pQKD Twin Cloud Edition



Quantum-Enhanced VPN Connectivity for AWS Cloud. Delivering ETSI QKD Compatibility, Genuine Quantum Entropy, and Post-Quantum Security.

> Created by: Ryszard Olejnik, Piotr Łuniewski, and Mirek Sopek

Quantum Blockchains, Inc.



Table of Contents

pQKD Twin Cloud Edition		
I. Solution presenta	ation	3
II. Installation of pQ	KD Twin on AWS	4
III. Configuration of	the QKD Emulator – pQKD Device on the Client Side	13
IV. Client software ir	nstallation	18
1. Installation of	WireGuard	18
2. Installation of	Java	19
3. Installation an	d Configuration of VPN (qVPN)	19
V. qVPN configurati	on on AWS	21
Support		27



pQKD Twin Cloud Edition

I. Solution presentation

pQKD Twin Cloud Edition is a solution for high-security connectivity between an AWS VPC (Virtual Private Cloud) and a client network, based on the principles of quantum cryptography realized through QKD emulation.

It follows standard VPN principles, enhanced with secure symmetric key exchange provided by QKD emulation technology. This technology ensures full ETSI QKD compatibility, genuine quantum entropy from a quantum random number generator, and uses a standard post-quantum key encapsulation mechanism (FIPS 203 - Module-Lattice-Based Key-Encapsulation Mechanism Standard known also as CRYSTALS Kyber).

The solution consists of the following elements:

- 1. An EC2 server instance on AWS (with the necessary services installed)
- 2. A client computer (with the necessary services installed)
- 3. A pQKD device.

The solution schema is presented below:



Figure 1 pQKD Twin Cloud Edition schematics



As shown in Figure 1, the VPN network is based on the efficient and secure WireGuard VPN. In our solution, we do not modify the essential security components of the VPN. WireGuard establishes a UDP connection (on port 51920), creating a new virtual network interface in the system. Authentication and encryption key exchange are still performed using the RSA algorithm. However, for maximum security, the transmitted standard key is encrypted with a presharedKey—identical on both the server and client sides—via an XOR operation, typical for One-Time Pad (OTP) mechanisms.

pQKD Twin (on the cloud side) and the pQKD device (on the client network/computer side) provide mechanisms for distributing the presharedKey.

Consequently, our solution is a hybrid approach, combining the WireGuard system with a post-quantum key exchange based on quantum entropy. The service requests key generation by connecting to the pQKD service on AWS via the KME (Key Management Entity) port.

The pQKD service communicates through a TCP link (port 8000) with the client service and its pQKD device (where the key is generated). The keys obtained on both the AWS server and client sides are then incorporated into the WireGuard VPN on each end.

On the AWS side, there is an EC2 instance with the following services installed: 1. WireGuard

- 2. A runner service for communication with WireGuard, pQKD, and the client
- 3. A software-based implementation of pQKD

This server configuration has been saved as an AMI image on AWS.

On the client side, the following components are present:

1. WireGuard

- 2. A service for communication with WireGuard, pQKD, and the AWS server
- 3. A pQKD device connected to the client computer

II. Installation of pQKD Twin on AWS

We select the image (Images \rightarrow AMIs) named: "pQKD Twin_Cloud_Edition."

9																		0
Dashboard <	Amazo	n Machin	e Ima	ges (AMIs) (1/3) Into										C	Recycle	Bin EC2 Image Builder	Actions •	inch instance from AMI
EC2 Global View	Owned	by me 🔻	Q F	ind AMI by attribute or tag														< 1 >
Instances		Name Ø	•	AMI name	v	AMI ID	•	Source	v	Owner	v	Visibility	v	Status	v	Creation date	9 Platform	V Root
Instances				tgateway		ami-06ab13462688deaf2		168420631900/tgateway		168420631900		Private		Available	e Q Q	2024/10/31 09:56 GMT+1	Linux/UNIX	ebs
Instance Types	_		_	avonAMI		ami-0620a7f528d6c18c7	8	168420631900/gvpnAMI	_	168420631900		Private		Available	00	2024/12/19 15:28 GMT+1	Linux/UNIX	ebs
Launch Templates				pQKD_Twin_Cloud_Edition		ami-0c093795242722a2t	6	168420631900/pQKD_Twin_Cloud_Edi.		168420631900		Private		@ Pending	QQ	2024/12/20 15:19 GMT+1	Linux/UNIX	ebs
Spot Requests	1																	
Savings Plans																		
Reserved Instances																		
Dedicated Hosts																		

Figure 2



After choosing the right image, click the "Launch Instance from AMI" button:

Launch an instance Info Imazon EC2 allows you to create virtual machines, or instances, that run on the AW5 Cloud. Quickly get started by following the simple steps below.	▼ Summary
Name and tags Info	
QvpnWirequard Add additional tags	Software Image (AMI) pQKD Twin Cloud Edition ami-0c095795242722azb
▼ Application and OS Images (Amazon Machine Image) Info	Virtual server type (instance type) t2.micro
An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below	Firewall (security group) New security group
Q Search our full catalog including 1000s of application and OS images	Storage (volumes) 1 volume(s) - 8 GiB
AMI from catalog Recents My AMIs Quick Start	Cancel
Name Q pQKD_Twin_Cloud_Edition Browse more AMIs	두_ Preview code
Description Including AMIs from pQKD Twin Cloud Edition AWS, Marketplace and	
Image ID the Community ami-0c093795242722a2b	
Username ① root	
Published Architecture Virtualization Root device type ENA Enabled 2024-12-20T14:19:40.0002 x86_64 hvm ebs Yes	
▼ Instance type Info Get advice	
Instance type t2.micro Free tier eligible Family: L1 vxPU 1 GiB Memory On-Demand Windows base pricing: 0.0134 USD per Hour Image: Compare instance type On-Demand Windows base pricing: 0.0124 USD per Hour Image: Compare instance types On-Demand Windows base pricing: 0.0124 USD per Hour Compare instance types	
Additional costs apply for AMIs with pre-installed software	
▼ Key pair (login) →	
You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.	
test C Create new key pair	



Then fill in the fields highlighted in red as follows:

- 1. Instance name
- 2. Instance type (vCPUs, CPU, disk size)
- 3. Select or generate the key for SSH terminal access

Next, click the "Launch Instance" button.

You should then see the following screen:

		QUANTUM < BLOG	CKCHAINS		
EC2 > Instances > Launch an Instance					0 0 5
Success Success/hilly initiated launch of instance (<u>1.028248/bb/di/02256b</u> Launch log)				
Next Steps Q. What would you like to do next with this instance, for example	r "create alarm" ar "create backsp"				< 1 2 3 4 >
Create billing and free tier usage alerts To manage costs and avoid surprise bills, set up email collifications for billing and free tier usage thresholds. Create billing alerts [3]	Connect to your instance One your instance is noning, log into it from your local computer. Connect to instance (2) Learn more (3)	Connect an RDS database Configure the connection between an IC2 instance and a database to allow traffic flow between them. Connect an RDS database (2) Contex a rest (2) database (3) Lower connection (3) database (3)	Create EBS snapshot policy Create a policy that automates the creation, retention, and detection of EBS snapshot Create EBS snapshot policy [3]	Manage detailed monitoring Enable or disable detailed monitoring for the instance. If you enable detailed monitoring park houses IC2 conside displays monitoring parks with 1 monitoria parks. Manage detailed monitoring &	Create Load Balancer Create a application, network pateway or classic Elastic Load Balancer Create Load Balancer (2)
Create AWS budget AWS budgets slow you to create budgets, forecast spend, and take action on your creats and usage from a single bootion. Create AWS budget (2)	Manage CloudWatch alarms Create or update Anazon CloudWatch alarms for the instance. Manage CloudWatch alarms (2)	Diaster recovery for your instances Recover the instances you just funched into a different Availabil? Zure or a different Algoin using All'S Lastic Diaster Recovery DESL. Diaster recovery for your instances [3]	Monitor for suspicious nutrime activities Anzaro GuardDay evalues you to continuously monitor for matchions nutrime activity and unauthorized behavior, with near mail dive wideling in an o-Net activities executing across your Anzaro ECI verticulas. Member for suspicious nutrime activities 🕐	Get Instance screenshot Capture a screenshot Capture a screenshot from the instance and view it as an image. This is useful for tradicishooting an unreachable instance. Get instance screenshot [3]	Get system log Veve his instanci's system log to traditishoot insues. Get system log (*

Which informs about creation of the instance based on the stored image.

After opening "EC2 \rightarrow instance" in AWS we shall see the created instance:

•																		٦
Dashboard K EC2 Global View	Ins	stances (1/5)	nfo attribute or too (once-constitue)			All states	•						Lest updated less than a minute ago	Connect	Instance state 🔻	Actions	th instances) Ø
Events Instances		Name Ø	v Instance ID	Instance state	٧	Instance type w	Status check	Alarm status	Availability Zone	v	Public IPv4 DN5	v Public IPed	v Elastic IP	IPv6 IPs	v Monitoring v	Security group name	v Key name	
Instances Instance Types		gupntest gateway	i-02ccb02db7ac9c9d1 i-0def04701989a12d7	Stopped	ର ର ଭ ର	t2.micro t2.micro	-	View alarms +	eu-central-1b eu-central-1c		-	-	-	-	disabled	launch-wizard-15 launch-wizard-8	test myARS	
Launch Templates Spot Requests		server2	i-0538780f65d40f4b0	Stopped	00	t2.micro	-	View alarms +	eu-central-1b		-	-	-	-	disabled	launch-wizard-14	test	
Savings Plans Reserved Instances		qvpnWirequar	rd i-028248dbdfd0276fb	@ Running	00	t2.micro	Initializing	View alarms +	eu-central-1b		ec2-3-64-126-113.eu-c	3.64.126.113	-	-	disabled	launch-wizard-16	test	
Dedicated Hosts Capacity Reservations															_			
▼ Images																		
AMI Catalog																		
 Elastic Block Store Volumes 																		

Figure 5

This instance is automatically launched. A "Public IPv4" number is also assigned, which may change after restarting the instance or stopping and resuming it. Therefore, we must associate a fixed IPv4 address with this instance, which will be needed in further system configuration.

To do this, open **"Network & Security"** → **"Elastic IPs"** in the AWS menu:



Now, click on 'Allocate Elastic IP address':

Elastic IP address settings Info	
Public IPv4 address pool	
 Amazon's pool of IPv4 addresses 	
Public IPv4 address that you bring to your AWS address that you bring to you bring to your AWS address that you bring to you brin	int with BYOIP. (option disabled because no pools found) Learn more 🖸
Customer-owned pool of IPv4 addresses created f customer owned pools found) Learn more	n your on-premises network for use with an Outpost. (option disabled because no
Allocate using an IPv4 IPAM pool (option disabled	cause no public IPv4 IPAM pools with AWS service as EC2 were found)
Network border group Info	
Q, eu-central-1 Global static IP addresses WWS Global Accelerator can provide global static IP a the Amsong related natured: A sam areas IA	X Sesses that are announced worldwide using anycast from AWS edge locations. This can help improve the availability and latency for your user traffic by using
Q eu-central-1 Ciclobal static IP addresses WKS Global Accelerator can provide global static IP an the Amazon global network. Learn more [2] Create accelerator [2]	x sees that are announced worldwide using anycast from AWS edge locations. This can help improve the availability and latency for your user traffic by using
Q eu-central-1 Stobal static (P addresses WS Global Accelerator can provide global static (P a the Amazon global network, Learn more (*) Create accelerator (*) Tags - optional	x sees that are announced worldwide using anycast from AWS edge focations. This can help Improve the availability and latency for your user traffic by using
Q eu-central-1 Clobal static IP addresses MKS Global Accelerator can provide global static IP a diversity of the static IP and the static IP and the Amazon global network. Learn more [*] (reate accelerator [*]) Trags - optional h tag is a label that you assign to an AWS resource. Ex	Esses that are announced worldwide using anycast from AWS edge locations. This can help improve the availability and latency for your user traffic by using
Q eu-central-1 Cickel static IP addresses NVS Global Accelerator can provide global static IP and the Anaxon global returnok. Learn more [] (reate accelerator []) Trags - optional At ag is a label that you assign to an AWS resource. Ex- No tags associated with the resource.	Esses that are announced worldwide using anycast from AWS edge locations. This can help improve the availability and latency for your user traffic by using tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.
Q eu-entral-1 Global static IP addresses WS Global Accelerator can provide global static IP an the Anazon global network. Learn more [3 Create accelerator [3] Tags - optional A tag is a label that you assign to an AWS resource. Eu No taga associated with the resource. Add new tag	esses that are announced worldwide using anycast from AWS edge locations. This can help improve the availability and latency for your user traffic by using tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Figure 7

After clicking the "Allocate" button, a Public IPv4 address is generated (as shown in Figure 7); in our example, it is 18.157.51.43.

Now we assign the generated IP address to our newly created instance. To do this, right-click the IP address (the link in blue) and select:



Dashboard 🗸	Elastic IP addresses (1/1)
EC2 Global View	
Events	Q Fina resources by attribute or tag
 Instances 	▼ Allocated IPv4 addr ▼ Type ▼ Allocation ID ▼ Reverse DNS record ▼
Instances	- <u>18,157 51 42</u> Duship [P] eipalloc-Oaac5157fd8cf535f - Allocate Elastic IP address -
Instance Types	View details
Launch Templates	Release Elastic IP addresses
Spot Requests	Associate Elastic IP address
Savings Plans	Disassociate Elastic IP address
Reserved Instances	Update reverse DNS
Dedicated Hosts	Enable transfers
Capacity Reservations	Disable transfers
Images	Accept transfers
AMIs	
AMI Catalog	
Elastic Block Store	
Volumes	
Snapshots	
Lifecycle Manager	
Network & Security	=
Security Groups	① View IP address usage and recommendations to release unused IPs with Public IP insights [2]
Elastic IPs	18 157 51 43
Placement Groups	

The screen appears:

esource type	
Instance	
) Network interface	
If no private IP address is specified, the Elastic IP address will be associated with the primary private IP address. Istance	
Q i-028248dbdfd0276fb	× O
rivate IP address ne private IP address with which to associate the Elastic IP address.	
Q. Choose a private IP address	
172.31.34.254	
secify whether the Elastic IP address can be reassociated with a <u>different resource if it aiready</u> associated with a resource.	

Figure 9

We select our instance and click on 'Associate':

e																	<u>1</u>
Dashboard <	i.	Instance	s (1/5)	Info									Last updated C Connect	Instance state 🔻	Actions 🔻 📕	Launch Instances	•
EC2 Global View Events		Q. Find In	istance by	y attribu	te or tag (case-sensitive)			All sta	ates 🔻							< 1 >	8
▼ Instances		Na:	ne Ø	v	Instance ID	Instance state	∇	Instance type	▼ Status check	Alarm	n status	Availability Zone	Public IPv4 DNS	v Public IPv4 v	Elastic IP	IPv6 IPs	
Instances		□ gat	eway		i-0def04701989a12d7	⊖ Stopped	ର ପ	t2.micro	-	View	alarms +	eu-central-1c			-	-	
Instance Types		□ ser	ver2		i-0538780f65d40f4b0	 Stopped 	QQ	t2.micro	-	View	alarms +	eu-central-1b	-	-	-	-	
Launch Templates		C qka	lvpn		i-04dd77b254c68140f	⊖ Stopped	QQ	t3.xlarge	-	View	alarms +	eu-central-1b	-	-	-	-	
Spot Requests			ntest		i-02ccb02db7ac9c9d1	⊖ Stopped	QQ	t2.micro	-	View	alarms +	eu-central-1b	-	-	-	-	
Savings Plans	L	TAD 🔽	nWirequ	ard	i-028248dbdfd0276fb	Running	QQ	t2.micro	Initializing	View	alarms +	eu-central-1b	ec2-18-157-51-43.eu	-c 18.157.51.43	18.157.51.43	-	_
Reserved Instances		4														1	•
Dedicated Hosts																	
Capacity Reservations																	

Figure 10

By navigating to '*EC2* \rightarrow *Instances*', we can verify that our server has been assigned the "Elastic IP.":

					QU			CHAINS					
•													() () ()
Dashboard <	Î	tances (1/5)	Info							Last updated C Connect	Instance state 🔻	Actions	Launch instances
EC2 Global View Events		Find Instance by	attribute or tag (case-sensitive)			AL	l states 🔻						< 1 > @
▼ Instances		Name Ø	▼ Instance ID	Instance state	v	Instance type	e 🔻 Status check	Alarm status	Availability Zone	Public IPv4 DNS	v Public IPv4	7 Elastic IP	IPv6 IPs
Instances] gateway	i-0def04701989a12d7	⊖ Stopped	ର ପ	t2.micro	-	View alarms +	eu-central-1c	-	-	-	-
Instance Types		server2	i-0538780f65d40f4b0	Stopped	QQ	t2.micro	-	View alarms +	eu-central-1b	-	-	-	-
Launch Templates	0] qkdvpn	i-04dd77b254c68140f	G Stopped	ର ପ	t3.xlarge	-	View alarms +	eu-central-1b	-	-	-	-
Spot Requests] qvpntest	i-02ccb02db7ac9c9d1	Stopped	QQ	t2.micro	-	View alarms +	eu-central-1b	-	-	-	-
Savings Plans		qvpnWirequa	rd i-028248dbdfd0276fb	Running	ର ପ	t2.micro	Initializing	View atarms +	eu-central-1b	ec2-18-157-51-43.eu-c	18.157.51.43	18.157.51.4	
Reserved Instances	4												 ,
Dedicated Hosts													

.....

Figure 11

We still need to open the necessary ports for our services so they can communicate publicly. To do this, we select (using the checkbox) our server and edit the "Secure" security group:

aws III Q Search		[Alt+S]					D 4 0	🚯 Frankfurt 🔻	olejnikrys @ 1684-2063-1900
G VPC									
									💼 😡 🗉
Dashboard EC2 Global View Events V Instances	Instances (1/5) Info Q. Find Instance by attribute or tog	r (case-sensitive) ice ID Instance state	v Instance type	ates ▼ ▼ Status check	Alarm status Avail	Last updated less than a minute ago) Connect Instance state ▼ blic IPv4 DNS ▼ Public IPv4	Actions V	Launch Instances ▼ < 1 > © IPv6 IPs
Instances	dypnwirequard 1-0262	CHaddardu276rb O Running	Q Q t2.micro	2/2 checks passed	View atarms + eu-ce	entral-10 ecz	-10-157-51-45.00-c 10.157.51.4:	0.157.51.43	
Instance Types	gateway 1-00em	1200f65d40f4b0 O Stopped	C C t2.micro	-	View alarms + eu-co	entral-IC -	-	-	-
Spot Requests	akten L0444	177b254r68140f @ Stopped			View alarms + eu-ce	entral-1b -		-	-
Savings Plans	gyphest i-02cct	b02db7ac9c9d1	Q Q t2.micro	-	View alarms + eu-co	entral-1b -	-	-	-
Reserved Instances	(0.11411							
Dedicated Hosts									
Images AMIs AMI Catalog Elastic Block Store Volumes									
Snapshots	I-028248dbdfd0276fb (gypn	Wirequard)			=				@ ~
Lifecycle Manager									
▼ Network & Security	Details Status and alarn	ns Monitoring Security	Networking Storage	a Tags					
Security Groups Elastic IPs	▼ Security details								
Placement Groups Key Pairs Network Interfaces	IAM Role -		Own C 1	er ID 168420631900		Launch time Mon Dec 23 20	024 12:10:31 GMT+0100 (Central Europe	an Standard Time)	
Load Balancers	Security groups	ch-wizard-16)							

Figure 12

After clicking the link to the security group, we edit it (adding ports) as follows:

-0abad6eaa7484914a -	launch-wizard-16				Actions V
Details					
Security group name	Security group ID Sg-Oabad6eaa7484914a	Description	1 -wizard-16 created 2024-12- 5.439Z	VPC ID	70b9f4f7 [2
Dwner 168420631900	Inbound rules count 1 Permission entry	Outbound a 1 Permission	rules count n entry		
Inbound rules Outbound rule	25 Sharing - new VPC associations	- new Tags			
Inbound rules (1)				C Manage tags	Edit inbound rules
Q search □ Name ▼ Sec	curity group rule ID 🛛 🗸 🛛 IP version	⊽ Туре	▼ Protocol	▼ Port range	< 1 > 83▼ Source

Figure 13

We now click on *'Edit inbound rules'*:

		QUANTUM	BLO	CKCHAINS		
it inbound rules 🖬	fo					
ound rules control the incoming) traffic that's allowed to reach	the instance.				
Inbound rules Info						
Security group rule ID	Type Info	Protocol Info	Port range	Source Info	Description - option	al Info
gr-052b27b7eae3652a0	SSH	тср	22	Custom 🔹 🔍		Delete
				0.0.0.0/0 ×		
Add mile						

Click the "Add rule" button and add the respective ports (UDP for WireGuard, TCP for the service):

nbound rules Info					
Security group rule ID	Type Info	Protocol Info	Port range	Source Info	Description - optional Info
			Info		
sgr-052b27b7eae3652a0	SSH	ТСР	22	Custom 🔻	Q
					0.0.0.0/0 X
	Custom UDP	▼ UDP	51920	Custom 🔻	Q 0.0.0.0/0 X Delete
	Custom TCP	ТСР	8000	Custom 🔻	Q 0.0.0.0/0 X
	Custom ICMP - IPv4		All	Anyw	Q 0.0.0.0/0 Delete
					0.0.0.0/0 ×
					0.0.0.0/0 ×

Figure 15

Support for "*Custom ICMP-IPv4*" (external pings, e.g., useful during system startup) has also been added. After clicking the "Save rules" button, we have:





After selecting "*EC2* \rightarrow *Instances*", our server appears as follows:

EC2 > Security Groups > s	sg-Oabad6eaa7/	484914a - launo	ch-wizard-16																D 0	9 57
Dashboard 🔇	Instances	(1/5) mo													Last updated	Connect	Instance state 🔻	Actions	nch instances	
EC2 Global View	O Findles	stonce hy attribu	te or tao (rece, sencitive)				All states								and that a finitum age				< 1 >	0
Events			Instance ID	Instance state		Instance	type w Status	charle	Alarm status	Availability Zona	-	Public IPol DNS	w Dui	blie IPv4	w Elastic IP	IPv6 IPs	w Monitarion w	Security aroun name	- Key D	
▼ Instances	gypr	Wirequard	i-028248dbdfd0276fb	Ø Running	00	t2.micro	@ 2/2	checks passed	View atarms +	eu-central-1b		e(2-18-157-51-43.eu-		157.51.43	18.157.51.43	-	disabled	launch-wizard-16	test	-
Instance Types	_ gate	way	i-0def04701989a12d7	⊙ Stopped	0,0	t2.micro	-		View alarms +	eu-central-1c		-	-		-	-	disabled	launch-wizard-8	myAR	δ
Launch Templates	serve	er2	i-0538780f65d4014b0	⊖ Stopped	QQ	t2.micro	-		View alarms +	eu-central-1b		-	-		-	-	disabled	launch-wizard-14	test	
Spot Requests	c qkdv	apra .	i-04dd77b254c68140f	Stopped	ର୍ ର୍	t3.xlarge			View starms +	eu-central-1b		-	-			-	disabled	launch-wizard-12	test	
Savings Plans Reserved Instances	C gype	ntest	i-02ccb02db7ac9c9d1	Stopped	۹Q	t2.micro			View atarms +	eu-central-1b		-	-		-	-	disabled	launch-wizard-15	test	
Dedicated Hosts	4																			•
Capacity Reservations																				
▼ Images																				
AMIs AMI Catalan																				
Flastic Block Store																				
Volumes																				
Snapshots																				
Lifecycle Manager																				
 Network & Security 																				
Security Groups Flastic IPs	I-028248d	lbdfd0276fb	(gypnWireguard)							=									0	~
Placement Groups																				
Key Pairs	Details	Status an	nd alarms Monit	oring Security	Networ	king	Storage Tags													
Network Interfaces																				
Load Balancing	▼ Securi	ity details																		
Target Groups	IAM Role							Owner ID 168420631900						Launch tin Mon Dec 2	ne 3 2024 12:10:31 GMT+010	0 (Central Europea	n Standard Time)			
Trust Stores New	Security g	roups																		
	10 sg-0al	bad6eaa748491	4a (launch-wizard-16)																	
Auto Scaling Groups																				
	▼ Inbou	nd rules																		
Settings	Q Filte	r rules)												< 1 >	
	Name		s	curity group rule ID	Port	range	Protocol	Source	Secu	rity groups	1	Description								
	-		5	r-0a55b30d98d2fb252	8000		TCP	0.0.0/0	laun	h-wizard-16		-								
	-		9	r-052b27b7eae3652a0	22		тср	0.0.0.0/0	laune	h-wizard-16		-								
	-		s	r-0464224ab4e824f7b	5192	0	UDP	0.0.0.0/0	laure	h-wizard-16		-								
	-		9	r-0d263e375e99b2b44	All		КМР	0.0.0.0/0	laun	h-wizard-16		-								



We connect to the server using "EC2 Instance Connect" or SSH.

The user in both cases is either "ubuntu" or "root", for example:

QUANTUM < BLOC	KCHAINS
2 > Instances > I-028246dbdfd0276fb > Connect to instance Connect to instance Info Connect to your instance I-028246dbdfd0276fb (qvpnWirequard) using any of these options	
EC2 Instance Connect Session Manager SSH client EC2 serial console Instance ID Image: Ho28248dbdfd0276fb (qvpnWirequard) Image: Ho28248dbdfd0276fb (qvpnWirequard) Image: Ho28248dbdfd0276fb (qvpnWirequard)	
Connection Type Connect using the C2 Instance Connect Connect using the C2 Instance Connect traveser-based client, with a public IPv4 or IPv6 address.	Connect using EC2 Instance Connect Endpoint Connect using the EC2 Instance Connect browser-based client, with a private IPv4 address and a VPC endpoint.
Public IPv4 address Public IPv4 address Inv6 address Ver6 address Username Enter the username defined in the AMI used to launch the instance. If you didn't define a custom username, use the default username, root. Q ubuntyd X	/
O Note: In most cases, the default username, root, is correct. However, read your AMI usage instructions to check if the	- : AMI owner has changed the default AMI username.

After clicking the "Connect" button, the terminal appears:

aws		Q Search	[Alt+S]		2	\$ @	٢	Frankfurt 🔻	olejnikrys @	1684-2063-1900 🔻
🕜 VPC										
										0
Console	is r	ready								
:\$ logi	n:									
: 020	2404	hdfd0276fh (gyppWiroguard)								×
Publici	2480	157 51 43 PrivatelPs: 172 31 34 254								
Fublici	rs. 16.	137.31.43 Flivateirs. 172.31.34.234								
> Clou	dShell	Feedback		© 202	24, Amazor	n Web Service	s, Inc. or its af	filiates. Priv	cy Terms	Cookie preferences

Cancel Connect

Cancel

Figure 19

Of course, the terminal can also be accessed externally via SSH (see the "SSH Client" tab). The connection instructions are provided in that tab:

onnect to instance Info nnect to your instance I-028248dbdfd0276fb (qvpnWirequard) using any of these options									
EC2 Instance Connect	Session Manager	SSH client	EC2 serial console						
Instance ID IIII I-028248dbdfd0276fb (qvpnWirequard)								
 Open an SSH client. 									
Locate your private ke	ey file. The key used to laur	nch this instance is t	est.pem						
 Run this command, if chmod 400 "test. 	necessary, to ensure your pem"	ey is not publicly vi	ewable.						
4. Connect to your insta c ec2-18-157-51-4	nce using its Public DNS: 3.eu-central-1.compute.an	nazonaws.com							
Example:									
🗖 ssh -i "test.pem" root@	ec2-18-157-51-43.eu-cent	ral-1.compute.amaz	onaws.com						
Note: In most cases, t	the guessed username is co	rrect. However, read	i your AMI usage instructions to check if the AMI owner has changed the default AMI username.						
<u> </u>									

Figure 20



III. Configuration of the QKD Emulator – pQKD Device on the Client Side

The default configuration (factory reset) of the pQKD device uses a network interface with the IP address 192.168.1.80. A factory reset is performed by holding the button (on the back of the device) while connecting the power supply until the blue LED starts flashing).

You need to configure your computer to use the same network (i.e. assign it the IP address from the same network, e.g. 192.168.1.200), then connect the pQKD device using its ETH0 (rightmost) LAN connector to your computer using a network cable.



Figure 21. Desktop and Rack Mounted pQKD. Use the rightmost (ETH0) LAN RJ-45 connector.

After powering on the pQKD device, wait until the blue LED lights up steadily.

In your browser, enter the address: http://192.168.1.80. This will open a login page.

The default login credentials are:

Username: admin Password: admin





Figure 22

After logging in, a configuration wizard for the device will launch:

	pQKD - KME ID: AliceKME												
START	Administrator account Netw	vork interfaces Servers parameters	QKD target configuration	KME configuration	Generating certificates	Generate postquantum keys 7	Set Certificates and Keys						
	Please follow the steps to finish configuration of your device or first select start configuration .												
		Load /	Alice profile \longrightarrow	Load Bob profile	\longrightarrow								

Figure 23

Select the profile: "Load Bob profile". In the next step, set your own unique password for logging.



Set your access password.

In step ("NETWORK INTERFACES), set a custom IP address for the ETHO interface that matches the network your computer will operating on. The ETH1 port will not be used in this configuration, so it can remain unchanged.

ST	RT Administrator Network Interfa	es Servers QKD target KME configuration Ceneral certific	ting Cenerate Set Certificates END	
dministrator account > Network interfaces > SERVERS PA	RAMETERS			
		1. Configuration Server $\!$		
		2. Source KME server 🗸		
		← Back Next →		

Press next step button:





Open "2. Source KME server":

SOURCE KME_ID	Peer_1		eth0 🗸		
master SAE_ID	Peer_1	QKD notification host port	8083		
KME ethernet socket	th0 🗸	QKD secure ethernet socket	eth0 🗸		
KME host port	082	QKD secure host port	8084	-	
KME protocol type	nttp 🗸				

Figure 25

Additionally, change the Source Identifier (KME_ID)* to something like Peer_1 and the Master Identifier (SAE_ID) to Peer_1. The number must be used in the name. Based on this, the allocated pool of ports on AWS.

	START	Administrator Net	work interfaces Servers parameter	S Configuration KME configu	ration Cenerating Cene certificates postquant	rate Set Certificates um keys and Keys	END	
ne > Administrator account > Network	k interfaces > Servers parame	eters > QKD TARGET ress KME host	CONFIGURATION	on address QKD notifical	ion port QKD secure ad	dress QKD secure	e port – QKD public k	ey file Client certificate p12 fil
ceKME AliceSAE https	192.168.1.80	8082	192.168.1.80	8083	192.168.2.80	8084	Alice.public	qbck-client.p12
			Click to disp	olay and edit position.				Upload target Add target
					_			
				← Back Next	\rightarrow			



In the next step, "QKD Target Configuration", we configure access to the pQKD system located on AWS (pQKD Twin), which is accessed via the qVPN client installed on the client computer. After selecting:

	pQKD - H	KME ID: AliceKME	
	Target	definition	
KME ID	AliceKME	QKD notification port	8083
SAE ID	AliceSAE	QKD secure address	192.168.2.200
KME protocol type	http	QKD secure port	8084 2
KME host address	192.168.1.200	QKD public key file	Upload file
KME host port	8082	Client certificate p12 file	Upload file
QKD notification address	192.168.1.200	Client private key password	
		•	
	Back	Modify Delete	
Constitut © Quantum Biockchains Inc. All tight reserved	Version 73		

Figure 26

Change HTTPS to HTTP and adjust the addresses to match your target network (your computer's ip).

Next, accept the modified values and proceed step by step, confirming "KME Configuration", "Generating Certificates", and finally "Generate Post-Quantum Keys".

		Sector Contraction	PQKD - KI	ME ID: A	liceKME		
home - Administrator account - Natwork interfaces - 5	start A	Antohistrator Network Interfaces Dave	nem QHD target configuration	RME comfiguration	Contracting Contractor positionation Provide Automatic Regional Contractor Regional Co	Contracting END	
QKD private key QKD public key	Bobl.private				random token size in byt Cipher AES mot	64 64 66 GCM 256	
							Generate keys
			- Back	Nest			

Figure 27

Change the key names, for example, from "Bob.private" to "Bob1.private" and from "Bob.public" to "Bob1.public". After renaming, generate the key by clicking the "Generate Keys" button and download the Bob1.public key to your computer (it will be needed for server configuration).

	QL	JANTUM < BLOCKCHAINS		
		pQKD - KME ID: AliceKME		
	START Administrator account Network interfaces	Samen Ord Larger Kield configuration Control and Control of Contro	Certificates END O	
home > Administrator account > Network interfaces >	Servers parameters > QKD target configuration > +	KME configuration > Generating certificates > Generate postquantum keys > SET CERTIFIC	CATES AND KEYS	
QKD private key file	Upload file Bobl.private	server private key password		
QKD public key file	Opload file Bobl.public	client certificate pizite	upload file qbck-client.pl2	
	Upload file			
		Hack Next		

Proceed to the "Set Certificates and Keys" step. Set the newly generated keys (Bob1.private, Bob1.public) in the corresponding fields:

Figure 28

Accept the changes and proceed to the end of the wizard. Confirm the settings to save them. The device will restart automatically.

Now you can reconfigure your computer's network settings (IP address) to match the network you were previously using (in the example 192.168.1.90). To access the pQKD device setup from now on, you must use the IP address you configured (see).

If needed, you can adjust individual pQKD device parameters later by navigating through the appropriate menu options.

All pQKD connections within the client's private network connected to the pQKD device use HTTP.

All other external connections are encrypted with the post-quantum cryptographic algorithm "CRYSTALS Kyber" (FIPS 204 Key Encapsulation Mechanism).

IV. Client software installation

The system can currently be installed on Linux computers (Ubuntu, Debian, etc.). The installation files must be downloaded from: <u>https://www.quantumblockchains.io/decks/qVPN_Linux.zip</u>

The installation proceeds in the following steps:

1. Installation of WireGuard

The WireGuard installation instructions are available at: <u>https://www.wireguard.com/install/</u>

For Ubuntu run:

```
sudo apt update
sudo apt install wireguard
```



WireGuard must be installed without generating RSA keys (we are setting them upfront). Ensure that WireGuard Tools (wg-quick) are also installed. You can verify this with the following command:

wg-quick

The command should execute successfully, displaying the command's help output.

2. Installation of Java

Run the following commands:

sudo apt update sudo apt install default-jre

You can verify the installation with:

java -version

3. Installation and Configuration of VPN (qVPN)

Copy the file qvpn.tar to your system, then extract it using the command:

tar -ux qvpn.tar

This will unpack the necessary files for further configuration.

A directory named qVPN will be created. Enter the directory using the command:

cd qVPN

Inside this directory, you'll find the following files:

init.config
qvpn.jar
qvpn.service
qvpn.sh
READ.ME
wgc.conf

Configuration Before Installation:

Before installation, you must configure *qvpn* service by editing the *wgc.conf* file:

```
#
# client
#
[Interface]
QKD_Host = 192.168.1.200
QKD_EndPointProxy = 18.157.51.43:8000
QKD_TimeOut = 10
Address = 10.0.0.2/32
DNS = 1.1.1.1
```



[Peer] QKD_IDENT = Peer_1 QKD_KME = 8082, 192.168.1.90:8082 QKD_EQKD = 8083, 192.168.1.90:8083 QKD_QKD = 8084, 192.168.1.90:8084 Endpoint = 18.157.51.43:51920 AllowedIPs = 172.31.0.0/16 PersistentKeepalive = 25

Configuration File Explanation:

The configuration file format is similar to a standard WireGuard configuration file. However, it includes new commands prefixed with *QKD*. In addition to these new commands, you can still use all standard WireGuard commands.

[Interface] Section

- QKD_Host: Enter your computer's IP address (used for communication with the pQKD device). In the example, this is the address in the default pQKD network (i.e. 192.168.1.200).

- QKD TimeOut: Maximum response time (in seconds) for pQKD to reply.

- QKD_EndPointProxy: The endpoint address (communication channel) exposed on AWS for auxiliary communication with the client.

- Address: The local address of the virtual network interface created on the computer. It's a good practice to establish a clear IP structure, e.g., `10.0.0.1` for the VPN server, `10.0.0.2` for Peer 1, 10.0.0.3 for Peer 2, and so on.

[Peer] Section

- QKD_IDENT: Identifier assigned to the Peer.
- QKD KME: Parameters for communication with the Key Management Entity (KME).

- QKD_EQKD: Parameters for communication with the emulated QKD service (for QKD notification channel).

- QKD QKD: Parameters for communication with the QKD service.

These configurations (IP addresses and ports) must match those set in the pQKD device configuration.

Ensure consistency across all settings to guarantee proper communication and integration between qVPN and pQKD.

NOTICE:

Due to the definition of multiple ports on the qVPN server side, automatic port assignment has been introduced via the *QKD_IDENT* parameter.

The value of this parameter should follow the syntax:

<name> <number>

Examples: Peer 1, Bob 1, User 1, etc.



The numbers should be assigned sequentially: 1, 2, 3, In the example above, the identifier is Peer 1. Name consistent with pQKD configuration (Figure 25).

Installation

After configuring the configuration file, run the installation command:

sudo sh qvpn.sh install

The qVPN service will also start automatically after restarting the computer. To uninstall the service, use:

sudo sh qvpn.sh uninstall

The program and configuration files are installed in the directory:

/usr/bin/qVPN

After uninstallation, all files are removed from this location.

Editing Configuration Parameters (wgc.conf)

You can modify the wgc.conf file in two ways:

1. Uninstall qVPN, Uninstall the service, Prepare the updated configuration file, Reinstall the service.

2. Direct File Editing: Edit the `wgc.conf` file located at:

/usr/bin/qVPN

Restart the service with the following command:

sudo systemctl restart qvpn

This ensures the changes take effect immediately.

V. qVPN configuration on AWS

To configure the qVPN server, open a terminal to the server instance on AWS (see Chapter II, Figure 19). After opening the terminal, the console will appear:





After entering the login (admin) and password (admin), proceed to enter commands.

You can view a list of available commands by typing: ? :



Figure 30

1. 'editconfig' – Used to edit the configuration file.

After executing this command, you gain access to the qVPN server configuration file. It's a simple line-based editor with basic commands:

- `? Display the list of editor commands
- `show` Display the contents of the configuration file
- `delete` Delete a specific line in the file:

delete <line number>

`insert` – Insert a new line before a specified line:

insert <line number> <line content>

`edit` – Edit a specific line:

edit <line number> <new line content>

`test` - Test the correctness of the configuration file

`save` – Save the configuration file

- `end` Exit the configuration editor
- 2. 'logout' Log out (end session)
- 3. `clear` Clear the screen
- 4. 'keylist' Display a list of post-quantum keys available in the system



- 5. `rmkey` Delete a specific key file
- 6. `reboot` Restart the system
- 7. `passwd` Change the password

The keys available on the server can be displayed using the following command: `keylist`:

aws I 🗰 Q sear	ch		[Alt+S]		۶.	۵	0	Frankf	urt ▼ olejnikrys @ 1684-2063	5-1900 🔻
G VPC										
										9
Console is ready										
:\$ login: admin :\$ password: :\$ keylist										
data/time	size bytes	name								
2024-10-17T12:17:43 2024-10-17T12:17:43 2024-10-17T12:17:43 2024-10-17T12:17:43	1568 3168 3168 1568	Alice.public Alice.private Bob.private Bob.public								
:\$										

Figure 31

We need to upload the previously generated public key `Bob1.public` (during pQKD configuration) to this resource. To do this, copy the file `Bob1.public`to the `~/` directory of the Ubuntu user using the following command:

scp -i key.pem Bob1.public ubuntu@18.157.51.43:~/

The key `test.pem`is obtained during the creation of the EC2 instance from the AMI image (see Figure 3).

Proper permissions must be set for the `test.pem` key (see Figure 20).

In the example above, we assume that both the `test.pem` key and the `Bob1.public file are located in the same directory.

Of course, the IP address `18.157.51.43 is just an example here. You should replace it with your generated IP address (see Figure 7). After copying the key, we have:



Figure32

Now we can use the key in the configuration.

Edit the configuration using the command:

editconfig

After executing this command, the screen will appear as follows:



In this mode, we have entered the simple command editor (a list of available commands can be displayed using `?`).

After issuing the `show` command, the configuration will be displayed:

aws		Q Sea	rch		[Alt+5]	۶.	ф	0	٢	Frankfurt 🔻	olejnikrys @ 1684-2	2063-1900 🔻
G VPC												
												0
<pre>console :\$ logi :\$ pass :\$ keyl data/ti -2024-12 2024-14</pre>	: 15 r .n: ad word: .ist .me -30TG	reauy dmin : 99:30:50 12:17:43	size byt 15 15	es 70 68	name 30b1.public 11cc.public							
2024-10 2024-10 2024-10)-17T1)-17T1)-17T1	12:17:43 12:17:43 12:17:43	31 31 15	68 68 68	ALCE.pr/wate 300.pr/wate 300.public							
:\$ edit editcor 0000 # 0000 # 0003 [1] 0004 QH 0005 QH 0005 QH 0005 QH 0005 QH 0009 AC 0009 AC 0001 L5 0011 Sa 0012 # 0012 # 0012 Sa 0012 Chillion AC 0014 PC ables - editcor	confi fig: fig: conterf conter	ig show face] stenPortI meGevExcl TVATE_PC s = 10.0 Port = 55 rfig = ti 1.1.1.1 fig = ti 1.1.1.1 aRD -i = ipt: RWARD -i s	Proxy = 80 10 ange = 30 KEY = Ali 920 ue les -A FOR 100 - o wgs ables -D F enX0 -o w	00 ce.p -m oRWA gs -	rivate -i %i -j ACCEPT; iptables -t nat -A POSTROUTING -o enX0 -j MASQUERADE; iptables -A FORWARD -i wgs -j ACCEPT; ipta statestate RELATED,ESTABLISHED RD -1 %i -j ACCEPT; iptables -t nat -D POSTROUTING -o enX0 -j MASQUERADE; iptables -D FORWARD -i wgs -j ACCEPT; ip n statestate RELATED,ESTABLISHED -j ACCEPT	bles	-t na s -t	t -A F nat -C	POSTRO) POST	UTING -o en ROUTING -o	X0 -j MASQUERAD enX0 -j MASQUER	DE; İptab RADE; İpt
i-028	248d	lbdfd027	6fb (qvpn\	Nire	guard)							×
Public	IPs: 18.	157.51.43	PrivatelPs: 1	72.31	34.254							

Figure 34

Just like during the configuration of the qVPN client, here we are also dealing with WireGuard commands along with additional commands prefixed with `QKD_`. The names of these variables describe their purpose.

If changes are required, such as different ports, alternative routing, or other parameters, they can be edited directly here.

To add a user (qVPN client), you need to add a [Peer] section. For our previous configuration, it will look like this:

insert 40
insert 40 [Peer]
insert 40 QKD_IDENT = Peer_1
insert 40 QKD_PUBLIC_PC_KEY = Bob1.public
insert 40 AllowedIPs = 10.0.0.2/32



In the `insert` command, the line number 40 was used. Since the file has only 19 lines, the new lines will be added at the end of the file.

The file has been updated with the "Peer_1" configuration.

To verify the correctness of the entered data, use the command:

`test`

To save the changes, use:

`save`

After saving, exit the editing mode with: `end`

To apply the changes, restart the system with the following command: `reboot`:



If everything has been configured correctly, on the client side, after executing the command: 'ip'

you should see the network interface `wgc` listed:

ferred_lft forever 58:da0e/64 scope link noprefixroute	t forever preferred lft forever
---	---------------------------------

Figure 37

Our "VPN server" has a local IP address of 172.31.34.254, and the range of available addresses is defined in (Chapter III, Point 3) as: *AllowedIPs = 172.31.0.0/16*

Therefore, we should be able to ping the server using the following command:

ping 172.31.34.254



Figure 38

If there are any issues, you can inspect the qVPN service on the client side with the following command:



sudo systemctl status qvpn

To check the status and transmission info of WireGuard, use:

sudo wg show

Adding Additional Peers

Subsequent Peers can be added in a similar manner.

Of course, access to hosts in the AWS network can be configured using AWS tools (e.g., route tables, NAT, firewall rules, etc.).

Troubleshooting Connection Issues

If the connection is not established after installation, try uninstalling the qVPN client and reinstalling it.

The connection may take several tens of seconds to establish after starting the services. This delay is related to the refresh time of the post-quantum key.

Support

In case of any troubles contact our support team at: support@quantumblockchains.io