White Paper

Telco & Communications Service Providers
Cloud Service Providers
Federal & Defense

Arqit and Intel Test Post Quantum Cryptography (PQC) Solution

Arqit SKA-Platform* achieves quantum-secure and high-performance IPsec throughput on 4th Gen Intel® Xeon® Scalable processor-based servers

Optimized networking solutions are easy to configure and operate while also remaining fast, reliable, and secure. FD.io's Vector Packet Processor (VPP) is a fast, scalable layer 2-4 multi-platform network stack, that has become a networking staple across the world due to its high-performance. While FD.io does not have a production-ready Internet Key Exchange v2 (IKEv2) implementation, it can be combined with strongSwan, an open-source and multiplatform IPsec virtual private network (VPN) library, to provide a comprehensive and secure networking solution.

Intel has previously demonstrated the performance of VPP combined with strongSwan (VPP-SSwan), achieving a 1.89 Terabit no drop rate (NDR) IPsec tunnel in tests using a server based on the 4th Gen Intel® Xeon® Scalable processor[1].

However, the key exchange method used by VPP-SSwan to establish IPsec connections will be broken in the near future by cryptographically relevant quantum computers (CRQC). The risk posed is grave, as stated in the White House-published National Security Memorandum 10 (NSM-10)[2].

> "When it becomes available, a [CRQC] could jeopardize civilian and military communications, undermine supervisory and control systems for critical infrastructure, and defeat security protocols for most Internet-based financial transactions."

Even before quantum computers become available, it was possible for attackers to steal and store data today and decrypt it in the future when they can get a CRQC. Information that needs to be kept secret for a long time (state secrets, personal health and genomics data, trade secrets, financial data, etc.) are already at risk since they can be easily siphoned from public networks and stored encrypted in a data silo until it can be decrypted. This is known as a store now, decrypt later (SNDL) attack.

Arqit, an Intel® Network Builders community member, offers its SKA-Platform, which is a cloud-hosted or on-prem service that can secure networks with encryption that is unbreakable by a quantum computer. The SKA-Platform allows endpoints to upgrade the security of communication channels they create, such as the IPsec tunnels created by VPP-SSwan.

This document outlines how the Arqit SKA-Platform can be used to enhance the existing FD.io VPP-SSwan setup via highly secure quantum-safe authentication and symmetric key agreement (SKA) between endpoints in line with existing standards and recommendations, including NSM-10.

Table of Contents

Performance testing verifies that the SKA-Platform can be used to upgrade IPsec connections such that they are quantum secure while maintaining maximum performance, thereby showing how Arqit and Intel can provide a joint post-quantum cryptography (PQC) solution without compromising the 1.89 Terabit NDR IPsec tunnel throughput achieved by Intel[1], a leadership feat in the world of post-quantum encryption.

Additionally, the solution is optimal for small form factor devices where security was previously limited by size, weight and power (SWaP) requirements and the need for specialist hardware crypto devices. Arqit has developed the SKA-Platform as a full endpoint security solution for a range of device types, with robust methods for endpoint authentication, provisioning, key agreement, and management that are provably secure. This is a hybrid solution that provides defense in depth and crypto agility for future flexibility.

This document outlines how the SKA-Platform can interoperate with existing IPsec solutions such as VPP-SSwan, enabling an out-the-box PQC-ready solution on a variety of Intel® architecture-based hardware. High-grade compute, such as the 4th Gen Intel Xeon Scalable processors, provide extremely high encrypted network throughput rates, such as the 1.89 Tb NDR IPsec tunnel achieved previously while such encryption being PQC-ready. Additionally, the solution can also be fully deployed on a platform based on Intel® NetSec Accelerator Reference Design (i.e., a "server on a card"), opening possibilities across edge computing, IoT and small form factor devices.

Arqit's Solution: SKA-Platform

The following sections outline the device lifecycle, describing how an endpoint is registered with the SKA-Platform and subsequently agrees symmetric keys for connections with other endpoints. This is all facilitated by lightweight software at the endpoint, providing flexible tools to ensure any device can utilize the SKA-Platform.

Authentication

Authentication, or the ability to ensure the identity of an individual, has central importance in security. Arqit takes an approach to authentication that ensures the highest level of security available today, so that all devices within a network can be trusted.

This process begins with the delivery of a master authentication key, which serves as a root-of-trust and can be stored using secure or verified hardware. This key is generated in the SKA-Platform and is then either delivered securely out-of-band or encapsulated with keys generated through multiple post-quantum key encapsulation mechanism (KEM) algorithms.

In the latter scenario, the encapsulated key is delivered to the endpoint at point of registration. These algorithms are drawn from the candidates in NIST's Post-Quantum Cryptography Standardization Process[3] and are intended as quantum-safe replacements for public-key protocols.

These methods are not yet standardized so Arqit uses multiple KEMs of different types to increase assurance should any single KEM be weakened, and since this is a one-time registration process which is not time sensitive, KEMs with larger key sizes and compute requirements can be used (e.g. Classic McEliece cryptography system). All PQA KEM key exchanges are made over a TLS channel, ensuring at least classical protection, in line with recommendations from NIST and others that require hybrid cryptography.

Once the initial master authentication key has been delivered it is used by the endpoint to form the initial authentication key that will strongly authenticate with the SKA-Platform. In addition, the authentication key is ratcheted with each successive authentication, meaning a new authentication key is derived from the previous one in a way that cannot be reversed. This ensures that each authentication key has a relatively short lifetime (e.g., minutes or hours), configurable by the user, that mitigates spoofing attacks and simplifies key revocation. The authentication method used employs irreversible hash functions that are not breakable by any known classical or quantum algorithm.

This final symmetric authentication key forms the basis of the security association between the endpoint and the platform meaning that any information the platform sends to the endpoint can be considered quantum safe.

Symmetric Key Agreement

Almost all secure communication today is based on two parties sharing a symmetric key. The party sending data uses the key to encrypt data, and the recipient uses the same key to decrypt it. The encryption and decryption ciphers (e.g., AES256) are extremely efficient and are often optimized at the hardware layer. If the key has sufficient length (i.e., greater than 128 bits), these methods are known to be extremely secure and robust against even quantum-based attacks.

The problem with these methods is how two endpoints, Alice and Bob, agree to a shared symmetric key in the first place. This is known as the key distribution problem – if Alice must transmit the key to Bob in advance it creates the opportunity for a bad actor to steal the key and eavesdrop on their communication.

There are two widely used methods to solve this problem:

1. *Manual key delivery/pre-shared keys*. A trusted courier manually delivers the key to Alice and Bob without using a network. This can be highly secure but is also extremely impractical and expensive for large, disparate networks. These keys are also infrequently replaced, meaning large volumes of information can be decrypted if one is lost or stolen. This can be an O[n^2] solution in the worst case.

2. *Public-key protocols*. These rely on a mathematical problem that is difficult for a classical computer to invert, e.g. factorizing large integers. The most used protocol is Diffie-Hellman key exchange. While these methods are much more convenient than manual delivery, the functions they rely on will be efficiently inverted by quantum computers in the future making them much less secure than initially believed.

Arqit's alternative solution is *Symmetric Key Agreement* which combines the high security of manual key delivery with the convenience and scalability of public-key protocols.

This balance is achieved through the introduction of a split-trust[4] third party, the Arqit SKA-Platform, which assists Alice and Bob in creating symmetric keys on demand. Entities register once with the platform and dynamically agree on keys between themselves, leading to a much simpler solution. This method of key agreement is secure because it relies on symmetric cryptography, which is a type of PQC that's extremely secure against attacks including by quantum computers. It's also efficient and scalable due to the hub-and-spoke topology of a single platform coordinating key agreement among all endpoints.

Alice now wants to create a shared key with another endpoint, Bob, which they can use to secure communication between them. We assume that both Alice and Bob are fully authenticated and provisioned with the platform. Arqit has created its own protocol that allows Alice and Bob to create a shared symmetric key using the platform as a broker as seen below:

1. Alice and Bob use a confidential channel to create a shared secret using a traditional (not necessarily quantum-safe) method, e.g. over TLS.

2. Alice sends a request to the SKA-Platform over the quantum-safe channel using the session key, created when she authenticated using her authentication key, to create an intermediate key based on knowledge of Alice's ID from her authentication token and Bob's ID sent by Alice. The SKA-Platform takes a key from its HSM and hashes it with this information to create the intermediate key, which is then returned to Alice.

3. Bob sends a request to the server, also over a quantum-safe connection using his session key, and receives the same intermediate key based on his ID from his authentication token and Alice's ID sent by Bob.

4. Both Alice and Bob now hash the intermediate key with their shared secret and recover the same shared symmetric key.

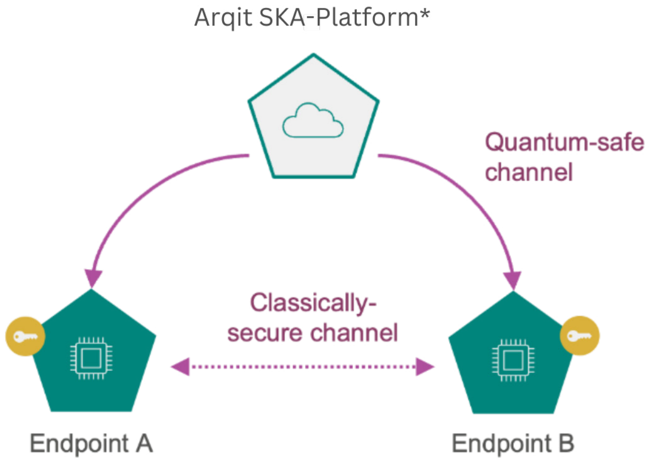


Figure 1. Schematic showing how two endpoints interact with the SKA to create a symmetric key.

Importantly, the platform does not have all the information it needs to create the same key as it does not know the secret that Alice and Bob shared in step 2. This is a split-trust mechanism, meaning that information is split between multiple channels. Any attack on the SKA-Platform would not result in the loss of encryption keys, keeping the data secure.

This shared symmetric key can now be used in many ways to secure the data passing between endpoints, e.g. in an IPsec tunnel, or at the application level to encrypt data with AES. A new key can be requested as often as required for the use case. Since the key is a standard 256-bit symmetric key it can also be easily mixed with other keys generated through other methods for a robust defense-in-depth approach.

Arqit SKA-Platform and Intel: Real World PQC Use Cases

The principles outlined in the previous sections can be applied to a wide range of real-world use cases. In writing this document, one specific architecture was developed for performance testing and proof-of-concept purposes, but this can be extended for deployment across edge computing, IoT, legacy systems and more.

Site-to-site PQC IPsec

The performance test results in the following section were produced using a standard site-to-site IPsec gateway between two GCP C3 hosts based on 4th Gen Intel Xeon Scalable processors. The underlying technology builds upon the VPP-SSwan work completed previously, where the two popular open-source projects, strongSwan and VPP, were combined by Intel to demonstrate easy-to-use functionality and incredibly fast packet processing speeds on Intel Xeon Scalable processor platforms.

The reason this IPsec VPN setup can be upgraded from classical cryptography to PQC is because strongSwan implements the RFC 8784 standard[5] from version 5.7.0 onwards. This allows a post-quantum pre-shared key (PPK) in addition to the authentication method that is already provided by IKEv2. This additional PPK is stirred into the exchange such that quantum resistance is provided to the IPsec security associations (SAs), protecting data-in-transit against quantum attack.

The SKA-Platform removes the cumbersome requirement of pre-sharing this key in typical systems, instead generating the key directly at the endpoints. The Arqit endpoint software is lightweight and easily deployable on any device, agreeing symmetric keys that are provided as PPKs to strongSwan as and when the IPsec VPN connection is required. Furthermore, keys can be automatically rotated extremely frequently, unlike typical pre-shared keys which are difficult to change once implemented in a system.

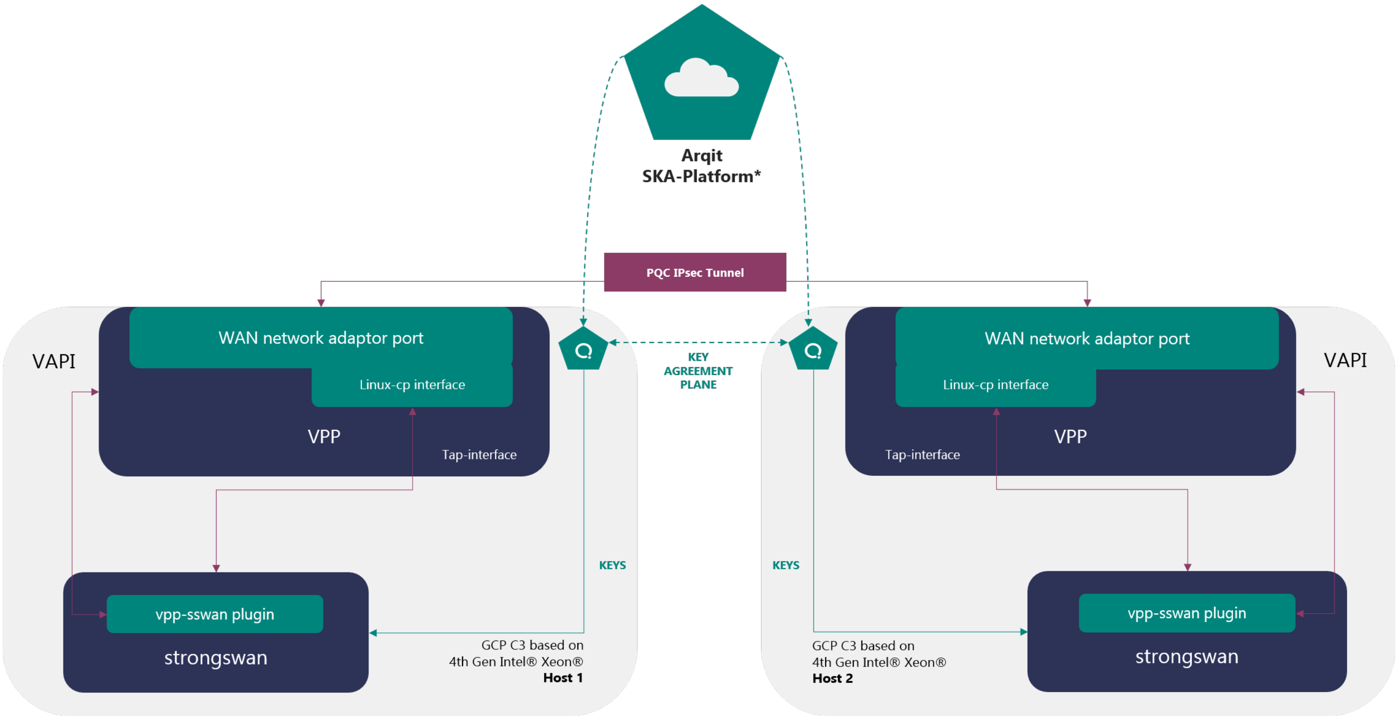


Figure 2. Example quantum-secure IPsec architecture using the SKA-Platform and VPP-SSwan on two Intel hosts.

Testing and Results

The throughput test deployment is not designed for hitting maximum performance but is instead architected such that the results prove that IPsec throughput remains unchanged when the SKA-Platform is incorporated into the system. The main aim was to prove that an out-of-the-box PQC-ready solution could be achieved without compromising performance. With the correct network configuration and hardware stack, it would then be possible to achieve the record breaking 1.89 Tb solution with PQC in place to mitigate against the quantum attack.

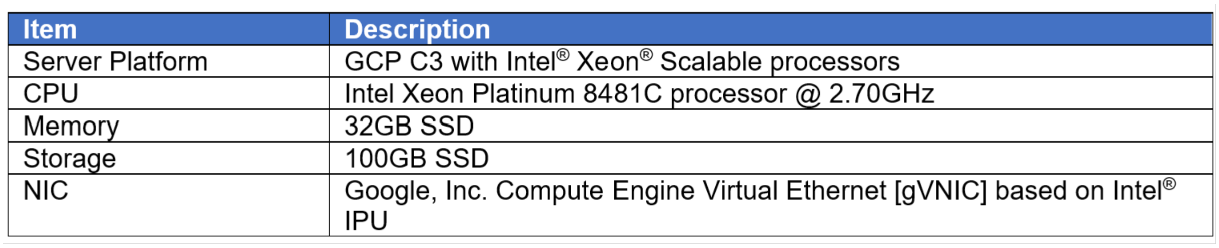
Deloyment Setup

The deployment architecture is exactly as laid out in Figure 2. Two Intel hosts establish an IPsec VPN tunnel using VPP-SSwan such that data can be transmitted between them. The AES256-GCM encryption algorithm was used along with a 256-bit elliptic curve group to provide the public key infrastructure (PKI) component needed to agree on the final key. However, two different scenarios were considered:

1. The two hosts do not use the SKA-Platform and therefore do not mix a PPK into the key material, so this is not quantum safe as it is fully dependent on the underlying PKI.

2. The two hosts use the SKA-Platform to agree on the shared PPK, which stirred into the key material to enable a quantum safe connection.

The specifications of the two Intel hosts used for testing can be found in Table 1. As noted, the individual elements in this case hold little importance as it is the comparison between the throughput measurements that provides real value.



Item	Description
Server Platform	GCP C3 with Intel® Xeon® Scalable processors
CPU	Intel Xeon Platinum 8481C processor @ 2.70GHz
Memory	32GB SSD
Storage	100GB SSD
NIC	Google, Inc. Compute Engine Virtual Ethernet [gVNIC] based on Intel® IPU

Table 1. Testbed system setup.

Parameter	Value	Description
Time	300 seconds	A total of five minutes throughput runtime was used for each test.
Omit (TCP slowstart)	5 seconds	The first five seconds of each test were omitted to avoid results from the TCP slowstart[7] period.
Maximum segment size (MSS)	1420 bytes	Set to the MTU (1460 bytes for the Ethernet adaptors) minus 40 bytes as standard.
Type-of-service	IPTOS_THROUGHPUT 0x08	The type-of-service for outgoing packets[8], set to maximize throughput for these tests.

Table 2. Configuration for the iperf3 throughout measurement.

Throughput Testing Process

The throughput tests were executed using iperf3[6], a simple tool for measuring IP network bandwidth. Once the IPsec tunnel was established in each case, one host acts as the iperf3 client and the other the server, with the tool transmitting data for five minutes. The parameters can be found in Table 2.

Results

The results (see Figure 3) demonstrate that IPsec throughput was unchanged with and without the SKA-Platform, proving that a PQC solution can be achieved without compromising performance.

The maximum throughput achieved during testing was identical in both cases, with a value of 1.55 Gbps achieved. The average throughput was 1.44 Gbps and 1.37 Gbps with and without the SKA-Platform respectively, again proving Arqit can provide PQC capability without any reduction in throughput performance.

The results would be identical if not for random fluctuations in network performance (e.g., a throughput measurement in a given second could differ from the average by up to ~0.3 Gbps).

Overall, it is clear that a quantum secure IPsec tunnel can be achieved without compromising performance if Arqit SKA provides a dynamically generated PPK.

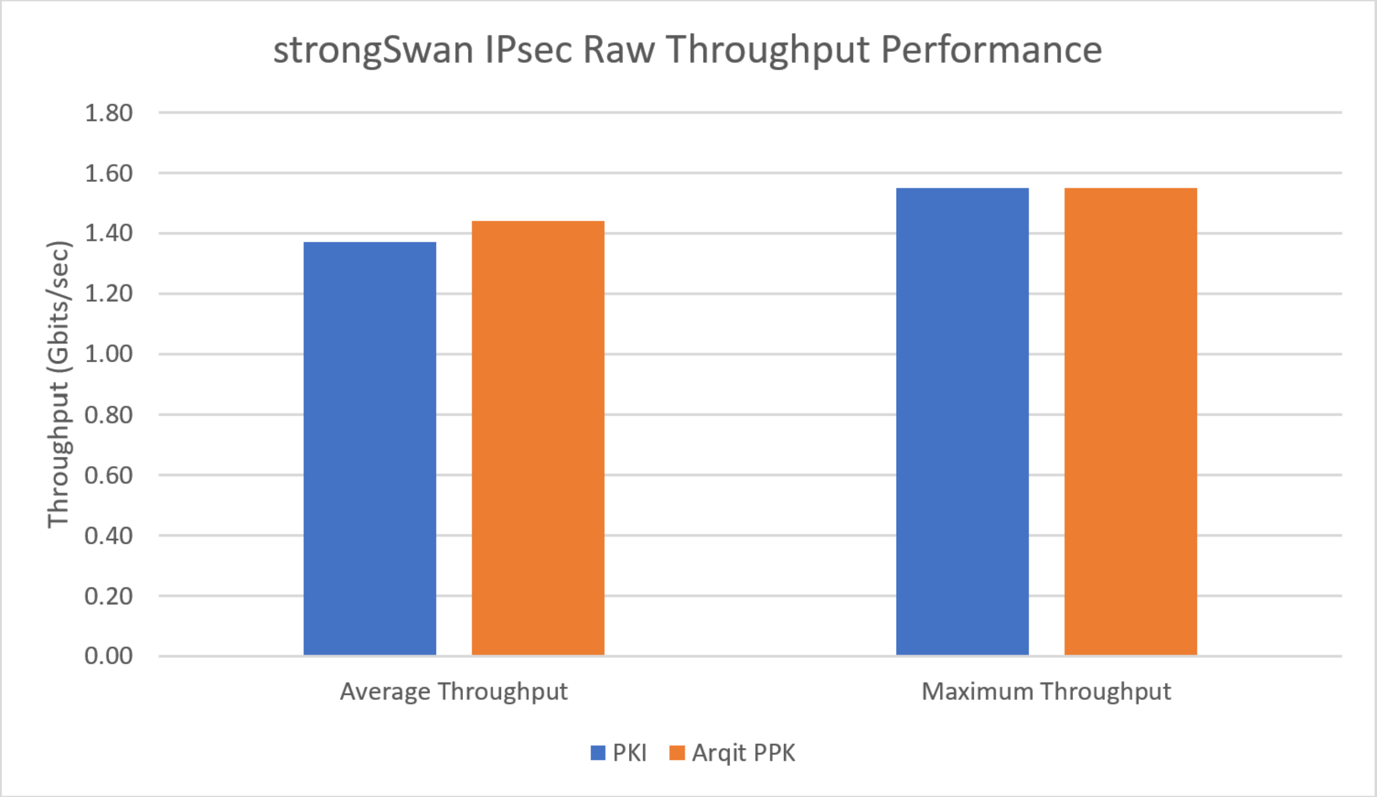


Figure 3. IPsec throughput performance with and without the SKA-Platform.

Summary

The Arqit SKA-Platform is a cloud-hosted or on-prem service that can secure networks with encryption that is unbreakable by a quantum computer. This allows endpoints to upgrade the security of communication channels they create such as IPsec tunnels.

VPP-SSwan is a software solution that combines the advantages of two popular open-source projects, strongSwan and FD.io VPP. strongSwan provides a user-friendly interface for setting up secure communication channels using VPNs, while VPP is a high-performance networking stack that uses hardware acceleration to achieve incredibly fast packet processing speeds on platforms based on Intel Xeon Scalable processors.

By merging these three technologies, this integration provides users with an easy-to-use VPN solution that delivers exceptional performance and is also quantum safe. With VPP's packet processing capabilities, VPP-SSwan can handle large volumes of network traffic with low latency, making it ideal for use in high-performance computing environments. The Arqit SKA-Platform and lightweight endpoint software enhances the VPN solution, enabling post-quantum security without compromising performance.

In addition to high-performance architectures, the lightweight nature of the software solution means small form factor devices can now be provided with this technology. While these endpoints and systems were previously limited by SWaP requirements and hardware crypto devices, this solution now provides quantum secure symmetric encryption to small form factor for the first time.

Furthermore, the Arqit SKA-Platform and VPP-SSwan support a wide range of protocols and encryption algorithms, offering users the flexibility to select the best option for their specific needs. With strongSwan's user-friendly interface and Arqit's easy-to-use SKA-Platform and SDK, setting up and managing VPN connections is simple, even for those with limited networking experience.

Overall, the Arqit SKA-Platform and VPP-SSwan is an excellent choice for anyone looking for a quantum safe VPN solution that combines ease-of-use with high-performance networking capabilities and lightweight software requirements.

Learn More

Arqit Website

strongSwan

FD.io VPP

4th Gen Intel Xeon Scalable processors

Intel NetSec Accelerator Reference Design

Intel Network Builders

[1] Intel, "FD.io VPP-SSwan and Linux-CP – Integrate StrongSwan with World's First Open Sourced 1.89 Tb IPsec Solution Technology Guide" (Network & Edge Platform documents, Intel Corporation, 2023)

[2] White House, "National Security Memorandum on Promoting United States Leadership in Quantum Computing While Mitigating Risks to Vulnerable Cryptographic Systems" (official memorandum, Washington, DC: White House, 2022)

[3] "Post-Quantum Cryptography", NIST, https://csrc.nist.gov/projects/post-quantum-cryptography

[4] Despite the introduction of a third party, we consider the SKA to be split trust because it does not have enough information to know the key agreed by Alice and Bob.

[5] IETF, "Mixing Preshared Keys in the Internet Key Exchange Protocol Version 2 (IKEv2) for Post-quantum Security" (Internet Standards Track document, Internet Engineering Task Force, 2020).

[6] iperf3, https://iperf.fr/ (iperf Official Website, 2024).

[7] TCP Slowstart, https://en.wikipedia.org/wiki/TCP_congestion_control#Slow_start (Wikipedia, 2024).

[8] IETF, "Type of Service in the Internet Protocol Suite" (Internet Standards Track document, Internet Engineering Task Force, 1992).

Notices & Disclaimers

Performance varies by use, configuration and other factors. Learn more on the Performance Index site.
Performance results are based on testing as of dates shown in configurations and may not reflect all publicly available updates. No product or component can be absolutely secure.
Intel does not control or audit third-party data. You should consult other sources to evaluate accuracy.
Intel technologies may require enabled hardware, software or service activation.
No product or component can be absolutely secure.
Your costs and results may vary.