

Schemes of IPsec integrated with quantum key distribution

Chun-Hui Zhang(张春辉), Wen-Xuan Zhang(张文轩), Xing-Yu Zhou(周星宇), Yuan Cao(曹原), Jun Wang(汪军), Jian Li(李剑), and Qin Wang(王琴)

Citation: Chin. Phys. B, 2026, 35 (**4**): 040305. DOI: 10.1088/1674-1056/ae3301
http://cpb.iphy.ac.cn; http://iopscience.iop.org/cpb

What follows is a list of articles you may be interested in

Encoding converters for quantum communication networks

Hua-Xing Xu(许华醒), Shao-Hua Wang(王少华), Ya-Qi Song(宋雅琪), Ping Zhang(张平), and Chang-Lei Wang(王昌雷)

Chin. Phys. B, 2025, 34 (**5**): 050310. DOI: 10.1088/1674-1056/adcdeb

Multi-protocol quantum key distribution decoding chip

Chun-Xue Zhang(张春雪), Jian-Guang Li(李建光), Yue Wang(王玥), Wei Chen(陈巍), Jia-Shun Zhang(张家顺), and Jun-Ming An(安俊明)

Chin. Phys. B, 2025, 34 (**5**): 050303. DOI: 10.1088/1674-1056/adb686

Free-space discrete-variable quantum key distribution in a mountainous environment

Xing-Ran Chen(陈星燃), Jian-Hong Shi(史建红), and Hai-Long Zhang(张海龙)

Chin. Phys. B, 2025, 34 (**5**): 050301. DOI: 10.1088/1674-1056/adbd25

Asymmetric mode-pairing quantum key distribution with advantage distillation

Hai-Tao Wang(汪海涛), Chun Zhou(周淳), Yi-Fei Lu(陆宜飞), Chen-Peng Hao(郝辰鹏), Yan-Mei Zhao(赵燕美), Yan-Yang Zhou(周砚扬), Hong-Wei Li(李宏伟), and Wan-Su Bao(鲍皖苏)

Chin. Phys. B, 2025, 34 (**4**): 040305. DOI: 10.1088/1674-1056/adbb5c

Mask-coding-assisted continuous-variable quantum direct communication with orbital angular momentum multiplexing

Zhengwen Cao(曹正文), Yujie Wang(王禹杰), Geng Chai(柴庚), Xinlei Chen(陈欣蕾), and Yuan Lu(卢缘)

Chin. Phys. B, 2025, 34 (**2**): 020308. DOI: 10.1088/1674-1056/ad9ff7

Schemes of IPsec integrated with quantum key distribution

Chun-Hui Zhang(张春辉)[1,2,3], Wen-Xuan Zhang(张文轩)[1,2,3], Xing-Yu Zhou(周星宇)[1,2,3], Yuan Cao(曹原)[1,2,3], Jun Wang(汪军)[4], Jian Li(李剑)[1,2,3], and Qin Wang(王琴)[1,2,3,†]

[1] *Institute of Quantum Information and Technology, Nanjing University of Posts and Telecommunications, Nanjing* 210003, *China*
[2] *Key Laboratory of Broadband Wireless Communication and Sensor Network Technology, Ministry of Education, Nanjing University of Posts and Telecommunications, Nanjing* 210003, *China*
[3] *Telecommunication and Networks, National Engineering Research Center, Nanjing University of Posts and Telecommunications, Nanjing* 210003, *China*
[4] *E-Quantum Information Technology Co., Ltd., Nanjing* 211112, *China*

(Received 16 October 2025; revised manuscript received 18 November 2025; accepted manuscript online 4 January 2026)

With the rapid advancement of quantum computing, traditional security protocols based on classical encryption algorithms are increasingly vulnerable to potential quantum attacks. The current IPsec protocol, which relies on classical cryptographic methods, is insufficient to withstand such threats, thereby compromising the security of long-term data transmission. To address this issue, we propose integrating quantum key distribution (QKD) into the internet protocol security (IPsec) protocol, thereby enhancing its resilience against quantum computing attacks. Here, two schemes that merge QKD-generated keys with classical cryptographic keys are designed to enhance both security and stability. Furthermore, we conduct a comprehensive evaluation of the performance of various QKD protocols implemented with the scheme, along with an assessment of its overall efficacy across a topological network configuration. This approach not only ensures secure data transmission in the era of quantum computing but also highlights the potential application value of integrating QKD with IPsec, providing valuable insights for the design and implementation of future quantum-secure communication systems.

Keywords: quantum key distribution, internet protocol security, internet key exchange

PACS: 03.67.Dd, 03.67.Hk

DOI: 10.1088/1674-1056/ae3301 **CSTR:** 32038.14.CPB.ae3301

1. Introduction

Technological advancements have ushered humanity into the information age, making information transmission through the internet significantly easier. Since information is generally not intended to be accessed by unauthorized individuals, network security has emerged as an important field, serving as a fundamental requirement for ensuring the confidentiality of sensitive data. Currently, network security primarily relies on cryptography for protection. However, with advancements in quantum mechanics, the introduction of quantum computing has enhanced computational capabilities for a small set of specific computational problems.[1] Widely used key exchange algorithms, such as Rivest–Shamir–Adleman (RSA)[2] and Diffie–Hellman (DH),[3] rely on computational complexity to ensure information security. However, quantum computing could break these encryption methods in a fraction of the time that would take classical computers thousands of years.[4] If quantum computers become widely accessible in the future, current cryptographic systems could be rendered obsolete almost instantly.

In modern communications, internet protocol security (IPsec)[5] is a suite of protocols and services designed to secure IP networks and operates at the network layer of the open systems interconnection (OSI) model. IPsec includes key components such as the internet key exchange (IKE), authentication header (AH), and encapsulating security payload (ESP) protocols. These components work together to ensure data confidentiality, integrity, source authentication, and protection against replay attacks during data transmission. IPsec is extensively used by enterprises and organizations to secure internal communications and is commonly employed as a virtual private network (VPN) technology. The IKE protocol, a crucial component of IPsec, establishes keys through negotiation based on public-key cryptography. In this process, algorithms such as DH are utilized, but they are highly susceptible to potential threats posed by quantum computing.

To defend against potential threats posed by quantum computing, researchers have proposed quantum key distribution (QKD)[6] and post-quantum cryptography (PQC).[7] PQC relies on mathematical problems that remain computationally intractable even for quantum computers as its security foundation, whereas QKD ensures security based on the principles of quantum mechanics.[8] Although PQC offers lower implementation costs, it lacks information-theoretic security, making QKD a promising direction for future development. QKD allows two communicating parties to establish symmetric keys, and by utilizing the one-time pad technique,[9] unconditional security in information transmission can be guaran-

†Corresponding author. E-mail: qinw@njupt.edu.cn

© 2026 Chinese Physical Society and IOP Publishing Ltd. All rights, including for text and data mining, AI training, and similar technologies, are reserved.

teed. In 1984, Bennett and Brassard introduced the first QKD protocol, known as the BB84 protocol.[6] To defend against detector-side attacks, Lo *et al.*[10] proposed the measurement-device-independent QKD (MDI-QKD) protocol, wherein both communicating parties send their optical pulses to an untrusted third party for measurement. This approach eliminates security assumptions regarding the measurement device, rendering MDI-QKD resistant to detector-side attacks. However, due to attenuation in fiber channels and the absence of quantum repeaters, these protocols cannot surpass the Pirandola–Laurenza–Ottaviani–Banchi (PLOB) bound,[11] which describes a linear relationship between key rate and channel transmittance.

In 2018, Lucamarini *et al.*[12] presented the twin-field QKD (TF-QKD) protocol, which successfully breaks the PLOB bound. Later, many variants were developed to further enhance performance and security.[13–15] However, the practical implementation of TF-QKD requires phase locking and tracking,[16,17] which makes the protocol more complex to deploy. To address this challenge, Zeng *et al.*[18] proposed the mode-pairing QKD (MP-QKD) protocol and Yin *et al.*[19] proposed the asynchronous MDI-QKD (AMDI-QKD), which reduce the technical difficulty and can also surpass the PLOB bound. Additionally, several notable experimental implementations and theoretical improvements have been reported.[20–24]

Recent theoretical advances in quantum networks leveraging QKD protocols have garnered significant attention,[25–31] particularly regarding the integration of QKD with existing network security protocols.[32–43] Notable demonstrations include: Marksteiner and Maurhart's[35] practical solution using rapidly rekeyed IPsec links to address QKD bandwidth limitations in 2015; and Sibson *et al.*'s[38] 2024 field trial of chip-based QKD over 28.5 km of deployed G.652 fiber, integrated via RFC 8784[44] with Juniper routers to support concurrent IPsec tunnels consuming independent keys.

These works collectively validate the feasibility of coupling quantum keys with classical security protocols. However, they remain limited in scope. For instance, JPMorgan's metropolitan-field trial[39] treated QKD as an external key source without modifying the IKE protocol internals. The IPSeQ protocol[41] introduced dynamic key requests during IKE authentication but restricted QKD usage to replacing IPsec session keys, leaving the foundational IKE master key derivation unchanged. Meanwhile, high-throughput hybrid encryptors[43] combined QKD with post-quantum and classical key exchange mechanisms but did not isolate or optimize the pure QKD contribution within IKE.

In contrast, our work presents a deeply integrated architecture that embeds QKD-generated keys directly into the core key hierarchy of both IKEv1 and IKEv2, specifically enhancing SKEYID in IKEv1 and SKEYSEED in IKEv2, thereby strengthening the entire handshake against future quantum attacks. We further provide the first unified performance analysis across three representative QKD protocols, quantify their impact on key consumption and VPN scalability, and validate system feasibility in a 24-node network topology. This end-to-end, protocol-aware integration offers practical insights beyond black-box key replacement approaches.

This paper is organized as follows. Section 2 describes the proposed integration schemes; Section 3 introduces the QKD protocols under evaluation; Section 4 compares the communication performance of IPsec integrated with different QKD protocols; Section 5 analyzes the scheme's behavior in a realistic network topology; and Section 6 concludes with a summary and discussion.

2. Scheme for integrating IPsec with QKD

The IKE protocol has two versions, IKEv1[45] and IKEv2.[46] IKEv1 involves a relatively complex information exchange process, whereas IKEv2 simplifies the procedure and removes certain potential security risks. Since both versions are still in use, we provide integration schemes for each of them. The IKE protocol is divided into two phases. In the first phase, the communicating parties negotiate to establish a security association (SA), known as the IKE SA. In the second phase, they negotiate to establish the IPsec SA, which is used by IPsec. The aforementioned SA includes negotiation of the security protocols used by the communicating parties, the encapsulation mode for data transmission, the encryption and authentication algorithms, and the keys for data transmission.

Below, we present a general application model for a quantum secure communication system based on the IPsec protocol and QKD, along with schemes integrating both versions of the IKE protocol with QKD-generated keys.

2.1. Quantum secure communication system

The application model of a quantum secure communication system based on the IPsec protocol and QKD in a terminal-to-gateway secure access scenario is illustrated in Fig. 1. The gateway device communicates directly with the QKD system or QKD network to obtain quantum cryptographic service keys online via the key acquisition interface Ak. The terminal device and the QKD system or network acquire quantum cryptographic service keys through the terminal key negotiation service, also via the Ak interface. The terminal device receives terminal charging keys either by loading them from a key storage medium or through an online update. The terminal key encryption service, in conjunction with the terminal device, cryptographically protects the quantum en-

cryption service key using the terminal charging key during transmission.

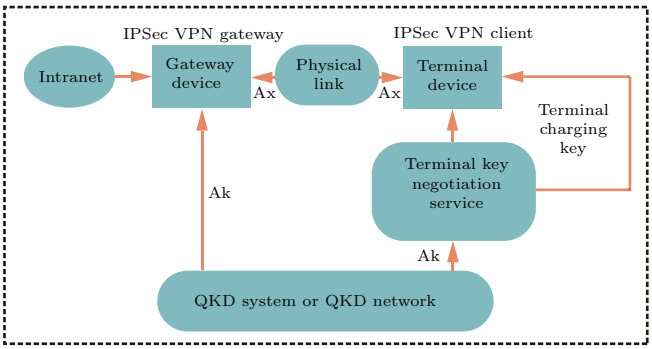


Fig. 1. Terminal-to-gateway application model of a quantum secure communication system.

Between the terminal and gateway devices, the quantum encryption key is combined with the IKE protocol to establish a bidirectional SA, thereby providing a secure encrypted transmission channel. The key exchange and encrypted service message transmission interface between the terminal and gateway devices is denoted as Ax.

2.2. The scheme of IKEv1 with QKD keys

In the following, the negotiation processes of the IKE SA and IPsec SA, as well as the scheme integrating QKD keys, are presented. The discussion mainly focuses on aspects related to key generation, while other message interactions during the negotiation process are not considered. First, after a series of related message interactions are completed, a function is used to generate a seed key

$$\mathrm{SKEYID} = \mathrm{prf}(\mathrm{Hash}(\mathrm{Ni}|\mathrm{Nr}), \mathrm{CKYi}|\mathrm{CKYr}), \tag{1}$$

where Hash is the hash function; Ni and Nr are random numbers shared between the sender and the receiver during message interaction; Ni|Nr denotes the concatenation of Ni and Nr; CKYi and CKYr are the cookies of the sender and the receiver. $\mathrm{prf}(K,S)$ is a pseudorandom function, such as AES-XCBC-PRF-128. In this function, K is the 128-bit key and S is a message of arbitrary length. The function outputs 128 bits, and the subsequent key consumption calculations in this paper are based on this output. Therefore, this paper outlines the calculation process of the function. AES-XCBC-PRF-128 is a pseudorandom function built on the AES algorithm.[47] First, three seed keys (K_1, K_2, K_3) are generated using AES encryption

$$K_1 = \mathrm{AES}(K, \mathrm{Const}_1), \tag{2}$$

$$K_2 = \mathrm{AES}(K, \mathrm{Const}_2), \tag{3}$$

$$K_3 = \mathrm{AES}(K, \mathrm{Const}_3), \tag{4}$$

where Const_1, Const_2, and Const_3 are fixed constants. The specific process of AES is detailed in Ref. [47]. The message S is divided into 128-bit chunks: $S = S_1, S_2, \ldots, S_n$, with 128 bits per block. If the last block S_n is less than 128 bits, it is padded with zeros to make it 128 bits. After initializing an intermediate value $X_0 = 0^{128}$ (a 128-bit all-zero vector), each block of the message S_i, where $i = 1, 2, \ldots, n-1$, is processed as follows:

$$X_i = \mathrm{AES}(K_1, X_{i-1} \oplus S_i). \tag{5}$$

Finally, the last block is processed based on the length of S

$$X_n = \mathrm{AES}(K_2, X_{n-1} \oplus S_n), \tag{6}$$

$$X_n = \mathrm{AES}(K_3, X_{n-1} \oplus S_n), \tag{7}$$

if S_n is a full 128-bit block, use Eq. (6); if S_n is an incomplete 128-bit block, use Eq. (7). X_n is the output of prf.

Next, SKEYID is used to generate the subsequent keys $\mathrm{SKEYID_d}$, $\mathrm{SKEYID_a}$, and $\mathrm{SKEYID_e}$ by

$$\mathrm{SKEYID_d} = \mathrm{prf}(\mathrm{SKEYID}, \mathrm{CKYi}|\mathrm{CKYr}|0), \tag{8}$$

$$\mathrm{SKEYID_a} = \mathrm{prf}(\mathrm{SKEYID}, \mathrm{SKEYID_d}|\mathrm{CKYi}|\mathrm{CKYr}|1), \tag{9}$$

$$\mathrm{SKEYID_e} = \mathrm{prf}(\mathrm{SKEYID}, \mathrm{SKEYID_a}|\mathrm{CKYi}|\mathrm{CKYr}|2), \tag{10}$$

where the values 0, 1, and 2 are represented by a single octet; $\mathrm{SKEYID_d}$ is used for the generation of the session key in the Child SA; $\mathrm{SKEYID_a}$ is the working key used by the IKE SA to authenticate the integrity of its messages as well as the identity of the data source; $\mathrm{SKEYID_e}$ is the working key used by the IKE SA to protect the confidentiality of its messages. The above corresponds to the first phase of IKEv1. We then combine the generated keys $\mathrm{SKEYID_d}$, $\mathrm{SKEYID_a}$, and $\mathrm{SKEYID_e}$ with QKD keys as

$$\mathrm{QSKEYID_d} = \mathrm{prf}+(\mathrm{QKD_d}, \mathrm{SKEYID_d}), \tag{11}$$

$$\mathrm{QSKEYID_a} = \mathrm{prf}+(\mathrm{QKD_a}, \mathrm{SKEYID_a}), \tag{12}$$

$$\mathrm{QSKEYID_e} = \mathrm{prf}+(\mathrm{QKD_e}, \mathrm{SKEYID_e}), \tag{13}$$

where $\mathrm{QKD_d}$, $\mathrm{QKD_a}$, and $\mathrm{QKD_e}$ are keys generated by QKD and prf+ is the pseudorandom function plus, which calls the prf function multiple times and concatenates the results of each call to generate key material of the desired length. The workflow is usually as

$$\begin{aligned} \mathrm{prf}+(K,S) &= T_1|T_2|T_3|T_4\ldots, \\ T_1 &= \mathrm{prf}(K, S|0x01), \\ T_2 &= \mathrm{prf}(K, T_1|S|0x02), \\ T_3 &= \mathrm{prf}(K, T_2|S|0x03), \\ T_4 &= \mathrm{prf}(K, T_3|S|0x04). \end{aligned} \tag{14}$$

$\mathrm{QSKEYID_d}$, $\mathrm{QSKEYID_a}$, and $\mathrm{QSKEYID_e}$ are generated based on the QKD keys. $\mathrm{QSKEYID_d}$ substitutes the original $\mathrm{SKEYID_d}$ in the IPsec protocol, serving as the derived key for the session key. Similarly, $\mathrm{QSKEYID_a}$ replaces $\mathrm{SKEYID_a}$ as

the authentication key, while $\mathrm{QSKEYID_e}$ replaces $\mathrm{SKEYID_e}$ as the encryption key.

In the second phase of IKEv1, we use $\mathrm{QSKEYID_d}$ to derive the keys KEYMAT that are required for the IPsec protocol

$$\mathrm{KEYMAT} = \mathrm{prf}(\mathrm{QSKEYID_d}, \mathrm{protocol}|\mathrm{SPI}|\mathrm{Ni}|\mathrm{Nr}), \quad (15)$$

where protocol and SPI are selected from the negotiated IKE SA proposal payload. Since the number of bits output by the prf function does not satisfy the required KEYMAT key length, it must be extended using a feedback concatenation method

$$\begin{aligned} &\mathrm{KEYMAT} = K_1|K_2\ldots, \\ &K_1 = \mathrm{prf}(\mathrm{QSKEYID_d}, \mathrm{protocol}|\mathrm{SPI}|\mathrm{Ni}|\mathrm{Nr}), \\ &K_2 = \mathrm{prf}(\mathrm{QSKEYID_d}, K_1|\mathrm{protocol}|\mathrm{SPI}|\mathrm{Ni}|\mathrm{Nr}). \end{aligned} \quad (16)$$

Although KEYMAT is derived from $\mathrm{QSKEYID_d}$ and already contains QKD keys, we further combine it with another QKD key to improve security

$$\mathrm{QKEYMAT} = \mathrm{prf}+(\mathrm{QKD}_k, \mathrm{KEYMAT}). \quad (17)$$

The above describes the scheme for combining the IKEv1 protocol with QKD keys. Based on the definition of the pseudorandom function employed in this work, the aforementioned key segments $\mathrm{QKD_d}$, $\mathrm{QKD_a}$, $\mathrm{QKD_e}$, and $\mathrm{QKD_k}$ each have a length of 128 bits.

2.3. The scheme of IKEv2 with QKD keys

Similarly, the two phases of the IKEv2 protocol involve negotiating the IKE SA and the Child SA for IPsec. Our focus is on the key generation part, while the remaining message interactions are not considered. The IKE protocol first generates a seed key through a pseudorandom function

$$\mathrm{SKEYSEED} = \mathrm{prf}(\mathrm{Ni}|\mathrm{Nr}, g^{ir}), \quad (18)$$

where Ni and Nr are random numbers shared during the message interaction between the sender and the receiver, and g^{ir} is a shared bit string obtained through the Diffie–Hellman algorithm. We use QKD keys to protect the SKEYSEED seed key

$$\mathrm{QSKEYSEED} = \mathrm{prf}(\mathrm{QKD_s}, \mathrm{SKEYSEED}). \quad (19)$$

The resulting QSKEYSEED seed key is then used with prf+ to derive all the keys used in the Child SA

$$\begin{aligned} &\mathrm{SK_d}|\mathrm{SK_{ai}}|\mathrm{SK_{ar}}|\mathrm{SK_{ei}}|\mathrm{SK_{er}}|\mathrm{SK_{pi}}|\mathrm{SK_{pr}} \\ &= \mathrm{prf}+(\mathrm{QSKEYSEED}, \mathrm{Ni}|\mathrm{Nr}|\mathrm{SPIi}|\mathrm{SPIr}), \end{aligned} \quad (20)$$

where $\mathrm{SK_d}$ is the key used to derive the keys for the Child SA among the generated keys. The remaining keys have specific roles: $\mathrm{SK_{ai}}$ and $\mathrm{SK_{ar}}$ are used as integrity protection keys to authenticate the messages exchanged subsequently; $\mathrm{SK_{ei}}$ and $\mathrm{SK_{er}}$ are used to encrypt and decrypt subsequent exchanges; $\mathrm{SK_{pi}}$ and $\mathrm{SK_{pr}}$ are involved in payload generation. SPIi and SPIr are the security parameter indexes.[46] In the second stage, the key in the Child SA is derived using $\mathrm{SK_d}$ and the random numbers Ni and Nr. To improve the process of deriving IPsec keys, we integrate QKD keys as follows:

$$\mathrm{QKEYMAT} = \mathrm{prf}+(\mathrm{SK_d}, \mathrm{QKD_k}|\mathrm{Ni}|\mathrm{Nr}). \quad (21)$$

By incorporating QKD keys in this way, the security of the IPsec process does not rely solely on pseudorandom functions and the Diffie–Hellman algorithm, whose probability of being subjected to quantum attacks and being cracked is greatly reduced. The proposed scheme utilizes key segments $\mathrm{QKD_s}$ and $\mathrm{QKD_k}$, each of which has a length of 128 bits. Compared with the IKEv1 scheme, the IKEv2 scheme utilizes fewer key segments, attributable to its streamlined process and enhanced security features. This distinction effectively highlights the differences between the two IKE versions.

2.4. The integration process of QKD and IPsec

This section describes the detailed integration process of QKD with IPsec. As shown in Fig. 2, this diagram illustrates the integration of QKD keys within the IPsec protocol framework, highlighting the secure communication process between the sender and receiver. In this model, QKD keys are generated between the sender and receiver to establish the key material for the IPsec SA.

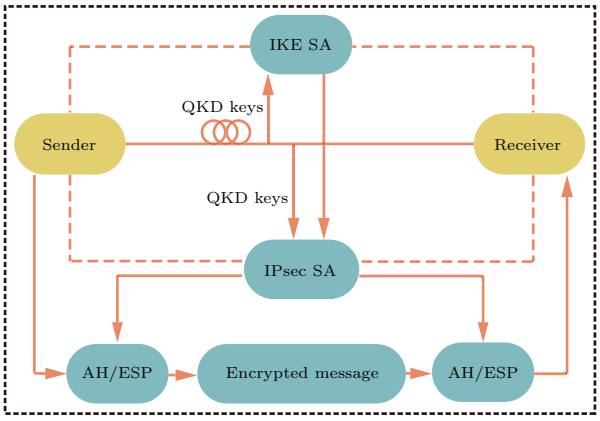


Fig. 2. Process of integrating QKD keys into the IPsec protocol.

The process begins with the negotiation of the IKE SA between the sender and receiver, during which the QKD-generated keys are incorporated, as shown in Eqs. (11)–(13). Subsequently, the QKD keys are integrated again during the negotiation of the IPsec SA over the established IKE SA, as shown in Eq. (17). Finally, using the obtained IPsec SA, the sender and receiver encrypt IP packets for transmission via the AH or ESP protocols employing AES-256-GCM, thereby integrating QKD keys into the IPsec framework.

3. Integration with specific QKD protocols

We select three representative QKD protocols to illustrate the performance of the proposed combination of IKE and QKD keys. The principles of these protocols, together with the key rate calculations, are briefly outlined below.

The BB84 protocol, involving Alice and Bob as the legitimate users and Eve as the potential eavesdropper, aims to establish a shared secret key between Alice and Bob. In BB84-QKD, Alice transmits a sequence of single-photon states to Bob via a quantum channel, and Bob randomly chooses bases to measure these states. After post-processing steps such as error correction and privacy amplification, Alice and Bob generate the final key. Due to practical limitations in generating ideal single-photon sources, experiments typically use approximate sources such as weak coherent state (WCS) light. However, the presence of multi-photon components introduces vulnerability to photon-number-splitting (PNS) attacks.[48] To counter PNS attacks, the decoy-state technique has been introduced,[49–51] which allows a more accurate estimation of the single-photon gain and bit error rate. The final key rate formula,[52] derived from the GLLP framework,[53] can be expressed as

$$R = q\left\{\mu \mathrm{e}^{-\mu} Y_1^Z\left[1-H_2\left(e_1^X\right)\right]-Q_Z^\mu f H_2\left(E_Z^\mu\right)\right\}, \quad (22)$$

where q represents the probability that Alice selects a signal state and prepares it in the Z basis, while Bob also measures it in the Z basis. μ denotes the intensity of signal states; Y_1^Z is the single-photon gain in the Z basis; e_1^X is the single-photon bit error rate in the X basis; f is the error correction efficiency; $H_2(x)$ is the binary Shannon entropy function; Q_Z^μ is the total gain of the Z basis data; E_Z^μ is the quantum bit error rate.

The relationship between the key rate and transmission efficiency was initially expressed as $R \sim O(\eta^2)$ using the GLLP formula. Later, the introduction of the decoy-state method enhanced the key rate to $R \sim O(\eta)$. Subsequently, TF-QKD breaks the PLOB bound, achieving a scaling of $R \sim O(\sqrt{\eta})$. The key reason is that both users send coherent-state pulses that produce single-photon interference at the central measurement node. In this case, only one detector click is required as a successful event. This is equivalent to one of the photon paths propagating only to the intermediate node and experiencing half of the total transmission loss, effectively doubling the transmission distance while still allowing the bit information to be transmitted to the other user. Among the many variants of TF-QKD, the sending-or-not-sending TF-QKD (SNS-TF-QKD) protocol is proposed to enhance performance and tolerate large misalignment errors,[13] in which Alice and Bob randomly decide to send a phase-randomized coherent state with probability p or send nothing in a Z window. The formula of the final key rate is[54]

$$R = 2p(1-p)\mu \mathrm{e}^{-\mu} s_1^Z[1-H_2(e_1^{\mathrm{ph}})]-Q_Z^\mu f H_2\left(E_Z^\mu\right), \quad (23)$$

where s_1^Z is the counting rate and e_1^{ph} is the phase-flip error rate.

In order to achieve stable interference between coherent states, SNS-TF-QKD requires active phase locking and real-time phase tracking. In contrast, MP-QKD can surpass the PLOB bound without such demanding stabilization. Taking time-bin phase encoding as an example, Alice and Bob each send weak coherent pulses with random intensities and phases in every round. Charlie then announces whether a single detection click occurred. Provided that the time interval between two clicked pulses is within the laser coherence time, Alice and Bob can pair them during post-processing to form a key-generation event.[55] Unlike conventional MDI-QKD, which relies on pre-defined pulse pairs and coincidence detection and therefore discards all single-click events, MP-QKD treats detection outcomes as post-selectable resources. This post-selected pairing recovers events that would otherwise be wasted, significantly boosting the key rate. The final key rate formula is given by[55]

$$R = \frac{1}{N}\{n_{Z_1}[1-H_2(e_{Z_1}^{\mathrm{ph}})]-n_Z f H_2(E_Z)\}, \quad (24)$$

where n_{Z_1} is the single-photon effective detection number in the raw key, N is the total number of pulses, $e_{Z_1}^{\mathrm{ph}}$ is the single-photon phase error rate, and n_Z is the length of the raw key. More details of the calculation are given in Appendix A.

4. Results

In this section, we present the performance of IPsec integrated with different QKD protocols. We also consider the finite-key effects using the Chernoff bound,[56] setting the total number of pulses $N = 1\times10^{13}$ and the failure probability $\epsilon = 1\times10^{-10}$. The system parameters are listed in Table 1. The key rate curves of the QKD protocols are plotted in Fig. 3 with global optimization of the key rate. For the SNS-TF-QKD protocol, the duty cycle is set to 80%, since a portion of the optical signals is allocated for phase locking. It can be seen that although the BB84-QKD protocol has a higher key rate at short distances, its key rate at long distances is lower than those of the MP-QKD and SNS-TF-QKD protocols, and its maximum transmission distance is also shorter.

Table 1. System parameters used in the simulations. η_{d}: detection efficiency of the detectors; p_{d}: dark count rate of the detectors; f: error correction efficiency; α: fiber loss coefficient (dB/km); e_{d}: optical misalignment error; l: pairing-window interval of MP-QKD; M: total number of phase slices of SNS-TF-QKD; F: system clock rate (Hz).

η_{d}	p_{d}	f	α	e_{d}	l	M	F
50%	1×10^{-8}	1.16	0.2	0.02	1×10^{5}	16	1×10^{9}

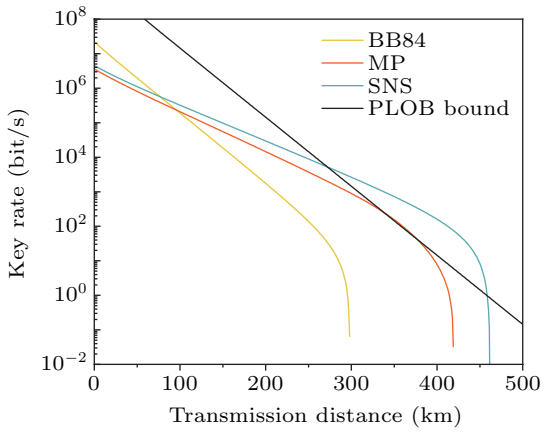


Fig. 3. Key rates of the selected QKD protocols. The yellow, red, and blue curves represent BB84-QKD, MP-QKD, and SNS-TF-QKD, respectively. The black curve denotes the PLOB bound.

To demonstrate the performance of the scheme, we set the key refresh time to one second. Thus, the key consumption ratio R_{key} and the number of supported VPN tunnels N_{vpn} can be expressed as

$$R_{\text{key}} = \frac{L_{\text{T}}}{R}, \tag{25}$$

$$N_{\text{vpn}} = \frac{R}{L_{\text{T}}}, \tag{26}$$

where L_{T} denotes the total length of QKD key segments in the IKEv1 or IKEv2 scheme, and R represents the secret key rate of the QKD protocol.

In Fig. 4(a), the key consumption ratios of the three QKD protocols, BB84-QKD, SNS-TF-QKD, and MP-QKD, are compared over the first 150 km for both the IKEv1 and IKEv2 protocols. The BB84-QKD protocol initially consumes the smallest proportion of keys up to a distance of 77 km, irrespective of the IKE version used. However, as the distance increases, its key consumption surpasses that of the SNS-TF-QKD and MP-QKD protocols. Notably, beyond 77 km, the SNS-TF-QKD protocol consistently consumes the smallest proportion of keys. The figure shows that the key consumption remains below 1% at shorter distances, allowing the deployment of multiple VPN tunnels to enhance throughput and reduce key refresh intervals, thereby improving security within hardware constraints.

Figure 4(b) extends the analysis beyond 150 km, showing that the SNS-TF-QKD protocol continues to outperform the others in terms of key consumption, followed by the MP-QKD protocol, with BB84-QKD consuming the highest proportion of keys. Beyond 250 km, the key generation rate of the BB84-QKD protocol can no longer support the key consumption required for even a single VPN tunnel. In contrast, the SNS-TF-QKD protocol remains viable at distances up to and beyond 350 km.

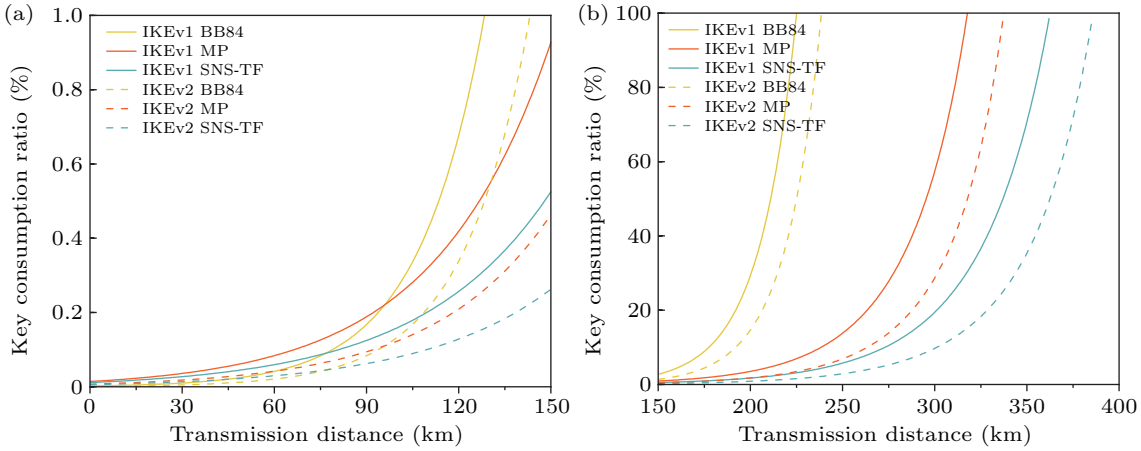


Fig. 4. QKD key consumption for the selected protocols. Solid lines correspond to the IKEv1 protocol, and dashed lines correspond to the IKEv2 protocol. The yellow, red, and blue lines represent BB84-QKD, MP-QKD, and SNS-TF-QKD, respectively.

Figure 5 illustrates the maximum number of VPN tunnels that can be supported simultaneously using three QKD protocols with a key refresh interval of 1 second. Figures 5(a) and 5(b) present the results for IKEv1 and IKEv2, respectively. As the transmission distance increases, the number of supported VPN tunnels decreases markedly for all three QKD protocols, with a sharper decline observed beyond 150 km. The insets provide a closer view of VPN tunnel variations between 200 km and 350 km, highlighting performance differences among the protocols at longer distances. Overall, the SNS-TF-QKD protocol supports the highest number of tunnels across both IKE versions, whereas the BB84-QKD protocol performs best at shorter distances. Although none of the protocols can fully support even a single VPN tunnel at extreme distances, increasing the key refresh interval could enable tunnel operation. Therefore, at longer distances, it is critical to optimize both the number of VPN tunnels and the key refresh interval, balancing throughput with key management to maximize system performance.

Table 2 compares the communication performance of BB84-QKD, MP-QKD, and SNS-TF-QKD at a transmission distance of 200 km. The table summarizes key consumption,

the maximum number of supported VPN tunnels, and the corresponding maximum throughput achievable under IKEv1 and IKEv2. Throughput values assume 8.31 Gbps per VPN tunnel, based on realistic experimental communication systems,[57] and reflect the maximum achievable data rate constrained by the key generation rates of each protocol. The results indicate that higher key rates directly translate into support for a greater number of VPN tunnels and increased throughput. For instance, the SNS-TF-QKD protocol, with a key rate of 29575 bps, can support up to 57 VPN tunnels with 473.67 Gbps throughput under IKEv1, and 115 VPN tunnels with 955.65 Gbps under IKEv2, demonstrating its suitability for high-bandwidth secure applications. In contrast, the BB84-QKD protocol, with a key rate of 1746 bps, achieves lower throughput and a smaller number of supported VPN tunnels, limiting its practicality for long-distance secure communications. These findings provide valuable insights into the scalability and performance trade-offs of different QKD protocols when integrated with IPsec security frameworks.

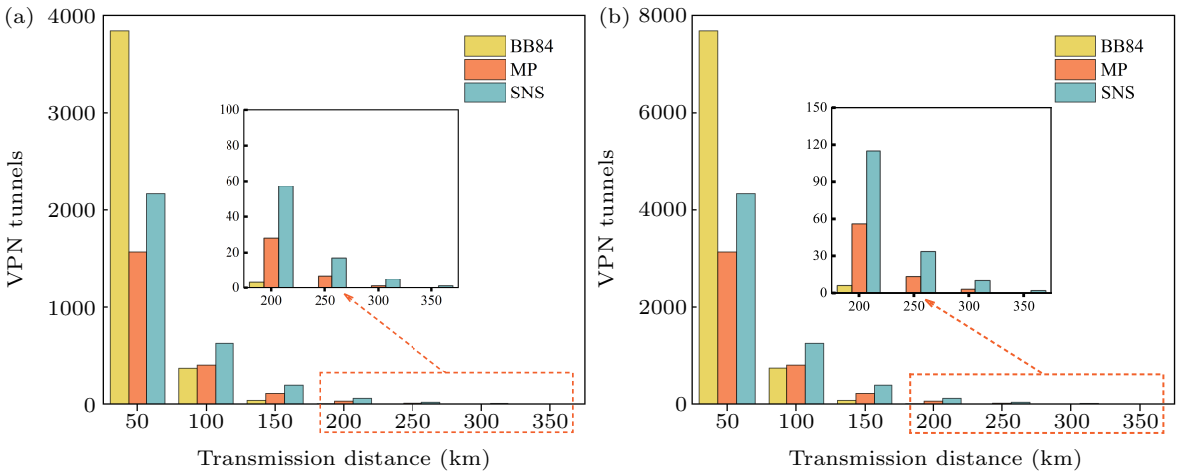


Fig. 5. Maximum number of VPN tunnels supported by the selected QKD protocols. (a) IKEv1 results; (b) IKEv2 results. The yellow, red, and blue columns correspond to BB84-QKD, MP-QKD, and SNS-TF-QKD, respectively.

Table 2. Comparison of communication performance of different protocols.

Protocols	Distance (km)	R (bps)	IKEv1			IKEv2		
			L_T (bit)	N_{vpn}	Throughput (Gbps)	L_T (bit)	N_{vpn}	Throughput (Gbps)
BB84	200	1746	512	3	24.93	256	6	49.86
MP	200	14513	512	28	232.68	256	56	465.36
SNS	200	29575	512	57	473.67	256	115	955.65

5. Network performance analysis

5.1. Network performance in protocol selection

To evaluate the practical performance of our proposed scheme,[58] we conducted network simulations based on the USNET topology $G=(V,E)$ in Fig. 6, a standard testbed comprising 24 physical nodes interconnected by 43 physical links, where V and E denote the sets of nodes and links, respectively. The QKD protocol for each link was dynamically selected according to its transmission distance, following the criteria: BB84-QKD (yellow lines) for short-range connections (<77 km) and SNS-TF-QKD (blue lines) for long-haul links (>77 km). To quantify the network performance, we define the average throughput $\bar{T}$ as

$$\bar{T}=\frac{1}{n}\sum_{i=1}^{n}T_i, \tag{27}$$

where T_i denotes the throughput of the i-th logical link, and $n=C_{24}^2=276$ represents the total number of node pairs. The throughput evaluation follows these principles. For physically adjacent nodes connected by a single fiber, T_i equals the throughput of that physical link. For non-adjacent node pairs, the bottleneck throughput is calculated as

$$T_i=\min\{t_1,t_2,\ldots,t_n\},$$

where $t_1,t_2,\ldots,t_n$ are the throughputs of constituent links along the Dijkstra-optimized path.

As demonstrated in Table 3, the adaptive protocol selection strategy exhibits significantly higher average throughput compared to individual QKD protocols under both IKEv1 and IKEv2 configurations. Among the individual QKD protocols, BB84-QKD demonstrates the highest average throughput performance, followed sequentially by SNS-TF-QKD and MP-QKD. This hierarchy primarily stems from three critical factors in our throughput calculation model: (1) the throughput is fundamentally constrained by the minimum link distance within the quantum communication network; (2) BB84-QKD

exhibits superior key generation efficiency at shorter transmission distances (< 77 km); (3) while both SNS-TF-QKD and MP-QKD maintain operational capability across extended distances, SNS-TF-QKD achieves a higher key rate compared to MP-QKD throughout the entire operational range. The comparative analysis reveals that the adaptive approach achieves a 2.9% throughput improvement over the best-performing standalone protocol (BB84-QKD) in IKEv1 and IKEv2 environments, respectively. The demonstrated performance gain substantiates that dynamic protocol adaptation mechanisms exhibit enhanced compatibility with hybrid quantum-classical network infrastructures, establishing a configurable framework for cross-layer optimization between QKD and IPsec in heterogeneous communication environments.

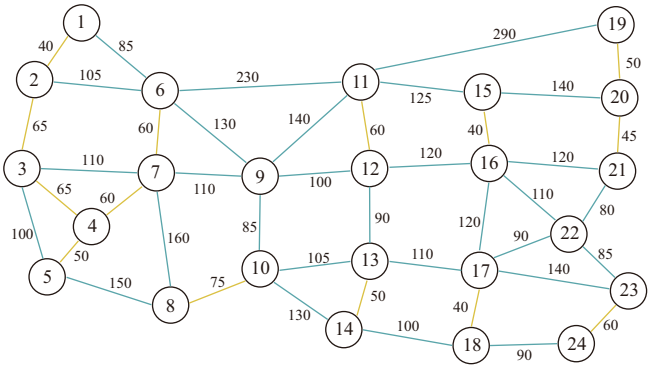


Fig. 6. USNET topology with adaptive QKD protocol allocation, where the yellow line and blue line represent BB84-QKD and SNS-TF-QKD, respectively.

Table 3. Throughput comparison of protocol selection strategies (Gbps).

Strategy	BB84	MP	SNS-TF	Proposed adaptive
IKEv1 $\bar{T}$	18850	7754	11852	19393
IKEv2 $\bar{T}$	37703	15512	23706	38792

5.2. Network performance in service requests

As demonstrated in the previous section, dynamically selecting QKD protocols can enhance the network's average throughput. Building on this result, we simulate service requests to illustrate the operation of the IPsec protocol integrated with QKD keys within the network.

We model a service request as $sr(V_{sr}^s, V_{sr}^t, K_{sr}) \in SR$, where V_{sr}^s and V_{sr}^t denote the source and destination nodes of service request sr, respectively, K_{sr} indicates the QKD keys required for the service request running the IPsec protocol, and SR represents the set of service requests.

For each service request, a unique set of QKD keys is assigned to ensure security. The request randomly selects a source and a destination node and determines the shortest path using Dijkstra's algorithm. $P_{sr}(k^1, k^2, \ldots, k^n, k_{sr}^1, k_{sr}^2, \ldots, k_{sr}^n)$ denotes the path selected by service request sr, where k^i and k_{sr}^i $(i \in E)$ indicate the total number of QKD keys and the current number of QKD keys in the ith link on path P_{sr}.

The system then checks whether every segment along path P_{sr} has sufficient QKD keys for IPsec encryption. A service request sr is considered successful only if all segments meet this requirement; otherwise, it fails. The set of successful service requests is denoted as SR_s. The success probability of service requests can be expressed as

$$\eta_s = \frac{|SR_s|}{|SR|}. \tag{28}$$

In addition, the key utilization in the ith link can be represented as

$$\eta_u^i = \frac{1}{k^i} \sum_{sr \in SR_s} K_{sr}. \tag{29}$$

The detailed procedure is summarized in Algorithm 1.

Algorithm 1 Service request provisioning algorithm.

Input: $G(V,E)$, SR.
Output: SR_s, $P_{sr}(k^1, k^2, \ldots, k^n, k_{sr}^1, k_{sr}^2, \ldots, k_{sr}^n)$ for each $sr \in SR_s$.

```
Initialize SR_s ← ∅;
for each service request sr(V_sr^s, V_sr^t, K_sr) ∈ SR do
    determine the shortest path using Dijkstra's algorithm and store it in P_sr;
    sufficient ← true;
    for each link i on path P_sr do
        if available keys k_sr^i < K_sr then
            sufficient ← false;
            break;
    if sufficient then
        for each link i on path P_sr do
            k_sr^i ← k_sr^i − K_sr;
        SR_s ← SR_s ∪ {sr};
    else
        mark service request sr as failed;
return SR_s, P_sr for sr ∈ SR_s.
```

As shown in Fig. 7, we conducted simulations with service request rates ranging from 1000 to 19000 per second for both IKEv1 and IKEv2. The results indicate that the success probability decreases as the service request rate increases. This decline occurs because the cumulative key demand of service requests exceeds the QKD key generation rate between nodes, resulting in insufficient keys to support all requests. Notably, the success probability for IKEv2 is higher than that for IKEv1, which is attributed to the lower key demand of the IKEv2 protocol compared to IKEv1.

We simulated the key utilization of links between each pair of nodes. Specifically, we conducted simulations for both IKEv1 and IKEv2 protocols at a service request rate of 11000 requests per second. The results are shown in Fig. 8, where the blue 3D bars represent the IKEv1 protocol and the red 3D bars represent the IKEv2 protocol. The results demonstrate that in both IKE protocol versions, links located at the edge of

the topology exhibit lower key utilization rates, while links positioned in the central region of the topology generally show higher utilization rates. Given that the nodes in our service request simulation were randomly selected, nodes in the central part of the topology are more likely to be chosen for service request paths. In contrast, paths rarely traverse nodes in peripheral regions, resulting in significantly lower key utilization rates, even under high service request rates. Therefore, in practical network deployments, it is feasible to adjust the QKD key generation rates between different nodes. This approach can effectively reduce costs and avoid unnecessary resource waste without compromising network performance.

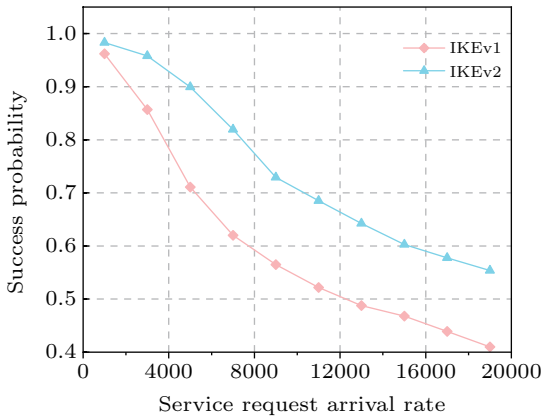


Fig. 7. Success probability of service requests. The blue line represents the IKEv1 protocol, and the pink line represents the IKEv2 protocol.

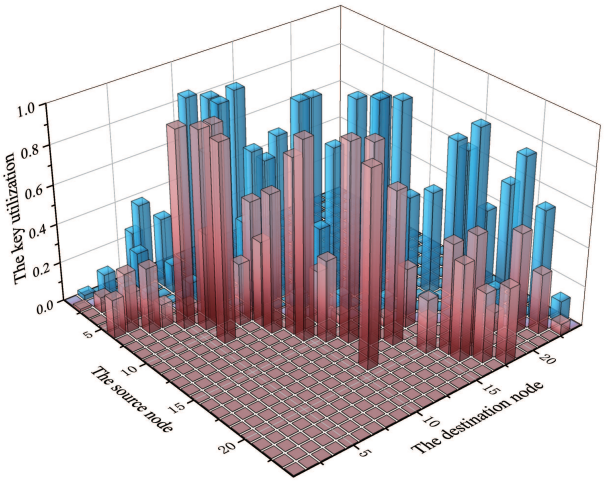


Fig. 8. Key utilization for each link. The blue 3D bars represent the IKEv1 protocol, while the red 3D bars represent the IKEv2 protocol.

6. Conclusion

In this paper, we present schemes for integrating IPsec with QKD protocols, employing QKD keys in both versions of the IKE protocol. We specifically evaluate three widely-used QKD protocols, performing detailed calculations of their practical communication performance under different conditions. Our results demonstrate that the combination of QKD and IPsec protocols provides sufficient key consumption at short distances. Even at extended ranges, such as over 350 km, secure communication can still be maintained by carefully selecting appropriate key refresh intervals. This highlights the flexibility and scalability of integrating QKD with traditional security protocols, ensuring robust protection in long-distance communications without compromising throughput or performance. We conducted a systematic performance evaluation of the IPsec protocol within network topology configurations, demonstrating that its performance can be significantly enhanced through adaptive integration with QKD protocols. In addition, we have demonstrated network performance under simulated service requests. Our work not only provides practical guidelines for maintaining secure communications over long distances by leveraging the advantages of QKD and IPsec together, but also offers valuable insights into the application and integration of quantum networks with classical networks,[37] paving the way for a more secure and interconnected future.

Appendix A: Calculation details

A1. BB84 protocol

Alice randomly selects the signal state of intensity μ, the weak decoy state of intensity ν, or the vacuum state with probabilities p_μ, p_ν, and $1-p_\mu-p_\nu$, respectively. The probabilities that Alice and Bob choose the X basis are denoted by p_X^{A} and p_X^{B}, respectively. The total number of events under the Z or X basis, when selecting the signal state, weak decoy state, or vacuum state, can be expressed as follows:

$$N_Z^\mu = p_\mu(1-p_X^{\mathrm{A}})(1-p_X^{\mathrm{B}})N, \tag{A1}$$

$$N_Z^\nu = p_\nu(1-p_X^{\mathrm{A}})(1-p_X^{\mathrm{B}})N, \tag{A2}$$

$$N_X^\mu = p_\mu p_X^{\mathrm{A}} p_X^{\mathrm{B}} N, \tag{A3}$$

$$N_X^\nu = p_\nu p_X^{\mathrm{A}} p_X^{\mathrm{B}} N, \tag{A4}$$

$$N_o = (1-p_\mu-p_\nu)N. \tag{A5}$$

The gains and quantum bit error rates can be obtained as in Ref. [52]

$$Q_o = 2p_{\mathrm{d}}(1-p_{\mathrm{d}}), \tag{A6}$$

$$Q_Z^\mu = Q_X^\mu = Q_0 + 1 - \mathrm{e}^{-\eta\mu}, \tag{A7}$$

$$Q_Z^\nu = Q_X^\nu = Q_0 + 1 - \mathrm{e}^{-\eta\nu}, \tag{A8}$$

$$T_Z^\mu = T_X^\mu = Y_0 + 2e_{\mathrm{d}}(1-\mathrm{e}^{-\eta\mu}), \tag{A9}$$

$$T_X^\nu = Y_0 + 2e_{\mathrm{d}}(1-\mathrm{e}^{-\eta\nu}), \tag{A10}$$

$$E_Z^\mu = T_Z^\mu / Q_Z^\mu, \tag{A11}$$

where $\eta = \eta_{\mathrm{d}} 10^{-\alpha L/10}$. Then, we can calculate the single-photon gain in the Z-basis, Y_1^Z, the single-photon gain in the

X-basis, Y_1^X, and the single-photon bit error rate in the X-basis, e_1^X by

$$Y_1^Z \geq \frac{\mu}{\mu\nu-\nu^2}\left(\underline{Q}_Z^{\nu}\mathrm{e}^{\nu}-\overline{Q}_Z^{\mu}\mathrm{e}^{\mu}\frac{\nu^2}{\mu^2}-\frac{\mu^2-\nu^2}{\mu^2}\underline{Q}_o\right), \tag{A12}$$

$$Y_1^X \geq \frac{\mu}{\mu\nu-\nu^2}\left(\underline{Q}_X^{\nu}\mathrm{e}^{\nu}-\overline{Q}_X^{\mu}\mathrm{e}^{\mu}\frac{\nu^2}{\mu^2}-\frac{\mu^2-\nu^2}{\mu^2}\underline{Q}_o\right), \tag{A13}$$

$$e_1^X \leq \frac{\mathrm{e}^{\mu}\overline{T}_X^{\mu}-\mathrm{e}^{\nu}\underline{T}_X^{\nu}}{(\mu-\nu)Y_1^Z}, \tag{A14}$$

where the overline and underline indicate the Chernoff bounds of the variables, which are given by the following formulas:[54,56]

$$\phi^L(X) = \frac{X}{1+\delta_1(X)}, \tag{A15}$$

$$\phi^U(X) = \frac{X}{1-\delta_2(X)}, \tag{A16}$$

where the values of $\delta_1(X)$ and $\delta_2(X)$ can be obtained by solving the following equations:

$$\left[\frac{\mathrm{e}^{\delta_1}}{(1+\delta_1)^{1+\delta_1}}\right]^{X/(1+\delta_1)} = \frac{\epsilon}{2}, \tag{A17}$$

$$\left[\frac{\mathrm{e}^{-\delta_2}}{(1-\delta_2)^{1-\delta_2}}\right]^{X/(1-\delta_2)} = \frac{\epsilon}{2}, \tag{A18}$$

Finally, we can obtain the secret key rate R

$$R = q\left\{\mu\,\mathrm{e}^{-\mu}Y_1^Z\left[1-H_2\left(e_1^X\right)\right]-Q_Z^{\mu}fH_2\left(E_Z^{\mu}\right)\right\}, \tag{A19}$$

where $q = p_{\mu}(1-p_X^{\mathrm{A}})(1-p_X^{\mathrm{B}})$.

A2. SNS-TF protocol

In the four-intensity decoy-state SNS-TF protocol,[54] Alice (Bob) randomly selects the signal state ρ_{μ} of intensity μ, two weak decoy states ρ_{ν} and ρ_{ω} of intensity ν and ω, or the vacuum state ρ_o with probabilities p_{μ}, p_{ν}, p_{ω}, and $p_o = 1-p_{\mu}-p_{\nu}-p_{\omega}$, respectively. If the signal state is chosen, Alice (Bob) sends the vacuum state with probability $1-p$. We denote the counting rate of source ij as $S_{ij} = n_{ij}/N_{ij}$ ($ij = \{oo, o\omega, o\nu, \omega o, \nu o\}$) when Alice selects source ρ_i and Bob selects source ρ_j. The expressions for n_{ij} and N_{ij} are given by

$$N_{oo} = \left[p_o^2+2p_op_{\mu}(1-p)\right]N, \tag{A20}$$

$$N_{o\omega} = N_{\omega o} = \left[p_op_{\omega}+p_{\mu}p_{\omega}(1-p)\right]N, \tag{A21}$$

$$N_{o\nu} = N_{\nu o} = \left[p_op_{\nu}+p_{\mu}p_{\nu}(1-p)\right]N, \tag{A22}$$

$$n_{oo} = 2p_{\mathrm{d}}(1-p_{\mathrm{d}})N_{oo}, \tag{A23}$$

$$n_{o\omega} = n_{\omega o} = 2\left[(1-p_{\mathrm{d}})\mathrm{e}^{-\eta\omega/2}-(1-p_{\mathrm{d}})^2\mathrm{e}^{-\eta\omega}\right]N_{o\omega}, \tag{A24}$$

$$n_{o\nu} = n_{\nu o} = 2\left[(1-p_{\mathrm{d}})\mathrm{e}^{-\eta\nu/2}-(1-p_{\mathrm{d}})^2\mathrm{e}^{-\eta\nu}\right]N_{o\nu}, \tag{A25}$$

where $\eta = \eta_{\mathrm{d}}10^{-\alpha L/20}$. Next, we have

$$n_{\mathrm{signal}} = 4Np(1-p)(1-p_{\mathrm{d}})\mathrm{e}^{-\eta\mu/2}\left[1-(1-p_{\mathrm{d}})\mathrm{e}^{-\eta\mu}\right], \tag{A26}$$

$$n_{\mathrm{error}} = 2Np^2(1-p_{\mathrm{d}})\mathrm{e}^{-\eta\mu}\left[I_0(\eta\mu)-(1-p_{\mathrm{d}})\mathrm{e}^{-\eta\mu}\right] + 2N(1-p)^2p_{\mathrm{d}}(1-p_{\mathrm{d}}), \tag{A27}$$

$$n_{\mathrm{t}} = n_{\mathrm{signal}}+n_{\mathrm{error}}, \tag{A28}$$

$$T_X = \frac{1}{\Delta}\int_{-\frac{\Delta}{2}}^{\frac{\Delta}{2}}(1-p_{\mathrm{d}})\mathrm{e}^{-2\eta\omega\cos^2\frac{\delta}{2}}\,\mathrm{d}\delta-(1-p_{\mathrm{d}})^2\mathrm{e}^{-2\eta\omega}, \tag{A29}$$

$$S_X = \frac{1}{\Delta}\int_{-\frac{\Delta}{2}}^{\frac{\Delta}{2}}(1-p_{\mathrm{d}})\mathrm{e}^{-2\eta\omega\sin^2\frac{\delta}{2}}\,\mathrm{d}\delta-(1-p_{\mathrm{d}})^2\mathrm{e}^{-2\eta\omega}+T_X, \tag{A30}$$

$$T_{\Delta} = T_X(1-2e_{\mathrm{d}})+e_{\mathrm{d}}S_X, \tag{A31}$$

where $\Delta = 2\pi/M$. Then, the single-photon gain in the Z-basis, s_1^Z, and the single-photon phase-flip error rate, e_1^{ph}, are calculated as

$$s_1^Z \geq \frac{1}{2\omega\nu(\nu-\omega)}\left[\nu^2\mathrm{e}^{\omega}(\langle\underline{S}_{o\omega}\rangle+\langle\underline{S}_{\omega o}\rangle) - \omega^2\mathrm{e}^{\nu}(\langle\overline{S}_{o\nu}\rangle+\langle\overline{S}_{\nu o}\rangle)-2(\nu^2-\omega^2)\langle\overline{S}_{oo}\rangle\right], \tag{A32}$$

$$e_1^{\mathrm{ph}} \leq \frac{\langle\overline{T}_{\Delta}\rangle-\frac{1}{2}\mathrm{e}^{-2\mu_1}\langle\underline{S}_{oo}\rangle}{2\mu_1\,\mathrm{e}^{-2\mu_1}s_1^Z}, \tag{A33}$$

where

$$\langle\underline{S}_{ij}\rangle = \frac{\phi^L(n_{ij})}{N_{ij}}, \tag{A34}$$

$$\langle\overline{S}_{ij}\rangle = \frac{\phi^U(n_{ij})}{N_{ij}}, \tag{A35}$$

$$\langle\overline{T}_{\Delta}\rangle = \frac{\phi^U\left(\frac{T_{\Delta}p_{\omega}^2N}{M}\right)}{\frac{p_{\omega}^2N}{M}}. \tag{A36}$$

The secret key rate R is

$$R = 2p(1-p)\mu\,\mathrm{e}^{-\mu}s_1^Z\left[1-H_2(e_1^{\mathrm{ph}})\right]-Q_Z^{\mu}fH_2(E_Z^{\mu}), \tag{A37}$$

where $E_Z^{\mu} = n_{\mathrm{error}}/n_{\mathrm{t}}$ and $Q_Z^{\mu} = n_{\mathrm{t}}/N$.

A3. MP protocol

In the i-th round, Alice (Bob) prepares a coherent state $|\sqrt{k_a^i}\exp(\mathrm{i}\theta_a^i)\rangle$ ($|\sqrt{k_b^i}\exp(\mathrm{i}\theta_b^i)\rangle$) with probability p_{k_a} (p_{k_b}), where $k_a^i \in \{\mu_a, \nu_a, o\}$ and $k_b^i \in \{\mu_b, \nu_b, o\}$. Let $k = (k_a^i+k_a^j, k_b^i+k_b^j)$ denote the group of intensities in the i-th and j-th rounds, with $k_a = k_a^i+k_a^j$ and $k_b = k_b^i+k_b^j$. We define

$$y = (1-p_{\mathrm{d}})\exp\left(-\frac{k_a^i\eta_a+k_b^i\eta_b}{2}\right), \quad \omega = \sqrt{k_a^i\eta_ak_b^i\eta_b}, \quad \theta^i = \theta_a^i-\theta_b^i, \tag{A38}$$

where $\eta_a = \eta_b = \eta_{\rm d}10^{-\alpha L/10}$. The response probabilities for only the L or R detector are given by[55]

$$q^{L,\theta^i}_{k^i_a k^i_b} = y\left(\mathrm{e}^{\omega\cos\theta^i} - y\right), \tag{A39}$$

$$q^{R,\theta^i}_{k^i_a k^i_b} = y\left(\mathrm{e}^{-\omega\cos\theta^i} - y\right). \tag{A40}$$

The average response probability during each round is

$$p = \Pr(C^i = 1) = \sum_{k^i_a k^i_b} p_{k^i_a} p_{k^i_b} \Pr(C^i = 1 \mid k^i_a k^i_b), \tag{A41}$$

where C^i denotes the response of the i-th round, $C^i = 0$ represents an invalid event, $C^i = 1$ represents an effective event, and

$$\Pr(C^i = 1 \mid k^i_a k^i_b) = 2y\left[I_0(\omega) - y\right]. \tag{A42}$$

The expected pair number generated during each round is

$$r_p(p,l) = \left\{\frac{1}{p[1-(1-p)^l]} + \frac{1}{p}\right\}^{-1}. \tag{A43}$$

The number of effective detections in the Z-pair and '0'-pair is

$$\begin{aligned} n^k_Z \;=\; & \frac{Nr_p}{p^2}\sum_k \Pr(k^i_a k^i_b k^j_a k^j_b)\Pr(C^i = 1 \mid k^i_a k^i_b) \\ & \times \Pr(C^j = 1 \mid k^j_a k^j_b), \end{aligned} \tag{A44}$$

where $k \in \{(\mu_a,\mu_b),(\mu_a,\nu_b),(\mu_a,o),(\nu_a,\mu_b),(\nu_a,\nu_b),(\nu_a,o),(o,\mu_b),(o,\nu_b),(o,o)\}$, and

$$\Pr(k^i_a k^i_b k^j_a k^j_b) = p_{k^i_a} p_{k^i_b} p_{k^j_a} p_{k^j_b}. \tag{A45}$$

The number of error effective detections is

$$\begin{aligned} m^k_Z \;=\; & (1-e_{\rm d})\frac{Nr_p}{p^2} \\ & \times \sum_{k_1} \Pr(k^i_a k^i_b k^j_a k^j_b)\Pr(C^i = 1 \mid k^i_a k^i_b)\Pr(C^j = 1 \mid k^j_a k^j_b) \\ & + (1-e_{\rm d})\frac{Nr_p}{p^2} \\ & \times \sum_{k_2} \Pr(k^i_a k^i_b k^j_a k^j_b)\Pr(C^i = 1 \mid k^i_a k^i_b)\Pr(C^j = 1 \mid k^j_a k^j_b), \end{aligned} \tag{A46}$$

where k_1 requires $k^i_a = k^i_b = o$ and k_2 requires $k^j_a = k^j_b = o$ for $k \in \{(\mu_a,\mu_b),(\mu_a,\nu_b),(\nu_a,\mu_b),(\nu_a,\nu_b)\}$. For $k \in \{(\mu_a,o),(\nu_a,o),(o,\mu_b),(o,\nu_b),(o,o)\}$, we have

$$m^k_Z = \frac{1-e_{\rm d}}{2} n^k_Z. \tag{A47}$$

For $k \in \{(2\mu_a,o),(2\nu_a,o),(o,2\mu_b),(o,2\nu_b)\}$, the number of effective detections is

$$\begin{aligned} n^k_X \;=\; & \frac{Nr_p}{p^2}\Pr(k^i_a k^i_b k^j_a k^j_b)\Pr(C^i = 1 \mid k^i_a k^i_b) \\ & \times \Pr(C^j = 1 \mid k^j_a k^j_b), \end{aligned} \tag{A48}$$

and $m^k_X = n^k_X/2$.

For $k \in \{(2\mu_a,2\mu_b),(2\mu_a,2\nu_b),(2\nu_a,2\mu_b),(2\nu_a,2\nu_b)\}$,

$$\begin{aligned} n^k_X \;=\; & \frac{Nr_p}{p^2}\frac{\Delta}{\pi}\Pr(k^i_a k^i_b k^j_a k^j_b)\Bigg[y^2\int_0^{2\pi}\frac{\mathrm{d}\theta^i}{2\pi}\int_{-\Delta}^{\Delta}\frac{\mathrm{d}\delta}{2\Delta} \\ & \times\left(\mathrm{e}^{\omega\cos\theta^i} + \mathrm{e}^{-\omega\cos\theta^i}\right)\left(\mathrm{e}^{\omega\cos(\theta^i+\delta)} + \mathrm{e}^{-\omega\cos(\theta^i+\delta)}\right) \\ & -8y^3 I_0(\omega) + 4y^4\Bigg]. \end{aligned} \tag{A49}$$

The quantity without considering detector error $e_{\rm d}$ is

$$\begin{aligned} m^{k,0}_X \;=\; & \frac{Nr_p}{p^2}\frac{2\Delta}{\pi}\Pr(k^i_a k^i_b k^j_a k^j_b)\Bigg[y^2\int_0^{2\pi}\frac{\mathrm{d}\theta^i}{2\pi}\int_{-\Delta}^{\Delta}\frac{\mathrm{d}\delta}{2\Delta} \\ & \times\left(\mathrm{e}^{\omega(\cos\theta^i - \cos(\theta^i+\delta))} + \mathrm{e}^{-\omega(\cos\theta^i - \cos(\theta^i+\delta))}\right) \\ & -4y^3 I_0(\omega) + 2y^4\Bigg], \end{aligned} \tag{A50}$$

$$m^k_X \;=\; (1-e_{\rm d})m^{k,0}_X + e_{\rm d}(n^k_X - m^{k,0}_X). \tag{A51}$$

The lower bound of $\langle yz_1\rangle$ is

$$\langle yz_1\rangle^L = \frac{Y^L_+ - Y^U_-}{a_1 a'_1 \tilde{b}_{12}}, \tag{A52}$$

where $\tilde{b}_{12} = b_1 b'_2 - b'_1 b_2$, and

$$\begin{aligned} Y^L_+ \;=\; & \min: \frac{a'_1 b'_2}{N^{(\nu,\nu)}_Z}\langle n^{(\nu,\nu)}_Z\rangle + \frac{a_1 b_2 a'_0}{N^{(o,\mu)}_Z}\langle n^{(o,\mu)}_Z\rangle \\ & + \frac{a_1 b_2 b'_0}{N^{(\mu,o)}_Z}\langle n^{(\mu,o)}_Z\rangle + \frac{a'_1 b'_2 a_0 b_0 - a_1 b_2 a'_0 b'_0}{N^{(o,o)}_Z}\langle n^{(o,o)}_Z\rangle, \end{aligned} \tag{A53}$$

$$\begin{aligned} Y^U_- \;=\; & \max: \frac{a_1 b_2}{N^{(\mu,\mu)}_Z}\langle n^{(\mu,\mu)}_Z\rangle + \frac{a'_1 b'_2 a_0}{N^{(o,\nu)}_Z}\langle n^{(o,\nu)}_Z\rangle \\ & + \frac{a'_1 b'_2 b_0}{N^{(\nu,o)}_Z}\langle n^{(\nu,o)}_Z\rangle, \end{aligned} \tag{A54}$$

$$N^k_Z \;=\; \frac{N}{2}\sum_k \Pr(k^i_a k^i_b k^j_a k^j_b), \tag{A55}$$

and a_n, a'_n, b_n, b'_n denote Poisson probabilities

$$\begin{aligned} a_n &= \frac{\nu_a^n\,\mathrm{e}^{-\nu_a}}{n!}, \quad b_n = \frac{\nu_b^n\,\mathrm{e}^{-\nu_b}}{n!}, \\ a'_n &= \frac{\mu_a^n\,\mathrm{e}^{-\mu_a}}{n!}, \quad b'_n = \frac{\mu_b^n\,\mathrm{e}^{-\mu_b}}{n!}. \end{aligned} \tag{A56}$$

The upper bound of the bit error rate in X_1 is

$$\langle e^{\rm bit}_{X_1}\rangle^U = \frac{T^U_+ - T^L_-}{a_1 b_1 \langle yz_1\rangle^L}, \tag{A57}$$

where

$$T^U_+ \;=\; \max: \frac{1}{N^{(2\nu,2\nu)}_X}\langle m^{(2\nu,2\nu)}_X\rangle + \frac{\mathrm{e}^{-2\nu}}{N^{(o,o)}_X}\langle m^{(o,o)}_X\rangle, \tag{A58}$$

$$T_{-}^{L} = \min : \frac{\mathrm{e}^{-\nu}}{N_X^{(0,2\nu)}}\langle m_X^{(0,2\nu)}\rangle + \frac{\mathrm{e}^{-\nu}}{N_X^{(2\nu,0)}}\langle m_X^{(2\nu,0)}\rangle, \quad \text{(A59)}$$

with

$$N_X^{(2\nu,2\nu)} = \frac{N\Delta}{\pi}p_\nu^4, \quad N_X^{(o,2\nu)} = N_X^{(2\nu,o)} = \frac{N}{2}p_\nu^2 p_o^2,$$
$$N_X^{(o,o)} = \frac{N}{2}p_o^4. \quad \text{(A60)}$$

The single-photon effective detection number in the raw key n_{Z_1} and the single-photon phase error rate $e_{Z_1}^{\mathrm{ph}}$ can be calculated using Eqs. (A15)–(A18)

$$n_{Z_1} = \phi^L\Big(\sum_{k_Z} N_Z^k k_a k_b \mathrm{e}^{-(k_a+k_b)}\langle yz_1\rangle^L\Big), \quad \text{(A61)}$$

$$e_{Z_1}^{\mathrm{ph}} = \phi^U\Big(n_{Z_1}\langle e_{X_1}^{\mathrm{bit}}\rangle^U\Big)/n_{Z_1}, \quad \text{(A62)}$$

where k_Z requires $k_a \in \{\mu_a, \nu_a\}$, $k_b \in \{\mu_b, \nu_b\}$. Additionally, we have

$$n_Z = \sum_{k_Z} n_Z^k, \quad \text{(A63)}$$

$$E_Z = \frac{1}{n_Z}\sum_{k_Z} m_Z^k. \quad \text{(A64)}$$

Finally, the secret key rate is

$$R = \frac{1}{N}\Big\{n_{Z_1}\big[1 - H_2(e_{Z_1}^{\mathrm{ph}})\big] - n_Z f H_2(E_Z)\Big\}. \quad \text{(A65)}$$

Acknowledgments

This work was supported by the Jiangsu Provincial Key R&D Program for Industrialization Prospects and Key Core Technology Project (Grant No. BE2022071), and the National Natural Science Foundation of China (NSFC) (Grant Nos. 62471248, 62201276, and 12074194).

References

[1] Shor P W 1999 *SIAM Rev.* **41**, 303
[2] Rivest R L, Shamir A and Adleman L 1978 *Commun. ACM* **21**, 120
[3] Diffie W and Hellman M E 2022 *Democratizing Cryptography: The Work of Whitfield Diffie and Martin Hellman* pp. 365
[4] Arute F, Arya K, Babbush R, *et al.* 2019 *Nature* **574**, 505
[5] Kent S and Seo K 2005 *Security Architecture for the Internet Protocol* (IETF)
[6] Bennett C H and Brassard G 1984 *Quantum Cryptography: Public Key Distribution and Coin Tossing*, in *IEEE Int. Conf. Computers, Systems and Signal Processing*, pp. 175
[7] Bernstein D J 2025 *Encyclopedia of Cryptography, Security and Privacy* (Springer) pp. 1846
[8] Wootters W K and Zurek W H 1982 *Nature* **299**, 802
[9] Vernam G S 1926 *J. AIEE* **45**, 109
[10] Lo H K, Curty M and Qi B 2012 *Phys. Rev. Lett.* **108**, 130503
[11] Pirandola S, Laurenza R, Ottaviani C and Banchi L 2017 *Nat. Commun.* **8**, 1
[12] Lucamarini M, Yuan Z L, Dynes J F and Shields A J 2018 *Nature* **557**, 400
[13] Wang X B, Yu Z W and Hu X L 2018 *Phys. Rev. A* **98**, 062323
[14] Ma X, Zeng P and Zhou H 2018 *Phys. Rev. X* **8**, 031043
[15] Cui C, Yin Z Q, Wang R, Chen W, Wang S, Guo G C and Han Z F 2019 *Phys. Rev. Appl.* **11**, 034053
[16] Liu J, Jiang Q, Ding H, Ma X, Sun M, Xu J, Zhang C H, Xie S, Li J and Zeng G 2023 *Sci. China Inf. Sci.* **66**, 189402
[17] Li W, Zhang L, Lu Y, Li Z P, Jiang C, Liu Y, Huang J, Li H, Wang Z and Wang X B 2023 *Phys. Rev. Lett.* **130**, 250802
[18] Zeng P, Zhou H, Wu W and Ma X 2022 *Nat. Commun.* **13**, 3903
[19] Xie Y M, Lu Y S, Weng C X, Cao X Y, Jia Z Y, Bao Y, Wang Y, Fu Y, Yin H L and Chen Z B 2022 *PRX Quantum* **3**, 020315
[20] Zhu H T, Huang Y, Liu H, Zeng P, Zou M, Dai Y, Tang S, Li H, You L and Wang Z 2023 *Phys. Rev. Lett.* **130**, 030801
[21] Zhou L, Lin J, Xie Y M, Lu Y S, Jing Y, Yin H L and Yuan Z 2023 *Phys. Rev. Lett.* **130**, 250801
[22] Zhu H T, Huang Y, Pan W X, Zhou C W, Tang J, He H, Cheng M, Jin X, Zou M and Tang S 2024 *Optica* **11**, 883
[23] Luo D, Liu X, Qin K, Zhang Z and Wei K 2024 *Phys. Rev. A* **110**, 022605
[24] Liu X, Luo D, Zhang Z and Wei K 2023 *Phys. Rev. A* **107**, 062613
[25] Guo M, Cao Y, Zhu J, Zhou X, Zhang C, Yu X, Zhao Y, Zhang J and Wang Q 2023 *J. Opt. Commun. Netw.* **15**, 700
[26] Cao Y, Zhao Y, Wang Q, Zhang J, Ng S X and Hanzo L 2022 *IEEE Commun. Surv. Tut.* **24**, 839
[27] Cao Y, Zhao Y, Zhang J and Wang Q 2022 *IEEE Commun. Mag.* **60**, 38
[28] Cao Y, Zhao Y, Zhang J, Wang Q, Niyato D and Hanzo L 2022 *IEEE Netw.* **36**, 14
[29] Cao Y, Zhao Y, Li J, Lin R, Zhang J and Chen J 2021 *IEEE J. Sel. Areas Commun.* **39**, 2701
[30] Huang C, Chen Y, Luo T, He W, Liu X, Zhang Z and Wei K 2024 *Sci. China Phys. Mech. Astron.* **67**, 240312
[31] Wei K, Li W, Tan H, *et al.* 2020 *Phys. Rev. X* **10**, 031030
[32] Elliott C, Pearson D and Troxel G 2003 *Proceedings of the 2003 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications* pp. 227
[33] Berzanskis A, Hakkarainen H, Lee K and Hussain M R 2009 *Method of Integrating QKD with IPsec*, U.S. Patent 7,602,919
[34] Zhang P, Chen N, Shen S, Yu S, Wu S and Kumar N 2022 *IEEE Wireless Communications* **31**, 141
[35] Marksteiner S and Maurhart O 2015 *9th Int. Conf. Quantum, Nano/Bio, and Micro Technologies*, pp. 35
[36] Cano A, Garcia C R, Frantz R, Monroy I T, Imaña J L and Vegas J J 2024 *2024 IEEE 32nd International Conference on Network Protocols (ICNP)* pp. 1
[37] Li J, Zheng P, Li Z, Xue K, Xie Z, Yu N, Sun Q and Lu J 2025 *IEEE Network* **39**, 180
[38] Sibson P, Kennard J, Crabtree T, Wright P, White C, Salas E H, Lord A, Grammel G, Mead B and Radev R 2024 *Optical Fiber Communication Conference* (Optica Publishing Group) pp. Th3B–4
[39] Alia O, Huang A, Luo H, Amer O, Pistoia M and Lim C 2024 *Optical Fiber Communication Conference* (Optica Publishing Group) pp. Th3B–5
[40] Lawo D C, Abu Bakar R, Cano Aguilera A, Cugini F, Imaña J L, Tafur Monroy I and Vegas Olmos J J 2024 *Future Internet* **16**, 300
[41] Gao X, Xue K, Li J, Li Z, Wu J, Yu N, Sun Q and Lu J 2025 *IEEE Comm. Mag.* **63** 148
[42] Cao Y, Yu X, Zhao Y, Zhang C, Zhou X, Zhang J and Wang Q 2025 *Chin. Phys. B* **34**, 010310
[43] Garcia C R, Cano A, Olmos J V, Rommel S and Monroy I T *2025 Optical Fiber Communications Conference and Exhibition (OFC)* (IEEE), pp. 1
[44] Fluhrer S, Kampanakis P, McGrew D and Smyslov V 2020 *RFC 8784*
[45] Harkins D and Carrel D 1998 *RFC 2409: The Internet Key Exchange (IKE)*
[46] Kaufman C 2005 *RFC 4306: Internet Key Exchange (IKEv2) Protocol* (RFC Editor)
[47] Joan D and Vincent R 2002 *Information Security and Cryptography* (Springer)
[48] Huttner B, Imoto N, Gisin N and Mor T 1995 *Phys. Rev. A* **51**, 1863
[49] Hwang W Y 2003 *Phys. Rev. Lett.* **91**, 057901
[50] Wang X B 2005 *Phys. Rev. Lett.* **94**, 230503
[51] Lo H K, Ma X and Chen K 2005 *Phys. Rev. Lett.* **94**, 230504
[52] Ma X, Qi B, Zhao Y and Lo H K 2005 *Phys. Rev. A* **72**, 012326
[53] Gottesman D, Lo H K, Lutkenhaus N and Preskill J 2004 *Security of Quantum Key Distribution with Imperfect Devices*, International Symposium on Information Theory
[54] Jiang C, Yu Z W, Hu X L and Wang X B 2019 *Phys. Rev. Appl.* **12**, 024061
[55] Wang Z H, Wang R, Yin Z Q, Wang S, Lu F Y, Chen W, He D Y, Guo G C and Han Z F 2023 *Commun. Phys.* **6**, 265
[56] Chernoff H 1952 *Ann. Math. Stat.* **23**, 493–507
[57] Alia O, Huang A, Luo H, Amer O, Pistoia M and Lim C 2024 arXiv:2405.04415 [quant-ph]
[58] Fan-Yuan G J, Lu F Y, Wang S, Yin Z Q, He D Y, Chen W, Zhou Z, Wang Z H, Teng J and Guo G C 2022 *Optica* **9**, 812