The crucial role of quantum random number generation in securing blockchains

Blockchains have generated a huge interest in the last few years, sparked by the arrival in 2009 of the Bitcoin white paper. This paper proposed the implementation of a distributed currency, where trust in individual actors was replaced by trust in distributed protocol based on strong cryptography and protocols, such as proof of work, that date back to the 1990s with Adam Back's Hashcash system for anti-spam.

What are blockchains?

Blockchains are part of a larger family of technologies called Distributed Ledger Technologies (or DLT). They are actually nothing more than a fancy distributed database with a limited purpose. They are a ledger. The information stored in that ledger includes things like sender, receiver, asset transferred, and a precise timestamp of the transaction.

Some of these ledgers add a functionality called a smart contract. It's a piece of software that is stored on the distributed system and that gets executed in a trusted manner by the system if transactions matching specified criteria are done. That execution is guaranteed, and since the code of the smart contract is visible in advance, it provides a very open, secure way of ensuring that some processes get run when transactions happen.

Note that I didn't say anything about the nature of the transaction. That's on purpose. Some blockchains like the Bitcoin network have a monetary value attached to the transactions. So these transactions tend to be used for payments.

But it's not mandatory (not even on the Bitcoin network). You can decide that a transaction of a small (but specified) amount means something. Say 0.0001 BTC exchanged between A and B means they talked. But if it's 0.0002 BTC then it means they met in person. And if it's 0.0003 BTC then it means they signed a contract (or whatever other meaning associated to the amount).

This is called tokenization. It's a very powerful use of blockchains that has lots of use cases in the enterprise world, and can replace notary acts automatically due to inherent trust brought by the platform.

Blockchains and cryptography

Most of the blockchains today have something in common: they make extremely high use of cryptography. Entities on blockchains (whether they are people or things) are identified by an address, usually a pair of cryptographically linked keys generated randomly. They are used, amongst other things, to digitally sign transactions. The digital signatures and cryptography are also used to ensure previous blocks aren't tampered with.

Block addresses are also generated with random numbers and cryptographic hashing algorithms. These functions require very strong random number generation, which means high and immediately available entropy to avoid delays in transaction processing.

ID QUANTIQUE SA
Chemin de la Marbrerie 3
1227 Carouge/Geneva
Switzerland
T +41 22 301 83 71
F +41 22 301 89 79
info@idquantique.com
www.idquantique.com

They are key to the trust of the blockchain. Any weakness in the randomness used could be exploited by an attacker to predict things and breach the system.

In the world of cryptography, algorithms are designed to be secure and strong. They are built in the open, leveraging Kerckhoffs's principle that states that a cryptosystem should be secure even if everything about the system, except the key, is public knowledge. This implies extremely secure keys.

The importance of quantum random number generation for blockchain

Since the keys are supposed to be random, it is vital that they are generated by systems for which the randomness is permanent, measurable and as absolute as possible. There must be no way to influence the generation of keys to trigger some bias in the randomness. There should be instant availability of these keys in the case of a system restart to avoid downtime in processing (in particular for session keys or changing keys on the fly). And there should be alerts in case something is wrong with the generation of keys.

So the simple solution of generating numbers with PRNGs doesn't work. While the bandwidth is high, there's a stored state that, if captured, breaks the whole system. There's cyclical patterns on large sets (and with expected volumes of blockchain transactions, large sets are going to be common). Entropy at start time isn't good. So long initialization times. It just doesn't do the job.

Hardware RNGs need to be used. But chaos based generators, while they fit the unpredictability, and strong, immediate entropy requirements, are impossible to properly model. So controlling the randomness through alignment with a stochastic model isn't possible.

The only ones that fit the requirements while allowing control and reactions in case something is wrong are the random number generators based on physical phenomena that have a proper model including a statistical distribution model that can be used to ensure that the device is functioning properly.

In terms of looking at Quantum Random Number Generators (QRNGs), there are several types that can be considered. In the datacenter, an appliance generating randomness for a pool of devices is going to be cost effective, and better for management. But we are starting to see much smaller devices now with form factors that are compatible with insertion on silicon. This can lead to mobile devices like cell phones to include high entropy QRNGs to power the randomness of their blockchain stack or dedicated hardware.

It's a bright future for QRNGs in the blockchain world!

Source: Gilles Gravier, Director, Senior Advisor – Blockchain and Open Source at Wipro

ID QUANTIQUE SA
Chemin de la Marbrerie 3
1227 Carouge/Geneva
Switzerland
T +41 22 301 83 71
F +41 22 301 89 79
info@idquantique.com
www.idquantique.com