Linen: Blockchain Key Registry for Distributed Energy Resources

WIVITY INC.

Future power grids leverage Distributed Energy Resources (DERs), but the current industry practices and regulation are not sufficient to ensure that such critical resources are effectively protected from cybersecurity threats. A particular problem is that the communication keys used in DER devices can have very different levels of security and protections, but such important information is not easily available to communication endpoints. To address this problem, in this position paper we propose a new solution called Linen that is a blockchain-based key registry for DER devices. Linen provides an easily accessible repository for querying security-critical information about DER devices and their keys. Our solution also provides high availability and strong integrity protection for the stored information due to use of permissioned blockchain technology, and a possibility to implement new business models using smart contracts that are enabled by blockchain technology.

1 INTRODUCTION

Distributed Energy Resources (DERs) are expected to play an important role in generating power for the grid. While this development helps in transition towards clean energy, it also opens up new threats and risks. One particular security problem is that if the deployed DER devices can be compromised by malicious actors, the consequences can be severe both economically and in terms of public safety. Thus, it is very important that the deployed DER devices are sufficiently protected.

Unfortunately, the current industry practices, and the corresponding regulation, are lagging behind. The current de-facto solution is to mandate that all certified DER devices have private/public key pairs and communication to them happens only over protected channels like TLS connections. However, such certification practices do not capture the fact that DER devices may support very different security mechanisms including various means of key generation, provisioning, storage, access control and physical attack hardening. While this information may in principle be available in manufacturer factsheets, the provided data is not programmatically accessible, easily comparable or integrity-protected which makes automated trust decisions by communication endpoints very difficult in practice.

To address this problem, in this position paper we propose a new solution called Linen, in which the main idea is to *complement* the current device certification infrastructures and security evaluation practices with blockchain technology. In our solution the blockchain functions as an integrity-protected and immutable registry that allows vendors to record easily-comparable information about registered DER devices, their key credentials, and their security properties. Energy grid communication endpoints can query this information and perform safe and automated trust decisions.

We leverage a *permissioned* blockchain (e.g., one based on the Hyperledger Fabric platform [1]). We opt for a permissioned chain as they offer strong security guarantees, good performance and efficient execution. The strong integrity and availability guarantees of permissioned chains rely on Byzantine-fault tolerant consensus protocols that have been studied and used for decades. We emphasize that this is in contrast to *permissionless* blockchain systems like Bitcoin that rely on potentially problematic trust assumptions and suffer from inefficient execution (see Section 2.4 for more details).

Compared to current industry practices, the solution that we put forward in this position paper offers numerous benefits:

(1) Linen provides a single data repository for obtaining information about the trustworthiness of DER devices, in particular their private keys.
(2) Linen can be queried programmatically to enable seamless service integration.
(3) Linen offers a strong integrity guarantee for all recorded information and high availability due to use of permissioned blockchain technology.
(4) Linen distributes trust and does not require a single trusted authority.
(5) Linen can enable new business models and applications that are implemented as smart contracts executed on the chain.

This paper is organized as follows. In Section 2 we provide background information for the rest of the paper. Section 3 describes the problem that we address in detail. We present the main ideas and benefits of our solution in Section 4. Section 5 provides a brief analysis. We end the paper with discussion in Section 6 and conclusions in Section 7.

2 BACKGROUND

This section provides background information on distributed energy resources, public key infrastructure, security evaluation practices, and blockchains technology.

2.1 Distributed Energy Resources

Distributed Energy Resources [7] are projected to play an increasingly large role in generating power for the grid. There are many reasons for this trend, including the decreasing cost of solar installations and policy changes such as Title 24 (aka the California Solar Mandate) [6], which requires all new homes under 4 stories to have solar power. California alone is expected to add 250,000 new DER facilities each year.

The advent of DERs has a big effect on how the power grid is managed. Rather than controlling a few large generation facilities under direct control, utilities will now need to manage millions of small sites located in residential homes and commercial facilities, either directly or via aggregators as shown in Figure 1.

Industry standards such as IEEE 2030.5 [10] and IEEE 1547 [11] were created to help utilities communicate with DERs for the purpose of controlling power output, and thereby giving (public) utilities some of the same control they have over utility-owned facilities. Some of these standards also specify how communication links must be secured because malicious control of DER facilities could disrupt power delivery. Hackers have already caused major outages in other countries such as Ukraine [22] so it is imperative to understand the potential threats in the new DER environment.

Author's address: Wivity Inc. atom@wivity.com.

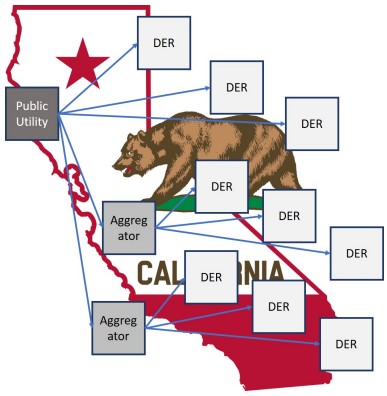

Fig. 1. Distributed Energy Resources (DER) in California.

DER deployments increase the *attack surface* for adversaries in many ways, including:

- *Larger equipment ecosystem.* Hundreds of DER manufacturers employ thousands of people to design security systems, program sensitive security credentials in DER equipment, and manage the communication networks they run on. DER equipment is manufactured all over the world and there are no regulations to control the supply chain or network management from a security standpoint.
- *Increased influence of the private sector.* Whereas traditional generation facilities are controlled by public utilities, many DER networks are controlled by private companies who either manufacture DERs, own DERs, or run management services for DERs. These companies are not subject to the same regulatory standards as utilities.
- *Consumer electronic design practices.* Hundreds of companies are entering the DER manufacturing business, especially in energy storage. Their design experience often comes from consumer electronic companies that in aggregate have a poor record in cybersecurity. DER communication standards partially address this vulnerability by mandating secure protocols but leave other security holes unaddressed.
- *More facilities.* By definition DER means more facilities, which means more physical locations an attacker can access. Most DER facilities will not be worth attacking at the site because the generation amount is too small, but larger commercial facilities could be large enough to warrant a physical attack.

There are several ways an attacker could take advantage of these new attack surfaces and disrupt the distributed grid, including:

- *Distributed network attack.* A hacker can attack DER devices over the network by exploiting a common vulnerability in many devices. For example, if a device has an open port with a default username and password a program can log into these devices over the Internet and inject software that can remotely shut down the DER.
- *Centralized network attack.* A hacker can steal credentials (e.g., a phishing attack) to infiltrate a network that controls a large number of DER devices. The hacker can log into the network with the stolen credentials and instruct all connected DER devices to shut down.
- *Supply chain attack.* A hacker can compromise the supply chain that manufactures and delivers DER devices. If a hacker has access to security credential provisioning processes, the hacker can steal private keys and build devices that masquerade as legitimate DER devices.

This position paper primarily addresses the latter of the above threats. Specifically, it brings transparency to how the supply chain provisions and manages security credentials in the Public Key Infrastructure (PKI) specified by DER communication standards. This transparency enables governing organizations to measure the security of credentials and thus impose requirements.

2.2 Public Key Infrastructures

Public Key Infrastructures (PKIs) leverage asymmetric cryptography and a set of trusted authorities called Certificate Authorities (CAs). In a typical PKI system, a *subject* like a user, device or server, has an asymmetric key pair that consists of a public and private part. During an enrollment phase, the subject creates a certification request to a CA. The request contains the subject's public key and attributes like the name of the user, the model of the device or the domain name of the server. The CA is expected to check the validity of the request (e.g., that the subject actually owns the requested domain name) and then issue a certificate for the subject.

The issued *certificate* is a data structure that contains the public key of the subject, the requested attributes, and additional information signed by the private key of the CA. The primary security function of a certificate is that it provides a secure *mapping* between the public key of the subject and attributes like its name, model and domain. A high-level view is shown in Figure 2. Any third party who knows the public key of the CA (and trusts the CA) can verify the correctness of this mapping by verifying the certificate. A common usage of certificates is in secure web communication: by verifying a certificate a web client learns that the public key of the server belongs to the entity who owns a specific domain name which allows the web client to establish a secure channel to the correct web server.

The trustworthiness of certificates relies on two key factors. The first is the *unforgeability* of CA signatures: if a malicious entity can forge a CA signature, the mapping between subjects and certified attributes can no longer be trusted. The second is the appropriate *validation* of the certification request: if a CA does not correctly validate all the attributes, the mapping between the public key and the attributes cannot be trusted. For these reasons, commonly used standards like X.509 certificates [9] allow certificates to include a link to a document called Certification Policy (CP) that is a human readable document that defines the practices the CA follows to protect its signing key (e.g., storage inside hardware security module) [13]. The CP can also define how the CA validates certificate request attributes and other similar organizational matters.

2.3 Security Evaluation Systems

Security evaluation systems like Common Criteria for general software (and hardware) systems [8] and FIPS-140 for cryptographic key

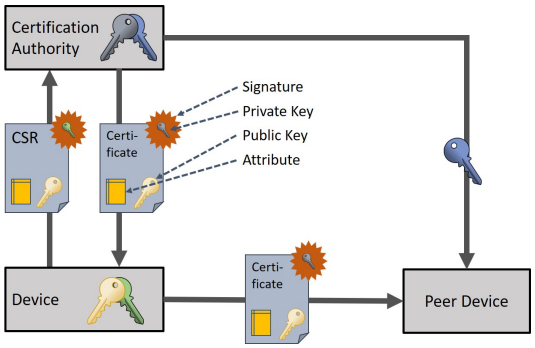

Fig. 2. Public Key Infrastructure (PKI)

modules [19] define the design, build and testing *processes* that must be followed during the creation of the system. Typically, security evaluation systems use numeric evaluation levels over a small range like 1 to 5. Each level has its own requirements regarding how the design of the system must be documented, how the system must be tested etc. If a system has been assigned a specific security evaluation level, it means that an accredited auditing authority has verified that its design and testing processes meet the requirements of that particular level. However, it is commonly acknowledged that a software and hardware system with higher security evaluation level is not necessarily "more secure" than another comparable system with a lower evaluation level (or one with no security evaluation).

2.4 Blockchain Technology

The concept of blockchains was introduced ten years ago by the Bitcoin system [17]. In short, blockchain is a decentralized data structure where transactions such as payments are organized into blocks. The blockchain is integrity-protected and immutable due to an underlying *consensus mechanism* [3]. When the blockchain is used to implement a digital currency, the consensus mechanism guarantees that the same coins cannot be spent multiple times – the so called *double-spending attack*. When the blockchain is used to implement another application besides a currency, the consensus mechanism ensures that any data written on the chain cannot be later modified and that all system participants have the same view of the latest state of the chain.

Two different types of consensus mechanisms and thus two different types of blockchain systems exist. The first is *permissionless* chains like Bitcoin. The second is *permissioned* chains like Hyperledger Fabric [1]. Below we review how such systems work and what are their main security and performance properties.

2.4.1 Permissionless chains. Permissionless systems like Bitcoin are completely free of trusted authorities. Anyone is free to participate in the consensus process *without permission*, hence the name "permissionless". The most widely used permissionless consensus scheme is Proof-of-Work (PoW). There are also other permissionless consensus schemes like Proof-of-Stake (PoS) [3], but since these schemes (to a large extent) have similar issues and trust assumptions, for simplicity and brevity we only discuss Proof of Work here.

PoW combines computationally difficult puzzles and economic incentives to achieve consensus. Thousands of *miners* contribute to the consensus process by computing hashes and the miner that finds a suitable hash first can propose the next block of the chain. This process does not provide immediate consensus, because sometimes more than one miner can find suitable hashes roughly at the same time, and as a result the chain forks. But thanks to economic incentives that are implemented as *block rewards* one chain branch eventually becomes longest and all the system participants can safely trust what is recorded on that chain. Because of this eventual consensus mechanism, permissionless consensus schemes like PoW are slow. In Bitcoin, for example, transaction finalization takes up to one hour.

The security of PoW systems relies on several *trust assumptions*. Probably the best-known trust assumption is that the majority of the mining power must not be malicious. Otherwise an adversary that controls the majority could launch a *51% attack*, where the adversary creates a long fork in the chain, and thus violates chain integrity and double spends money. Another (not equally-well-known) trust assumptions is good message dissemination. Recent research has shown that this assumption does not always hold, and thus systems like Bitcoin can be attacked by manipulating network routing [2].

Although systems like Bitcoin have thousands of consensus participants (miners), the mining power is concentrated in a small number of large *mining pools*. Most Bitcoin miners decide to join a pool due to economic incentives that make individual mining non-profitable.[1] According to a recent study [16], a majority of the mining power is controlled by a couple of pools and there has been moments when a single pool (Ghash.io) has controlled the majority of the mining power.

In summary, although existing permissionless systems like Bitcoin may seem highly decentralized, in actuality such systems provide a low degree of decentralization. Whether it is possible to build a permissionless system that remains highly decentralized is currently an open research question. According to a recent research work [14], full decentralization in permissionless systems is most likely impossible.

2.4.2 Permissioned chains. Permissioned chains rely on a trusted authority that can appoint a set of consensus nodes, as shown in Figure 3. These consensus nodes could be each operated by a different company, public institution or other reputable entity. Because one *needs permission* from the trusted authority to become a new consensus node, these systems are called "permissioned".

In a setting where a relatively small number of pre-defined consensus nodes exist, consensus can achieved efficiently and securely using a so called *Byzantine-fault tolerant* (BFT) consensus protocol. The term Byzantine-fault means that some of the nodes can deviate from the correct protocol execution in arbitrary ways, which covers node compromise by an external attacker, malicious inside administrator and other similar attacks.

[1] Because block rewards in Proof-of-Work systems are rare events, it might take years before a single miner (e.g., running one ASIC) finds a block. If unlucky, a single miner might waste resources (electricity) and investment (mining equipment) for years and get no reward at all. For this reason, most miners decide to join a large pool where rewards are shared between all individual miners who do the work. This way, every participating miner gets a periodic and guaranteed return on investment.

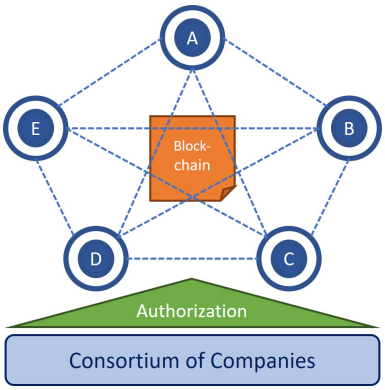

Fig. 3. Permissioned blockchain where consensus nodes are authorized by a trusted authority that is implemented as a consortium of companies.

BFT consensus protocols have been studied for decades. A classical theoretical work by Lamport et al. from the early 1980's shows that there are solutions to the BFT consensus problem only when there are less than 1/3 faulty nodes [15]. This is why *any* BFT consensus protocol can tolerate at maximum 1/3 compromised nodes. Another classical work in this field is an efficient and formally-proven BFT consensus protocol called Practical BFT (PBFT) by Castro et al. [5]. This protocol is commonly used in modern permissioned blockchain systems. Recently, there has been significant research into more optimized BFT consensus protocols that can have slightly better scalability or other improvements.

The trust assumptions of BFT protocols, and thus permissioned blockchains, are simple and well-understood. To violate consensus, the adversary needs to compromise at least 1/3 of the consensus nodes. For example, when 30 consensus nodes are used, the adversary needs to gain control of 10 different organizations. If each organization protects their consensus node appropriately, this is a difficult task for an external adversary. The other possible way to violate consensus is to have more than 1/3 of the organizations collude. If the organizations are chosen carefully and they are reputable ones, this is unlikely to happen.

Importantly, in permissioned chains the consensus nodes can be chosen based on the security needs of the application such that they are generally trusted by the users of the blockchain. This is very different from permissionless chains, where the users of the system have no other choice than to trust the few largest mining pools. Furthermore, in permissioned systems the consensus nodes do not have economic incentives to form a pool, so the level of decentralization does not decrease as it does in permissionless systems. Due to these reasons, a permissioned blockchain with, for example, 30 consensus nodes can provide a higher degree of decentralization and more trusted consensus maintainers than a typical permissionless system like Bitcoin.

In addition, permissioned blockchains are significantly faster. A typical transaction finalization takes 1-2 seconds, instead of one hour in Bitcoin [3]. The operation of permissioned chains is also more efficient, as expensive mining (hashing) is not required.

The main claimed limitation of permissioned blockchains is that they need a trusted authority who will appoint and authorize the consensus nodes. Whether this issue is a problem in practice *depends on the application* of the blockchain. When the goal of the blockchain system is to implement a currency that is completely free of banks, governments and similar authorities, then a permissioned system is problematic. However, when goal of the blockchain is to implement a distributed system that is governed by a consortium of companies that can jointly act as a trusted authority (which is our focus in this position paper), then this is not a problem.

2.4.3 Smart contracts. A smart contract [20] is a program that is executed by the consensus participants and its execution results are stored on the blockchain as new transactions. In other words, smart contracts generalize digital currency transactions to arbitrary blockchain applications. A typical smart contract controls an account and it allows the contract participant to load digital currency to that account. The contract's code defines how the money is transferred out of the contract.

Smart contracts enable the implementation of new types of financial applications that benefit from the advantages of blockchain technology. Compared to traditional contracts and financial applications, smart contracts can enable better business automation (a smart contract can be executed automatically by the blockchain system when its execution conditions are met), increased transparency (the business logic and transaction correctness can be verified by anyone from the chain), high availability (the contract's execution cannot be prevented by a single contract participant), and better privacy (business partners can enter contracts without using or revealing their real identities). The current business interest around blockchain technology is focused on permissioned blockchains and smart contract applications — and that is our focus in this position paper as well.

3 PROBLEM STATEMENT

In this section we provide a detained explanation of the problem that this position paper addresses.

3.1 Current Situation

Deployments of distributed energy resources (DERs) are on the rise. While their adoption can be an important step towards clean energy and new business models, this trend also opens critical infrastructures to new attack vectors. If the deployed DER devices can be compromised by malicious actors, the consequences can be severe both economically and in terms of public safety (see Section 2.1 above). Therefore, it is increasingly important that the deployed DER devices, including their communication and information storage capabilities, are appropriately protected.

Some states like California have addressed this challenge with regulations like CA Rule 21 that define requirements like the minimum level of device authentication or data integrity in the communication between DER devices and utilities. The industry has also developed communication standards (such as IEEE std 2030.5) that have approached this problem by adopting well-known security mechanisms like using TLS connections for communication

integrity and PKIs for device authentication. For example, each certified DER device has a key pair and using that key pair enables secure communication over a TLS channel with the rest of the grid infrastructure.

3.2 Limitations of Industry Practices

Unfortunately the current industry practices, and the corresponding regulation, are insufficient. There is no nation-wide, let alone international, standard that would precisely define what kind of protection these devices must implement [18]. In addition, the above listed regional regulations, communications standards, and industry practices rely on security mechanisms like TLS that are not sufficient to protect all aspects of DER deployments.

3.2.1 Limitation of device certification. A particular problem with the current industry practices is that different DER devices may implement very different security mechanisms but the level of protection that the device provides is not captured by the current device certification practices, and thus the security level is not accurately conveyed to the communication endpoint that needs to decide whether to trust the device or not. The main differentiating factors of DER device security are:

- *Key generation and provisioning.* Some DER devices use keys for TLS that are generated on the device by a cryptographic random number generator that derives strong randomness from the physical environment. Other devices use keys that are derived from limited entropy or are provisioned to the device in plaintext.
- *Key storage and access.* Some DER devices store keys in a processing environment based on dedicated memory or memory encryption that is strongly isolated from the rest of the device and only specific code running on the device can access the private key. Other devices may store the key in the main memory of the device and the private part of the key is accessible to all code running on the device.
- *Hardening against attacks.* Some DER devices may be hardened against physical (intrusive) attacks and side-channel attacks like power consumption monitoring, while other devices may not provide similar physical and side-channel attack protections.
- *Manufacturing process.* Some DER devices may be manufactured using processes that are documented in great detail and evaluated by accredited auditing authorities (e.g., Common Criteria's Security Evaluation Level or FIPS-140), while other devices may be manufactured using processes that are not documented or reviewed. In some cases, the manufacturing process of a DER device can be outsourced to another company or different country and the company that is considered the manufacturer of the device actually has limited visibility to the actual manufacturing process of the device.

The fact that a DER device has a key pair that has been certified by a trusted certification authority (CA) *does not capture* such important differences that determine how trustworthy the DER device is.

This problem cannot be easily solved by using standard certification mechanisms alone. While X.509 certificates include additional fields like KeyUsage that determine whether the signed key can be used for signing or encryption, such information does not help the user of the certificate to determine the trustworthiness of the key, but only its usage. X.509 certificates also allow the CAs to define and publish Certification Practice Statements (CPS) that can define information like how the CA keys are generated, stored and used, but these mechanisms do not help in assessing the certified DER device keys.

3.2.2 Limitations of security evaluations. If the certified device is manufactured following a process that has been evaluated (e.g., Common Criteria [8] or FIPS [19]), such a security evaluation level can be included in a certificate. However, relying on evaluation levels alone has the following problems:

- *High cost.* Security evaluations like Common Criteria and FIPS are very costly and time-consuming processes. For example, a typical Common Criteria security evaluation level 3 or 4 auditing process takes on average one or two years and costs hundreds of thousands of dollars. Not every DER device model can afford to go through such a process.
- *Difficulty of interpretation.* The fact that a specific device model has been evaluated means that its design, manufacturing and testing have been documented and reviewed in a certain level of detail, but it does not directly determine the security of the evaluated device. For example, a device designated as"evaluation level 3" is not necessarily more secure than another device with "evaluation level 2".
- *Missing information.* The security of a device is a multifaceted issue and reducing it to a single numeric value is often misleading and missing significant valuable information.

3.2.3 Limitations of fact sheets. DER device vendors and chip manufacturers typically publish extensive details about their devices and components in the form of fact sheets and other similar documents. Such documents can include very detailed information about aspects like key generation, storage, physical attack protections and more. However, relying on the user of a device certificate checking all the details from a corresponding fact-sheet has the following problems:

- *Difficult programmatic access.* These documents are typically published as PDF files that are intended for humans to read, which makes their automated parsing and processing by a program very difficult. Therefore, such decision-making about the trustworthiness of a DER device cannot be easily integrated into software applications and services.
- *Challenging comparison.* Different device vendors and chip manufacturers tend to provide different types of information, at different levels of abstraction and detail, which makes such information sources difficult to compare, even for a trained professional.
- *Fragmentation.* This information is scattered around various websites and offline storages.
- *Lack of integrity and cryptographic binding.* Once a fact sheet is published, there is no guarantee that its contents will not be modified maliciously later or completely replaced by another document.

Chip makers and distributors also offer chips with credentials that are pre-provisioned using an audited process. This addresses the cost and integrity issues, but does not solve the other aforementioned issues such as fragmentation and interpretation.

Summary. Due to these reasons, the communication endpoints that programmatically parse DER device certificates, cannot easily and securely determine the actual trustworthiness of the device. This is the main problem that we address in this position paper.

4 SOLUTION OVERVIEW

In this section we provide an overview of our solution.

4.1 Main Idea

We propose a new system, in which the main idea is to *complement* the current device certification infrastructures and security evaluation practices with *blockchain technology*. In our solution, herein referred to as Linen (after Hyperledger Fabric [1]), the blockchain functions as an integrity-protected and immutable registry that allows vendors to record easily-comparable information about registered DER device keys and their security properties.

We emphasize that this position paper *does not* propose significant changes to the current device manufacturing, security evaluation and certification practices, as such changes would be difficult, slow and expensive to adopt. Instead, the main goal of our solution is to (a) benefit from the advantages of the emerging blockchain technology, (b) promote the adoption of already known good practices across the chip and DER device manufacturing industry, and (c) combine these elements into a solution that is appropriate for secure DER device deployments.

4.2 Building Blocks

The first main building block of Linen is a *permissioned blockchain* that is operated by a consortium of companies. This consortium's sole responsibility is to make governance decisions on the base-level blockchain and will cooperate with other consortia in the industry such as the SunSpec Alliance.

We opt for a permissioned chain as such chains can provide good security and performance with well-accepted trust assumptions (see Section 2.4 for details). In particular, we use a permissioned blockchain that uses a Byzantine-fault tolerant (BFT) consensus protocol, which means that the adversary has to compromise more than 1/3 of the consensus nodes to violate the integrity of the chain, and thus manipulate any information stored in the registry. As a secondary protection, each consensus node can be hardened, for example, using trusted execution technology like Intel SGX [12] to make such attacks very difficult in practice.

The second main building block of Linen is a predefined *data format* that allows chip manufacturers and DER device vendors to publish information on how keys on registered devices are protected. Informally said, this format is *less detailed* than a typical fact sheet, but *more detailed* than a single numeric value like a security evaluation level. The format also enables other security-related facts such as ownership, physical security, and life-cycle information. The format is designed for electronic processing, instead of human reading, and it includes fields that allow specification of key generation method, key provisioning method, key storage type, key access control, side-channel hardening, physical attack hardening and so on. The full specification is outside the scope of this position paper.

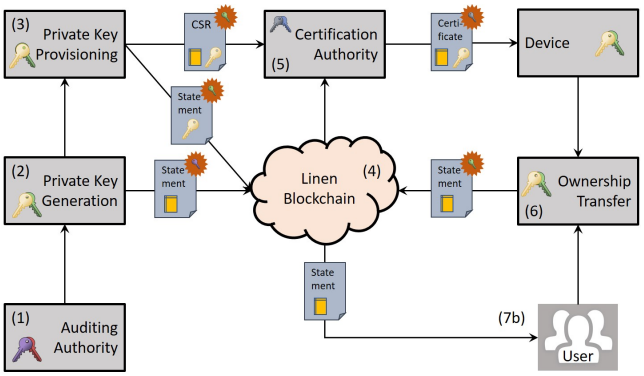

Fig. 4. Overview of Linen's main operational phases

4.3 Operation Overview

The operation of the Linen system consists of the following main phases, and is illustrated in Figure 4.

(1) *Initialization phase.* The consortium is formed and it chooses which members will run consensus nodes. The consortium creates a blockchain instance using the Hyperledger Fabric platform and its tools. The consortium designates first accreditation authorities.
(2) *Auditing phase.* Companies that want to provision keys on a device get their provisioning processes audited by an accredited authority. The auditing authority has a key pair and using the private key it issues a signed statement for a successful audit. The signed statement contains details like the company that owns the provisioning process (semiconductor manufacturer, distributor, solution provider, or device manufacturer) and other metadata associated with the process being audited. Any entity that has and trusts the public key of the auditing authority can verify this signed statement.
(3) *Provisioning phase.* When a provisioning process produces a new batch of key pairs on devices, it registers each key pair on the Linen blockchain. For each key pair, the provisioning entity creates a new transaction that contains (a) security details using the above mentioned predefined format and (b) signed audit statement, and posts the transaction to the consensus nodes.
(4) *Block recording phase.* For each received transaction, the consensus nodes first verify that the transaction is correctly formatted. After that, one of the consensus nodes includes the transaction to a new block that it proposes to the rest of the consensus nodes. After a successful execution of the consensus protocol, the new block is signed by sufficiently many consensus nodes and it is now a permanent (immutable) part of the Linen chain.

(5) *Certification phase.* A certification authority (CA) issues a device certificate for each device. The owner of the key pair sends a CSR to the CA. The CA has access to information on the blockchain. The CA pulls up the security information associated with the device in the blockchain. The CA issues a certificate from the CSR, including security-related information from the blockchain, in an extension of the certificate.
(6) *Distribution phase.* The device is sold to a distributor or customer, and the seller marks the change of ownership transaction on the blockchain. Thus, custody of the device is immutably recorded on the blockchain.
(7) *Operational phase.* The device is used in the field. A communication endpoint (e.g. a cloud server) establishes a TLS connection to the device. The connection establishment includes the following checks: (a) The endpoint requests the certificate of the device. The device sends a certificate associated with the device address. (b) The endpoint can evaluate the extensions in the certificate to determine authorization. In addition, the communication endpoint can query the blockchain to obtain additional information about the key like its generation and storage method and physical attack protections.
(8) *Decommission phase.* The device has reached end-of-life and this is recorded on the blockchain. If an endpoint receives a request from an EOL device it rejects the request.

4.4 Main Benefits

Linen provides the following main benefits over the current industry practices that were reviewed in Section 2:

(1) *Central data repository.* All necessary information about registered device keys is stored in one data repository, the blockchain, instead of being scattered around the Internet in various websites and fact sheets. We note that from a physical point of view, a permissioned blockchain is a distributed data structure where data resides in multiple locations, but from a logical point of view it can be seen as a single data structure.
(2) *Easy programmatic access.* All information stored on the registry is easily accessible and extendable programmatically, in contrast to PDFs that are intended for human reading. This allows easy automation and seamless service integration.
(3) *Comparable information.* Since all information must be stored using a predefined format, Linen encourages chip manufacturers and device vendors to publish information that is easily comparable (unlike currently available fact sheets and other similar documentation).
(4) *Data immutability.* Once data is published on the chain, it cannot be modified, due to the immutability property of blockchain. This is in contrast to the current industry practice, where device details are published on the vendor website and can be later modified.
(5) *Distribution of trust.* The solution requires no single trusted entity. The permissioned blockchain is maintained by a consortium of companies and to violate its integrity and immutability guarantees, more than 1/3 of the consensus nodes must be compromised (see section Analysis for details). Thus, one or few malicious organizations or administrators cannot launch an insider attack.
(6) *High availability.* The solution also provides high availability. Since the blockchain is a distributed data structure, communication endpoints can successfully query information from it, even if one or few organizations would happen to be offline or otherwise unreachable.
(7) *New business models.* Finally, the use of blockchain allows new business applications that are implemented as smart contracts. One possible application is a smart contract that tracks a DER device's response to utility commands. The full details of such new business models are outside the scope of this position paper.

5 ANALYSIS

In this section we outline the main security, availability and performance properties of Linen. A full analysis with precise security definitions and performance measurements is outside the scope of this position paper.

5.1 Security

The main security property that Linen provides is *strong integrity* for all key and provisioning information that is recorded on the chain. When a provisioning process produces a new batch of devices (see Section 4.3 above), it places information about each provisioned key on the chain. Once this information is included to a new block, it cannot be later modified or erased unless the adversary manages compromise at least 1/3 of the consensus nodes. Such integrity property is achievable due to use of a BFT protocol [5].

To consider a concrete example, let us assume a deployment where a consortium of 30 companies runs Linen such that each company operates one consensus node. To violate the integrity of recorded key information, the adversary would need to compromise consensus nodes in 10 separate organizations. Alternatively, an insider attack like a malicious administrator in one organization would need to convince 10 other organizations to collude to violate integrity. Recent research has suggested optimizations that may allow up to 100 consensus nodes for permissioned chains. For example, the LibraBFT consensus protocol [4] that is based on the HotStuff protocol [21] and the consensus protocol used in the Hyperledger Fabric system [1] support up to 100 consensus nodes (in lab conditions). If such consensus protocols are used, the adversary would need to compromise as many as 33 separate organizations that run consensus nodes. For more details, refer to Section 2.4.

To increase trust in each individual organization, an organization can harden their consensus node by using trusted execution environments (TEEs) like Intel's SGX [12]. In such deployments, even 1/3 malicious administrators cannot launch an attack without compromising their local TEEs. The consortium can also require organizations to implement security measures such as FIPS standards.

5.2 Availability

The primary availability property of Linen is a *strong liveness* condition that ensures that new blocks can be created (i.e., new key and provisioning information recorded) and existing blocks can be

read (i.e., previously stored information can be retrieved) when at least 2/3 of the consensus nodes are reachable. Following the same example above, this means that the system is available when at any given time less than 10 consensus nodes are offline at the same time.

5.3 Performance

The two main performance characteristics of blockchain systems are *throughput* and *latency*. Throughput determines how many new transactions can be added (that is, written) to the chain per time unit. Latency determines how long it takes before a new transaction that is posted to the consensus nodes is accepted and part of the immutable chain.

The exact throughput and latency numbers depend on several factors like the consensus protocol that is used, the speed of the links between the consensus nodes, the average size of transactions (key and provisioning information), the used block size and so on. Modern permissioned blockchain platforms like Hyperledger Fabric have demonstrated throughput as high as over 1000 transactions per second and latency of less than one 1 second (in optimal lab conditions).

In Linen, new transactions are added (written) to the chain at the time of key provisioning. This operation is typically not time critical in general, so low blockchain latency is not a mandatory requirement for Linen. As long as transactions are confirmed in a few seconds, the used blockchain is sufficiently fast for Linen. We emphasize that reading data from the blockchain is generally fast (normal Internet round-trip time) regardless of the used consensus scheme. Similarly, the number of provisioned keys per time unit is expected to be low, and thus Linen does not require a blockchain with very high throughput.

6 DISCUSSION

In this section we discuss deployment schedule and privacy considerations for Linen.

6.1 Deployment Schedule

The first cybersecurity requirements for DER devices will come into effect on January 22, 2020 as part of the California Public Utility Commission's Rule 21 tariff. Rule 21 requires all direct communication channels to the utilities to be protected by TLS with mutual authentication using X.509 certificates issued by a certificate authority. However, discussions are ongoing with the Smart Inverter Working Group and the SunSpec Alliance Cybersecurity Workgroup to impose additional cybersecurity requirements to mitigate various threats, including the ones mentioned above. Most of these requirements will probably not be required by January 22, 2020, but if desired Linen is fully capable of being deployed in the possible timelines that may be considered. The first version of Linen for product SKU-level security is on schedule for completion in 2019.

Deployment of Linen for individual private key security will follow these steps:

(1) Design extension of Linen for individual private key security.
(2) Create consortium and ratify Linen initial specification.
(3) Approve first nodes.
(4) Implement Linen extension.
(5) Create provisioning audit/certification process.
(6) Go live with main-net and start inserting provisioning transactions on the blockchain.

6.2 Privacy Considerations

The most straightforward way to use Linen is that the provisioning entity publishes separate information about each provisioned device on the blockchain. However, such simple usage may not be ideal for all use scenarios, because it can leak information. For example, a provisioning entity may not want to leak the exact number of provisioned devices to its competitors. To address this problem, a few possible techniques can be used.

The first technique is that the provisioning entity does not publish separate information for each device, but rather information per device type, class or batch. The second technique is that the ability to read data stored on the blockchain is limited to authorized entities only. The third technique is to complement the consensus nodes with SGX enclaves that can process transactions privately and include them in encrypted format into created blocks. These techniques can also be combined based on the needs of the use case. The precise privacy requirements will be determined in discussions with stakeholders and the exact deployed privacy mechanisms shall be decided later.

7 CONCLUSION

In this position paper we have presented a new solution, called Linen, to improve the security of Distributed Energy Resource (DER) deployments. Our main idea is to complement the existing auditing and certification practices with a permissioned blockchain that records security-relevant information about key pairs that are provisioned to DER devices. The communication endpoints can parse the client certificate or query the chain to determine the trustworthiness of each connected DER device. Compared to current regulations and industry practices, Linen provides many advantages including distribution of trust, high availability, strong integrity, central data repository, programmatic access and new business models.

REFERENCES

[1] Elli Androulaki, Artem Barger, Vita Bortnikov, Christian Cachin, Konstantinos Christidis, Angelo De Caro, David Enyeart, Christopher Ferris, Gennady Laventman, Yacov Manevich, et al. 2018. Hyperledger fabric: a distributed operating system for permissioned blockchains. In *Proceedings of the Thirteenth EuroSys Conference*. ACM, 30.

[2] Maria Apostolaki, Aviv Zohar, and Laurent Vanbever. 2017. Hijacking bitcoin: Routing attacks on cryptocurrencies. In *2017 IEEE Symposium on Security and Privacy (SP)*. IEEE, 375–392.

[3] Shehar Bano, Alberto Sonnino, Mustafa Al-Bassam, Sarah Azouvi, Patrick McCorry, Sarah Meiklejohn, and George Danezis. 2017. Consensus in the age of blockchains. *arXiv preprint arXiv:1711.03936* (2017).

[4] Mathieu Baudet, Avery Ching, Andrey Chursin, George Danezis, François Garillot, Zekun Li, Dahlia Malkhi, Oded Naor, Dmitri Perelman, and Alberto Sonnino. 2018. State machine replication in the Libra Blockchain. (2018).

[5] Miguel Castro, Barbara Liskov, et al. 1999. Practical Byzantine fault tolerance. In *OSDI*, Vol. 99. 173–186.

[6] California Energy Commission. 2019. Building Energy Efficiency Standards: Title 24. (2019). https://www.energy.ca.gov/programs-and-topics/programs/building-energy-efficiency-standards.

[7] North American Electric Reliability Corporation. 2017. Distributed Energy Resources: Connection Modeling and Reliability Considerations. (2017). https://www.nerc.com/comm/Other/essntlrlbltysrvcstskfrcDL/Distributed_Energy_Resources_Report.pdf.

[8] Common Criteria. 2019. Common Criteria Portal. (2019). https://www.commoncriteriaportal.org/.

[9] Russell Housley, William Polk, Warwick Ford, and David Solo. 2002. Internet X. 509 public key infrastructure certificate and certificate revocation list (CRL) profile. (2002).

[10] IEE. 2013. IEEE 2030.5-2013 - IEEE Adoption of Smart Energy Profile 2.0 Application Protocol Standard. (2013). https://standards.ieee.org/standard/2030_5-2013.html.

[11] IEEE. 2018. IEEE 1547-2018 - IEEE Standard for Interconnection and Interoperability of Distributed Energy Resources with Associated Electric Power Systems Interfaces. (2018). https://standards.ieee.org/standard/1547-2018.html.

[12] Intel. 2019. Software Guard Extensions. (2019). https://software.intel.com/en-us/sgx.

[13] Stephen Kent, Derrick Kong, Karen Seo, and Ronald Watro. 2012. RFC 6484: Certificate policy (cp) for the resource public key infrastructure (rpki). (2012).

[14] Yujin Kwon, Jian Liu, Minjeong Kim, Dawn Song, and Yongdae Kim. 2019. Impossibility of Full Decentralization in Permissionless Blockchains. *arXiv preprint arXiv:1905.05158* (2019).

[15] Leslie Lamport, Robert Shostak, and Marshall Pease. 1982. The Byzantine generals problem. *ACM Transactions on Programming Languages and Systems (TOPLAS)* 4, 3 (1982), 382–401.

[16] Loi Luu, Yaron Velner, Jason Teutsch, and Prateek Saxena. 2017. Smartpool: Practical decentralized pooled mining. In *26th {USENIX} Security Symposium ({USENIX} Security 17)*. 1409–1426.

[17] Satoshi Nakamoto. 2008. Bitcoin: A peer-to-peer electronic cash system. (2008).

[18] Kevin L Stamber, Andjelka Kelic, Robert A Taylor, Jordan M Henry, and Jason E Stamp. 2017. Distributed Energy Systems: Security Implications of the Grid of the Future. *Sandia National Laboratory, January* (2017).

[19] Federal Information Processing Standards. 2001. Security Requirements for Cryptographic Modules. (2001). https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.140-2.pdf.

[20] Gavin Wood et al. 2014. Ethereum: A secure decentralised generalised transaction ledger. *Ethereum project yellow paper* 151, 2014 (2014), 1–32.

[21] Maofan Yin, Dahlia Malkhi, Michael K Reiter, Guy Golan Gueta, and Ittai Abraham. 2018. Hotstuff: BFT consensus in the lens of blockchain. *arXiv preprint arXiv:1803.05069* (2018).

[22] Kim Zetter. 2016. Wired: Inside the Cunning, Unprecedented Hack of Ukraine's Power Grid. (2016). https://www.wired.com/2016/03/inside-cunning-unprecedented-hack-ukraines-power-grid/.