Run OpenQKDNetwork/CQPToolkit on AWS

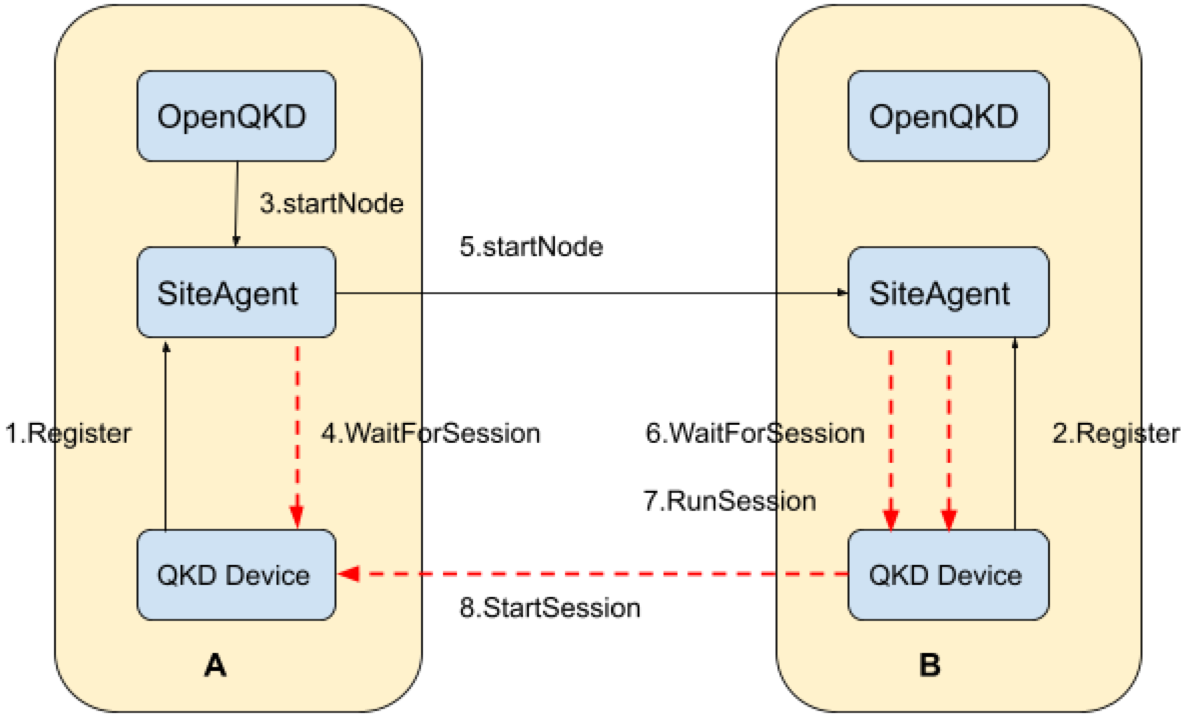
Problem

We can not run real QKD hardware devices on AWS. The real QKD hardware device can only run in the local network behind the firewall. And during the key generation process, SiteAgent on AWS needs to make a GRPC connection to the QKD device that is behind the firewall.

Solution

We can use remote SSH tunnel/port forwarding to solve the problem.

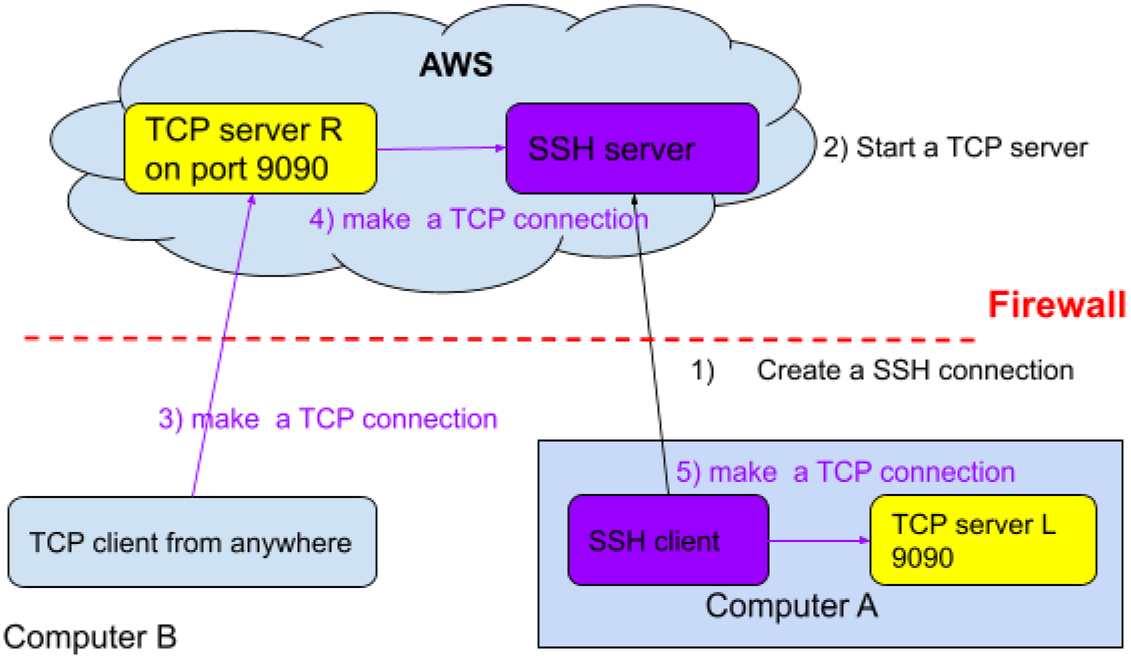
Network connections in key generation process



When running a QKD Device on a local network that is behind a firewall, SiteAgent can not make a GRPC connection to the QKDDevice for steps 4, 6, 7. Also for step 8, QKDDevice on site B needs to make a GRPC connection to QKDDevice on site A, this may not be possible due to the network restriction.

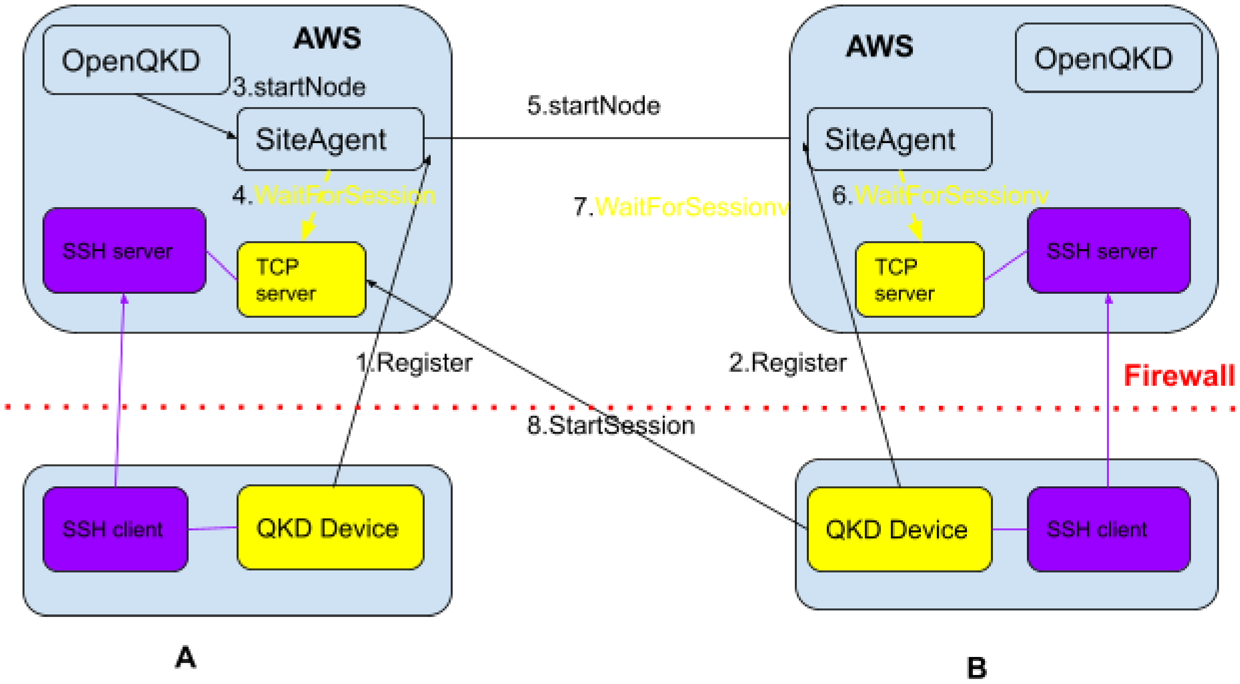
Kaiduan Xie, Institute for Quantum Computing, University of Waterloo

How does the SSH tunnel work?



The basic idea is that the SSH client asks the SSH server to start a TCP server on the same machine as the SSH server, for example on port 9090. Any TCP connection to the TCP server R on the SSH server is forwarded to the local TCP server L that listens on port 9090. *Please note that local TCP server does not need to run the same machine as that of the SSH client, the only requirement is that the SSH client can make TCP connection to local TCP server L. Also please note that the TCP server started by SSH server only forwards the traffic and it does not know the application logic of the local TCP server at all*.

SSH tunnel on AWS for OpenQKDNetwork/CQPToolKit

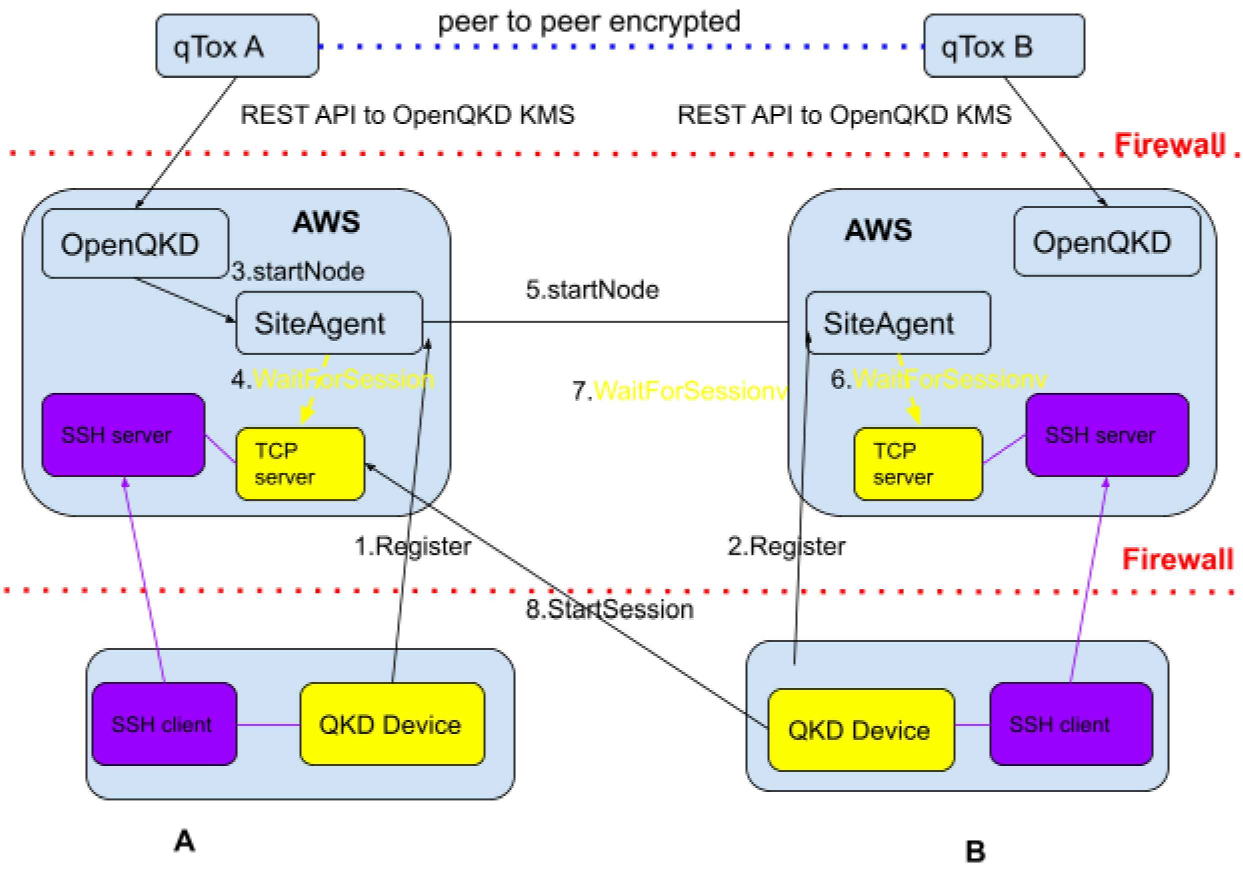


1. On each node, a SSH tunnel is created to the SSH server on the AWS EC2 instance before the QKD device registers to SiteAgent on AWS EC2. The SSH server starts a TCP server on the AWS EC2 instance.
2. When the QKD device registers, it tells SiteAgent its ***control address including IP address and port, which is the TCP server*** started by the SSH server.

OpenSSH command to start SSH tunnel

1. Assume the IP address of the SSH server is *45.55.2.5* for site A, and the QKD device runs on address *192.168.2.1:9000.*
2. The following command is invoked on the QDKDevice.
   ssh -R 9000:192.168.2.1:9000 -N -f user@45.55.2.5.
3. SSH server will start a TCP server listening on port 9000 on 45.55.2.5.
4. When the QKD device registers, the QKD device configures its control address as *45.55.2.5:9000* which is the listening address of the TCP server started by the SSH server.

Run QTox with OpenQKDNetwork/CQPToolkit



With the above setup, we can run qTox in *different networks from anywhere*. QTox gets the key via OpenQKDNetwork KMS REST API, and two qTox can run in different networks and the communication is peer-to-peer encrypted with the OpenQKDNetwork key.

Kaiduan Xie, Institute for Quantum Computing, University of Waterloo