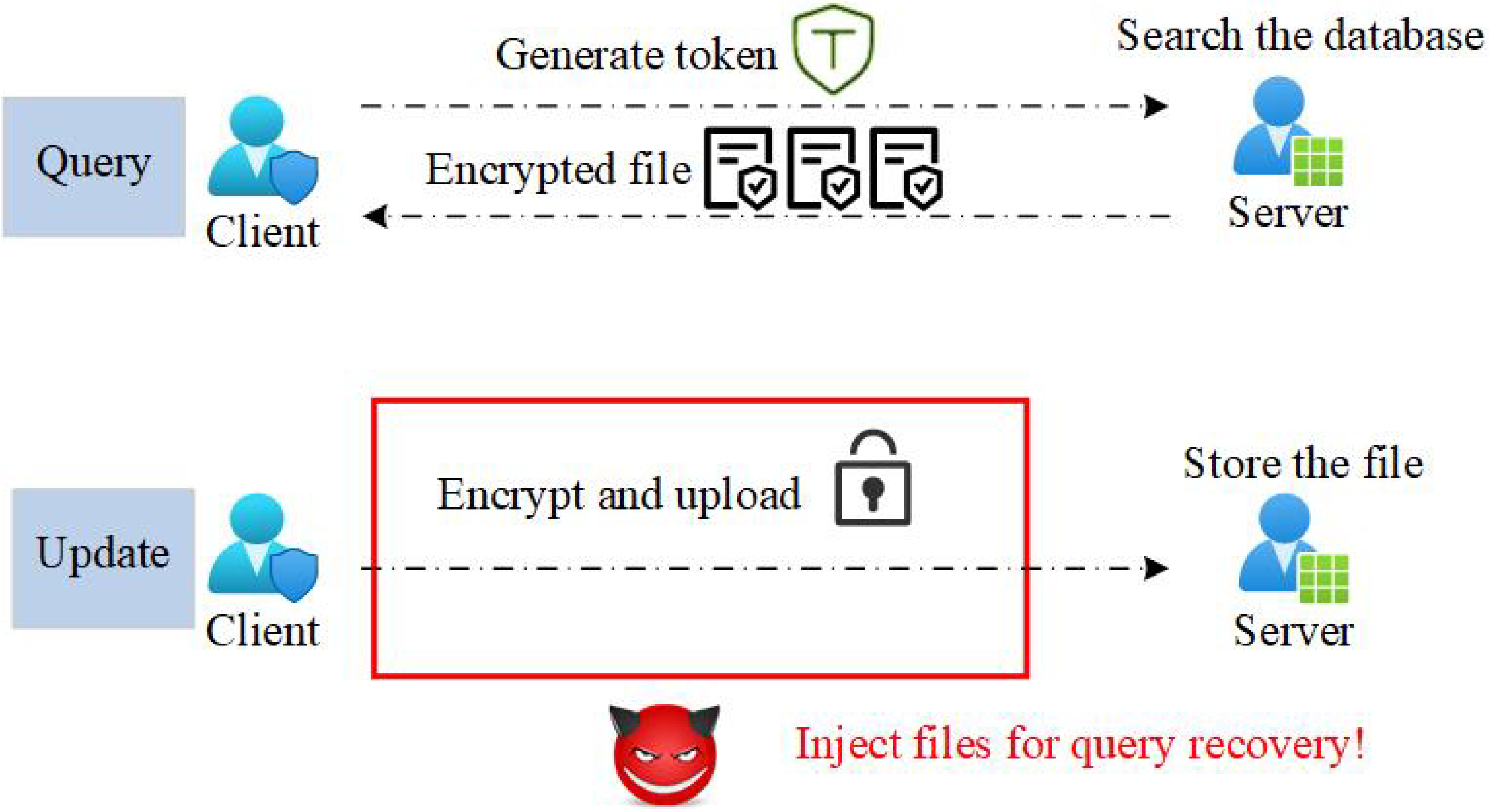
Contents

Ⅰ. Motivations

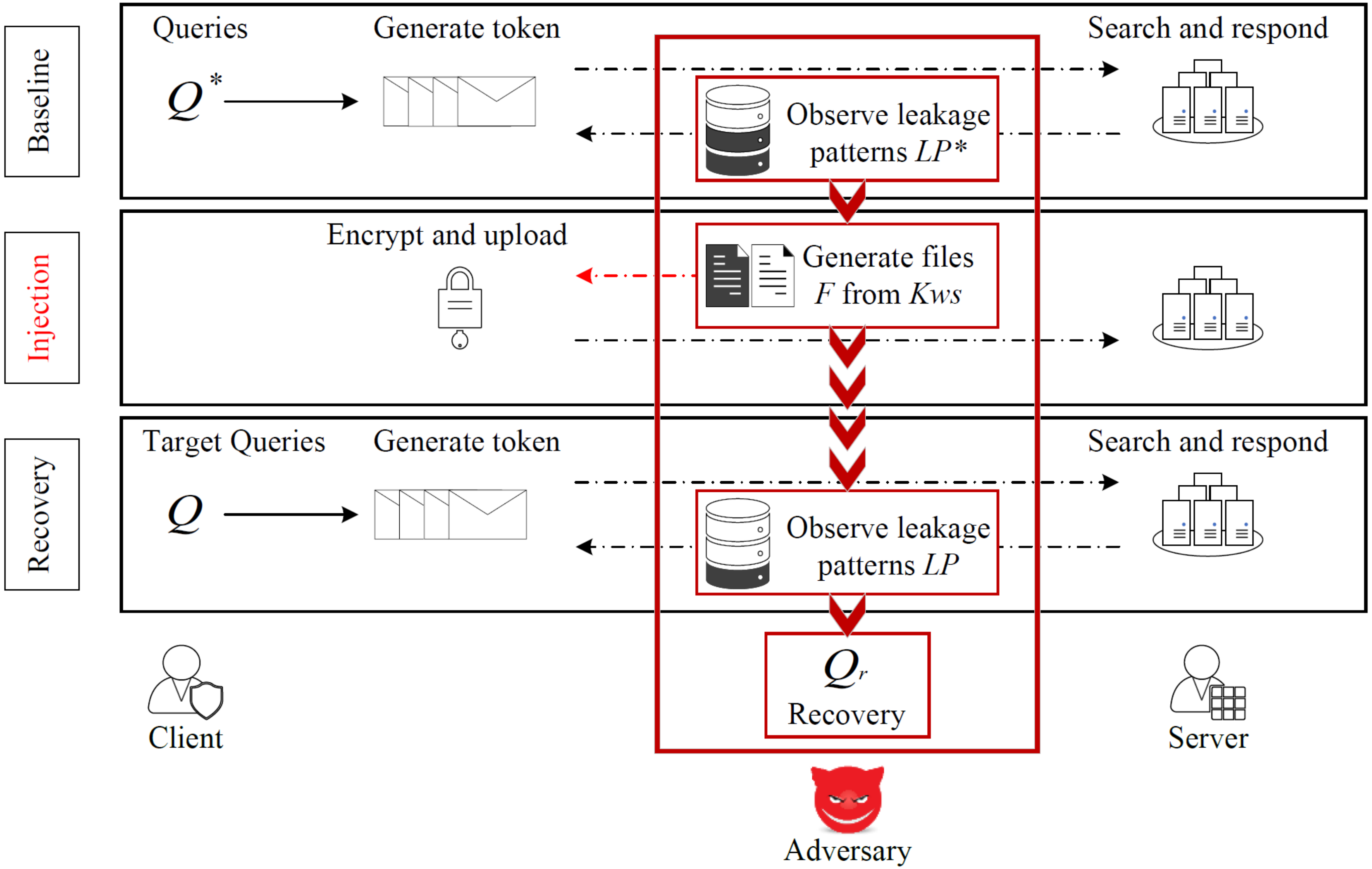
Dynamic Searchable Encryption (DSE)



Threats faced by DSE



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 rsp).

- 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.

Comparisons

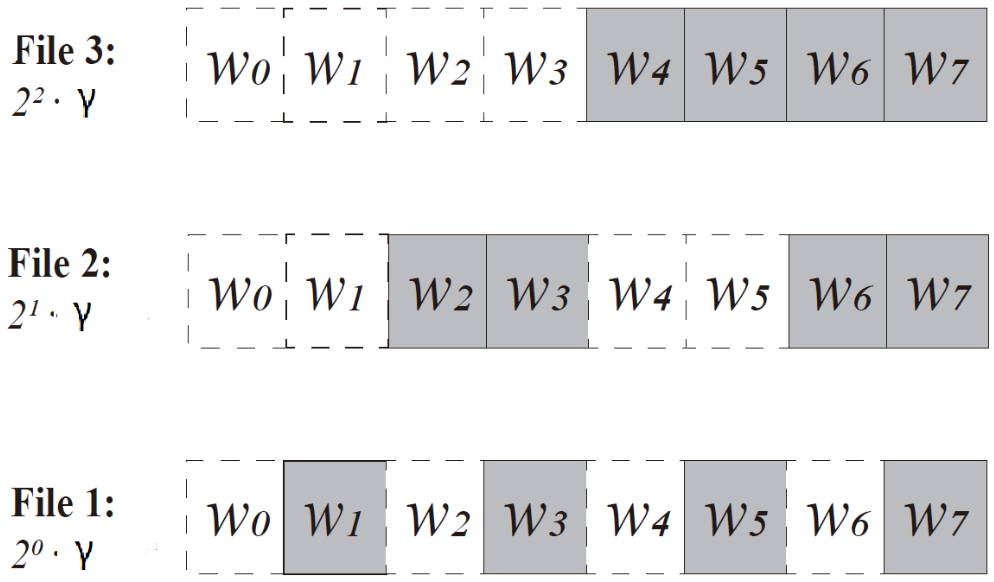
- Parameters range: #W is the number of known keywords, $m \geq 1$, $offset \gg \#W, \gamma \geq \#W/2$.
- Optimal injection length and injection size.

Attack	Injection length	Injection size
[ZKP16]	$O(\log \#W)$	$O(\#W \log \#W)$
[PWL+20] (Multiple-round attack)	$O(\#W\log \#W)$	$O(\#W^2)$
[PWL+20]* (Single-round attack)	$O(m\#W)$	$O(m\#W^2)$
[BKM20] (Decoding attack)	$O(\#W)$	$O(offset \cdot \#W^2)$
[BKM20]* (Search attack)	$O(\#W\log \#W)$	$O(\#W^2)$
Ours (BVA)	$O(\log \#W)$	$O(\gamma\#W)$
Ours (BVMA)	$O(\log \#W)$	$O(\#W \log \#W)$

BVA

Observe rsp of unknown queries before injection → **Inject** logarithmic files with different size. → **Recover** the query q with rsp_q

$\widetilde{rsp}_1$
$\widetilde{rsp}_2$
$\widetilde{rsp}_3$
… …

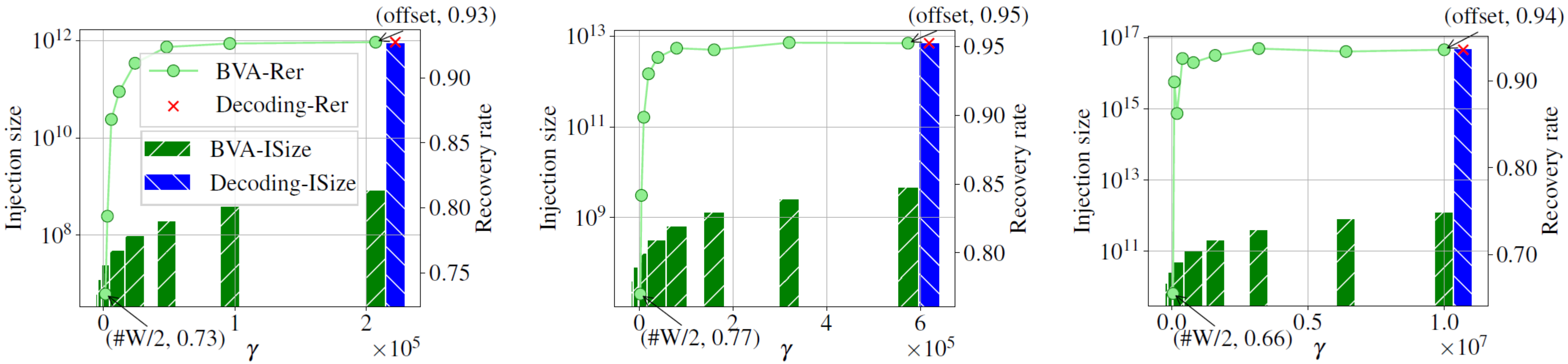


If $rsp_q - \widetilde{rsp}_l = k \cdot \gamma$, recover q as w_k.

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

Experiments on BVA

	Enron	Lucene	Wikipedia
#Keyword	3,000	5,000	100,000
#File	30,109	113,201	6,154,345
QI	GTrend [22]	GTrend	Pageview [33]
Coverage	260 weeks	260 weeks	75 months



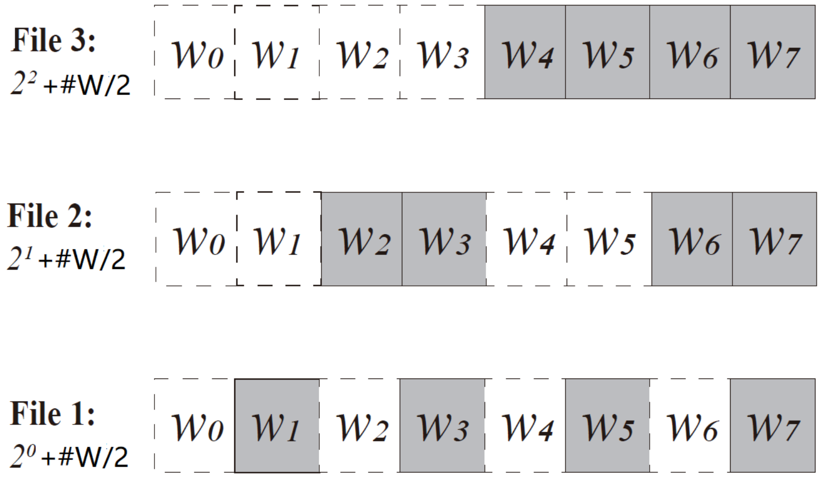
(a) Enron (b) Lucene (c) WikiPedia

- Set $\boldsymbol{\gamma = O(\#W)}$ **is enough** to achieve practical recovery, e.g., exceed 60% recovery in three datasets.
- **Less injection size** than decoding attack of [BKM20].

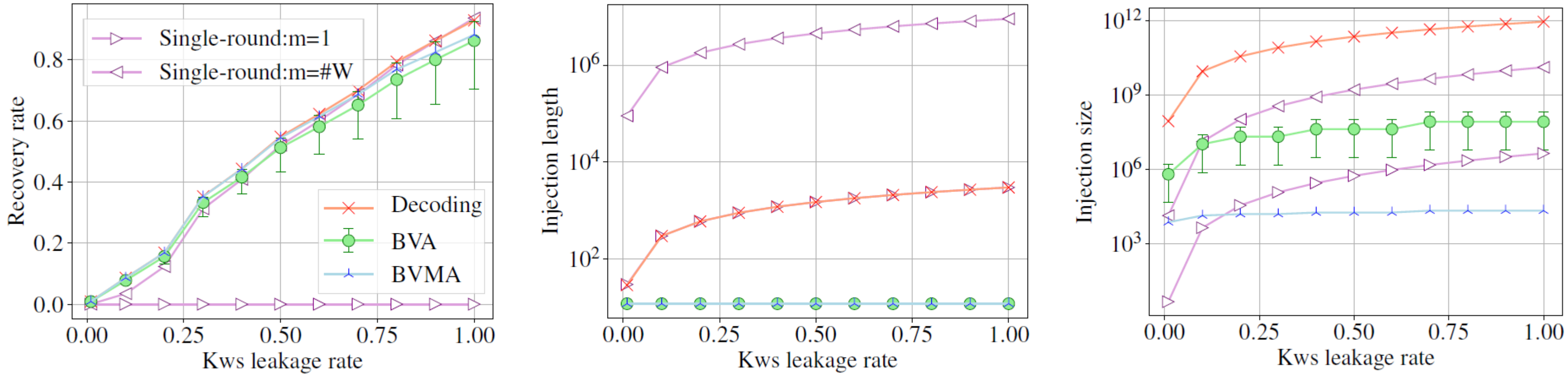
BVMA

- 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., $O(\#W \log \#W)$.



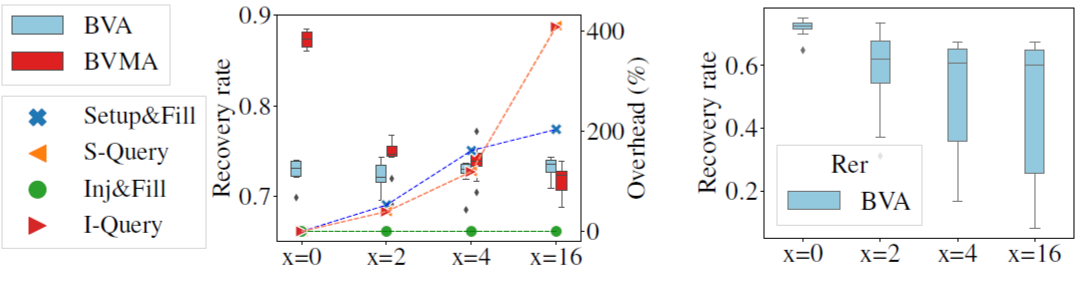
Experimental comparison



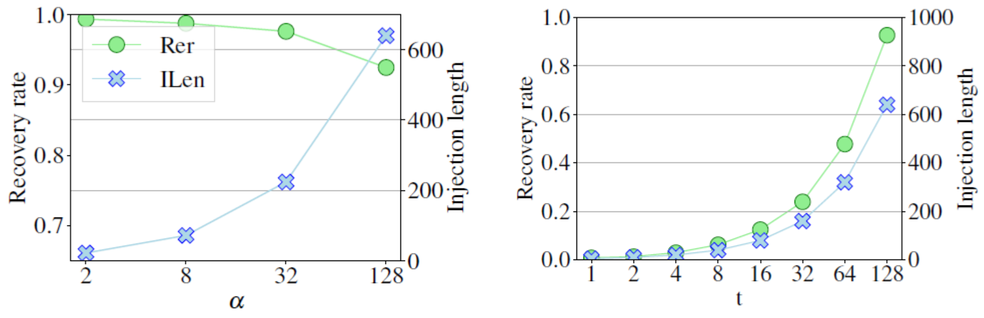
(a) Recovery accuracy (b) Injection length (c) Injection size

- Similar high recovery rate (around 80%).
- Less injection length and injection size (save >99% injection costs).

Against padding



(a) Padding. (b) Padding & ORAM.



(a) Recovery rate for different α. We set $t = \alpha$ in this case. (b) Recovery rate for different t. We set $\alpha = 128$ in this case.

Attacks against static padding (SEAL, [DPP+20])

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

- Effectively bypass these paddings.

Face client active update

Modified attack

Observe the largest rsp_{max}

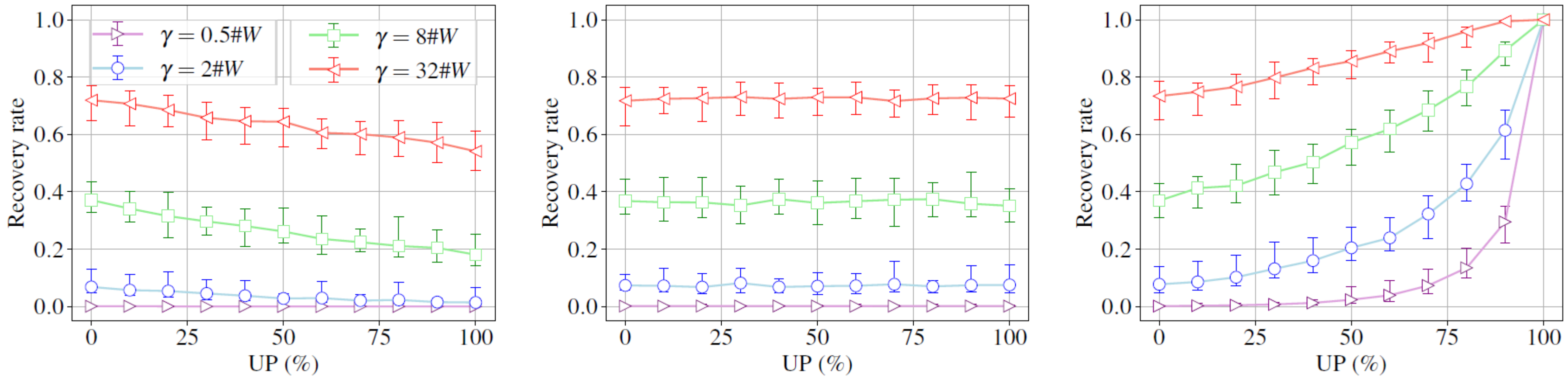
Predict the total updated size $Tsize_u$.
Set $\gamma > rsp_{max} + Tsize_u$ and inject files as BVA.

Recover the query q with rsp_q as the keyword $w_{\lfloor rsp_q/\gamma \rfloor}$

Here, we set the upper bound of γ.

A small γ is actually enough.

Evaluations against update



(a) All updates are add operations (b) Randomly selects add or delete (c) All updates are delete operations

- $\gamma = 32\#W$ 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