Blockchain on the SIM card

Lightweight

Markus Breuer

In spite of their great importance in industry, IoT edge devices and the data they capture and transfer are often quite vulnerable. A resource-saving blockchain client can provide a remedy here.

The slightly shopworn gag told in the IoT sector according to which the "S" in "IoT" stands for "Security" is an exaggeration, but to a certain extent it is still true today: many IoT devices are insecure. While attackers often abuse consumer devices for their botnets, many professional IoT products continue to harbor security risks too.

The risks are even greater when you take a device used as a standalone solution up to now and connect it to the Internet: to manipulate a device, you used to require physical access to it, but now it can be accessed from anywhere in the world. This means that taking your time with digitalization – something which many German companies have been preaching and practicing over the last few years – does not necessarily signal an unwillingness to change. The new security risks are very real, and they must be detected and eliminated.

Security is a catch-all term in the IoT context. In many cases, it simply means preventing the complete and – in the worst case – unnoticed hijacking of a networked device. Hijackings of this kind are the basis for simple botnets consisting of networked thermostats and kitchen devices, but they also led to Stuxnet, a computer worm used to perpetrate a surprisingly effective cyber attack on the Iranian nuclear program in 2010. Security includes confidentiality, however. Attacks are directed against connected devices and the transmission channels they use – and they give unauthorized persons access to information which should be out of bounds. Information on operating states, utilization rates and maintenance intervals for machines can be just as much a part of the "crown jewels" of a company as their construction plans.

The upshot of this is that data itself is now becoming a more and more popular target for attacks. The intention behind this is frequently not to spy out the data but to manipulate it. It is an established fact that it is usually easier to modify the data flowing out of a device than it is to hijack the device completely. Promising attacks on companies can also be staged in this way, for example by trying to blackmail them afterwards. The industry has been discussing this attack vector for over ten years now, but the increasing digitalization of decision-making processes as well as business processes means that more and more sectors are now being affected (see ix.de/za9f for details).

Sealing instead of encoding

Various different approaches can be used to prevent the manipulation of IoT data. The most widespread method is to encode data transfer, ideally in the context of a public key infrastructure (PKI) with private and public codes. A transmission

Algorithms make the decisions

Many business decisions are now already being made on the basis of data only. This applies not only for digital sectors and their established processes. In mail order trade, for example, people are now often nothing more than executing elements. Decisions concerning the shipping processes are made by data-based algorithms whose instructions will virtually never be questioned by a worker in a high-bay warehouse.

The number of applications for data-based decisions and the business volume that depends on them is growing fast. Predictive Maintenance is an established model at the intersection between IoT and Machine Learning. Here, neural networks first learn how operating parameters of a machine act in relation to each another when the device is working perfectly, but they also learn which patterns of operating data occur just before it fails. Maintenance then no longer takes place at regular intervals, but only when the trained neural network detects impending failure.

When this model is used in a factory or for a fleet of construction vehicles, we have to be able to rely completely on the correctness of the data underlying the decisions made. If the data flow does not reproduce the actual operating state of the devices to be maintained, maintenance takes place too frequently, leading to high costs, or it takes place too late or not at all, leading to device failure. Both of these situations are dangerous in particular when Predictive Maintenance is combined with the equally popular Machine as a Service approach. In this arrangement, the user of a machine no longer owns it, and he doesn't pay a monthly flat rate for it either. Instead, the costs are calculated according to actual use. The customer is presented with a commitment of availability comparable with an IT-SLA, and this is in turn can only be guaranteed in a cost-effective way by means of Predictive Maintenance.

False or manipulated data therefore lead to incorrect decisions, and these – along with device failure – lead to high costs and the corresponding negative economic consequences. This is precisely how the Stuxnet bot proceeded: the control logic was given incorrect data on the speed of the uranium centrifuges, which were therefore operated at 84,600 rpm instead of 63,000 rpm, then slowed down and accelerated to an excessive speed again. The result was that the highly sensitive machines broke down within a very short time.

In step with the growth of the IoT market, these risks are becoming more and more significant. It is no longer just a question of timetable boards displaying incorrect times or individual computers being hacked. When devices destroy themselves and infrastructures collapse, it can cause actual hazards in the real world. In the worst case, these can be life-threatening. Bruce Schneier put the altered situation into the following words: "The Internet is no longer a web that we connect to. Instead, it's a computerized, networked, and interconnected world that we live in. This is the future, and what we're calling the Internet of Things."

channel protected in this way reliably prevents the manipulation of the data transported, and in contrast to just a few years ago, it is now a standard feature of most IoT devices.

The disadvantage of this is that a link protected by encoding is always an end-to-end transfer. For messenger services, for example, this is a hallmark of quality, but it soon leads to complications in more complex networks.

In complex data ecosystems, there are always several recipients, and what's more, they can all belong to different companies. This means that often a large number of transfer channels with end-to-end encoding is required. Once one of the nodes involved has been compromised, the trustworthiness of the data received can no longer be guaranteed. In more complex networks, therefore, encoding is not a reliable approach that allows you to detect or even prevent data manipulation with any degree of security.

An alternative approach to making data trustworthy is to make it unchangeable using the blockchain method, or more generally, the distributed ledger technology (DLT). After all, the principal benefit which DLT promises to users is to make the data unchangeable once it has been recorded and then sound the alarm when it detects a change. If special machine data were saved in a blockchain, in particular immediately after the measurement process, its integrity could be ensured for all recipients, virtually by sealing the data in much the same way as a registrar certifies a document.

The crucial factor here is that anchoring has to take place as close as possible to the source. The risk of manipulation increases in step with the distance between data measurement and blockchain. Also, DLT alone does not guarantee that the data saved is correct, but only that it was not changed after entry.

In contrast to what is usually assumed, this does not require the data to be saved "publicly" in the blockchain. For verification purposes, it is quite sufficient to save an unchangeable hashtag that does the anchoring. The actual payload can be saved in forms which are not publicly accessible - in databases, for example.

X-TRACT

- Many IoT devices rightly have the reputation of still not being sufficiently protected against attacks.
- In industry, the goal of attackers is often not the complete hijacking of the devices, which are usually not very powerful. Instead, their aim is to manipulate the data the devices collect and transfer.
- Lightweight blockchain implementations executed directly within the edge device can help to ensure the necessary trustworthiness of data and protect it against potential manipulation.

Blockchains as resource-wasters?

When confronted with this suggestion, blockchain sceptics will no doubt remark that the saving and anchoring of every single data record in a classical blockchain requires extremely laborious computing operations that consume a great deal of energy, giving this approach a catastrophic energy balance.

On the device itself, too, operating a blockchain client requires considerable computing power, energy and network bandwidth. Depending on the device in-

volved, this may be manageable. Modern automobiles provide plenty of computing power and energy, for example. A Raspberry Pi connected to the network is also capable of operating an Ethereum or IOTA client. However, this would make excessive demands on a small battery-operated sensor with a reasonably priced 32-bit MCU. For years into the future, systems of this kind along with even less powerful ones will account for the by far highest percentage of all devices connected to the IoT.

This is why especially lightweight clients that can make do with few resources have to be used. A platform often mentioned in the IoT context is IOTA. This is not a standard blockchain but a distributed ledger with comparable assurances. That the two terms are mentioned in one breath is not only because they have the first three letters of their acronyms in common: in comparison with clients of classical blockchains, IOTA clients are relatively easy on resources, so it is worthwhile to take a closer look at them.

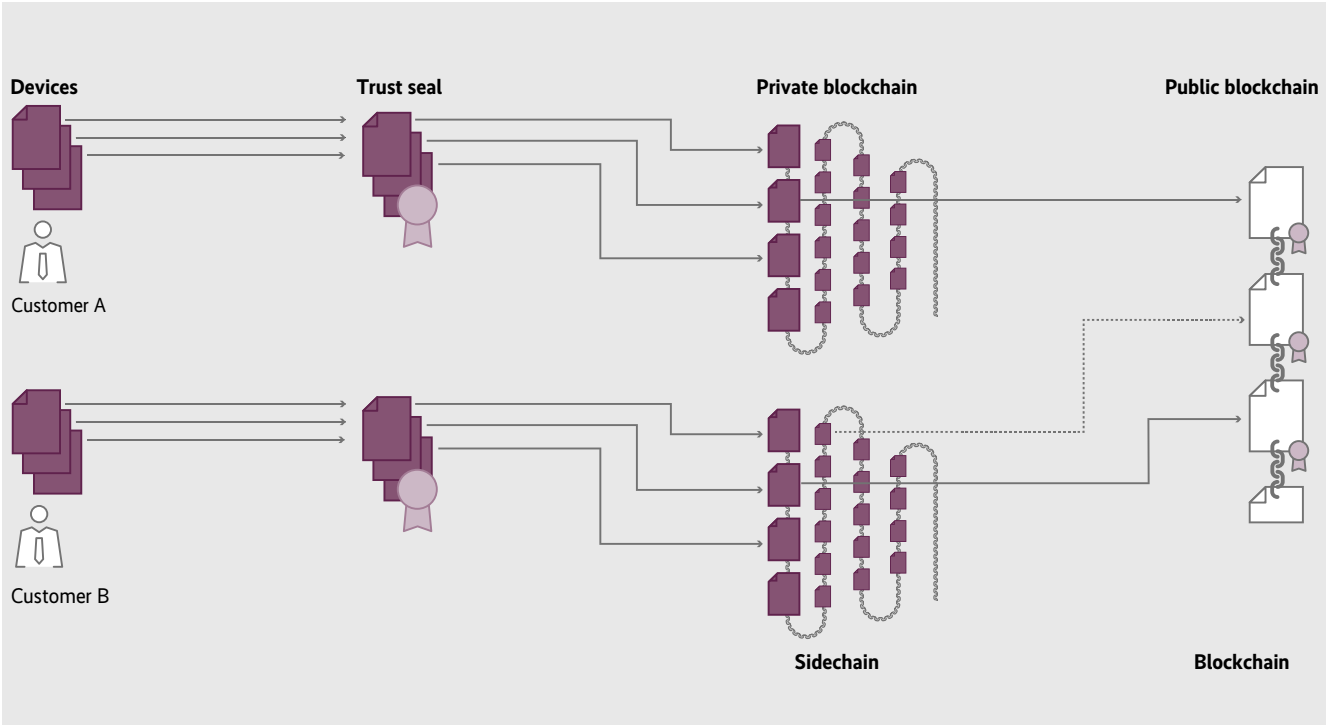
There are not many dependable analyses of IOTA implementation on battery-operated devices. This is because, up to now at least, IOTA appears to have been rarely used in situations in which an electricity network is not available and the Internet connection is slow. However, one of the few publications on this subject comes to the general conclusion that battery-operated IoT devices are currently not suitable for IOTA operations, whereas more powerful Raspberry Pis are (the link to the paper can be found under ix.de/za9f). Now as before, however, IOTA nodes all have the disadvantage that the protocol was originally designed to protect transactions of value. As IOTA uses a Proof-of-Work (PoW) method here, it seems natural to start from an energetic overhead, but one which is lower than the one we find with the PoW methods used by Bitcoin or Ethereum. However, ensuring that data packages remain unmanipulated actually makes considerably fewer demands on the technology used.

For the latter purpose, the IOTA Foundation offers the MAM (Masked Authenticated Messaging) protocol. The author has not been able to find any independent analyses of the relevant resource requirements and energy consumption levels yet. However, the use of MAM will no doubt be effective enough since at least one commercial battery-operated product now uses the MAM protocol: the Bluetooth sensor RuuviTag.

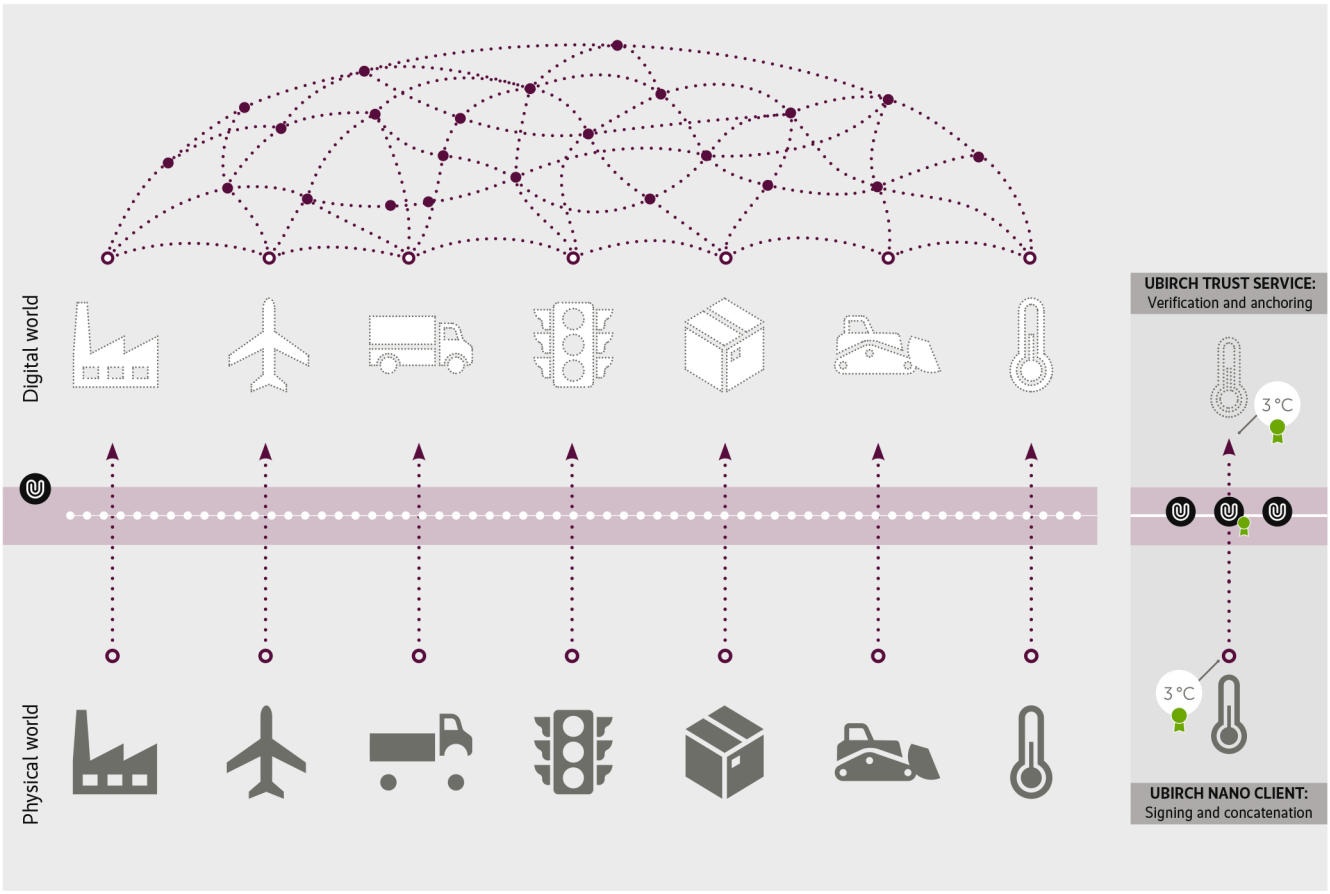
SIM(ple) implementation

The open-source Ubirch protocol takes a different approach to mastering the above problems resulting from the use of blockchain technology on hardware that is less powerful and more limited energetically (see ix.de/za9f for details). The implementation of a complete blockchain client on the edge device is also dispensed with here. Instead, each device uses the Ubirch Nano Client to produce a kind of blockchain with the block size 1: hashes of the data are signed cryptographically using a code that is only present on the device and is linked to the preceding data package (see Figure 1).

This technique dispenses with Proof of Work und Proof of Stake. Instead, the data flow, which contains the signed and concatenated hashes only, is moved to an endpoint in the cloud backend of the company and a two-stage block architecture then anchors the data on several public DLT platforms. These hashes of the data packages are used to build up internal Merkle tree structures. Only the root hashes of these structures are saved at regular intervals on the public DTLP platforms described. The purpose of this work-sharing between edge device and backend is to allow tasks requiring a lot of computer power to take place on the cloud servers and the public DLT platforms as completely as possible and reduce the load that hashing and signature exert on the IoT devices to a minimum (see Figure 2). This procedure



With the two-stage Ubirch protocol, only the root hashes are saved in public blockchains, whereas the signed and concatenated hashes stay with the company. In this way, the data is sealed and protected from manipulation (Figure 1).



The Nano Client allows the data collected in the physical world to be signed on less powerful devices too. This data then comes together in a trustworthy way in the digital world of the Internet of Things (Figure 2).

makes the client component on the IoT devices very undemanding, and it reduces the memory requirements too.

IOTA and the Ubirch protocol are not in direct competition with each other: they have differing applications and therefore differing merits – in fact, the two companies even work together as the approaches complement each other in a number of applications. Although these approaches allow DLT to be implemented on IoT edge devices with few resources too, another obstacle comes between theoretical feasibility and actual rollout as it is necessary to install additional software on the edge devices. The memory space available on many IoT devices is almost completely occupied. Adding libraries after the fact often meets with little enthusiasm among product developers. The same is true for the requirement to incorporate "third-party code". Also, if the product has already been completely developed and the provision of reliable data is required after the fact, implementation becomes even more complicated.

For this reason, Giesecke+Devrient (G+D), the second largest manufacturer of SIM cards worldwide, has cooperated on developing a compact version of the client software for the Ubirch protocol. This version is called SIGNiT, and it all fits onto one SIM card. It uses the SIM as a kind of additional hardware security module. This means that the edge device only needs to have one SIM card slot or eSIM. The sealing functions of SIGNiT can be activated using AT commands. The SIM card itself executes the code and saves all confidential information, and this additionally hardens the procedure.

Of course, this type of firmware installation is not always suitable for cable-bound or Wi-Fi-capable devices as these rarely have an additional SIM slot. In contrast, IoT devices deployed in a widely distributed way or used outdoors are often connected to the mobile phone network. Wi-Fi is not always available outside of buildings and urban environments, whereas alternative networks like Sigfox and LoRaWAN come up against the limits of performance in many applications.

Conclusion

Experts expect the significance of mobile phone networks in the industrial sphere to increase as a result of the features of the new 5G standard. 5G networks are suitable for connecting many independent devices in a restricted area such as a factory, and they guarantee the bandwidths industry requires (see ix.de/za9f). Business models such as Predictive Maintenance and Machine as a Service rely on trustworthy data, and concepts such as Connected Manufacturing cannot function either if the networked devices cannot trust the data exchanged.

As a result, the option of using a SIM card with the necessary functions or installing the software on an eSIM to give networked devices the ability to seal data reliably can help to guarantee the correctness of the data and do its bit to ensure the success of Industrie 4.0. (jvo@ix.de)

Sources

Further literature and sources are available under ix.de/za9f

Markus Breuer

is COO at Ubirch GmbH. He is interested in the social impact of blockchain technology and the question of how to secure tens of billions of devices on the internet can be secured.