March 2022

QUANTUM SECURE MULTICAST

Niranjan M M

QUANTUM SECURE MULTICAST

AUTHOR:
Niranjan M M

ABSTRACT

The existing techniques describe methods to provide quantum resistant pre-shared key distribution to secure different protocols used in different deployments. But existing methods does not consider providing quantum resistant key distribution for securing the multicast traffic as well multicast routing protocols (e.g., PIM) used in different deployments. The techniques presented herein is to distribute post quantum keys to be used for securing the multicast traffic and multicast routing protocol.

DETAILED DESCRIPTION

The existing techniques describe methods to provide quantum resistant pre-shared key distribution to secure different protocols used in different deployments. But the existing methods does not consider providing quantum resistant key distribution for securing the multicast traffic as well multicast routing protocols (e.g., PIM) used in different deployments.

The techniques presented herein is to distribute post quantum keys to be used for securing the multicast traffic and multicast routing protocol. As per this method, Blockchain Provider (BP) runs QKS (Quantum Key Source) to generate unique Post Quantum Multicast Group Key PQMGK (ID, Key) per multicast group (and/or per multicast IP address). But QKS requires common seed secret value to generate unique pair of PQMGK (ID, Key), For this, BP generate common seed secret value using HKDF function, which will be shared with the multicast nodes during registration.

As part of the initial node bring up, multicast nodes registers with the Blockchain Provider. After successful registration, Blockchain Provider share common seed secret value with the nodes using secure mechanisms such as PKI (asymmetric key encryption) methods. And Blockchain Provider generates unique PQMGK ID per multicast group (if it not already generated) by using QKS. Further, this PQMGK ID is used to generate unique PQMGK Key which will be used for encryption/decryption of the multicast traffic/protocol (to provide confidentiality). Blockchain Provider would add transaction T with the attributes Multicast group name, Multicast group ID,

Multicast IP address and PQMGK ID to the Hyperledger, which will be used by the sender and receiver nodes to fetch PQMGK ID to generate PQMGK Key for encryption/decryption.

Figure-1 describe overall method to provide quantum resistant key distribution for securing multicast traffic as well as multicast routing protocols.

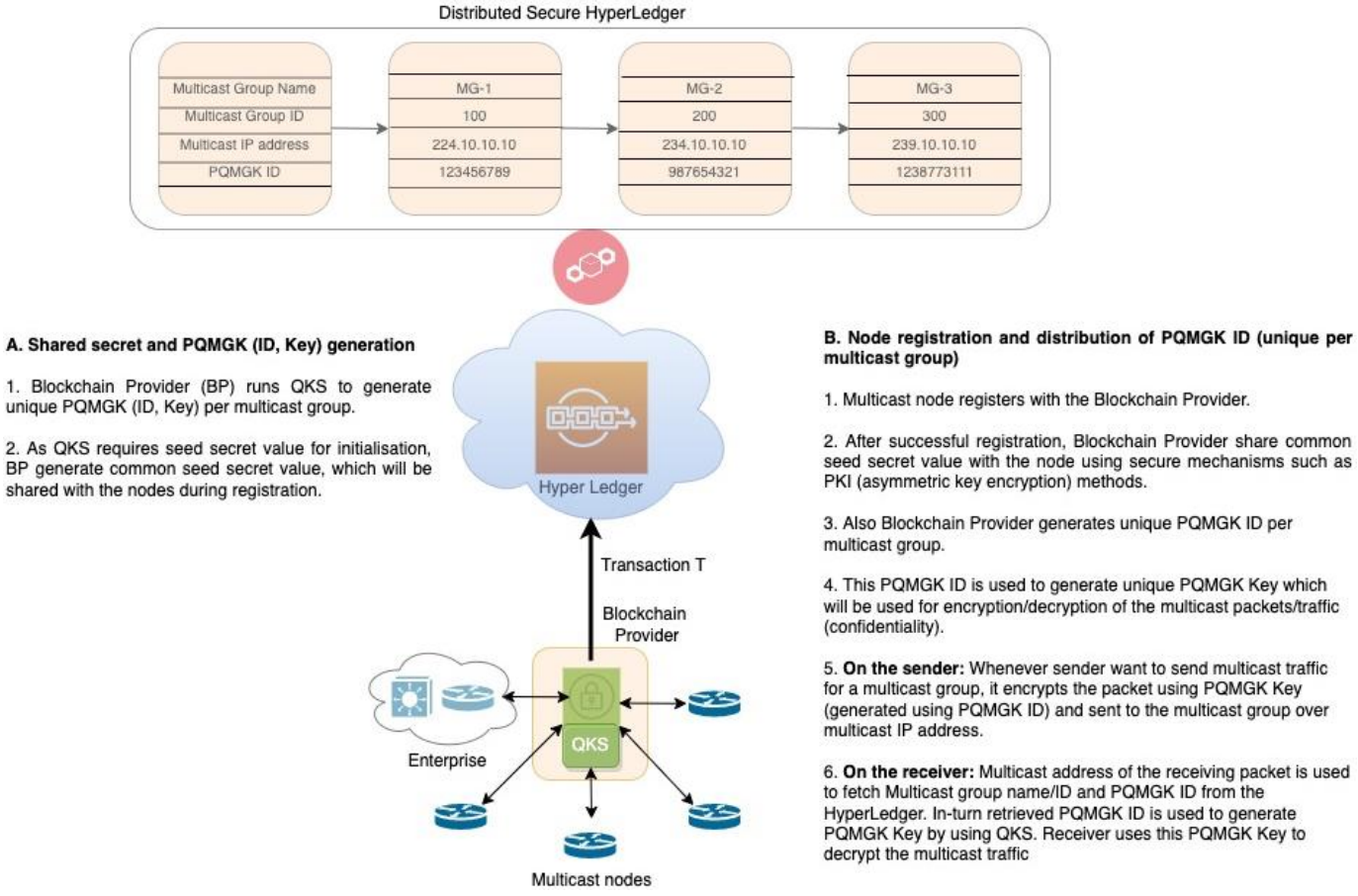

Figure-1

On the sender: Whenever sender want to send multicast traffic for a multicast group, it encrypts the packet using PQMGK Key (generated using PQMGK ID) and sent to the multicast group over multicast IP address.

On the receiver: Multicast address of the receiving packet is used to fetch multicast group name/ID and PQMGK ID from the Hyperledger. In-turn PQMGK ID is used to generate PQMGK Key by using QKS. Receiver uses this PQMGK Key to decrypt the multicast traffic.

The techniques presented herein is explained in detail as below:

A. Multicast Node Registration:

Let us say, all the nodes in the deployment (i.e., in a particular multicast group) are enabled with Hyperledger functionality. Nodes in the multicast group are authenticated with Blockchain Provider (BP) using secure credentials. Along with authentication, trustworthiness of the nodes is established using simple attestation protocol.

- To attest the node/instance/device, Blockchain Provider (BP) retrieves the Public Attestation Identity Key PK_AIK_NODE of the node and sends a "nonce" to that node (say, target node).
- The target node answers with a quote that contains this "nonce" and its current Platform Configuration Registers (PCR) values that are stored inside the local TPM of the target node.
- This quote is signed by the Private Attestation Identity Key SK_AIK_NODE of the target node to ensure the integrity of the response.
- After the receipt of the payload from the target node, BP compare the "nonce" to check the freshness of the attestation and if this matches the expected value, compare the received PCR values with a library of trusted node configurations. Note here, vTPM (or software TAM) is used for the virtual/cloud/container deployments.

B. Exchange common secret seed value:

Blockchain Provider (which acts as a Key Server) generates common secret seed value and share it with the multicast nodes using PKI (asymmetric key algorithm) method during node registration. The nodes independently use common secret seed value to initialise the QKS. Note: The Quantum Key Source (QKS) require common secret seed value to be used for initialisation, so that it can generate unique pair of PQMGK (ID, Key).

C. Exchange Post Quantum Multicast Group Key ID (PQMGK ID):

Along with common secret seed value, Blockchain Provider generates PQMGK ID (unique per multicast group) using QKS and distribute to the registered nodes by adding transaction T (with attributes as multicast group name/ID, multicast IP address and the PQMGK ID) into the Hyperledger. BP maintains list of all the multicast groups in distributed Hyperledger. BP help with

accounting; lawful intercept and they maintain immutable records. Markle tree hash algorithms (double SHA-256) are used for generation of public keys for enhanced security and enterprise consensus algorithms such as Proof of elapsed time (PoET) or Practical Byzantine Fault Tolerance (PBFT) are used to synchronise database among all nodes in the deployment. Permissioned ledger is used to allow only authenticated and trusted nodes to participate and share the multicast services.

For example: Let us say, Node-1 and Node-2 are part of same multicast group MG-1 with multicast IP address 224.10.10.10.

- On Node-1: Whenever Node-1 wants to send multicast traffic to the multicast group MG-1, it fetches PQMGK ID corresponding to the multicast group MG-1 (and/or multicast IP address) from the Hyperledger.
- On Node-1: It uses this PQMGK ID to generate the PQMGK Key from its QKS, to encrypt the multicast packet/traffic and sent over Multicast IP address.
- On Node-2: Upon receiving the encrypted multicast packet/traffic data from the Node-1 with Multicast IP address 224.10.10.10, it fetches PQMGK ID corresponding to the received Multicast IP address from the Hyperledger.
- On Node-2: Using PQMGK ID, it generates PQMGK Key from its QKS to decrypt the received multicast packet/traffic.
- Note: QKS can generate unique pair of PQMGK (ID, Key) if it is initialised with common secret seed value shared by the Blockchain Provider as part of initial registration.

The techniques presented here propose method to provide quantum resistant security to the multicast traffic using QKS where-in, QKS generate unique pair of PQMGK (Key, ID), which does not need any explicit negotiation of messages such as Anonce, Snonce etc., so that each endpoint can generate Post Quantum Pre-Shared Key independently. Hence resistant to cryptanalysis attacks. This method uses per-multicast group PQMGK ID maintained in the Hyperledger to generate PQMGK Key to encrypt the traffic for that multicast group, so that along with quantum secure, this method provides optimised/efficient method to provide per-multicast group secure communication. Moreover, to better control over the multicast groups and to provide privacy to the multicast group, this method incorporates policy-based access so that only registered multicast groups can communicate each other.