IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 16 January 2024, accepted 11 June 2024, date of publication 1 July 2024, date of current version 10 July 2024.

*Digital Object Identifier 10.1109/ACCESS.2024.3421333*

COMMENT

# Comment on "Expressive Public-Key Encryption With Keyword Search: Generic Construction From KP-ABE and an Efficient Scheme Over Prime-Order Groups"

**KOON-MING CHAN, SWEE-HUAY HENG, SYH-YUAN TAN, AND SHING-CHIANG TAN**
Faculty of Information Science and Technology, Multimedia University, Malacca 75450, Malaysia
Corresponding authors: Koon-Ming Chan (koonming1996@gmail.com) and Swee-Huay Heng (shheng@mmu.edu.my)

This work was supported by the Telekom Malaysia Research and Development under Grant RDTC/221045.

**ABSTRACT** The public key encryption with keyword search (PEKS) scheme is a cryptographic primitive that allows a cloud server to search for a ciphertext without knowing the corresponding keyword used in the search. An expressive PEKS scheme is a variant of the PEKS scheme that supports conjunctive and disjunctive searches (expressive search). Utilising the expressive properties of an attribute-based encryption (ABE) scheme, most of the expressive PEKS schemes can be constructed from an ABE scheme. In this paper, we first give a brief review of the transformed expressive PEKS scheme by Shen et al. in 2019. Then, we present a keyword guessing attack on Shen et al.'s transformed expressive PEKS scheme and show that an adversary can correctly guess the supposedly hidden keyword.

**INDEX TERMS** Expressive PEKS, keyword guessing, key-policy ABE.

## I. INTRODUCTION

With the proliferation of cloud infrastructure, many applications have moved towards outsourcing their data to the cloud service provider. When outsourcing data to a third party, user privacy becomes a crucial factor that must be considered. The most traditional way of protecting data privacy is by turning the data into ciphertext through encryption; however, the ciphertext will extensively restrict the capability of performing the search. In order to search through ciphertext, Boneh et al. [1] put forward the notion of public key encryption with keyword search (PEKS) to overcome this challenge.

The PEKS scheme proposed by Boneh et al. [1] only supports a single keyword search , such as `title = sales report`, which makes it less practical in terms of its functionality. To further improve on this, a variant of the PEKS scheme called the expressive PEKS scheme was proposed. In an expressive PEKS scheme, the user is allowed to search for multiple keywords in a conjunctive and disjunctive manner. For instance, a conjunctive keyword search can be `title = sales report` $\wedge$ `sender = richard` while a disjunctive keyword search can be `title = sales report` $\vee$ `title = purchasing report`. Such keyword expressivity greatly improves the search flexibility as compared to the PEKS scheme. Fig. 1 shows the general structure of a PEKS scheme. Note that a general PEKS scheme only supports a single keyword search thus the trapdoor is a simple single keyword trapdoor. The difference between a general PEKS scheme and an expressive PEKS scheme is in their search capability. In an expressive PEKS scheme, it allows multiple keyword search in a conjunctive and disjunctive manner and thus the trapdoor would be more complex. It is generally presented in the form of a tree as depicted in Fig. 1. Lai et al. [2] proposed the first expressive PEKS based on the attribute-based encryption (ABE) proposed by Lewko et al. [3].

A keyword guessing attack (KGA) is an attack that allows the adversary to exhaustively guess the keyword encrypted in the ciphertext or trapdoor. A successful KGA

The associate editor coordinating the review of this manuscript and approving it for publication was Jenny Mahoney.

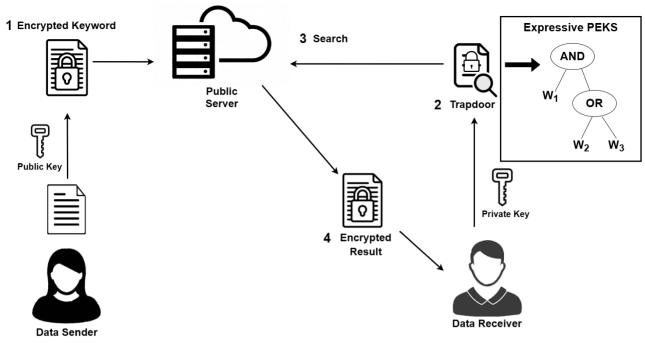


**FIGURE 1.** PEKS scheme [4].

reveals the supposedly confidential keyword (e.g. `title = retrenchment`) in a search query, which indirectly reveals the content of the encrypted data to search for. There are two kinds of keyword guessing attacks (KGA) which are offline KGA and online KGA [4]. In offline KGA, the adversary can obtain the searchable ciphertext or the trapdoor through a public channel to perform keyword guessing. Whereas in an online KGA, the adversary first uploads the specially crafted ciphertexts to the server and then eavesdrops on the public channel. The adversary can determine the keyword if the user queried the crafted ciphertext. These kinds of attacks are the main threat to any searchable encryption scheme. Many efforts have shown that designing and constructing a secure PEKS scheme against KGA is a challenging work [5], [6], [7].

### A. CONTRIBUTIONS

Shen et al. [8] proposed a generic approach to construct a secure expressive PEKS scheme from a secure key-policy attribute-based encryption (KP-ABE) scheme. More precisely, if the underlying KP-ABE achieves anonymity and ciphertext indistinguishability under chosen plaintext attack (ANO-IND-CPA), the resulting expressive PEKS scheme achieves ciphertext indistinguishability under chosen keyword attack (IND-CKA). In this paper, we show that the proposed scheme by Shen et al. is vulnerable to offline keyword-guessing attack which is a weaker form of chosen keyword attack.

### B. PAPER ORGANIZATION

Section II briefly lists some important definitions and security notion. In Section III, we give a brief review of Shen et al.'s transformed expressive PEKS scheme. In Section IV, we show how an adversary can perform the offline KGA on the transformed expressive PEKS scheme. We conclude the paper in Section V.

### C. RELATED WORK

Attribute-based encryption (ABE) is a cryptographic primitive that can provide fine-grained access control to the data user. There is an access policy and attribute set in every ABE scheme, the access policy states the rules that must be fulfilled in order to access the encrypted file and the attribute set is a tuple of entities that specifies who has the right to access. An ABE scheme has two variants: the key-policy ABE (KP-ABE) scheme [9] and the ciphertext-policy ABE (CP-ABE) scheme [10]. In a KP-ABE scheme, the key is associated with the access policy and the data is encrypted with a set of attributes. The data user can decrypt the ciphertext if and only if the access policy in its key fulfils the attribute set in the ciphertext. Whereas in a CP-ABE scheme, it is vice versa to the KP-ABE scheme. The key in a CP-ABE scheme is associated with the attribute set and the data is encrypted with an access policy.

With the fine-grained access control mechanism offered by the ABE scheme, many works have focused on constructing a secure and efficient expressive PEKS scheme from the ABE scheme. Lai et al. [2] proposed the first expressive PEKS scheme based on the KP-ABE scheme. Their proposed scheme is proven secure in the standard model but their scheme construction is based on composite order group which will lead to a higher computational cost and the ciphertext length is linear to the number of keywords. Lv et al. [11] proposed an expressive PEKS scheme that supports conjunctive, disjunctive and negation search (e.g. $\neg$(`sender = richard`)) but their scheme construction is also in composite order group and hence inefficient. Cui et al. [12] proposed the first expressive PEKS scheme in prime order groups based on a KP-ABE scheme but the scheme construction showed high communication and computational costs. Zhu et al. [13] proposed an expressive PEKS scheme that allows testing whether the ciphertext contains the same keywords that are encrypted by different public keys. Wang et al. [14] proposed an expressive PEKS scheme that supports attribute revocation. Peng et al. [15] proposed an expressive PEKS scheme that dynamically updates the access policy. Meng et al. [16] stressed that the main drawbacks of the existing expressive PEKS scheme are the pairing cost in the test phase and the ciphertext length increases linearly with the number of keywords. To tackle the problem, they proposed an expressive PEKS scheme based on the KP-ABE scheme in a prime order asymmetric bilinear group that offered constant ciphertext length and constant pairing cost. Shen et al. [8] proposed a more efficient expressive PEKS scheme in a prime order group based on the KP-ABE scheme in terms of the computational cost.

Wang et al. [17] proposed the first expressive PEKS scheme based on the CP-ABE scheme. Yin et al. [18] proposed an expressive PEKS scheme based on the CP-ABE scheme that allows the data owner to perform fine-grained search authorisation for a data user. Trinh [19] proposed an expressive PEKS scheme based on the CP-ABE scheme that not only supports expressive search but also supports multi-writer and multi-reader. Wu et al. [20] also proposed an expressive PEKS scheme based on the CP-ABE scheme but their scheme construction is in a prime order group which is more efficient. Huang et al. [21] proposed an expressive

PEKS scheme based on the CP-ABE scheme that supports conjunctive, disjunctive, negation, and threshold operators.

## II. PRELIMINARY

### A. BILINEAR PAIRING

Define $G$ as a group of prime order $p$. A bilinear pairing over $G$ and $G_T$ is defined as $e : G \times G \rightarrow G_T$. A bilinear pairing $e$ must satisfy the following properties:

1) Bilinear: for all $g \in G$ and all $a, b \in Z_p$, $e(g^a, g^b) = e(g, g)^{ab}$.
2) Non-degenerate: $e(g, g) \neq 1$.
3) Computable: For any $g_1, g_2 \in G, e(g_1, g_2)$ can be computed efficiently.

### B. DEFINITION OF PEKS SCHEME

A PEKS scheme consists of the following algorithms [1]:

**KeyGen**($k$): This algorithm takes in a security parameter $k$ and outputs a pair of public and private key $(pk, sk)$ for the user.

**PEKS**($pk, w$): This algorithm is run by the sender. With the input of a public key $pk$ and a keyword $w$, it outputs a searchable ciphertext $C_w$.

**Trapdoor**($sk, w$): This algorithm is run by the receiver. With the input of a private key $sk$ and a keyword $w$, it outputs a trapdoor $T_w$.

**Test**($T_w, C_{w'}$): This algorithm is run by the server. With the input of a trapdoor $T_{w'}$ and a searchable ciphertext $C_w$, it outputs 1 if $w = w'$ else 0.

### C. DEFINITION OF KP-ABE SCHEME

A KP-ABE scheme consists of the following algorithms [9]:

**Setup**($k$): This algorithm is run by a trusted central authority (TCA). It inputs a security parameter $k$ and outputs a master key $MK$ and the public parameter $PP$.

**KeyGen**($PP, MK, AS$): This algorithm is run by a TCA. It takes in the public parameter $PP$, the master key $MK$, and an access structure $AS$ as an input and outputs a private key associated with the access structure $sk_{AS}$ for a user.

**Encrypt**($PP, M, ATS$): This algorithm is run by a sender. It takes in the public parameter $PP$, a message $M$, and an attribute set $ATS$ as the input and outputs a ciphertext associated with the attribute set $CT_{ATS}$.

**Decrypt**($PP, sk_{AS}, CT_{ATS}$): This algorithm is run by a receiver. It takes in the public parameter $PP$, a private key $sk_{AS}$, and a ciphertext $CT_{ATS}$ as the input and outputs the message $M$ if and only if the attribute set $ATS$ associated with the ciphertext $CT_{ATS}$ fulfilled the access structure $AS$ associated with the private key $sk_{AS}$.

### D. DEFINITION OF EXPRESSIVE PEKS SCHEME

An expressive PEKS scheme consists of the following algorithms [2]:

**KeyGen**($k$): This algorithm takes in a security parameter $k$ and outputs a pair of public and private key $(pk, sk)$ for the user.

**Encrypt**($pk, WS$): This algorithm is run by the sender. With the inputs of a public key $pk$ and a keyword set $WS$, it outputs a searchable ciphertext $C_{WS}$.

**Trapdoor**($pk, sk, P$): This algorithm is run by the receiver. It takes in a public key $pk$, a private key $sk$, and a search predicate $P$ as input and outputs a trapdoor $T_P$ of predicate $P$.

**Test**($pk, T_P, C_{WS}$): This algorithm is run by the server. It takes in a public key $pk$, a trapdoor $T_P$, and a searchable ciphertext $C_{WS}$ as input and outputs 1 if the keyword set $WS$ associated with the ciphertext $C_{WS}$ fulfilled the predicate $P$ associated with the trapdoor $T_P$ else 0.

### E. SECURITY MODEL OF EXPRESSIVE PEKS

The security of the expressive PEKS scheme is defined by the security notion of indistinguishability against chosen keyword attacks (IND-CKA) [8]. The security notion guaranteed that an expressive PEKS scheme should not leak any information about the keyword set $WS$ in ciphertext $C_{WS}$. The security model is defined through the following adversarial game:

1) **Initiate**. Adversary $A$ chooses two challenge keyword sets $WS_0$, $WS_1$ with the same length.
2) **Setup**. The challenger $Ch$ runs the $KeyGen$ algorithm to generate $pk$ and $sk$ and only gives the $pk$ to the adversary.
3) **Phase 1**. The adversary $A$ can query for the trapdoor $T_P$ for any predicate $P$ with the challenger $Ch$ as long as $WS_0$ and $WS_1$ do not fulfil $P$.
4) **Challenge**. The challenger $Ch$ randomly chooses a bit $b \in \{0, 1\}$ and computes the challenge ciphertext $C_{WS_b}$ and sends to the adversary.
5) **Phase 2**. The process is the same as Phase 1.
6) **Guess**. The adversary outputs its answer $b' \in \{0, 1\}$. The adversary wins the game if $b' = b$.

We define adversary $A's$ advantage as follows:

$$Adv_A^{IND-CKA} = |Pr[b = b'] - \frac{1}{2}|. \quad (1)$$

## III. REVIEW OF Shen et al.'s SCHEME

In this section, we give a brief review of Shen et al.'s transformed expressive PEKS scheme [8]. Shen et al. [8] proposed a generic transformation where an anonymous KP-ABE scheme can be transformed into an expressive PEKS scheme. However, we note that their transformed expressive PEKS scheme is vulnerable to KGA.

Shen et al.'s transformed expressive PEKS scheme [8] consists of the following algorithms:

- **KeyGen**($k$): This algorithm takes a security parameter $k$ as an input. It outputs the following elements:
  1) A bilinear group $(G, G_T)$ of order prime $p$.
  2) A bilinear map $e : G \times G \rightarrow G_T$.
  3) Choose a random generator $g \in G$, three random elements $u, h, w \in G$, and a random number $\alpha \in Z_p$.

4) Set public key $pk = (p, G, G_T, e, g, u, h, w, e(g, g)^{\alpha})$ and a private key $sk = \alpha$.

- **Trapdoor**$(pk, sk, P)$: This algorithm is executed by the receiver by inputting the public key $pk$, the private key $sk$, and a search predicate $P$. It outputs the following elements:
  1) Generate an access structure $AS$ from the search predicate $P$.
  2) Choose a vector $\overrightarrow{y} = (\alpha, y_2, \cdots, y_n)^{\perp}$ where $y_2, \cdots, y_n \in Z_p$.
  3) Compute $\overrightarrow{\lambda} = (\lambda_1, \lambda_2, \cdots, \lambda_l)^{\perp} = MA\overrightarrow{y}$, where $MA$ is the share-generating matrix in $AS$.
  4) Pick $l$ random numbers $t_1, t_2, \cdots, t_l \in Z_p$.
  5) For every $\tau \in [l]$, calculate $K_{\tau,0} = g^{\lambda_\tau} w^{t_\tau}$, $K_{\tau,1} = (u^{\rho(\tau)}h)^{-t_\tau}$, and $K_{\tau,2} = g^{t_\tau}$.
  6) Output trapdoor $T_P = (MA, \{K_{\tau,0}, K_{\tau,1}, K_{\tau,2}\}_{\tau\in[l]})$.
- **Encrypt**$(pk, WS)$: This algorithm is executed by the sender by inputting the public key $pk$, and a keyword set $WS$. It outputs the following elements:
  1) Choose $k + 1$ random numbers $s, r_1, r_2, \cdots, r_k \in Z_p$.
  2) Calculate $C = e(g, g)^{\alpha s}$, $C_0 = g^s$, for every $\tau \in [k]$ compute $C_{\tau,1} = g^{r_\tau}$ and $C_{\tau,2} = (u^{W_r}h)w^{-s}$.
  3) Output the searchable ciphertext $SE_{WS} = (C, C_0, \{C_{\tau,1}, C_{\tau,2}\}_{\tau\in[k]})$.
- **Test**$(pk, SE_{WS}, T_P)$: This algorithm is executed by the server by inputting the public key $pk$, a searchable ciphertext $SE_{WS}$, and a trapdoor $T_P$. Let $I_{AS}$ be the minimum subset meeting $AS$ generated from $P$. The server calculates $I_{AS}$ from the access structure $MA$ and checks if there is a $I \in I_{AS}$ satisfying

$$C = \prod_{i\in I}(e(C_0, K_{i,0})e(C_{\tau,1}, K_{i,1})e(C_{\tau,2}, K_{i,2}))^{\omega_i} \quad (2)$$

where $\Sigma_{i\in I}\omega_i MA_i = (1, 0, \cdots, 0)$ and $MA_i$ is the $i^{th}$ row of $MA$. It outputs 0 if no element in $I_{AS}$ satisfies the above equation or otherwise 1.

## IV. THE KEYWORD GUESSING ATTACK

In this section, we show how an adversary $A$ can mount an offline KGA on Shen et al.'s expressive PEKS scheme [8]. Recall that in an offline KGA, the adversary can obtain the searchable ciphertext or the trapdoor through a public channel to perform keyword guessing. An offline KGA is considered a more critical attack as compared to an online KGA because offline KGA is relatively easier to mount on a searchable encryption scheme. An adversary $A$ can perform a keyword-guessing attack as follows:

1) When $A$ captures a searchable ciphertext $SE_{WS} = (C, C_0, \{C_{\tau,1}, C_{\tau,2}\}_{\tau\in[k]})$, $A$ uses $C_{\tau,2}$ to compute $J = e(C_{\tau,2}, g)$. $A$ will get

$$\begin{aligned} J &= e(C_{\tau,2}, g) \\ &= e((u^{W_r}h)w^{-s}, g) \\ &= e((u^{W_r}h), g)e(w^{-s}, g) \end{aligned} \quad (3)$$

2) $A$ guesses a keyword $W'$ and computes

$$J' = e((u^{W'}h), g)e(w, C_0)^{-1} \quad (4)$$

where $C_0 = g^s$.

3) If $J = J'$, then $A$ guesses correctly the keyword embedded in the ciphertext. Otherwise, go back to step 2.

The correctness of equality for equations (3) and (4) when $W_r = W'$ is as below:

$$\begin{aligned} J &= J' \\ e((u^{W_r}h), g)e(w^{-s}, g) &= e((u^{W'}h), g)e(w, g^s)^{-1} \\ e(u^{W_r}, g)e(h, g)e(w^{-s}, g) &= e(u^{W'}, g)e(h, g)e(w, g^s)^{-1} \\ e(u, g)^{W_r}e(h, g)e(w, g)^{-s} &= e(u, g)^{W'}e(h, g)e(w, g)^{-s} \end{aligned}$$

It is also easy to see that the attack can be run in polynomial time as it has a complexity of $O(N)$ where $N$ is the keyword space.

### A. DISCUSSION

Shen et al. showed that if a KP-ABE scheme is anonymous under the chosen plaintext attack (ANO-IND-CPA) then the transformed expressive PEKS scheme is IND-CKA secure. As the keywords in expressive PEKS are the attributes in KP-ABE, the keyword guessing attack above can be similarly applied to Shen et al.'s KP-ABE scheme in breaking the ANO-IND-CPA security. We highlight that our results do not falsify Shen et al's transformation technique, but the security of their instantiated KP-ABE scheme and the subsequently transformed expressive PEKS scheme. Also, it is already known that the security property of IND-CKA alone is not sufficient to resist the KGA based on past research [7], [22], [23].

## V. CONCLUSION

We mounted an offline KGA on Shen et al.'s expressive PEKS scheme which allows the adversary to correctly guess the keyword encrypted inside the ciphertext. We would like to emphasise that it is important for a searchable encryption scheme such as PEKS or expressive PEKS to be secure against KGA as it is regarded as the basic security requirement.

## REFERENCES

[1] D. Boneh, G. Di. Crescenzo, R. Ostrovsky, and G. Persiano, "Public key encryption with keyword search," in *Proc. Int. Conf. Theory Appl. Cryptograph. Techn.*, Interlaken, Switzerland. Berlin, Germany: Springer, 2004, pp. 506–522.
[2] J. Lai, X. Zhou, R. H. Deng, Y. Li, and K. Chen, "Expressive search on encrypted data," in *Proc. 8th ACM SIGSAC Symp. Inf., Comput. Commun. Secur.*, May 2013, pp. 243–252.
[3] A. Lewko, T. Okamoto, A. Sahai, K. Takashima, and B. Waters, "Fully secure functional encryption: Attribute-based encryption and (hierarchical) inner product encryption," in *Proc. 29th Annu. Int. Conf. Theory Appl. Cryptograph. Techn.* Cham, Switzerland: Springer, 2010, pp. 62–91.
[4] K.-M. Chan, S.-H. Heng, W.-C. Yau, and S.-C. Tan, "Trapdoor privacy in public key encryption with keyword search: A review," *IEEE Access*, vol. 10, pp. 21584–21598, 2022.

[5] W.-C. Yau, S.-H. Heng, and B.-M. Goi, "Off-line keyword guessing attacks on recent public key encryption with keyword search schemes," in *Proc. Int. Conf. Autonomic Trusted Comput.*, Oslo, Norway. Cham, Switzerland: Springer, Jul. 2008, pp. 100–105.
[6] W.-C. Yau, R. C.-W. Phan, S.-H. Heng, and B.-M. Goi, "Keyword guessing attacks on secure searchable public key encryption schemes with a designated tester," *Int. J. Comput. Math.*, vol. 90, no. 12, pp. 2581–2587, Dec. 2013.
[7] J. W. Byun, H. S. Rhee, H.-A. Park, and D. H. Lee, "Off-line keyword guessing attacks on recent keyword search schemes over encrypted data," in *Proc. 3rd VLDB Workshop Secure Data Manag.*, Seoul, South Korea. Cham, Switzerland: Springer, Sep. 2006, pp. 75–83.
[8] C. Shen, Y. Lu, and J. Li, "Expressive public-key encryption with keyword search: Generic construction from KP-ABE and an efficient scheme over prime-order groups," *IEEE Access*, vol. 8, pp. 93–103, 2020.
[9] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-based encryption for fine-grained access control of encrypted data," in *Proc. 13th ACM Conf. Comput. Commun. Secur.*, Oct. 2006, pp. 89–98.
[10] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-policy attribute-based encryption," in *Proc. IEEE Symp. Secur. Privacy*, May 2007, pp. 321–334.
[11] Z. Lv, C. Hong, M. Zhang, and D. Feng, "Expressive and secure searchable encryption in the public key setting," in *Proc. 17th Int. Conf. Inf. Secur.* Cham, Switzerland: Springer, 2014, pp. 364–376.
[12] H. Cui, Z. Wan, R. H. Deng, G. Wang, and Y. Li, "Efficient and expressive keyword search over encrypted data in cloud," *IEEE Trans. Dependable Secure Comput.*, vol. 15, no. 3, pp. 409–422, May 2018.
[13] H. Zhu, L. Wang, H. Ahmad, and X. Niu, "Key-policy attribute-based encryption with equality test in cloud computing," *IEEE Access*, vol. 5, pp. 20428–20439, 2017.
[14] S. Wang, D. Zhao, and Y. Zhang, "Searchable attribute-based encryption scheme with attribute revocation in cloud storage," *PLoS ONE*, vol. 12, no. 8, Aug. 2017, Art. no. e0183459.
[15] T. Peng, Q. Liu, B. Hu, J. Liu, and J. Zhu, "Dynamic keyword search with hierarchical attributes in cloud computing," *IEEE Access*, vol. 6, pp. 68948–68960, 2018.
[16] R. Meng, Y. Zhou, J. Ning, K. Liang, J. Han, and W. Susilo, "An efficient key-policy attribute-based searchable encryption in prime-order groups," in *Proc. 11th Int. Conf. Provable Secur.*, Xi'an, China. Cham, Switzerland: Springer, Oct. 2017, pp. 39–56.
[17] C. Wang, W. Li, Y. Li, and X. Xu, "A ciphertext-policy attribute-based encryption scheme supporting keyword search function," in *Proc. 5th Int. Symp. Cyberspace Saf. Secur.*, Zhangjiajie, China. Cham, Switzerland: Springer, 2013, pp. 377–386.
[18] H. Yin, J. Zhang, Y. Xiong, L. Ou, F. Li, S. Liao, and K. Li, "CP-ABSE: A ciphertext-policy attribute-based searchable encryption scheme," *IEEE Access*, vol. 7, pp. 5682–5694, 2019.
[19] V. C. Trinh, "A ciphertext-policy attribute-based searchable encryption scheme in non-interactive model," *J. Comput. Sci. Cybern.*, vol. 35, no. 3, pp. 233–249, 2019.
[20] Q. Wu, X. Ma, L. Zhang, and Y. Chen, "Expressive ciphertext policy attribute-based searchable encryption for medical records in cloud," *Int. J. Netw. Secur.*, vol. 23, no. 3, pp. 461–472, 2021.
[21] Q. Huang, G. Yan, and Q. Wei, "Attribute-based expressive and ranked keyword search over encrypted documents in cloud computing," *IEEE Trans. Services Comput.*, vol. 16, no. 2, pp. 957–968, Mar. 2023.
[22] H. S. Rhee, J. H. Park, W. Susilo, and D. H. Lee, "Trapdoor security in a searchable public-key encryption scheme with a designated tester," *J. Syst. Softw.*, vol. 83, no. 5, pp. 763–771, May 2010.
[23] Y. Lu, G. Wang, and J. Li, "Keyword guessing attacks on a public key encryption with keyword search scheme without random Oracle and its improvement," *Inf. Sci.*, vol. 479, pp. 270–276, Jan. 2019.

**KOON-MING CHAN** is currently pursuing the Ph.D. (I.T.) degree with the Faculty of Information Science and Technology, Multimedia University, Malaysia. His research interests include cryptography and information security.

**SWEE-HUAY HENG** received the Doctor of Engineering degree from Tokyo Institute of Technology, Japan. She is currently a Professor with the Faculty of Information Science and Technology, Multimedia University, Malaysia. Her research interests include cryptography and information security. She served as the technical program committee member for many international security conferences. She was the Program Chair of ProvSec 2010, CANS 2010, and ISPEC 2019.

**SYH-YUAN TAN** is currently a Postdoctoral Researcher with Multimedia University, Malaysia. Before this role, he was a Senior Cryptography Developer with Zamna Technologies and a Research Associate with Newcastle University, U.K. His research primarily revolves around cryptography and information security, with a particular interest in zero-knowledge protocols and provable security techniques. In addition to his professional roles, he is a member of Malaysian National Mirror Committee for ISO/IEC, where he focuses on *Cryptography* and Security Mechanisms. He also contributes to Malaysian Society for Cryptology Research. Furthermore, he is part of the Committee of Malaysian Cryptographic Standards (MySEAL).

**SHING-CHIANG TAN** received the B.Tech. (Hons.) and M.Sc. (Eng.) degrees from Universiti Sains Malaysia, Malaysia, in 1999 and 2002, respectively, and the Ph.D. degree from Multimedia University, Malacca, Malaysia, in 2008. He is currently a Professor with the Faculty of Information Science and Technology, Multimedia University. His current research interests include computational intelligence (artificial neural networks, evolutionary algorithms, fuzzy logic, and decision trees), and their applications, data classification, condition monitoring, fault detection and diagnosis, and biomedical disease classification. He was a recipient of the Matsumae International Foundation Fellowship, Japan, in 2010.