IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 6 October 2022, accepted 20 October 2022, date of publication 25 October 2022, date of current version 1 November 2022.

*Digital Object Identifier 10.1109/ACCESS.2022.3216665*

RESEARCH ARTICLE

# Loki: A Physical Security Key Compatible IoT Based Lock for Protecting Physical Assets

**SIBI CHAKKARAVARTHY SETHURAMAN[1], (Member, IEEE), ADITYA MITRA[1], (Student Member, IEEE), KUAN-CHING LI[2], (Senior Member, IEEE), ANISHA GHOSH[1], (Student Member, IEEE), M. GOPINATH[1], (Student Member, IEEE), AND NITIN SUKHIJA[3], (Senior Member, IEEE)**

[1]Center of Excellence, Artificial Intelligence and Robotics (AIR) and School of Computer Science and Engineering, VIT-AP University, Amaravati, Andhra Pradesh 522237, India
[2]Department of Computer Science and Information Engineering, Providence University, Taichung 43301, Taiwan
[3]Computer Science Department, Slippery Rock University, Slippery Rock, PA 16057, USA

Corresponding authors: Sibi Chakkaravarthy Sethuraman (sb.sibi@gmail.com) and Kuan-Ching Li (kuancli@gm.pu.edu.tw)

This work was supported in part by the Vellore Institute of Technology–Andhra Pradesh University under the Center of Excellence in Cyber Security Research Grant.

**ABSTRACT** The twenty first century has witnessed an enormous rise in data produced per person and it has also witnessed newer and advanced forms of digital attacks and instinctively, witnessed a rise in the need for data protection. However, the essential assets are still physical and needs to be protected. Usually vaults, lockers, safes and so on and used for the safe keeping of the physical assets. However, studies have shown they are vulnerable to various attacks. This paper proposes a novel and robust physical lock for safekeeping of physical assets called Loki. A Physical Security key is used to authenticate the lock and it uses a cloud-server architecture. It employs best cloud security practices, proper use of cryptography and trusted computing to mitigate all common risks. The cloud architecture runs a Virtual Machine (VM) to securely authenticate using Fast IDentity Online (FIDO2) specifications. The physical authenticator data is stored in the cloud for security and only accessed when an unlock is requested. The cloud allows web-based physical key management for adding more keys or removing keys. The whole system has been implemented in a Internet of Things (IoT) scenario.

**INDEX TERMS** Physical security key, fast IDentity online, FIDO, FIDO2, IoT lock, smart lock.

## I. INTRODUCTION

Physical assets like currency, jewellery, or important documents are some of the most valuable things we need to protect from being stolen or tampered with. With the advent of modern technology, it has been seen that protecting data is extremely important [1], [2], [3]. However, data is not the only asset we have – we have physical assets to be protected. There have been various types of locks and vaults, like numeric, biometric [4], password protected and so on. However, the use of sophisticated locks becomes complicated and insecure for daily use. Moreover, the older generation is not so accustomed to technology to use modern-day locks like the pin or password-based ones. Most locks have one or the other vulnerability which can be exploited for unauthorized access to the assets.

Implementing cybersecurity features into the security of physical assets has been a challenge for quite a considerable amount of time, giving rights to vaults and lockers with pin/password security, fingerprints, access cards for different personnel in a corporate structure, etc. However, using the same set of physical security keys for securing both physical and digital assets can be instrumental in implementing access control for both seamlessly.

Passwordless authentication has been instrumental in eliminating most attacks that traditional passwords are vulnerable to [5], [6]. FIDO [7] specifications allow the usage of physical security keys with the public key cryptosystem. The private key never leaving the physical security key ensures safety against most digital forms of attack [8]. Moreover, keeping

The associate editor coordinating the review of this manuscript and approving it for publication was Sedat Akleylek.

all data in motion encrypted using symmetric encryption keys are instrumental in preventing man-in-the-middle attacks [9]. Thus, this makes the lock secure against most known attacks.

Various organizations like Google have used FIDO2 security keys for secure Multi-Factor Authentication (MFA), Microsoft (for Passwordless authentication), several social media platforms like Facebook, Twitter, Instagram, Reddit, Cryptocurrency wallets like Binance and Coinbase, online password managers like 1Password and so on [10], [11]. FIDO2 is the new standard for online authentication [12]. FIDO2 is the third iteration after being a successor to FIDO U2F and FIDO UAF. It essentially represents a universal way to implement password less identity on top of existing identity verification infrastructure. A passwordless system is a new approach to verification that removes passwords as a weak point both for security and for social engineering attacks like phishing. However, it is indeed possible to use FIDO2 for authentication in the case of physical locks like vaults and safes or even doors in households.

With traditional locks and keys, the major problem that we saw is the limited number of keys and difficulty to reset them in case of loss of keys. With FIDO2 physical security keys, every user can have their key, and it is easier to reset in case of loss of one. It is even possible to enable or disable security keys remotely. Security keys can be disabled when not needed to prevent unnecessary use. Moreover, a single key can be used to manage all credentials of a particular person: a single FIDO2 compatible key can be used to authenticate the online accounts and the physical assets. Not only door locks, but this can also have various other uses like in vaults and lockers, corporate environments, and smart padlocks. This can have multiple security features like secure authentication and remote authentication disabling. For example, even if both the lock and the key are stolen, it is possible to deactivate the physical security keys until they are recovered remotely. It is possible even to reset the lock back to factory state remotely. Hence, in this paper the use of FIDO2 [13] and Universal 2nd Factor (U2F) [14], [15] compatible physical keys for locks has been proposed. It can be used in various use cases, starting from door locks to vaults and lockers, making them more secure. This brings the latest cybersecurity trends for protecting Physical assets. Also, there is no existing work that applies FIDO for physical security.

The contributions in this paper:

- First work to implement FIDO2 specifications and Passwordless authentication for securing physical assets.
- Multiple keys and easy key management for single lock.
- Seamless access control and complete security against most of the known attacks
- Remote device lock in case of loss of keys or theft.
- Remote addition and deletion of keys.

Rest of the paper is organized as follows: Section II explains the background concepts and the existing related research. Section III discusses about the proposed work and provides the system level architecture. Section IV outlines the core functionality and process such as Lock Registration, Key management and Authentication. Section V presents a prototype of the Loki along with the validation of the results of the proposed Loki and also provides an analysis of Loki. This section also discusses about the security validation and use cases of Loki. Finally, the paper concludes in Section VI.

## II. BACKGROUND AND RELATED WORK

Most digital locks have leveraged PIN/Password security and/or biometrics for safekeeping of physical assets [16]. Mechanical locks mainly implement the use of keys for the same. Such locks are primarily used in vaults, safes, lockers, and sometimes on doors. These have become instrumental in protecting various physical assets like documents, currency, jewellery, etc. However, it is to be noted that such locks have numerous vulnerabilities. For example, passwords and pins, once typed, can be extracted by an attacker using suitable technology. One common way is to use Forward-Looking Infra-Red (FLIR) devices which are thermal cameras and can be used to capture the thermal residue of the user on the keyboard after typing [17]. Similarly, it usually leaves a smudge on the screen on touchscreen-based devices to enter passwords, pins, and patterns. This smudge can be captured even using a standard high-end mobile camera with proper settings. It has also been seen that it is pretty challenging to remove the smudges, even after wiping [18]. Fingerprints authentication systems can exceptionally be vulnerable to a wide variety of attacks, including using an artificial clone of the fingerprint, printed images, or even extracting the fingerprint from the authorized user by social engineering [19]. Mechanical locks and vaults with physical keys are vulnerable to keys being stolen and used later by malicious users. Moreover, it is not easy to control access to the locks in these scenarios. Sometimes even restricting access to a particular person may imply changing the lock pin or password for everyone [16]. Table. 2 shows the comparison of the recent related works with Loki.

Existing studies in the literature shows that FIDO UAF and World Wide Web Consortium (W3C) Verifiable Credentials can be used to present a user-centric and decentralized digital identity system [12], [21]. It has made digital identity highly trustworthy both for the user and the service provider who may be authenticating the user. The entire system was implemented for a banking scenario to show how secure it could be, and has also allowed users to generate on-demand identities that could contain only the necessary information [12]. Their model presented the service provider with the authenticated information from the source directly. Another paper presented the application of FIDO protocol to enable multi-factor authentication in banking scenarios. It allowed a single gesture phishing-resistant multi-factor authentication. It involves the keys and biometrics to stay on the user's device and no server-side secrets. It also ensures no third-party protocol is involved [22]. A study proposed a promising approach to maintain security even after a FIDO authentication is done. A continuous FIDO authentication browser extension allows the Relying Party (RP) and the authenticator

**TABLE 1. Loki vs other physical locks.**

| Features | **Loki** | Pin based Lock [20] | Biometric Lock [4] | Smart Lock [16] |
|---|---|---|---|---|
| Speed of operation | Yes | No | Yes | Slower than Loki |
| User defined keys | Yes | NA | NA | No |
| Access can be blocked when lock/key is stolen | Yes | No | No | No |
| Remote addition or removal of keys | Yes | NA | NA | No |
| Support for biometric authentication | Yes | No | Yes | No |
| Secure against smudge attacks | Yes | No | Yes | Yes |
| Secure against biometric cloning | Yes | Yes | No | NA |
| Secure against mechanical lock picking | Yes | Yes | No | Yes |
| Can enroll new keys/change credentials when one is stolen | Yes | Vulnerable | Vulnerable | No |
| Same key support for multiple locks (Easy key management) | Yes | NA | NA | No |
| Same key for protecting digital assets | Yes | NA | NA | No |

**TABLE 2. Comparison of Loki with the recent related works.**

| Research | Conventional Methods [16] | UCIM [12] | **Current paper - Loki** |
|---|---|---|---|
| Security Logic | Password/Identity | Passwordless | Passwordless |
| Authentication protocol | Application Specific | UAF | Web Authentication (WebAuthn) and CTAP |
| Communication protocol | HTTP | HTTP/HTTPS | HTTPS only |
| Library utilized | Application specific | Application Specific | FIDO2, Fernet |

to continuously exchange verification in the background. It has been validated using an Android-based roaming authenticator communicating via BLE [23]. Another research [24] presented a large-scale lab study of FIDO2 single-factor authentication and collected insights about the perception, acceptance, and concerns about passwordless authentication among the end-users. Their results showed that users are willing to accept a replacement of text-based passwords with a security key for single-factor authentication [24]. A study proposed FIDO-based password management. It allowed the user to log in to the system with a password and biometric information using physical security keys. This method was even backwards compatible with legacy password-based authentication. Their approach also guaranteed registration and authentication in polynomial-times [21]. The problem formulation of the current paper is listed below:

- Vulnerabilities of locks used for safe keeping of physical assets
- Difficulty in access control and key management in such locks
- Different Authentication methods of physical assets and digital assets.

It is known that the existing physical locks are vulnerable to many attacks. There are various types of locks, one is more vulnerable than another, and it all depends on the use cases. Loki advances all the problems and offers a lock compatible with FIDO protocols, allowing the use of physical security keys. The public key cryptosystem protects the authentication system and the private keys never leave the security key. The protocols have been tested and proven successful in protecting digital assets. Moreover, the same physical security key can also be used for protecting digital assets in the corporate environment. Access control can be done quickly by provisioning and de-provisioning keys remotely by the administrator. We believe that this study is the first to use FIDO protocols for securing digital assets. The conceptual workflow of Loki is depicted in Figure. 1. Table 1 presents a comparative perspective of Loki and other physical locks.

Loki does not have any of the vulnerabilities discussed above as it uses FIDO2 specifications for authentication in a client-server model [11]. The proposed system can used in various places for the safekeeping of digital assets. For example, corporate entities using Azure Active Directory can use FIDO2 compatible physical security keys for access control to digital assets. FIDO2 is a new age technology for MFA and Passwordless authentications in digital asset security. FIDO2 leverages Client to Authenticator Protocol (CTAP) [12] and public key cryptosystem for secure authentication. FIDO is a relatively newer technology [13] and there is a high possibility that it will be a highly used authentication technology online shortly. FIDO2 is becoming instrumental in making the secure physical posture [12], [13] better by enforcing the use of physical security keys or other FIDO compatible physical devices for authentication. Loki can be easily set up without any hassle of complex pin/password settings and configurations. This has been made possible using FIDO Device Onboard (FDO) specifications for automated and secure IoT device provisioning [25]. For unlocking a lock using Loki, a physical security key is needed. Keys can be managed easily and seamlessly using an online portal by the administrator. Hence, Loki also has added security features like remotely wiping the keys in case of a key has been compromised. Further, biometric authentication, a fingerprint-compatible physical security key like the YubiKey Bio [26] is to be used.

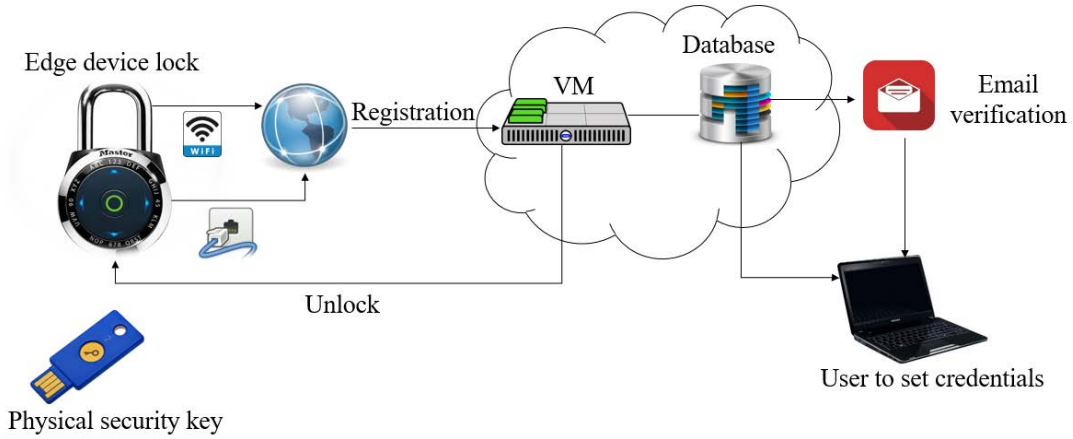



**FIGURE 1.** Loki Workflow overview.

### A. PROPOSED SOLUTION AND NOVELTY OF LOKI

It is evident from the above discussion that there are physical-mechanical locks that uses physical keys. However, there is no option for key management in these kinds of locks. It is vulnerable to mechanical lock picking. Further, the key can be easily cloned, making it vulnerable. The lock can be easily unlocked with a stolen key, thus, making it very prone to thefts. Pin Based locks [27] are vulnerable to smudge attacks, thermal imaging attacks [8] and social engineering where one might be made to reveal the PIN. The only form of access control to this is that only the authorized users would know the pin, which is not secure. Biometric locks [28] are vulnerable to biometric cloning. Smart locks [29] do not have user-defined keys or support biometric authentication. There are locks that allow resettable keys for better security [30],however, to reset this type of a lock, the previous key is needed. It is also possible that the unauthorized person who might have the key may reset the keys, denying access to the authorized persons. Hence, all these types of security locks have one or the other drawback or vulnerability. We believe that Loki is the first physical locking system to introduce user-defined keys, seamless key management, access control, seal the lock remotely, use the same key for multiple locks and protect digital assets. It also leverages public key cryptosystem and trusted computing to secure the authentication process against the most known threats. As per our knowledge, Loki is the first physical lock to use FIDO.

## III. PROPOSED WORK

Loki can be used in multiple ways, from door locks to secure vaults and lockers. It interfaces with a servo motor that can be mechanically attached with physical locks or other electrical locks connected to the onboard pins left open. Loki can act as an independent lock (with the use of 3D printable attachments for locking) or interface with other mechanical or digital locks.

Loki offers a methodology for using FIDO2 compatible physical security keys (optionally adds biometrics) to protect physical assets. Robust cloud security principles and in-transit data security protect against most known attacks. Moreover, implementing a public key cryptosystem and trusted computing ensures the keys cannot be cloned (both physically and in case of repudiation). Moreover, in every unlock request, a different token and challenge combination is used, eliminating replay attacks. From a legitimate user's point of view, using Loki is as simple as using a standard mechanical lock. The difference is that in the case of Loki, the key is just a USB Physical security key.

### A. HARDWARE DESIGN OVERVIEW OF LOKI

A low-power Single Board Computer (SBC) running Ubuntu OS is used to communicate with the physical security key and the backend over the internet. The SBC is further connected to a screen that may be used to display the QR Code when Loki is in OOBE mode. The SBC is again connected to a microcontroller for serial communication over USB. This microcontroller can be further interfaced with actuators for the lock, and it also provides an interface to connect existing electrical locking mechanisms (See Figure. 2 for detailed circuit diagram). For development purposes, the devices listed in Table 3 are used

**TABLE 3.** Hardware components list.

| Component | Used device |
|---|---|
| SBC | Raspberry Pi 4 |
| Microcontroller | Arduino Mega (ATmega 2560) |
| Actuator | Servo Motor |
| Physical security key | Yubico Security Key |
| Cloud Instances | Azure Cloud platform |

### B. SOFTWARE DESIGN OVERVIEW OF LOKI

The proposed system essentially works on the Cloud and Edge computing models [31]. An edge computing

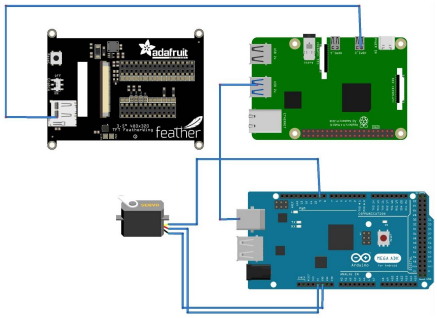


**FIGURE 2.** Circuit diagram of Loki.

microcontroller on the lock interacts with the cloud server to authenticate using FIDO2 specifications. However, in more secure regions and regions without internet connectivity, an on-premises server can be used instead of cloud. The lock contains an onboard running Ubuntu as the Edge server in the background. A web browser, preferably Mozilla Firefox runs in the foreground with which the user can interact. The locking mechanisms are controlled with actuators connected to an onboard connected to another onboard (ARM64 microcontroller). The cloud architecture uses a VM and a SQL Database to hold the public keys of the registered devices and acts as a web server for access control and other operations. The cloud architecture is hosted on Microsoft Azure. The following protocols and specifications have been used to develop Loki:

- The device is developed in Cloud and Edge computing models
- FIDO2 specifications [12], [13] that encompass W3C WebAuthn and FIDO Alliance's CTAP.
- FDO.

## IV. LOCK REGISTRATION, KEY MANAGEMENT AND AUTHENTICATION BY LOKI

Registration includes setting up the Loki device from Out of Box Experience (OOBE) mode to the usable mode. This is done by scanning a QR that is displayed on the screen as the device is turned on for the first time. It is then followed by the user registering their email address and verifying the same using a one-time password sent to it. Further a physical security key is added to the account which can be used to unlock Loki. The authentication process simply includes plugging in the physical security key to the USB port of Loki and seamlessly unlocking it. Key management includes allowing one physical security key for multiple online services, as well as using multiple physical keys, corresponding to multiple users for Loki.

The device must be registered first before using, and security keys can be added or removed remotely. The Cloud VM has firewall rules allowing inbound traffic to the Flask program's port. This VM acts all three, the rendezvous server, the Relying Party, and the target web server. However, this rendezvous server is tied with this target web server only for the time being. An SQL server is also used to hold records of the device.

### A. DEVICE REGISTRATION

The device shows a QR code that can be scanned to register the device out of the box. The registration process includes registering the username and email ID of the user. The email ID is verified with an OTP before saving to the database. When registration is complete, the OOBE is exited. Hence, the device is provisioned. Fig. 3 shows the device registration workflow.

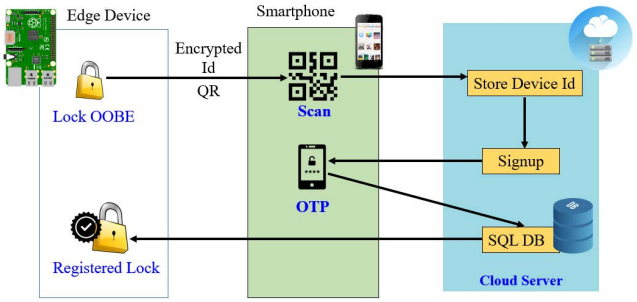



**FIGURE 3.** Device Registration Workflow.

### B. LOGGING IN AND KEY REGISTRATION

The user attempting to log in must enter his username and authenticate using an OTP sent to his email. For key addition, the webpage will prompt the user for the key to be connected to the USB port or scanned with Near-Field Communication (NFC). Fig. 4 shows the key registration workflow.

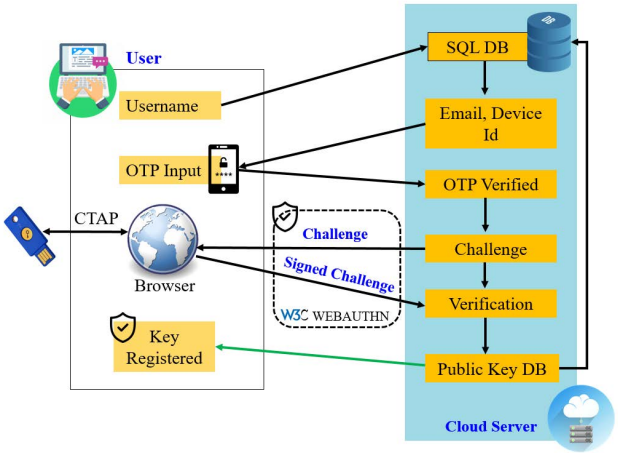



**FIGURE 4.** Login and key registration workflow.

### C. AUTHENTICATION

As the user taps on the 'unlock' button on the device, it prompts for the registered security key. It unlocks if the correct key is inserted into the device USB port. Fig. 5 shows the authentication workflow.

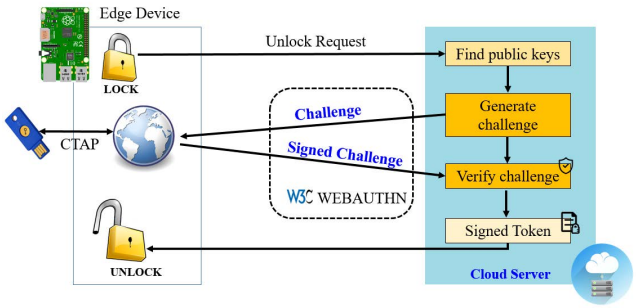



**FIGURE 5.** Authentication Workflow.

### D. DELETING KEYS AND FACTORY RESETTING LOKI

When the user wants to delete the keys, he logs in to the dashboard with his username. The delete keys button clears all saved security keys of the device. The 'Reset Device' button on the dashboard is used to delete the keys and then remove the record from the database, thus resetting the device.

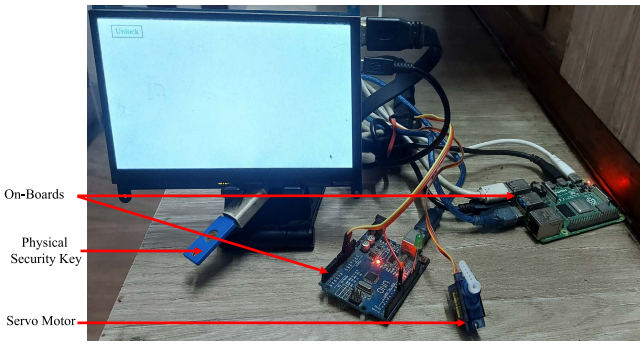



**FIGURE 6.** Loki - A Physical Security Key compatible lock.

### E. ATTACK MODEL USED TO VALIDATE LOKI

Figure. 6 shows the experimental testbed of Loki. Loki follows the FIDO2 specifications, which implies it uses public-key cryptosystem for device attestation. The private cryptographic keys never leave the physical security key, making it secure. The device identities are stored in Azure SQL Database, which is always encrypted, making sure it is decrypted only during transactions. Virtual network firewall rules ensure that only the backend virtual machine can access the database. Only the virtual machine can access the database with SQL Authentication. The public cryptographic keys, residing on the virtual machine's disks, have Azure Disk Encryption to prevent unauthorized access. Microsoft Anti-malware helps protect the backend against known threats and malware in real-time.

## V. EXPERIMENTAL VALIDATION OF LOKI

### A. EXPERIMENTAL SETUP

The lock or the client device has been implemented on a SBC (Single Board Computer) with processor Broadcom BCM2711, quad-core Cortex-A72 (ARMv8) 64-bit Systems on Chip (SoC) with a clock speed of 1.5GHz. It has a 4GB of LPDDR4 primary memory and connectivity of 2.4 GHz and 5.0 GHz IEEE 802.11b/g/n/ac wireless LAN (Wi-Fi). It further has a Gigabit Ethernet port and can be connected to the internet without wireless network, making it more secure. The operating system used was Ubuntu 20.04. A servo motor was used to emulate the lock which was further connected to an Arduino ATmega2560 based microcontroller. The cloud server is running on a standard Azure B1s Virtual Machine with 1 GiB RAM and 1 vCPU. It is running Ubuntu Server 18.04-LTS. It has been secured with firewall and network security groups to keep malicious users out.

### B. ATTACK ANALYSIS

- Malware: Microsoft Anti-malware service scans the backend virtual machine in real-time for any known threat or malware and it is mitigated.
- Phishing: FIDO2 specifications enforce verification of the Relying Party identity at every step, making sure phishing is not possible in this model. Moreover, SSL certificates are used to verify ownership of the backend and mitigate the risk of phishing. Hence neither the database, not the email of the user can be hacked.
  - A phished website cannot replay the FIDO key inputted by the legitimate user. And a phished server requesting the FIDO keys will result in an IP error on the client-side. For example, when a request is made to a phished server running the same code as the legitimate server, the Invalid Domain error is thrown.
- Man-in-the-middle (MITM): All communications between the edge device and the cloud backend is encrypted with symmetric encryption keys. Thus, MITM attacks are mitigated, and the authorization tokens are protected.
- Distributed Denial-of-Service (DDoS): The virtual machine on Azure is protected by Azure DDoS protection. DDoS may be caused by attacks like bot attacks [32]. It is covered with active traffic monitoring and always-on detection. Moreover, automatic attack mitigation is helpful for backend resilience. This is tested using the Hulk tool to increase the server's load and monitor it using Azure insights. It is observed that after a high number of requests are made, Azure blocks the incoming requests from the client attempting the DDoS while normal operations are unaffected.
- SQL Injection: Advanced Threat Protection for Azure SQL Database detects anomalous activities indicating unusual and potentially harmful attempts to access or exploit databases. Advanced Threat Protection can identify Potential SQL injection, Access from unique locations or data center, access from an unfamiliar principal or potentially harmful application, and brute force SQL credentials. Moreover, the SQL database is accessible only from the backend virtual machine, not allowing

users to access it directly. Thus, the databases are protected against all known forms of attacks against it.

- Attack on Edge device: The edge device does not provide any interface for an adversarial actor to perform an attack. The only interface the Edge device provides is a single USB port for the security key.
- Device cloning: The security keys are protected with Physically Unclonable Functions (PUFs) and hence are resilient to device cloning.

### C. PERFORMANCE ANALYSIS

It is seen that Loki takes significantly less time, about 500 ms (plus user input times), for registering and authenticating the security keys. Hence, it can be claimed that it is quite efficient and can be implemented in real-world scenarios. 5 keys have been registered and authenticated. The time taken for the operations (Key registration and Authentication & unlocking) are given in the Table. 4 and Table. 5.The analysis was done with the edge device which have an internet connectivity of about 100 Mbps. However, the challenge being of 16 bytes, it is tested to be working with similar speed under low internet conditions.

**TABLE 4.** Key registration time.

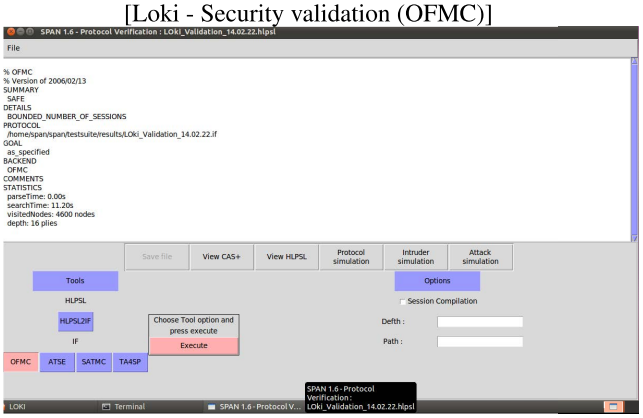
| Sl.No. | Time to create challenge | Time to verify and register key | Total time for processing |
|---|---|---|---|
| 1 | 302ms | 288ms | 590ms |
| 2 | 311ms | 277ms | 588ms |
| 3 | 311ms | 64ms | 375ms |
| 4 | 412ms | 312ms | 724ms |
| 5 | 346ms | 274ms | 620ms |
| Average | 336.4ms | 243ms | 579.4ms |

**TABLE 5.** Authentication and unlock time.

| Sl.No. | Time to create challenge | Time to verify & unlock | Total time for processing |
|---|---|---|---|
| 1 | 212ms | 319 ms | 531ms |
| 2 | 193ms | 203ms | 396ms |
| 3 | 224ms | 324ms | 548ms |
| 4 | 234ms | 304ms | 538ms |
| 5 | 260ms | 209ms | 469ms |
| Average | 224.6ms | 271.8ms | 496.4ms |

The backend virtual machine is on elastic scaling that scales the backend VMs up or down when needed. It is further going through an elastic load balancer which is instrumental in distributing the load among all the VMs in the VM set. Availability sets and Availability zones make the backend resilient to most outages. According to the Service Level Agreements (SLA), the monthly uptime should be over 99.99% and using Availability sets, Availability zones can be instrumental in keeping the service up even in case of a disaster or a catastrophic failure in a data center. The edge device in the lock will work as long as there is internet and power connectivity. For uninterrupted usage, proper error and exception handling and recovery have been implemented in both the edge devices and the cloud VMs. Thus, it is claimed that Loki, in the experimental setup, runs fast enough and is reliable enough against downtimes and failures.

[Loki - Security validation (OFMC)]




[Loki - Security validation (ATSE)]

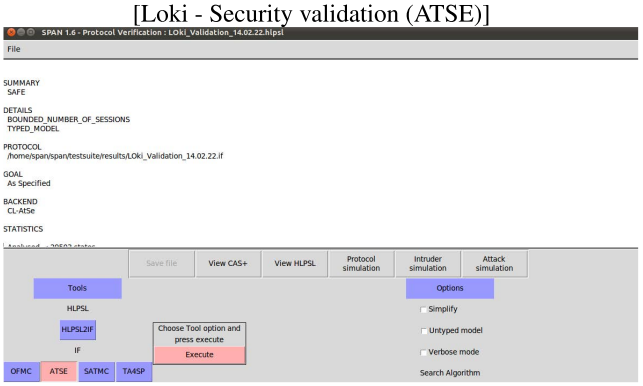



**FIGURE 7.** AVISPA security validation.

### D. SECURITY VALIDATION OF LOKI

Security validation of the proposed system is performed using the Automated Validation of Internet Security Protocols (AVISPA) tool [33], [34]. AVISPA Security tool analyzes the theoretical workflow of the security logic and returns any security flaw or vulnerability that could be exploited on the same. Modular and expressive formal language specifications such as High-Level Protocol Specification Language (HLPSL) and CAS+ is provided by AVISPA to specify protocols and their security characteristics of them. Further, the validation is performed with four distinct back-end tools: OFMC, ATSE, SATMC, and TA4SP. OFMC and ATSE tools afford security analysis by applying various automated state-of-the-art analysis approaches. Further, the roles are defined in Dolev–Yao (dy) adversary model, which involves characteristics verification related to the internet security protocols. The attack analysis is performed on all the attacks as specified in section V-B.

The security logic of Loki is specified in HLPSL format with hlpsl extension as Intermediate File (IF). It is then forwarded to OFMC and ATSE tools as input. The back end tools (OFMC and AtSe) perform the execution of the Loki. SAFE status is attained for the Loki while reviewing the summary

(See Figure. 7) of AVISPA's backend tools OFMC and ATSE which implies that the Loki is secure without any violation. Reports of AVISPA reported that the executed operations are safe and does not have any known vulnerability.

Loki has been validated against standard test conditions and under attack models. It has been seen that Loki is resilient against most attacks and can resume regular operation even under attack conditions.

### E. USE CASES UTILIZING LOKI

Loki can be useful in the following scenarios:

- *Home usage*: Every member may have a different key so that whenever a person comes, he can unlock the door with his own key. Also, access control can be implemented in a home. Also, the same key can be used in multiple locks.
- *Vaults and Lockers*: Every authorized user can have a different key, even access logging can be enabled easily. Moreover, these are FIDO2 compliant security keys. In corporate scenarios, the same key can be used to provide access to physical assets and digital assets and devices, using frameworks like Azure Active Directory (AD).
- If a lock or a key is stolen, the keys can be remotely wiped so that they cannot be unlocked.
- Transferring physical assets from one site to another: In many cases, a physical asset is to be moved from one place to another, and the logistics party is unreliable. It is also not possible to physically transfer the key to the locker. In such cases, this can be useful. The asset can be locked with this locker and moved. As the recipient party confirms that the package has been received, the sender can remotely wipe the device, and the device goes back to OOBE. Now, the recipient can register their own keys and unlock with it.
- Loki uses a client-server computing model to connect the cloud server to the edge server, which is physically interfaced with a lock using actuators. The cloud holds the necessary information to authenticate a user for unlocking and provides a web interface for the user and key management. Keeping all data in-transit encrypted ensures attacks against any adversary. Loki is protected from all common attacks, both physical and digital. The security is validated with various tools. The performance analysis of Loki shows it is pretty fast for a day to day use and the protection of physical assets. Loki provides an interface to connect various actuators as required by a lock to be operated by Loki. A user of Loki may choose to use the cloud server (in the case of home users), or a corporate entity may choose to set up the Loki backend on their on-premise servers for key management.

## VI. CONCLUSION AND FUTURE WORK

This paper presents a standard that allows FIDO2 compliant physical security keys to be used with IoT based physical locks and can be used for the safekeeping of tangible assets. This enables users to manage keys with registered locks easily. In corporate scenarios, this standard can allow access control to physical assets. A reference implementation has been demonstrated with an SBC to show the concept. The performance analysis and the demonstrated attack models (AVISPA security validation and real-time validation) confirm that the proposed standard is utmost secured in all the validated conditions and performs well. The paper also discusses the advantages of Loki over other locks, vaults, and lockers and exhibits how Loki is more secure. Finally, Loki has been implemented to develop a lock for the safekeeping of physical assets, and it can also be used for other forms of physical security like door locks and so on. According to the best of our knowledge, Loki is the first physical lock to implement FIDO2 based security, enabling seamless access control where users can be authorized or unauthorized from accessing the assets remotely.

As future work, this work can be further improved for smartphone-based remote unlock features. Other physical locks like car locks provide security with public-key cryptosystems and symmetric cryptosystems that may be proven to be beneficial.

Loki has been validated with various security testing tools. It can be concluded that Loki can be used to provide physical security keys based cryptographic security to physical locks and IoT based devices. This will be instrumental in providing proper security to physical assets while keeping proper access management for the same. However, it is to be noted that Loki has some minor limitations. It requires to be plugged in to a power source to work. This might be overcome with the use of battery. Another weakness includes it needs network connectivity to the server for functioning. If the server is hosted in a cloud across the internet, Loki would need internet connectivity. Otherwise, if it is hosted in an on-premises server, a wired or wireless connection to the same is required for proper functioning of Loki.

## ACKNOWLEDGMENT

The authors thank the management for motivating and supporting the Artificial Intelligence and Robotics (AIR) Research Centre, VIT-AP University, in building this project. The author Sibi Chakkaravarthy Sethuraman would also like to thank the Center for Excellence in Cyber Security, VIT-AP University.

## REFERENCES

[1] R. Kumar, P. Kumar, R. Tripathi, G. P. Gupta, T. R. Gadekallu, and G. Srivastava, "SP2F: A secured privacy-preserving framework for smart agricultural unmanned aerial vehicles," *Comput. Netw.*, vol. 187, Mar. 2021, Art. no. 107819.

[2] P. Kumar, R. Kumar, G. Srivastava, G. P. Gupta, R. Tripathi, T. R. Gadekallu, and N. N. Xiong, "PPSF: A privacy-preserving and secure framework using blockchain-based machine-learning for IoT-driven smart cities," *IEEE Trans. Netw. Sci. Eng.*, vol. 8, no. 3, pp. 2326–2341, Jul. 2021.

[3] P. Kumar, R. Kumar, G. P. Gupta, R. Tripathi, and G. Srivastava, "P2TIF: A blockchain and deep learning framework for privacy-preserved threat intelligence in industrial IoT," *IEEE Trans. Ind. Informat.*, vol. 18, no. 9, pp. 6358–6367, Sep. 2022.

[4] A. Roy, N. Memon, and A. Ross, "MasterPrint: Exploring the vulnerability of partial fingerprint-based authentication systems," *IEEE Trans. Inf. Forensics Security*, vol. 12, no. 9, pp. 2013–2025, Sep. 2017.

[5] W. Liang, S. Xie, D. Zhang, X. Li, and K.-C. Li, "A mutual security authentication method for RFID-PUF circuit based on deep learning," *ACM Trans. Internet Technol.*, vol. 22, no. 2, pp. 1–20, May 2022.

[6] Z. Xu, W. Liang, K.-C. Li, J. Xu, A. Y. Zomaya, and J. Zhang, "A time-sensitive token-based anonymous authentication and dynamic group key agreement scheme for Industry 5.0," *IEEE Trans. Ind. Informat.*, vol. 18, no. 10, pp. 7118–7127, Oct. 2022.

[7] K. Hu and Z. Zhang, "Security analysis of an attractive online authentication standard: FIDO UAF protocol," *China Commun.*, vol. 13, no. 12, pp. 189–198, Dec. 2016.

[8] D. Han, N. Pan, and K.-C. Li, "A traceable and revocable ciphertext-policy attribute-based encryption scheme based on privacy protection," *IEEE Trans. Dependable Secure Comput.*, vol. 19, no. 1, pp. 316–327, Jan. 2022.

[9] Y. Fan, J. Liu, K.-C. Li, W. Liang, X. Lei, G. Tan, and M. Tang, "One enhanced secure access scheme for outsourced data," *Inf. Sci.*, vol. 561, pp. 230–242, Jun. 2021.

[10] M. Morii, H. Tanioka, K. Ohira, M. Sano, Y. Seki, K. Matsuura, and T. Ueta, "Research on integrated authentication using passwordless authentication method," in *Proc. IEEE 41st Annu. Comput. Softw. Appl. Conf. (COMPSAC)*, Jul. 2017, pp. 682–685.

[11] S. Ghorbani Lyastani, M. Schilling, M. Neumayr, M. Backes, and S. Bugiel, "Is FIDO2 the kingslayer of user authentication? A comparative usability study of FIDO2 passwordless authentication," in *Proc. IEEE Symp. Secur. Privacy (SP)*, May 2020, pp. 268–285.

[12] R. Laborde, A. Oglaza, S. Wazan, F. Barrere, A. Benzekri, D. W. Chadwick, and R. Venant, "A user-centric identity management framework based on the W3C verifiable credentials and the FIDO universal authentication framework," in *Proc. IEEE 17th Annu. Consum. Commun. Netw. Conf. (CCNC)*, Jan. 2020, pp. 1–8.

[13] D. W. Chadwick, R. Laborde, A. Oglaza, R. Venant, S. Wazan, and M. Nijjar, "Improved identity management with verifiable credentials and FIDO," *IEEE Commun. Standards Mag.*, vol. 3, no. 4, pp. 14–20, Dec. 2019.

[14] H. Luo, C. Wang, H. Luo, F. Zhang, F. Lin, and G. Xu, "G2F: A secure user authentication for rapid smart home IoT management," *IEEE Internet Things J.*, vol. 8, no. 13, pp. 10884–10895, Jul. 2021.

[15] K. Fan, H. Li, W. Jiang, C. Xiao, and Y. Yang, "Secure authentication protocol for mobile payment," *Tsinghua Sci. Technol.*, vol. 23, no. 5, pp. 610–620, Oct. 2018.

[16] B. Patil, P. Vyas, and R. K. Shyamasundar, "SecSmartLock: An architecture and protocol for designing secure smart locks," in *Information Systems Security*, V. Ganapathy, T. Jaeger, and R. Shyamasundar, Eds. Cham, Switzerland: Springer, 2018, pp. 24–43.

[17] T. Kaczmarek, E. Ozturk, and G. Tsudik, "Thermanator: Thermal residue-based post factum attacks on keyboard data entry," in *Proc. ACM Asia Conf. Comput. Commun. Secur.*, Jul. 2019, pp. 586–593.

[18] S. S. Ul hassan, A. Ghani, M. Bilal, and A. Jolfaei, "Multifactor pattern implicit authentication," *IEEE Consumer Electronics Magazine*, vol. 11, no. 1, pp. 26–32, Jan. 2021.

[19] A. Razaque, K. K. Myrzabekovna, S. Y. Magbatkyzy, M. Almiani, B. A. Doszhanovna, and A. Alnusair, "Secure password-driven fingerprint biometrics authentication," in *Proc. 7th Int. Conf. Softw. Defined Syst. (SDS)*, Apr. 2020, pp. 95–99.

[20] G. Singh, S. Butakov, and B. Swar, "Thermal print scanning attacks in theretail environments," in *Proc. Int. Siberian Conf. Control Commun. (SIBCON)*, Apr. 2019, pp. 1–6.

[21] H. Kim, D. Lee, and J. Ryou, "User authentication method using FIDO based password management for smart energy environment," in *Proc. Int. Conf. Data Mining Workshops (ICDMW)*, Nov. 2020, pp. 707–710.

[22] Z. P. Zwane, T. E. Mathonsi, and S. P. Maswikaneng, "An intelligent security model for online banking authentication," in *Proc. Conf. IST-Afr.*, 2021, pp. 1–6.

[23] E. Klieme, J. Wilke, N. van Dornick, and C. Meinel, "FIDOnuous: A FIDO2/WebAuthn extension to support continuous web authentication," in *Proc. IEEE 19th Int. Conf. Trust, Secur. Privacy Comput. Commun. (TrustCom)*, Dec. 2020, pp. 1857–1867.

[24] S. Ghorbani Lyastani, M. Schilling, M. Neumayr, M. Backes, and S. Bugiel, "Is FIDO2 the kingslayer of user authentication? A comparative usability study of FIDO2 passwordless authentication," in *Proc. IEEE Symp. Secur. Privacy (SP)*, May 2020, pp. 268–285.

[25] A. Jabbari and J. B. Mohasefi, "A secure and LoRaWAN compatible user authentication protocol for critical applications in the IoT environment," *IEEE Trans. Ind. Informat.*, vol. 18, no. 1, pp. 56–65, Jan. 2022.

[26] C. Bischoff, E. Gerlitz, and M. Smith, "Vision: I don't want to use my phone! A cognitive walkthrough for YubiKeys," in *Proc. IEEE Eur. Symp. Secur. Privacy Workshops (EuroSPW)*, Sep. 2020, pp. 160–165.

[27] M. Zhou, Q. Wang, X. Lin, Y. Zhao, P. Jiang, Q. Li, C. Shen, and C. Wang, "PressPIN: Enabling secure PIN authentication on mobile devices via structure-borne sounds," *IEEE Trans. Dependable Secure Comput.*, early access, Feb. 16, 2022, doi: 10.1109/TDSC.2022.3151889.

[28] Y. Liu, T. Zhou, Z. Yue, W. Liu, L. Y. Han, Q. Li, and X. Yang, "Secure and efficient online fingerprint authentication scheme based on cloud computing," *IEEE Trans. Cloud Comput.*, early access, Aug. 10, 2021, doi: 10.1109/TCC.2021.3103546.

[29] S. Yu, N. Jho, and Y. Park, "Lightweight three-factor-based privacy-preserving authentication scheme for IoT-enabled smart homes," *IEEE Access*, vol. 9, pp. 126186–126197, 2021.

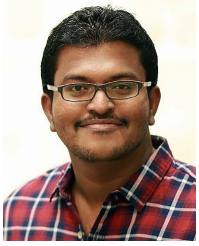
[30] Kwikset. *US20120312127a1—Device for Resetting Locks*. Accessed: Oct. 1, 2022. [Online]. Available: https://patents.google.com/patent/US20120312127

[31] C. Stergiou, K. E. Psannis, B.-G. Kim, and B. Gupta, "Secure integration of IoT and cloud computing," *Future Gener. Comput. Syst.*, vol. 78, pp. 964–975, Jan. 2018.

[32] V. A. Memos, K. E. Psannis, and Z. Lv, "A secure network model against bot attacks in edge-enabled industrial Internet of Things," *IEEE Trans. Ind. Informat.*, vol. 18, no. 11, pp. 7998–8006, Nov. 2022.

[33] A. Armando, D. Basin, Y. Boichut, Y. Chevalier, L. Compagna, J. Cuellar, P. H. Drielsma, P. C. Heám, O. Kouchnarenko, and J. Mantovani, "The AVISPA tool for the automated validation of internet security protocols and applications," in *Proc. Int. Conf. Comput. Aided Verification*, 2005, pp. 281–285.

[34] D. Das, S. C. Sethuraman, and S. C. Satapathy, "A decentralized open web cryptographic standard," *Comput. Electr. Eng.*, vol. 99, Apr. 2022, Art. no. 107751.

**SIBI CHAKKARAVARTHY SETHURAMAN** (Member, IEEE) received the Ph.D. degree from Anna University, in 2018. He is currently working as an Associate Professor with the School of Computer Science and Engineering, Vellore Institute of Technology–Andhra Pradesh (VIT-AP) University. He is the Co-ordinator of the Artificial Intelligence and Robotics (AIR) Research Center, VIT-AP University. He is the Lead Engineer for the project VISU, an advanced 3D printed humanoid robot developed by the VIT-AP. He is an active contributor of open source community and lead writer in top security magazines, such as Pentestmag and eforensics. He is an Active Reviewer of many reputed journals, including IEEE, Springer, IET, IGI Global, and Hindawi. He was a recipient of the DST Fellowship.

**ADITYA MITRA** (Student Member, IEEE) is currently pursuing the Bachelor of Technology degree with the Vellore Institute of Technology–Andhra Pradesh (VIT-AP) University. His research interests include deep learning approaches, the Internet of Things in consumer electronics, and embedded hardware.

**KUAN-CHING LI** (Senior Member, IEEE) received the Licenciatura degree in mathematics and the M.S. and Ph.D. degrees in electrical engineering from the University of Sao Paulo (USP), Brazil, in 1994, 1996, and 2001, respectively. He is currently a University Distinguished Professor with the Department of Computer Science and Information Engineering (CSIE), Providence University, Taichung, Taiwan, where he is also the Director of the High-Performance Computing and Networking Center, established by collaborations with industry. Besides publishing articles in renowned journals and conferences, he is the coauthor/co-editor of more than 30 books published by Taylor & Francis, Springer, IGI Global, and McGraw-Hill. He has been actively involved in several national and international conferences in several countries in various capacities. His research interests include parallel and distributed computing, big data, and emerging technologies. He is a fellow of IET.

**ANISHA GHOSH** (Student Member, IEEE) is currently pursuing the Bachelor of Technology degree with the Vellore Institute of Technology–Andhra Pradesh (VIT-AP) University. Her research interests include deep learning approaches, the Internet of Things in consumer electronics, and embedded hardware.

**M. GOPINATH** (Student Member, IEEE) is currently pursuing the Doctor of Philosophy (Ph.D.) degree with the Vellore Institute of Technology–Andhra Pradesh (VIT-AP) University. His research interests include cyber security, malware analysis, the Internet of Things in consumer electronics, and embedded hardware.

**NITIN SUKHIJA** (Senior Member, IEEE) received the B.S. degree (Hons.) in computer science engineering from the Institute of Technology and Management, India, in 2002, the Post Graduate Diploma degree in financial management from Symbiosis International, India, in 2005, the M.B.A. degree in information systems from San Diego State University, in 2009, the M.S. degree in computer science majoring in computing from National University, San Diego, CA, USA, in 2010, and the Ph.D. degree in computer science majoring in high performance computing from Mississippi State University, in 2015. He is currently an Associate Professor with the Department of Computer Science. He has 12 years of work experience, where he worked for several organizations and on different IT platforms facilitating research, teaching and curriculum development in diverse areas funded through the National Science Foundation, Department of Energy, and Department of Defense. His research is recognized by publications in high impact peer-reviewed IEEE and ACM conferences, journals and book chapters. His research efforts are directed towards the resilience modeling, analysis and development of the scheduling algorithms with the goal of insuring high performance for scientific and big data applications on current and future computing environments, such as clusters, grids, clouds, and others. His research interests include high-performance computing, dynamic load balancing, performance modeling, prediction and evaluation, robustness and resilience analysis, cyber security, and big data analytics. He was a recipient of research, career awards and fellowships. He is also serving as an organizing committee member and a reviewer for many esteemed conferences and journals, and as a NSF XSEDE Campus Champion for Slippery Rock University.