Offline/online attribute-based searchable encryption scheme from ideal lattices for IoT

Yang YANG, Guoyin ZHANG, Sizhao LI, Zechao LIU

Frontiers of Computer Science, DOI: 10.1007/s11704-023-3128-3

Problems & Ideas

- Problems in previous schemes:
  - Traditional searchable encryption schemes are not resistant to the threat of quantum algorithms.
  - Storing search trapdoor in previous searchable encryption schemes consumes large storage space.
  - Search, key generation and encryption algorithms are executed inefficiently.
- Ideas in our scheme:
  - Constructing attribute-based searchable encryption scheme from ideal lattices.
  - Designing a lightweight search trapdoor and an efficient search algorithm.
  - Accelerating the efficiency of execution of key generation and encryption algorithms utilizing offline/online techniques.

Main Contributions

- We propose an attribute-based searchable encryption scheme from ideal lattices (ABSEIL), which is believed to be resistant to attacks by quantum algorithms.
- The proposed scheme contains a lightweight search trapdoor and an efficient search algorithm.
- The complex arithmetic operations in the encryption and key generation algorithms of ABSEIL are pre-executed in the offline phase, and only a few arithmetic operations are involved in the online phase.
- ABSEIL enables data sharing between users without revealing any information about the underlying plaintext.

Better Performances

Table 1 Comparison of computational overhead

Scheme	KeyGen	Enc	Trapdoor	Search	Decrypt	ReKeyGen	ReEnc
Ref. [5]	T_{sp}	$12kn^2n_q + 2n$	T_{sp}	$6knn_q$	$6knn_q$	$2T_{sp}$	$72kn^2n_q^3$
Ref. [6]	$T_{pk} + T_b$	$6(k+2)n^2n_q\ell + T_{sp}$	$T_{pk} + T_b + T_{sp}$	$6nn_q(n+3\ell) + T_{ct}$	–	–	–
Ref. [7]	$T_{sp} + T_b$	$6(k+1)n^2n_q$	$2T_{sp}$	$24nn_q$	$12nn_q$	–	–
Ref. [8]	$T_{pk} + T_b$	$(k+2)n^2n_q$	–	–	$4n^2n_q^2 + T_{ct}$	$T_{pk} + T_{sp}$	$4n^2n_q^2 + T_{ct}$
Ours	$2n\tilde{n}_q\tilde{n}\ell + T_{pk} + \tilde{T}_{sl}$	$(k+2)n\tilde{n}n_q + n\tilde{n}\ell + \tilde{T}_{sp}$	$n\tilde{n}\ell + \tilde{T}_{sp}$	$2n\tilde{n}_q\tilde{n}$	$n(2\tilde{n}_q + \ell)\tilde{n} + T_{ct}$	$2T_{pk} + \tilde{T}_{sl}$	$4n\tilde{n}_q^2\tilde{n} + T_{ct}$

T_b denotes the overhead of basis delegation, $\tilde{n} = \log n$, $n_q = \lfloor \log q \rfloor$, $\tilde{n}_q = \lfloor \log_b q \rfloor$. T_{pk} and T_{ct} denote the computational overhead of Eval_{pk} and Eval_{ct}. $\tilde{T}_{sp}$ and $\tilde{T}_{sl}$ denote the computational overhead of *RSamplePre and RSampleLeft*. T_{sp} denotes the computational overhead of *SamplePre*. ℓ is the size of message space, n is the ring dimension.

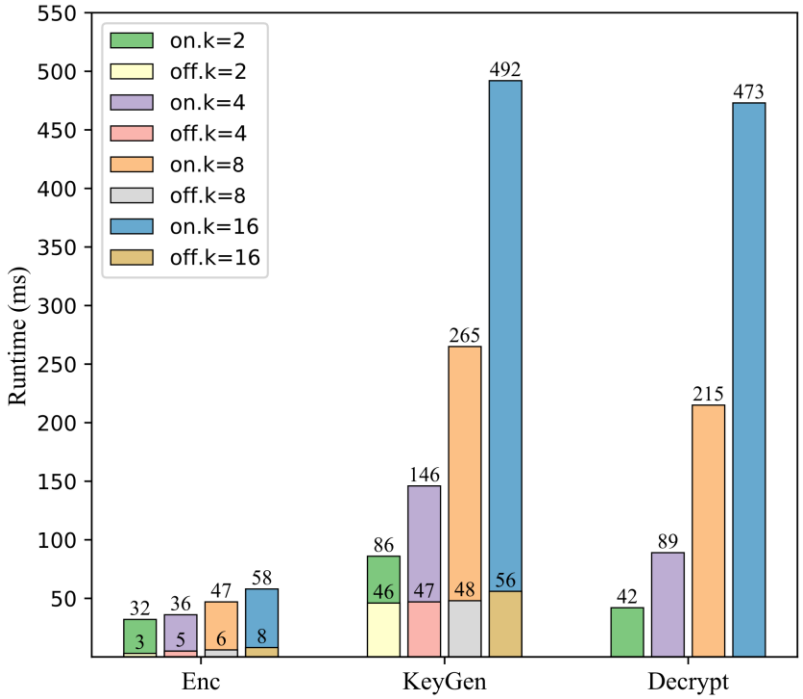

Fig.1 The runtime for *Enc*, *KeyGen and Decrypt* algorithms of ABSEIL when attribute number k grows.