trust yourself.

# P3KI Explained

## Decentralized Offline Authorization for IoT

*Version 1.3*
*2019-10-25*
*Gregor Jehle*

# Contents

# 1 IoT device on-boarding with offline authorization

A common practice for on-boarding IoT devices, especially but not exclusively in the consumer market, is based on ***Principle of First Use***. This means, whoever configures the device first, owns it.

For industrial and comparable scenarios this is not sufficent and stronger guarantees are often required. These could be one or more of the following measures:

- Ensure only certain personnel can on-board devices by having the device authorize its user
- Ensure only expected devices are on-boarded by having the user authorize the device

Using P3KI Core both types of authorization are possible. Furthermore, both can be achieved entirely offline, transparently upgrading existing bootstrapping processes.

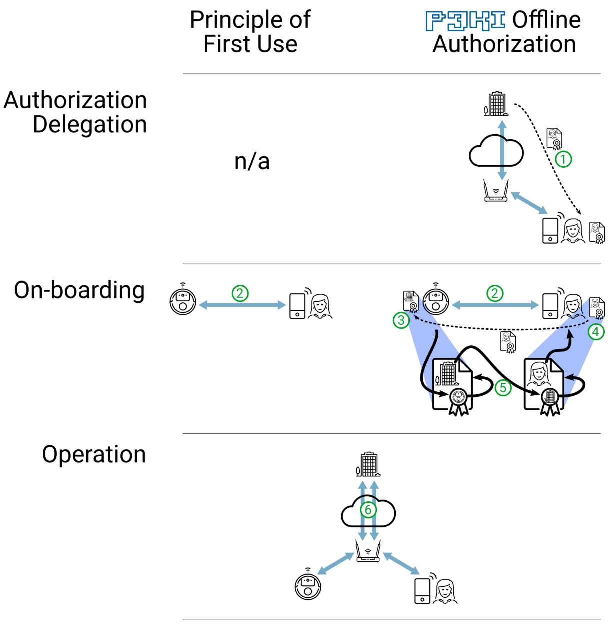

Figure 1: IoT device on-boarding and provisioning scenario

## 1.1 Classic On-Boarding

Figure 1 compares the classic approach on the left with a P3KI-based solution on the right using the example of an IoT-enabled vacuum cleaning robot. The classic approach is to open a WiFi hotspot on the robot and have the user connect (left, #2) their mobile phone or laptop computer to perform initial configuration (like providing their actual WiFi credentials to it). To perform this operation, the person doing the on-boarding is required to have physical access to the device. Once initial configuration has been achieved, all further communication and configuration will usually happen via some kind of online management interface (#6).

If they're on-boarding many devices at the same time or do not have easy physical access this classical process will fail. It will also not allow to limit who can do on-boarding or which devices are on-boarded.

## 1.2 Mutual Offline Authorization

Adding a simple prelude step where permission to perform on-boarding and initial configuration is delegated (right, #1) to a specific person or group of people remedies this. This peron will then open the same local-only WiFi hotspot or similar (right, #2) on the device (at which point neither they nor the device are connected to the internet). The device trusts a suitable authority with performing on-boarding operations (right, #3) and this authority previously delegated suitable permission to them (right, #1). They can now provide this delegated information (right, #4) as a proof of their authorization to the device.

The device can combine the proof with the base trust in its authority to form a chain of trust between itself and the person configuring it (right, #5). This chain is cryptographically secure and fully offline verifiable without involvement by any third-party. Furthermore the permissions authorized by such a chain can be arbitrarily specific. This allows limiting permissions down to individual operations.

# 2 Offline Authorization for Car-Sharing

A common problem with location-flexible car-sharing is the inability to return rented cars in underground parking locations. One of the problems here is that the car requires to stay in active communication with its operator's backend, both to report the end of a rental, and for the backend to prime the car for the next customer. While this usually works well enough in above ground scenarios (fig. 2, #1), it quickly fails for underground parking (#2).

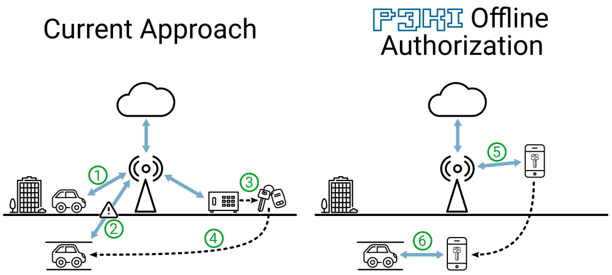

Figure 2: Offline authorization for underground parked car-sharing vehicles

There are workarounds available for car-sharing operators that offer services based on cars with fixed parking location. Such scenarios allow dedicated key lock-boxes installed near the underground parking location, but still above ground. Customers retrieve keys from the lock-box (#3), unlock the car (#4), and return the keys to the box after returning the car. This, however, only works with car-sharing schemes where the car need to be returned to a designated parking spot, which is rather inflexible.

## 2.1 Digital Offline Keys with P3KI

P3KI Core enables a fully flexible scenario where customers reserve a car with the operator's backend (eg. using an app on their smartphone) and the backend provides them with a suitable delegation valid for as long as the reservation lasts and suitable for unlocking the car.

This delegation can be thought of as a digital key to unlock the car. And much like its older physical sibling, this digital key can be verified by the car without having to communicate with its operator's backend servers.

## 2.2 Bi-directional Communication

Based on P3KI Core authorization, not just the above forward communication from backend to car but actual bi-directional ***store & forward*** communication going from car to the backend can be implemented. This means the car can have a backchannel to the operator via the customer's mobile phone. However, this backchannel will not be live but a ***store & forward*** channel, where the car can put information to be communicated to its backend on the phone, and once the phone is back within reach of a cell tower, it will forward this data on the car's behalf. Because of the offline verifiability properties of P3KI Core, the backend is also able to authorize this communication and ensure it came from a trusted car and wasn't modified by any third-party.

# 3 Personal Skill Certificates as Privacy Friendly as Paper

Personal certificates attesting to skills a person possesses are far older than modern computer technology. A piece of paper and a wax seal to reduce chances of tampering and creating forgeries are well understood by basically everyone. So are its privacy properties: as long as one does not hand someone their certificate, noone knows they're a state certified pipe welder and they decide who to show it to.

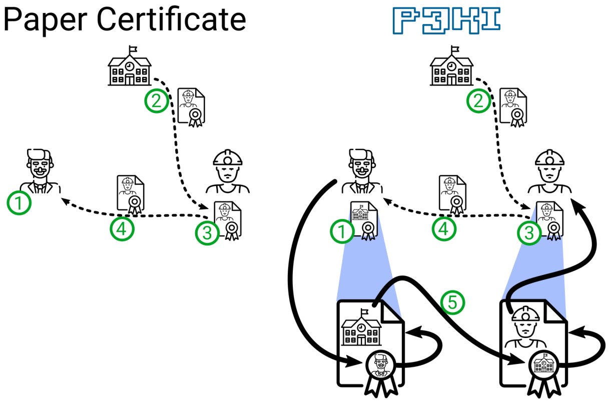

Figure 3: Paper-equivalent personal certificates with P3KI

## 3.1 Reliable Paper Processes

The classic paper-based process of training certification is straight forward: upon successful conclusion of a training the school issues a piece of paper (fig. 3, #2, left) that one then owns (#3, left). To prove one's skill to one's employer (#1, left), they simply present them with the certificate (#4, left) that the employer can then examine.

## 3.2 Digital Analogue with P3KI

Replicating the same process with P3KI is straight forward as well: upon successful conclusion of a training the school issues a signed delegation (#2, right) that one then own (#3, right). To prove one's skill to one's employer (#1, right), they simply present them with the delegation (#4, right) that the employer can then examine.

But we can do one better. If the employer also trusts the school to certify certain skills, they are now able to cryptographically verify the authenticity of the certification (#5, right) presented to them. It's also possible to do this reliably while offline, without having to query the school for confirmation.

All this is possible by effectively giving the same data privacy guarantees as the paper version but with the added benefit of cryptographic verification. And since P3KI Core delegations can be communicated via any transport available, that transport can also be a QR-Code printed on a piece of paper to enable the classic paper version of a certification to be cryptographically verified with the simple scan of a mobile phone camera.

# 4 Further Scenarios

## 4.1 Critical Infrastructure Use-Cases

Using P3KI Core it's possible to build access control and delegation systems that can still operate under the worst conditions imaginable: total loss of internet connectivity due to wide spread power loss.

Operators will still be able to authorize people in the field to enter facilities even if their central access control system is no longer available. The delegations neccessary to enable these authorizations can be communicated via any means available: email, text message, USB thumb drive, printouts, or RFC 2549.

Verification of such a delegation can be performed in the field without any third-party involvement even while entirely offline.

## 4.2 Certification for Software-based Products

Certification of physical products regarding safety is well established. Since physical properties usually change very slowly, re-certification cycles may be low frequency. For instance vehicle inspection periods in Germany are in the range from 12 to 36 months.

A tough problem to deal with, however, is the growing number of software-based components in classical electro-mechanical systems. For instance many medical devices lose their certification if a software or firmware update is installed. On the other hand, it's currently not possible to quickly revoke a certification to react to security vulnerabilities being discovered in a product, even if this vulnerability has implications to the products safety.

With P3KI Core it's possible to flexibly reduce certification to a lower level or rescind it entirely in extreme cases based on new information becoming available. A once rescinded certification can also be easily restored should a problem be resolved by installing a suitably certified software or firmware update. These certifications can be automatically and reliably checked, even in cases where devices are entirely offline.

## 4.3 Network-on-Card for Physical Access Control Systems

A building access control system based on P3KI Core enables several possibilities.

People already trusted with access to certain areas, can have the option of delegating all or part of their permissions to others. This is especially useful if one chooses to decentralize their access control system and instead allows department or team leads to on-board new employees directly, without having to involve (or even have) a central access control management department. Such on-the-spot delegation can happen basically instantly without the need to wait a week or two to get access cards.

Individual door access terminals also do not need to be connected to the network. Instead it's possible to communicate all access control updates required for individual terminals via smartcards or smartphones of the people unlocking the doors anyway. The only terminal that needs to be online is the main gate terminal where everyone or most people pass through.

This offers highly reliable and resilient access control systems, independent of central infrastructure or network communication.

## 4.4 Sub-Contractor Authorization

With today's plethora of logistics services companies, online shipping broker platforms are becoming more and more common. These work like auction platforms where one puts in a request to ship goods between the required locations and shipping companies bid on them.

Another very common occurence in these realms is sub-contracting, especially with jobs concerning last mile handling.

With P3KI Core support in a mobile phone application one is now able to authorize logistics personnel on the spot and ensure that the person picking up the high-value shipment is actually representing the company contracted for it, even if it's a sub-sub-contractor of that company. The best thing: it will work reliably, even if one or both mobile phones involved in the authorication verification are not online!

# 5 Comparing P3KI to Classic Public-Key Infrastructure (PKI)

All PKI-relevant scenarios require two parties (fig. 4, #3) that want to communicate safely and securely. To achieve this, they require mutual authorization and authentication to ensure they are talking to the right party and the respective party has sufficient permissions for that.

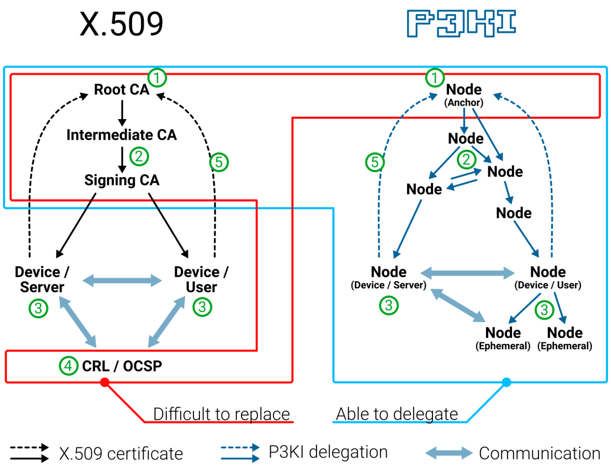

Figure 4: Comparing classic X.509-based PKI with P3KI Web-of-Trust

## 5.1 Trust Anchor

Both systems rely on a trust anchor (#1) to work efficiently. Such an anchor, be it the Root Certificate Authority (Root CA) of X.509-based systems or an anchor node in a P3KI Web-of-Trust, should not be actively operating but instead only be used to set up an intermediary (#2) and then should be placed into secure cold storage where its secret key material is protected from attackers.

## 5.2 Hierarchy vs. Web-of-Trust

X.509 has a top-down tree hierarchy. Techniques like cross-signing can be used to partially weaken this requirement to gain some degree of flexibility, but cross-signing is a late addition to the technology.

P3KI Core on the other hand does not require to form delegations in a tree-like structure but instead can express delegations between arbitrary nodes in arbitrary directions. This makes it especially flexible because it's easy to set up alternative trust paths beforehand to prepare for cases where an intermediate node gets attacked and its secret keys become compromised, or even set up replacements after the fact.

## 5.3 Distributed vs. Decentralized in the Context of Attacks

A targeted attack on PKI could look like this:

- The attacker compromises an intermediate CA to gain access to private key data suitable for issuing relevant certificates.
- Come the attack, they perform a distributed denial of service (DDoS) attack on certificate revocation list servers (CRL) and/or online certificate status protocol servers (OCSP) required to communicate certificate revocations to devices.
- While everyone is busy focusing on mitigating the DDoS, they march in through the front door using a valid certifacte they issued themselves.
- Even if they are detected, it's near impossible to communicate that fact to other parties because the infrastructure required for this (CRL/OCSP) is not availabe.

This highlights that X.509 may be a distributed system, but in cases where revocations are required it devolves into a centralized system around CRL and OCSP servers that cannot easily be changed on the fly, since they are specifically mentioned by name or address in every certificate issued to devices in the field.

Even if all is well and there is no attack, CRL and OCSP servers need to be actively queried. This does not work for systems with elements often or always offline.

A P3KI-based system offers various improvements on these fronts. To begin with, there is no infrastructure required to communicate revocations. Revocations as such do not exist with P3KI Core. Instead a node that issued a delegation simply publishes a new version of the data structure containing its delegations. This new version does not contain any delegation no longer wanted. The amount of data required to communicate this change is small and furthermore can be communicated via any means available. If nothing else, this information will trickle through the network, much like gossip, by nodes using the updated structure in their proofs to other nodes which will learn of the new version of the data that way and and from then on incorporate it into their proofs. To rephrase it: even if no explicit steps are taken to communicate updates, updates get transported by simply using updated data in proofs.

## 5.4 Changing One's Mind: Revocation vs. Rescinding

As already mentioned above, explicit revocations as such do not exist in P3KI-based systems. Rather nodes ***change their mind*** about who they trust and to which degree and publish this information. This means, not only can one adjust the degree of a delegation at the intermediate hop level without invalidating certificates down the line, they can also flexibly migrate downstream delegations to replacement nodes or selectively rescue part of their delegations after an attack on their PKI has been detected and mitigated.

With X.509 one will need to issue new certificates to all devices in the field after revocation of an intermediate CA. With P3KI they can rescind the intermediary node, set up a replacement and transfer an audited subset of the existing delegations to the replacement node. The only piece of information they now need to communicate (eg. simply by use in proofs) is the new delegation to their replacement and its delegation structure. This requires significantly less data than with X.509. Also, the updated data do not need to be distributed via an authoratative source but instead can be communicated by any transport and platform available.

# 6 About P3KI Core

P3KI Core enables your devices and services to trust each other – comparable to how humans do – and do so in fully decentralized and offline scenarios. For the first time it's possible to secure machine-to-machine communication in a way that still works during outages and even enables you to not only issue new trust but also communicate it, allowing you to stay operational when it really matters.

P3KI Core allows you to handle tasks not currently efficiently possible: safely and digitally unlock car sharing vehicles parked in underground parking lots, end-to-end authorized IoT device on-boarding processes, reliable sub-contractor authorization, delegatable physical access control with network-on-card communication, and many more.

P3KI Core is built around a Web-of-Trust architecture to offer ultimate flexibility for modelling your scenario. You decide how precisely you need to express trust. This can be delegations of arbitrary permissions, roles, and capabilities within your application. P3KI Core then takes care of the details based on a system founded in solid, mathematically proven Trust Policy Languages which ensure that you're able to delegate at most what you're trusted with yourself.

Any P3KI Core user can use their peer's delegations to form proofs of certain trust relationships existing. These proofs are fully offline verifiable, without the need for any additional central or online authority.

Our technology is based on research that started in 2006, produced two best-of-class diploma theses and has been under active development since 2014 by a growing consortium of partners encompassing five companies and an accumulated 30 years or work. It's the brainchild of internationally renowned cyber security researcher and whitehat hacker Felix 'FX' Lindner and is strongly informed by decades of experience[1,2,3,4,5,6,7] doing cyber security consulting[8] for large multi-nationals from the automotive to telecoms sector.

P3KI Core is fully transport agnostic. You decide how to store, exchange, and communicate trust based on your scenario's specific requirements. Making your solution work as a decentralized and offline-capable system is as simple as four easy steps:

- We analyze your requirements and existing solutions to identify how your systems can best benefit from P3KI Core.
- We design a custom Trust Policy Language that fits your requirements like a glove.
- You add our library or service to your system to gain decentralized access delegation.
- You get exclusive access to our latest Solution Engineering Tools as they become available to help you operate your system more easily and gain valuable insights.

We support a vast number of platforms from low-power ESP32[9], the ubiquitous ARM-based Raspberry Pi,[10] mobile phone platforms like Google®Android®or Apple®iOS®, to enterprise server hardware and cloud systems. You can either directly integrate our easy-to-use software library or pick our deployment friendly RESTful[11] microservice that will feel right at home in your microarchitecture.

To arrange for a complementary initial workshop, call Gregor Jehle at +49 (0)711 2205 1252 or send an email to gregor@p3ki.com.

[1] https://www.ciscozine.com/black-hat-usa-2009-router-exploitation/

[2] https://www.youtube.com/watch?v=fZAs9M-gLbE

[3] https://www.zdnet.com/article/hack-in-the-box-researcher-reveals-ease-of-huawei-router-access/

[4] https://www.heise.de/security/meldung/Huawei-bittet-Felix-FX-Lindner-um-Hilfe-1741721.html

[5] https://www.zdnet.com/article/researcher-describes-ease-to-detect-derail-and-exploit-nsas-lawful-interception/

[6] https://www.zeit.de/2014/16/blackout-energiehacker-stadtwerk-ettlingen/seite-2

[7] https://www.heise.de/security/meldung/Pwnie-Awards-Hacker-Oscar-fuer-deutschen-Whitehat-Hacker-FX-3784864.html

[8] https://recurity-labs.com/

[9] http://esp32.net/

[10] https://www.raspberrypi.org/

[11] https://en.wikipedia.org/wiki/Representational_state_transfer