SCIENCE CHINA
Information Sciences

• LETTER •

January 2024, Vol. 67, Iss. 1, 119101:1–119101:2
https://doi.org/10.1007/s11432-022-3808-7

Provably secure attribute-based authenticated encryption with keyword search from ideal lattices

Lisha YAO[†], Lin HOU[†], Jian WENG[*] & Feixiang ZHAO

College of Cyber Security, Jinan University, Guangzhou 510632, China

Received 17 October 2022/Revised 16 May 2023/Accepted 19 June 2023/Published online 18 December 2023

A public-key authenticated encryption with keyword search (PAEKS) [1, 2] system overcomes the problem of an inside adversary forging searchable ciphertexts and guessing the keyword information from the given token. This system allows the sender to encrypt and authenticate a keyword. However, traditional PAEKS schemes require the sender to specify the intended receiver in advance, losing much flexibility.

In this study, we formalize a new cryptographic primitive called attribute-based authenticated encryption with keyword search (AB-AEKS), which allows fine-grained control over search permissions by embedding access policies into secret keys. To instantiate this notion, we first provide a generic framework for constructing AB-AEKS from a signature scheme, an attribute-based keyword search (ABKS) scheme, and a non-interactive zero-knowledge (NIZK) proof system. Then we give a concrete scheme from ideal lattices for the provably secure AB-AEKS, which is inspired by a fully key-homomorphic attribute-based encryption scheme [3] and independent of the generic framework. It is worth noting that Li et al. [4] designed the first ABKS scheme from lattices with fine-grained control of the searchability. Although their construction could also resist insider keyword guessing attacks, their syntax, security model, and underlying assumptions differ from that of our AB-AEKS. This scheme can be used as a building block to construct our AB-AEKS scheme. For more related studies, please refer to Appendix A.

AB-AEKS. For a keyword space $\mathcal{W}$ and policy space $\mathcal{F}:\{0,1\}^l \rightarrow \{0,1\}$, we define an AB-AEKS scheme $\Pi = (\mathsf{Setup}, \mathsf{KeyGen}_s, \mathsf{KeyGen}_r, \mathsf{AB\text{-}AEKS}, \mathsf{Token}, \mathsf{Test})$ in following way.

$\mathsf{Setup}(1^\lambda, 1^l) \rightarrow (\mathrm{MPK}, \mathrm{MSK})$. Take as input the security parameter λ and the number of attributes l, output a master public key MPK and a master secret key MSK.

$\mathsf{KeyGen}_s(\mathrm{MPK}, \mathrm{MSK}, \mathrm{ID}) \rightarrow (\mathrm{PK}_s, \mathrm{SK}_s)$. Take as input the master public key MPK, master secret key MSK and an identity ID, output a pair of keys $(\mathrm{PK}_s, \mathrm{SK}_s)$ for the sender.

$\mathsf{KeyGen}_r(\mathrm{MPK}, \mathrm{MSK}, f) \rightarrow \mathrm{SK}_r$. Take as input the master public key MPK, master secret key MSK, and an access policy $f \in \mathcal{F}$, output a secret key SK_r for the receiver.

$\mathsf{AB\text{-}AEKS}\,(\mathrm{MPK}, \mathrm{PK}_s, \mathrm{SK}_s, \boldsymbol{x}, w) \rightarrow C$. Take as input the master public key MPK, sender's key pair $(\mathrm{PK}_s, \mathrm{SK}_s)$, an attribute vector $\boldsymbol{x} \in \{0,1\}^l$ of length l and a keyword $w \in \mathcal{W}$, output a ciphertext C.

$\mathsf{Token}\,(\mathrm{MPK}, \mathrm{SK}_r, w) \rightarrow K$. Take as input the master public key MPK, receiver's secret key SK_r and a keyword $w \in \mathcal{W}$, output a keyword token K.

$\mathsf{Test}\,(C, K) \rightarrow \{0,1\}$. Take as input the ciphertext C and token K, output 1 if $f(\boldsymbol{x}) = 0$ and C and K contain the same keyword; otherwise, output 0.

We define two security models for AB-AEKS that are indistinguishable under chosen-keyword attacks (IND-CKA) and unforgeable under insider keyword guessing attacks (UNF-IKGA), which are illustrated in Appendix B.

Generic framework. Let $\Pi_1 = (\mathsf{Setup}', \mathsf{KeyGen}', \mathsf{ABKS}', \mathsf{Token}', \mathsf{Test}')$ be an ABKS scheme, $\Pi_2 = (\mathsf{KeyGen}, \mathsf{Sign}, \mathsf{Ver})$ be a signature scheme and $\Pi_3 = (\mathsf{I}, \mathsf{P}, \mathsf{V})$ be a NIZK argument for the following NP relation: $R \overset{\mathrm{def}}{=} \{((\mathrm{mpk}, \mathrm{vk}, c), (\mathrm{PK}_s, \sigma)) : \exists\ \boldsymbol{x}, w,\ \text{s.t. } c \leftarrow \mathsf{ABKS}'(\mathrm{mpk}, \boldsymbol{x}, w) \wedge \mathsf{Ver}(\mathrm{vk}, \mathrm{PK}_s, \sigma) = 1\}$. The AB-AEKS scheme $\Pi = (\mathsf{Setup}, \mathsf{KeyGen}_s, \mathsf{KeyGen}_r, \mathsf{AB\text{-}AEKS}, \mathsf{Token}, \mathsf{Test})$ is described as follows.

$\mathsf{Setup}(1^\lambda, 1^l)$. Output $\mathrm{MPK} = (\mathrm{mpk}, \mathrm{vk}, \omega)$ and $\mathrm{MSK} = (\mathrm{msk}, \mathrm{sk})$, where $(\mathrm{mpk}, \mathrm{msk}) \leftarrow \mathsf{Setup}'(1^\lambda)$, $(\mathrm{vk}, \mathrm{sk}) \leftarrow \mathsf{KeyGen}(1^\lambda)$ and $\omega \leftarrow \mathsf{I}(1^\lambda)$.

$\mathsf{KeyGen}_s(\mathrm{MPK}, \mathrm{MSK}, \mathrm{ID})$. Upon input $\mathrm{MPK} = (\mathrm{mpk}, \mathrm{vk}, \omega)$, $\mathrm{MSK} = (\mathrm{msk}, \mathrm{sk})$ and ID, output $(\mathrm{PK}_s, \mathrm{SK}_s) = (\mathrm{ID}, \sigma)$, where $\sigma \leftarrow \mathsf{Sign}(\mathrm{sk}, \mathrm{ID})$.

$\mathsf{KeyGen}_r(\mathrm{MPK}, \mathrm{MSK}, f)$. Upon input $\mathrm{MPK} = (\mathrm{mpk}, \mathrm{vk}, \omega)$, $\mathrm{MSK} = (\mathrm{msk}, \mathrm{sk})$ and f, output $\mathrm{SK}_r = \mathrm{sk}'$, where $\mathrm{sk}' \leftarrow \mathsf{KeyGen}'(\mathrm{mpk}, \mathrm{msk}, f)$.

$\mathsf{AB\text{-}AEKS}\,(\mathrm{MPK}, \mathrm{PK}_s, \mathrm{SK}_s, \boldsymbol{x}, w)$. Upon input $\mathrm{MPK} = (\mathrm{mpk}, \mathrm{vk}, \omega)$, $(\mathrm{PK}_s, \mathrm{SK}_s) = (\mathrm{ID}, \sigma)$, $\boldsymbol{x}$ and w, output $C = (c, \pi)$, where $c \leftarrow \mathsf{ABKS}'(\mathrm{mpk}, \boldsymbol{x}, w)$ and $\pi \leftarrow \mathsf{P}(\omega, (\mathrm{mpk}, \mathrm{vk}, c), (\mathrm{ID}, \sigma))$.

$\mathsf{Token}\,(\mathrm{MPK}, \mathrm{SK}_r, w)$. Upon input $\mathrm{MPK} = (\mathrm{mpk}, \mathrm{vk}, \omega)$, $\mathrm{SK}_r = \mathrm{sk}'$ and w, output $K = k'$, where $k' \leftarrow \mathsf{Token}'(\mathrm{mpk}, \mathrm{sk}', w)$.

$\mathsf{Test}\,(C, K)$. Upon input $C = (c, \pi)$ and $K = k'$, output $0/1 \leftarrow \mathsf{Test}'(c, k')$ if $\mathsf{V}(\omega, (\mathrm{mpk}, \mathrm{vk}, c), \pi) = 1$; otherwise, output 0.

Clearly, the construction of AB-AEKS is correct if and only if underlying primitives are correct.

Theorem 1. Our general AB-AEKS scheme is secure if ABKS scheme Π_1 satisfies IND-CKA, signature scheme Π_2 satisfies existentially unforgeable under chosen-message attacks, and NIZK argument Π_3 satisfies the properties

[*] Corresponding author (email: cryptjweng@gmail.com)
[†] Yao L S and Hou L have the same contribution to this work.

of adaptive multi-theorem zero knowledge and knowledge soundness.

Proof. For the details on the proof, please refer to Appendix C.

Concrete construction. We present an AB-AEKS scheme based on the ring variant assumptions of learning with errors (ring-LWE) and inhomogeneous short integer solution (ring-ISIS) [5], which is independent of our general framework. We give the construction details of the scheme and compare the functional and storage aspects of the related schemes with ours in Appendix D.

$\mathsf{Setup}(1^\lambda, 1^l)$. The initialization algorithm performs the following operations.

(1) Run $(\boldsymbol{A}, \boldsymbol{R_A}) \leftarrow \mathsf{TrapGen}(\overline{\boldsymbol{A}}, H = 1, \sigma, q)$, where $\boldsymbol{A} = (\overline{\boldsymbol{A}}^{\mathrm{T}} | \boldsymbol{G} - \overline{\boldsymbol{A}}^{\mathrm{T}} \boldsymbol{R_A}) \in \mathcal{R}_q^{1\times m}$, $\overline{\boldsymbol{A}} \xleftarrow{R} \mathcal{R}_q^{m-k}$, $\boldsymbol{G} \in \mathcal{R}_q^{1\times k}$ and $\boldsymbol{R_A} \in \mathcal{R}_q^{(m-k)\times k}$.

(2) Select $l+1$ uniformly random row vectors $\widehat{\boldsymbol{A}}, \boldsymbol{B}_i \in \mathcal{R}_q^{1\times m}$ of ring elements for $i \in [1, l]$.

(3) Choose a random polynomial $u \in \mathcal{R}_q$, hash functions $H_1 : \{0,1\}^n \to \mathbb{Z}_q$, $H_2 : \{0,1\}^* \to \mathcal{R}_q^{1\times m}$, $H_3 : \{0,1\}^* \times \mathcal{R}_2 \to \mathcal{R}_q^{1\times m}$, and $H_4 : \mathcal{R}_q^m \times \mathcal{R}_2 \to \mathcal{R}_q$.

(4) Output a master public key $\mathrm{MPK} = (\boldsymbol{A}, \widehat{\boldsymbol{A}}, \{\boldsymbol{B}_i\}_{i=1}^l, u, H_1, H_2, H_3, H_4)$ and a master secret key $\mathrm{MSK} = \boldsymbol{R_A}$.

$\mathsf{KeyGen}_s(\mathrm{MPK}, \mathrm{MSK}, \mathrm{ID})$. The key generation algorithm for the sender performs the following operations.

(1) Compute $H_2(\mathrm{ID})$ and $\boldsymbol{F}_{\mathrm{ID}} = (\boldsymbol{A} | H_2(\mathrm{ID}))$.

(2) Run $\boldsymbol{T}_{\mathrm{ID}} \leftarrow \mathsf{DelTrap}(\boldsymbol{F}_{\mathrm{ID}}, \boldsymbol{R_A}, H' = 1, s')$.

(3) Output a sender's public key $\mathrm{PK}_s = \mathrm{ID}$ and secret key $\mathrm{SK}_s = \boldsymbol{T}_{\mathrm{ID}}$.

$\mathsf{KeyGen}_r(\mathrm{MPK}, \mathrm{MSK}, f)$. The key generation algorithm for the receiver performs the following operations.

(1) Compute $\boldsymbol{B}_f = \mathsf{EVAL}_{\mathrm{MPK}}(f, (\boldsymbol{B}_1, \ldots, \boldsymbol{B}_l))$.

(2) Run $\boldsymbol{T}_f \leftarrow \mathsf{DelTrap}(\boldsymbol{A} | \boldsymbol{B}_f, \boldsymbol{R_A}, H' = 1, s')$.

(3) Output a receiver's secret key $\mathrm{SK}_r = \boldsymbol{T}_f$.

$\mathsf{AB\text{-}AEKS}(\mathrm{MPK}, \mathrm{PK}_s, \mathrm{SK}_s, \boldsymbol{x}, w)$. The encryption algorithm performs the following operations.

(1) Compute a tag $H_1(w)$ and let $\boldsymbol{F}_w = \widehat{\boldsymbol{A}} + (\boldsymbol{0} | H_1(w)\boldsymbol{G})$, where the zero vector has dimension $m-k$.

(2) Choose two random polynomials $s \xleftarrow{R} \mathcal{R}_q$, $b \xleftarrow{R} \mathcal{R}_2$, and error terms $\boldsymbol{e}_0 \xleftarrow{R} D_{\mathcal{R}^{m-k},\tau}$, $\boldsymbol{e}_1 \xleftarrow{R} D_{\mathcal{R}^k,\gamma}$, $e \xleftarrow{R} D_{\mathcal{R},\tau}$, $\boldsymbol{e}_A \xleftarrow{R} D_{\mathcal{R}^m,\sigma}$.

(3) Choose arbitrarily $\boldsymbol{S}_i \xleftarrow{R} \{-1,1\}^{m\times m}$, compute $\boldsymbol{e}_i = \boldsymbol{S}_i^{\mathrm{T}} \boldsymbol{e}_A$ for $i \in [1, l]$.

(4) For $i \in [1, l]$, compute $\boldsymbol{C}_1 = \boldsymbol{F}_w^{\mathrm{T}} s + (\boldsymbol{e}_0^{\mathrm{T}} | \boldsymbol{e}_1^{\mathrm{T}})^{\mathrm{T}}, C_2 = u \cdot s + e + b \left\lfloor \frac{q}{2} \right\rfloor, \boldsymbol{C}_3 = \boldsymbol{A}^{\mathrm{T}} s + \boldsymbol{e}_A, \boldsymbol{C}_i = (\boldsymbol{B}_i + (\boldsymbol{0} | x_i \boldsymbol{G}))^{\mathrm{T}} s + \boldsymbol{e}_i$.

(5) Compute $H_2(\mathrm{ID})$ and $H_3(\mathrm{ID}, b)$, set $\boldsymbol{F}_{\mathrm{ID},b} = (\boldsymbol{A} | H_2(\mathrm{ID}) | H_3(\mathrm{ID}, b))$.

(6) Run $\boldsymbol{T}_{\mathrm{ID},b} \leftarrow \mathsf{DelTrap}(\boldsymbol{F}_{\mathrm{ID},b}, \boldsymbol{T}_{\mathrm{ID}}, H' = 1, s')$.

(7) Let $h = H_4(\boldsymbol{C}_1, b)$, capture $\boldsymbol{v} \leftarrow \mathsf{SamplePre}(\boldsymbol{F}_{\mathrm{ID},b}, \boldsymbol{T}_{\mathrm{ID},b}, H' = 1, h, \zeta, \alpha, \sigma)$ such that $\boldsymbol{F}_{\mathrm{ID},b} \boldsymbol{v} = h$.

(8) Output a ciphertext $C = (\boldsymbol{C}_1, C_2, \boldsymbol{C}_3, \{\boldsymbol{C}_i\}_{i=1}^l, \boldsymbol{v})$.

$\mathsf{Token}(\mathrm{MPK}, \mathrm{SK}_r, w)$. The token generation algorithm performs the following operations.

(1) The procedure for calculating $H_1(w)$ and $\boldsymbol{F}_w$ is the same as the above AB-AEKS algorithm.

(2) Choose randomly $\boldsymbol{p} \xleftarrow{R} D_{\mathcal{R}^m,\sigma}$, let $t = u - \boldsymbol{F}_w \boldsymbol{p}$.

(3) Run $\boldsymbol{\varphi}_f \leftarrow \mathsf{SamplePre}(\boldsymbol{A} | \boldsymbol{B}_f, \boldsymbol{T}_f, H' = 1, t, \zeta, \alpha, \sigma)$ such that $(\boldsymbol{A} | \boldsymbol{B}_f) \boldsymbol{\varphi}_f = t$.

(4) Output a keyword token $K = (\boldsymbol{\varphi}_f^{\mathrm{T}} | \boldsymbol{p}^{\mathrm{T}})$.

$\mathsf{Test}(C, K)$. Parse the ciphertext C related to an attribute vector $\boldsymbol{x}$ and the keyword token K connected with a policy circuit f. If $f(\boldsymbol{x}) \neq 0$, the test algorithm returns 0. Otherwise, it performs the following operations.

(1) Set $\boldsymbol{C}_f = \mathsf{EVAL}_C(f, (x_i, \boldsymbol{B}_i, \boldsymbol{C}_i)_{i=1}^l)$.

(2) Compute $d = C_2 - (\boldsymbol{\varphi}_f^{\mathrm{T}} | \boldsymbol{p}^{\mathrm{T}})(\boldsymbol{C}_3 | \boldsymbol{C}_f | \boldsymbol{C}_1)$.

Let $b_i = 1$ if each d_i is closer to $\left\lfloor \frac{q}{2} \right\rfloor$ than to 0; otherwise, $b_i = 0$. For $i \in [1, l]$, we have that $b = \sum_{i=1}^l b_i x^i \in \mathcal{R}_2$.

(3) Compute $H_3(\mathrm{ID}, b)$ and $h = H_4(\boldsymbol{C}_1, b)$ according to b, let $\boldsymbol{F}_{\mathrm{ID},b} = (\boldsymbol{A} | H_2(\mathrm{ID}) | H_3(\mathrm{ID}, b))$, then check whether $\boldsymbol{F}_{\mathrm{ID},b} \boldsymbol{v} = h \mod q$ and $0 < \|\boldsymbol{v}\| \leqslant t\zeta\sqrt{3mn}$ are true. If these two conditions hold, the test algorithm returns 1, else returns 0.

Theorem 2. If the hardness of Ring-LWE$_{n,q,m,D_{\mathcal{R},\tau}}$ and Ring-ISIS$_{q,m,\beta}$ problems holds, our proposed AB-AEKS scheme is proved to be selective-attribute IND-CKA and UNF-IKGA security in the random oracle model, respectively.

Proof. Please see Appendix E for detailed proof.

Acknowledgements This work was supported by National Key Research and Development Plan of China (Grant No. 2020YFB1005600), Major Program of Guangdong Basic and Applied Research Project (Grant No. 2019B030302008), National Natural Science Foundation of China (Grant No. 61825203), Guangdong Provincial Science and Technology Project (Grant Nos. 2017B010111005, 2021A0505030033), National Joint Engineering Research Center of Network Security Detection and Protection Technology and Guangdong Key Laboratory of Data Security and Privacy Preserving. Thanks also to the anonymous reviewers for their useful comments.

Supporting information Appendixes A–E. The supporting information is available online at info.scichina.com and link.springer.com. The supporting materials are published as submitted, without typesetting or editing. The responsibility for scientific accuracy and content remains entirely with the authors.

References

1 Huang Q, Li H B. An efficient public-key searchable encryption scheme secure against inside keyword guessing attacks. Inf Sci, 2017, 403: 1–14

2 Liu Z Y, Tseng, Y F, Raylin T, et al. Public-key authenticated encryption with keyword search: cryptanalysis, enhanced security, and quantum-resistant instantiation. In: Proceeding of ACM Asia Conference on Computer and Communications Security, Nagasaki, 2022. 423–436

3 Dan B, Craig G, Sergey G, et al. Fully key-homomorphic encryption, arithmetic circuit ABE and compact garbled circuits. In: Proceeding of the 33rd Annual International Conference on the Theory and Applications of Cryptographic Techniques, Copenhagen, 2014. 533–556

4 Li J, Ma M M, Zhang J, et al. Attribute-based keyword search from lattices. In: Proceeding of the 15th International Conference on Information Security and Cryptology, Nanjing, 2019. 66–85

5 Peikert C. A decade of lattice cryptography. FNT Theor Comput Sci, 2016, 10: 283–424