A Blockchain Solution for Enhancing Cybersecurity Defence of IoT

K. M. Giannoutakis[1], G. Spathoulas[2], C. K. Filelis-Papadopoulos[3], A. Collen[4],
M. Anagnostopoulos[5], K. Votis[6], N. A. Nijdam[7]
[16]*Information Technologies Institute, Centre for Research and Technology Hellas*, Thessaloniki, Greece
[25]*Dept. of Information Security and Communication Technology*, Gjøvik, Norway
[3]*Department of Computer Science, University College Cork*, Cork, Ireland
[47]*Centre Universitaire d'Informatique, University of Geneva*, Geneva, Switzerland
[16]{kgiannou, kvotis, }@iti.gr, [3]christos.papadopoulos-filelis@cs.ucc.ie,
[47]{georgios.spathoulas, marios.anagnostopoulos}@ntnu.no,
[25]{anastasija.collen, niels.nijdam}@unige.ch

***Abstract*—The growth of IoT devices during the last decade has led the development of smart ecosystems, such as smart homes, prone to cyberattacks. Traditional security methodologies support to some extend the requirement for preserving privacy and security of such deployments, but their centralized nature in conjunction with low computational capabilities of smart home gateways make such approaches not efficient. Last achievements on blockchain technologies allowed the use of such decentralized architectures to support cybersecurity defence mechanisms. In this work, a blockchain framework is presented to support the cybersecurity mechanisms of smart homes installations, focusing on the immutability of users and devices that constitute such environments. The proposed methodology provides also the appropriate smart contracts support for ensuring the integrity of the smart home gateway and IoT devices, as well as the dynamic and immutable management of blocked malicious IPs. The framework has been deployed on a real smart home environment demonstrating its applicability and efficiency.**

***Index Terms*—Blockchain, smart home, IoT, Cybersecurity**

I. Introduction

Blockchain technology has been adopted during the last years in many application domains in order to strengthen security issues in a decentralized manner. Due to its nature, this technology has direct applicability to Internet of Things smart homes environments, as it can support security-related use cases.

In this work, a blockchain security defence mechanism is proposed targeting IoT smart homes. The framework supports smart contracts for implementing registration of users and hardware elements that constitute a smart home, while realizing security related operations for ensuring the integrity of these elements and blacklisting of malicious IPs. The methodology introduces a blockchain technology agnostic solution that can be adopted easily for any installation with minimal requirements on the smart home gateways.

The solution proposed, developed under GHOST EU project, [1]–[3] and coupled with behavioral anomaly detection procedures, provides a novel defence mechanism for smart homes. Towards this, a private Ethereum network has been established, for demonstrating the applicability and efficiency of the proposed implementation on lightweight hardware (Gateways) that is used for managing the IoT devices.

The rest of the paper is organized as follows. Section II discusses the related work on the research field, whereas Section III presents background information regarding its applicability on a real deployment using the Ethereum Blockchain infrastructure. Section IV presents the proposed components for enhancing the security of smart home installations, while Section V summarizes the conclusions of the study.

II. Related Work

A 2016 literature review [4] identified various research papers that use the Blockchain technology in other areas beyond cryptocurrencies, including data storage management, identity management, rating systems, data and goods trading, with only four use cases being identified as applicable in the context of IoT. Nevertheless, the IoT domain and its applicability for Blockchain solutions continued to raise attention in research and industrial deployments [5], [6]. The main reason behind this trend is the structure of the IoT, currently evolving to the Internet of Everything, where Blockchain ensures reliability and security of the IoT systems through its traceability and openness characteristics [7].

The use of the blockchain technology also recently gained interest in the cyber security domain. In particular, the use of smart contracts was incorporated in the design and implementation of a DDoS defence mechanism in [8]. The use of the existing public infrastructure of Ethereum to advertise blacklisted IPs suspected to be involved in ongoing DDoS attacks is fully exploited in this work. The use of timeout notions while creating white/black/grey lists of IP addresses for firewall and corresponding nodes classification is discussed in [9]. Another interesting approach of using blacklists is discussed in [10], in the field of Vehicular Wireless Networks.

The use of Blockchain in a smart home installation of IoT devices, discussed in [11], demonstrates the implementation of the security in smart homes in terms of confidentiality, integrity, and availability with a comparatively low overhead introduction. A secure communication platform in a smart city was presented in [12] as part of a Blockchain based security framework. A cost-effective secure architecture for smarthomes with promising results on the resources consumption and not requiring any cloud storage is presented in [13]. The presented model utilises a hybrid version of the Blockchain, consortium blockchain, where the user's performance as a node has been eliminated. Singh et al. developed an efficient and secure smarthome architecture [14], which is based on the cloud computing and Blockchain technology. This architecture provides a network attack detection and response system, yet it lacks the validation through the real deployment in the smarthomes.

Using the Blockchain technology, which mainly provides trust between nodes, seems to be an effective approach to facilitate the future underlying infrastructure for IoT. Huckle et al. presented IoT scenarios where Blockchain can be used to enable sharing economies of different assets [15]. By using smart IoT devices it is possible to automatically restrict or grant access to assets like vehicles or buildings according to rules implied by smart contracts without the need for any human intervention. Lee and Lee proposed to use the Blockchain to certify IoT devices running on the latest and most secure firmware [16]. This is an interesting approach, that could also employ the creation of an open market between manufacturers, end-users, validators and penetration testers. However, there are limitations with respect to the application of the proper rules in the devices to provide the initiative for all stakeholders to push for more secure firmware installed on deployed devices.

III. Background

GHOST - Safe-Guarding Home IoT Environments with Personalised Real-time Risk Control is an European Union Horizon 2020 Research and Innovation funded project [2], providing the deployment context for the herein presented SC use cases. The GHOST architecture, presented in [1], envisions a singular solution, deployed on a home gateway device, which is capable of monitoring all IoT devices in a smart home environment. It enables endusers (home habitants) to become aware and understand the cybersecurity risks (threats and vulnerabilities), and to take informative decisions affecting their cyber-physical security and privacy. The solution is built up from several layers, including the Data Interception and Inspection layer responsible for aggregating and analysing network traffic; the Contextual Profiling layer that further analyses on the output from the previous layer to establish device behaviour and more specifically report on anomalies; the Risk Assessment layer that gathers all the information about anomalies and correlates them to current and potential risks; finally the Control and Monitoring layer that presents a graphical interface to the end user.

As a cybersecurity solution it is imperative to be able to trust and rely on its execution. Meaning, the software itself and everything in its digital periphery are to be included in the integrity management to ensure a safe and trusted environment. Based on this understanding, along its execution pipeline several key aspects were identified as critical to ensure trust of the system. Blockchain technology has been used in the context of GHOST in various functions, namely integrity management of users and devices, firmware validation of the gateway and IoT devices and enhanced decision making through collective IP blacklisting. The integration of the SCs are part of the Blockchain Defence Infrastructure that interacts with each of the aforementioned GHOST layers.

The GHOST Blockchain Defence Infrastructure consists of a deployed private Ethereum Blockchain on top of an interconnected grid of smart home installations of IoT devices. Each smart home features a device that will act as the smart home gateway middleware and simultaneously as a Blockchain node. Due to their limited processing power, this type of devices act as light nodes, therefore additional, full nodes with adequate processing power are required to act as miners within the network.

IV. Blockchain solution for cybersecurity of IoT

The component described herein is a blockchain based component that can be installed at a smart home gateway and provide enhanced cyber-security functionality for both the administrator of the installation and the smart home owner. The component can be applied in a context where there are one or more IoT service providers and multiple smart home installations. The component offers the following functionalities :

- Registration of Devices and Users and User Consent
- Firmware validation
- Collaborative IP blacklisting

1) Registration of Devices and Users and User Consent: The processes of registration of users and devices along with the management of the consent of users with respect to the service are tightly coupled and are implemented under the scope of the same smart contract. Data stored in the contract are anonymized hashes that can be retrieved by the distributed ledger and cross-checked with local stored information. This separation enables for deployment in public Blockchain environments, since publicly available information on the chain are stripped of description or other sensitive information. This approach simplifies registration of gateways and devices as well as permission handling.

The user access control is performed solely by the Smart Contract (SC). Three types of users are allowed:

- **Owner:** The Owner of the contract is the entity that deployed the SC on the Blockchain or gained the role by a transfer of ownership from a previous owner. The Owner is a superuser that can add or remove Administrators and Users, add or remove Gateways and IoT devices and add or remove certified software hashes. The Owner of the contract cannot be an owner of a Gateway. The user type for the Owner is 1.
- **Administrators:** The Administrator is an entity that has the same rights as the Owner, except that Administrators cannot add or remove other Administrators. The user

type for the Administrators is 1, the same as the Owner. It should be noted that the Owner is also considered an Administrator. Administrators are not allowed to be owners of a Gateways, since this would be controversial to their role.

- **User:** The User is an entity that can sign or unsign the Consent related to the ownership of a Gateway. Users have no other rights. The user type for the Users is 10.

The SC retains Gateways as an array of structures inside the contract. Each Gateway structure retains its corresponding IoT devices. Each Gateway structure is composed of the following members:

- **Gateway ID**: This can be any unique string or a hashed string derived from a string describing the Gateway uniquely, i.e: its MAC address (or the hash of the MAC address). Derivation is performed using the SHA3 (Keccak256) hashing method. Uniqueness of the ID is checked during addition of Gateways. The Gateway ID is a mandatory input for registering a Gateway to the smart contract. Since MAC addresses are not considered as sensitive data, it can be used as Gateway IDs in plain text.
- **Gateway status**: The status of the Gateway is described by an unsigned integer that can be assigned the values 0 or 1 corresponding to deleted or active.
- **Last modification**: The last modification is the timestamp of the last action performed on the members or properties of a Gateway. In practice the timestamp coincides with the timestamp of the block that included the transaction that modified the Gateway.
- **Owner (User)**: The address of the owner of the Gateway is retained in the Gateway structures. The address of the user owning the Gateway is required in order to link the Gateway to the user, as well as allow for restricting the signing / unsigning of the consent corresponding to that Gateway to its owner. The address of the owner is a mandatory input for registering a Gateway to the smart contract.
- **Consent**: The consent is expressed by the value of an unsigned integer. Consent can be 0 or 1.
- **IoT devices list**: The IoT devices are identified as unique strings (IDs) describing these devices (i.e. MAC address or the hash of the MAC address). These hashes are stored in an array.
- **Number of IoT devices**: A variable retaining the number of registered IoT active / inactive devices.

Following the same approach with the Gateways, the SC retains an array of IoT structures. Each IoT structure is composed of the following members:

- **IoT device**: This can be any unique string or a hashed string derived from a string describing the IoT uniquely, i.e: its MAC address (or the hash of the MAC address). Derivation is performed using the SHA3 (Keccak256) hashing method. Uniqueness of the ID is checked during addition of IoTs. The IoT ID is a mandatory input for registering a Gateway to the smart contract. Since MAC addresses are not considered as sensitive data, it can be used as Gateway IDs in plain text.
- **IoT status**: The status of the IoT is described by an unsigned integer that can be assigned the values 0 or 1 corresponding to deleted or active.
- **Last modification**: The last modification is the timestamp of the last action performed on the members or properties of an IoT. In practice the timestamp coincides with the timestamp of the block that included the transaction that modified the IoT.
- **IoT type**: The IoT type is expressed by the value of a hex string representing the type of an IoT device (i.e Motion Sensor, etc).
- **IoT uuid**: The IoT uuid are identified as the hashed value of the universal unique identifier with which system Administrators add the device in the local Gateway database.

The uniqueness of a Gateway, before addition to the array of Gateways in the smart contract, is enforced by a mapping (injection) G from Gateway ID to an unsigned integer $G : ID \rightarrow J$. This mapping is also used for random access to the array. The default value of the integer corresponding to a registered gateway belongs to $[1, N_{GW}] \subset \mathbb{N}$, with N_{GW} denoting the total number of Gateways, while if a Gateway is not already registered the value J is zero. Random access to the elements of the $Gateways$ array can be performed as $Gateways[G(ID)-1] = Gateways[J-1]$, for matrices with zero based indexing. Thus, addition of a Gateway requires $G(ID) = J = 0$. Each new Gateway is added in the end of the $Gateways$ array in position N_{GW} and its corresponding value in the mapping is set to $N_{GW} + 1$. In order to remove a Gateway with Gateway ID (ID) the mapping is set to $G(ID) = 0$ and the status of the Gateway is set to 0. This removal procedure does not involve moving or copying structures, thus enabling reduced computational complexity as well as cost for public blockchain deployments. To further improve performance, a variable denoting the number of registered Gateways N^r_{GW} (numOfGWs) is retained in the contract. This variable increases by 1 when a new Gateway is added and decreased by 1 on deletion. Retaining the number of registered Gateways in the contract, results in avoiding searching operations required by get type functions, thus reducing cost of deployment of the contract. It should be noted that deletion of a Gateway results in deletion of all its constituent IoT devices.

In the case of IoT devices, their registration follows the same procedure. Similarly, a mapping from IoT ID to an unsigned integer $I : ID \rightarrow J$. This mapping is also used for random access to the array of IoT structures. Random access to the elements of the $IoTs$ array can be performed as $IoTs[I(ID) - 1] = IoTs[J - 1]$. Addition can be performed only if $I(ID) = 0$. The deletion of a device is performed by setting the status of the IoT device to 0 and $I(ID) = J = 0$. In order to improve performance, each Gateway retains the number of registered IoT devices N^r_{IoT}, avoiding search operations during call of get type functions, thus reducing cost of deployment of the contract.

The registration / deletion of Gateways and IoT devices can be performed only by the Administrators.

The retrieval of information, concerning Gateways and IoT devices, from the SC can be performed with get type functions. These functions return info for all Gateways, info for a Gateway with a specific UID or info for all IoT devices of a Gateway with a specific UID.

The main functions that have been realised in the SC with regards to this functionality are:

- **Registration of Users and Administrators:** The Administrators are responsible for the registration of Users. The Administrators can be registered by the Owner of the SC. The Owner can also register Users. The contract retains a common registry to ensure uniqueness of Users and Administrators.
- **Registration of Gateways:** The Administrators, are able to register the Gateways and each Gateway is assigned to one registered User corresponding to its smart home. Each Gateway is described by a unique identifier, that can be derived, for example, by the MAC address or combination of other characteristics of the Gateway device. A User is identified by the Address on the blockchain. Moreover, administrators or the Owner of the smart contract can remove Gateways from the distributed ledger. After the registration of a Gateway, the User should sign the consent related to this Gateway. The contract retains a common registry to ensure uniqueness of Gateways.
- **Registration of IoT devices:** The registration of IoT devices is performed by the Administrators or the Owner of the contract. Each IoT is characterised by a unique id which can be derived its MAC address or combination of characteristics. The contract retains a common registry to ensure uniqueness of IoT devices across all Gateways.
- **Signing / Unsigning of consent form:** A User that owns a Gateway is able to sign the consent corresponding to its registered Gateway. The User side consent can be removed at any moment for a given Gateway.
- **Change of ownership of the contract:** The contract enables for change of ownership. The current Onwer can pass ownership of the contract to another blockchain address.

A visual representation of the main flows is depicted in Fig. 1.

2) Firmware Integrity: The second main functionality offered by the system is the validation of firmware of gateways and devices. The main concept is based on the fact that a significant number of cyber-security attacks relate to modifying one or more of the existing files of a system. The integrity of the firmware/software of each Gateway (and potentially of IoT devices that support that) is periodically checked against the SC. The administrator of the installation stores in the SC the hashes of all valid firmware/software versions released. A service that constantly runs in the Gateway calculates the hash of a predefined part of the filesystem and checks it against the set of hashes of the valid firmware versions stores in the SC. The component reports both the event of an invalid hash and the event of an interruption of the execution of the service (which could potentially be part of an attack).

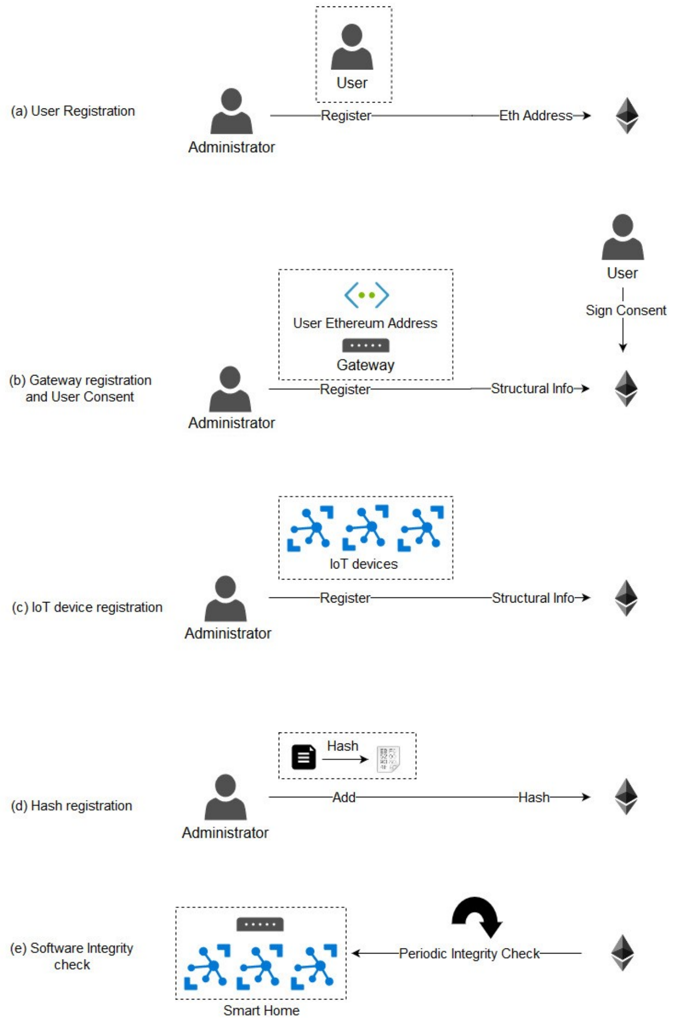

Fig. 1. An example flow of operations: (a) User registration, (b) Gateway registration from Administrator and signing of consent by the owning user, (c) Registration of constituent IoT devices, (d) Registration of hashes related to firmware, (e) Periodic check of software integrity of the Gateway and IoT devices.

The contract allows for registering hashes of firmware (or software) allowed for installation on the registered devices of the IoT system. These hashes are computed using a hashing scheme such as: IMO, $MD5$, $SHA224$, $SHA256$, $SHA384$ or $SHA512$ and are stored to the Blockchain through a mapping $H : (HASH) \rightarrow B$, with $HASH$ denoting the computed hash after rehashing with the $SHA3$ method and B a Boolean value denoting existence. The computed hashes are rehashed in order to ensure constant size of the hash that would be given as input to the mapping. The integrity of the installed firmware can be checked frequently by the Gateway or other devices against hashes stored on the Blockchain.

The registration or removal of hashes can be performed by the Administrators. Moreover, the data structure (mapping) retaining all hashes is not iterable and can be accessed only randomly with knowledge of the hashes.

3) Blocking Blacklisted IP Addresses: Due to the distributed structure of IoT systems it is common to have information (through which knowledge can be produced) generated in different nodes of a network. Combining such knowledge can be beneficial for the whole system, given that the integrity of the combined data can be preserved. The functionality of the system proposed is related to the combination of the information about problematic IP addresses between different smart home installations. Two different types of IPs blacklisting are envisaged public and private. The first one refers to the functionality of a collaboratively maintained knowledge base for public IPs reputation. Each smart home gateway report IPs as malicious to a single smart contract while the same contract can be queried for a reputation score for a specific IP. The reported reputation score is cumulatively calculated upon the reports of all gateways. The second type refers to a private blacklisting / whitelisting functionality where each gateway has its own smart contract to which it backups to and restores from its locally maintained black and white lists. This feature will enable gateways to double check the integrity of their local lists.

a) Public Blacklisting: For the public blacklisting functionality, a shared and publicly available knowledge of potentially malicious IP addresses is maintained in a smart contract. The contract will contain a list of records, each one corresponding to the event that an installation (of the component) has reported an IP as malicious. These records cannot be maliciously altered or deleted, as such an action would need to alter data already stored in the blockchain, which is extremely hard.

The calculation of bad reputation score for each external IP, is mainly based upon three factors:

- the number recent reports related to the specific IP
- the cardinality of distinct report submitters for those records
- the time elapsed since each on of those reports

The main concept for the calculation approach is that, when reports for a specific IP start appearing, then its bad reputation score starts increasing. The more such reports come by the larger the score becomes. If those originate from the same source then each subsequent reports has a reduced effect to the final score. This practically protects the system from malicious users that would like to intentionally increase the bad reputation score for a specific IP (given that they control a single or a few identities in the system). Each reports effect is gradually limited as time goes by. In order for the score of an IP to remain high it is required to continuously have new reports for the specific IP from varying sources. This feature enables the gradual restoration of the reputation of the IP of a legitimate host, that a malicious user has temporarily taken control of.

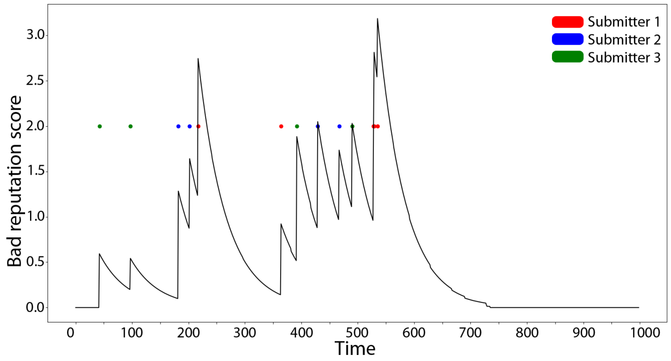

Fig. 2. Reputation score for a specific IP

The formula that calculates a bad reputation score for each IP is depicted in Equation 2. The approach of dividing the time in discrete time frames or steps has been applied to implement a scheme that takes into account more recent values with a higher weight. The score is calculated for a specific time period, a specific length of time steps denoted as t_p. If the current time step it t_n, then the score is:

$$score = \sum_{t=t_n-t_p}^{t=t_n} -\ln{(cf)} sr_t (\lambda)^{t_n - t} \tag{1}$$

The sr_t is equal to 1, if there is a record for the IP in time step t and equal to 0 otherwise. The summation does not accumulate values for the time steps at which no record exists for the specific IP.

The λ factor is a decay parameter that takes values in the range $(0, 1)$. The higher the value of λ is the strongest the memory of the scheme is. Lower λ values mean that the scoring scheme penalises old values in a more heavy way.

Finally, the $cf \in [0, 1]$ parameter stands for the cardinality factor and penalises the case where all records come from the same submitting address. It is equal to the ratio of the nember of distinct submitters to the total number of records for the specific IP that have been reported in the time window $[t_n - t_p, t_n]$.

$$cf = \frac{submitters}{records} \tag{2}$$

This parameter practically protects the reputation of IP from spamming accounts, that would want to harm the owner of an IP by repeatedly submitting blacklisting records for this IP.

Figure 2 shows the calculated reputation score for a particular malicious IP, given that the relevant reports for this IP are shown by the coloured dots. The colour of each dot represents the unique id of the submitter of the report. For instance, all red dots are representing submissions of Submitter 1, all blue dots come from Submitter 2 etc.

It is evident in the graph that the scoring scheme values reports according to how recent they are. This is why the score starts to decrease with time after a specific report, at least until a new report is submitted. The rate at which the scheme phases out the past reports is dictated by the decay parameter λ.

Additionally, not all reports contribute the same value to the total reputation score for a particular IP. If a submitter keeps sending reports for the same IP, then every new report is weighted less. This is evident in the case of the first two reports by Submitter 3, i.e. the first two green dots, or in the case of the first two submissions by Submitter 2, i.e. the first two blue dots approximately at $t = 200$. In contrast, when Submitter 2 keeps quiet for a period of time, his records start again to be valued more, at $t = 470$.

b) Private Black/Whitelisting: The private blacklisting and whitelisting of the IP addresses is a variation of the public blacklisting, where the reports for IP addresses have influence only on a per installation basis. Despite any public recommendation (i.e. Public blacklisting), a user still can have personalised settings and a set of rules. Each smart home gateway is associated with a smart contract where a private list of rules is recorded. Each rule in turn is encrypted together with a state indicating to which list it belongs (i.e. blacklist, whitelist or none for the purpose of resetting the state).

Practically the gateway can maintain a copy for each one of the its private whitelist and blackclist into a smart contract. It can then either add an IP, remove an IP or check if it exists in the smart contract. Through this mechanism the gateway can have an integrity guarantee for the IPs existing in its local lists.

V. Discussion and Conclusions

This paper proposed a decentralized approach for supporting IoT security in smart homes through blockchain technology. The mechanisms developed support a variety of functions for registering users and functional elements of a smart home, and more security oriented tools for ensuring the integrity of software and firmware installed, as well as dynamic and immutable IP blacklisting procedures.

The whole framework has been deployed and evaluated in real smart homes, under the purposes of the GHOST EU project, enhancing the cybersecurity defence mechanisms in a decentralized manner under a private blockchain network.

Acknowledgment

This work is partially funded by the European Union's Horizon 2020 Research and Innovation Programme through GHOST project (https://www.ghost-iot.eu/) under Grant Agreement No. 740923.

References

[1] J. Augusto-Gonzalez, A. Collen, S. Evangelatos, M. Anagnostopoulos, G. Spathoulas, K. M. Giannoutakis, K. Votis, D. Tzovaras, B. Genge, E. Gelenbe, and N. A. Nijdam, "From internet of threats to internet of things: A cyber security architecture for smart homes," in *2019 IEEE 24th International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD)*, Sep. 2019, pp. 1–6.

[2] A. Collen, N. A. Nijdam, J. Augusto-Gonzalez, S. K. Katsikas, K. M. Giannoutakis, G. Spathoulas, E. Gelenbe, K. Votis, D. Tzovaras, N. Ghavami, M. Volkamer, P. Haller, A. Sánchez, and M. Dimas, "Ghost - safe-guarding home iot environments with personalised real-time risk control," in *Security in Computer and Information Sciences*, E. Gelenbe, P. Campegiani, T. Czachórski, S. K. Katsikas, I. Komnios, L. Romano, and D. Tzovaras, Eds. Cham: Springer International Publishing, 2018, pp. 68–78.

[3] C. S. Kouzinopoulos, G. Spathoulas, K. M. Giannoutakis, K. Votis, P. Pandey, D. Tzovaras, S. K. Katsikas, A. Collen, and N. A. Nijdam, "Using blockchains to strengthen the security of internet of things," in *Security in Computer and Information Sciences*, E. Gelenbe, P. Campegiani, T. Czachórski, S. K. Katsikas, I. Komnios, L. Romano, and D. Tzovaras, Eds. Cham: Springer International Publishing, 2018, pp. 90–100.

[4] M. Conoscenti, A. Vetrò, and J. C. D. Martin, "Blockchain for the internet of things: A systematic literature review," in *2016 IEEE/ACS 13th International Conference of Computer Systems and Applications (AICCSA)*, Nov 2016, pp. 1–6.

[5] M. A. Khan and K. Salah, "Iot security: Review, blockchain solutions, and open challenges," *Future Generation Computer Systems*, vol. 82, pp. 395–411, 2018.

[6] Q. Wang, X. Zhu, Y. Ni, L. Gu, and H. Zhu, "Blockchain for the iot and industrial iot: A review," *Internet of Things*, vol. 10, p. 100081, 2020.

[7] L. Wei, J. Wu, C. Long, and Y. Lin, "The convergence of ioe and blockchain: Security challenges," *IT Professional*, vol. 21, no. 5, pp. 26–32, 2019.

[8] B. Rodrigues, T. Bocek, A. Lareida, D. Hausheer, S. Rafati, and B. Stiller, "A blockchain-based architecture for collaborative DDoS mitigation with smart contracts," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, ser. Lecture Notes in Computer Science, D. Tuncer, R. Koch, R. Badonnel, and B. Stiller, Eds., vol. 10356 LNCS. Cham: Springer International Publishing, 2017, pp. 16–29.

[9] M. Steichen, S. Hommes, and R. State, "ChainGuard — A firewall for blockchain applications using SDN with OpenFlow," in *2017 Principles, Systems and Applications of IP Telecommunications (IPTComm)*. IEEE, sep 2017, pp. 1–8.

[10] J. Tobin, C. Thorpe, and L. Murphy, "An Approach to Mitigate Black Hole Attacks on Vehicular Wireless Networks," in *2017 IEEE 85th Vehicular Technology Conference (VTC Spring)*, vol. 2017-June. IEEE, jun 2017, pp. 1–7.

[11] A. Dorri, S. S. Kanhere, R. Jurdak, and P. Gauravaram, "Blockchain for iot security and privacy: The case study of a smart home," in *2017 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*, March 2017, pp. 618–623.

[12] K. Biswas and V. Muthukkumarasamy, "Securing smart cities using blockchain technology," in *18th International Conference on High Performance Computing and Communications; 14th International Conference on Smart City; 2nd International Conference on Data Science and Systems (HPCC/SmartCity/DSS)*. IEEE, 2016, pp. 1392–1393.

[13] S. Arif, M. A. Khan, S. U. Rehman, M. A. Kabir, and M. Imran, "Investigating smart home security: Is blockchain the answer?" *IEEE Access*, vol. 8, pp. 117 802–117 816, 2020.

[14] S. Singh, I.-H. Ra, W. Meng, M. Kaur, and G. H. Cho, "Sh-blockcc: A secure and efficient internet of things smart home architecture based on cloud computing and blockchain technology," *International Journal of Distributed Sensor Networks*, vol. 15, no. 4, p. 1550147719844159, 2019.

[15] S. Huckle, R. Bhattacharya, M. White, and N. Beloff, "Internet of Things, Blockchain and Shared Economy Applications," *Procedia Computer Science*, vol. 98, pp. 461–466, jan 2016.

[16] B. Lee and J. H. Lee, "Blockchain-based secure firmware update for embedded devices in an Internet of Things environment," *Journal of Supercomputing*, vol. 73, no. 3, pp. 1152–1167, mar 2017.