Blockchains and consensus protocols: Snake oil warning

(Keynote talk)

Christian Cachin
IBM Research - Zurich
CH-8803 Rüschlikon, Switzerland
cca@zurich.ibm.com

Abstract

A blockchain is a public ledger for recording transactions, maintained by many nodes without central authority through a distributed cryptographic protocol. All nodes validate the information to be appended to the blockchain, and a consensus protocol ensures that the nodes agree on a unique order in which entries are appended. Consensus protocols for tolerating Byzantine faults have received renewed attention because they also address blockchain systems. However, amid the current hype around blockchains, cryptocurrencies, fintech startups, and novel consensus mechanisms, it is sometimes overlooked that assessing and gaining confidence in the resilience of a protocol is a difficult task.

We argue that developing consensus protocols is similar to engineering cryptographic systems, and that blockchain developers should look towards the established experience in cryptography and security with building trustworthy systems. Otherwise, it might be dangerous to entrust financial value to new protocols. Public discussion, expert reviews, broad validation, and standards recommendations should be employed, following the established practice in cryptography and security.

1. Blockchain consensus

Blockchains or *distributed ledgers* are systems that provide a trustworthy service to a group of *nodes* or parties that do not fully trust each other. They originate from cryptocurrencies, but stand in the tradition of distributed protocols for secure multiparty computation in cryptography and replicated services tolerating Byzantine faults in distributed systems. Generally, a blockchain acts as trusted, dependable and secure party, for maintaining shared state, mediating exchanges and acting as a trusted computing engine.

A plethora of new protocols for blockchains have been proposed recently, in particular for so-called *permissioned blockchains*, operated and governed by known entities.

Permissioned blockchains address many of the problems that have been studied in the field of distributed computing over decades, most prominently for developing *Byzantine fault-tolerant (BFT)* systems. Such blockchains can benefit from many techniques developed for reaching consensus, replicating state, broadcasting transactions and more, in environments where network connectivity is uncertain, nodes may crash or become subverted by an adversary, and interactions among nodes are inherently asynchronous.

2. How resilient is a protocol?

In the area of cryptography and computer security, it is generally difficult to evaluate any security mechanism. First and foremost, a "secure" solution should not interfere "too much" with the functionality, i.e., primary task one tries to accomplish. But more importantly, the security tool should ensure that one can accomplish this task in a way that is resilient to problems caused by the (adversarial) environment, by preventing, deterring, withstanding, or tolerating any influence that could hinder one from accomplishing the task.

Showing that the solution works in the absence of problems and attacks is easy. The task is achieved, and output is straightforward to verify. Assessing the security is the hard part. A security solution should come with a clearly stated security model and trust assumption, under which the solution should satisfy its goal. This is widely accepted today; it prompts the question of how to validate that the solution satisfies its goal.

Yet, the experimental validation of a security solution in the information technology space fails very often because no experiment can exhaustively test the solution in all scenarios permitted by the model. In a way, experimentation can only demonstrate the *failure* of a security mechanism.

Therefore one needs to apply mathematical reasoning and formal tools to reason why the solution would remain secure under *any* scenario permitted by the stated trust assumption. Without such reasoning, security claims remain vague.

In the domain of blockchain protocols, one can learn a lot from the history of cryptography. Already since the 19th century, Kerckhoffs' principle has been widely accepted, which states that "a cryptosystem should be secure even

This work has been supported in part by the European Commission through the Horizon 2020 Framework Programme (H2020-ICT-2014-1) under grant agreements number 643964-SUPERCLOUD and 644579 ESCUDO-CLOUD and in part by the Swiss State Secretariat for Education, Research and Innovation (SERI) under contracts number 15.0091 and 15.0087.

if everything about the system, except the key, is public knowledge." It implies that any security claim of the kind that a system embodies a superior but otherwise undisclosed design should be dismissed immediately.

Starting from the pioneering work in the 1980s, modern cryptography has developed formal treatment, security notions, and corresponding provably secure protocols. Cryptography research has concentrated on mathematically formalizing (a small number of) security assumptions, such as "computing discrete logarithms in particular groups is hard," and on building complex systems and protocols that rely on these assumptions, without introducing any additional insecurity. In other words, in a "provably secure" solution, an attack on the stated goal of the solution can be turned efficiently into a violation of some underlying assumption.

For assessing whether the formal models are appropriate and whether the security assumptions cover the situation encountered during deployment, human judgment is needed, best exerted through careful review, study, validation, and expert agreement. The AES block cipher, for instance, was selected in 2000 by the U.S. NIST after a multi-year public review process during which many candidates were debated and assessed openly by the world-wide cryptographic research community.

3. Snake oil

During the internet boom in the late 1990s there were many claims of new and "unbreakable" cryptosystems, all lacking substantiation. Many of them were covered in the *Snake Oil FAQ*[1] and in blog posts by Schneier about *snake oil*[2], alluding to the history of medicine before regulation:

> The problem with bad security is that it looks just like good security. You can't tell the difference by looking at the finished product. Both make the same security claims; both have the same functionality. (...) Both might use the same protocols, implement the same standards, and have been endorsed by the same industry groups. Yet one is secure and the other is insecure.

Expert judgment, formal reasoning, experience, public discussion, and open validation are needed for accepting a cryptosystem as secure.

A similar development has taken place with building resilient distributed systems, whose goal is to deliver a service while facing network outages, communication failures, timing uncertainty, power loss and more. The Chubby database of Google [1] and Yahoo!'s ZooKeeper [2], developed for synchronizing critical configuration information across data centers, support strong consistency and high availability through redundancy and tolerate benign failures and network outages. Those systems started from well-understood, mathematically specified, and formally verified protocols in the research literature (e.g., Paxos [3]). Yet it has taken considerable effort during development and testing and frequent exercising of failure scenarios during deployment to achieve the desired level of resilience in practice.

Over the recent years countless proposals for new features in distributed ledger systems and completely new blockchain protocols have appeared. Most of them come without formal expression of their trust assumption and security model. There is no agreed consensus in the industry on which assumptions are realistic for the intended applications, not to mention any kind of accepted standard or validation for protocols. The field of blockchain protocols is in its infancy today, but already appears at the peak of overstated expectations [4]. Many fantastic and bold claims are made in the fintech and blockchain space by startups, established companies, researchers, and self-proclaimed experts alike. Snake-oil claims appear and confuse the public opinion.

Instead, broad agreement on trust assumptions, security models, formal reasoning methods, and protocol goals is needed. Developers, investors, and users in the industry should look towards the established scientific methodology in cryptography and security with building trustworthy systems, before they entrust financial value to new protocols.

Acknowledgments

This talk is based on a forthcoming survey of blockchain consensus protocols, jointly authored with Marko Vukolić.

We thank our colleagues at IBM Research - Zurich and the members of the IBM blockchain teams for interesting discussions and valuable comments.

References

[1] T. D. Chandra, R. Griesemer, and J. Redstone, "Paxos made live: An engineering perspective," in *Proc. 26th ACM Symposium on Principles of Distributed Computing (PODC)*, 2007, pp. 398–407.

[2] P. Hunt, M. Konar, F. P. Junqueira, and B. Reed, "ZooKeeper: Wait-free coordination for internet-scale systems," in *Proc. USENIX Annual Technical Conference*, 2010.

[3] L. Lamport, "The part-time parliament," *ACM Transactions on Computer Systems*, vol. 16, no. 2, pp. 133–169, May 1998.

[4] D. Furlonger and R. Valdes, "Hype cycle for blockchain technologies and the programmable economy, 2016," http://www.gartner.com/smarterwithgartner/3-trends-appear-in-the-gartner-hype-cycle-for-emerging-technologies-2016, Jul. 2016.

[1] http://www.interhack.net/people/cmcurtin/snake-oil-faq.html

[2] https://www.schneier.com/crypto-gram/archives/1999/0215.html