Trust Zone Formation for Building Automation Networks using Building Information Modeling

Arne Wall, Björn Butzin, Dirk Timmermann
University of Rostock
Institute of Applied Microelectronics and Computer Engineering
18051 Rostock, Germany, Tel.: +49 381 498-7271
Email: arne.wall@uni-rostock.de

2020 IEEE Global Conference on Artificial Intelligence and Internet of Things (GCAIoT) | 978-1-7281-8420-3/20/$31.00 ©2020 IEEE | DOI: 10.1109/GCAIOT51063.2020.9345857

***Abstract*—Modern Building Automation Systems (BAS) consist of sensors and actuators that are connected via an IP-based network and offer their functionality via RESTful APIs. Because a single device can be exploited by an attacker to perform attacks within the local network, we put devices into isolated groups. These groups are isolated MAC-layer Trust Zones to reduce the attack surface in contrast to a BAS with fully connected devices. We propose an algorithm that leverages the so far neglected potential of Building Information Modeling (BIM) to compute Trust Zones. We assure unimpaired operation of all applications while limiting the number of infrastructure devices. The proposed mechanisms are demonstrated considering sensors and actuators that are connected via wired Ethernet and the IEEE 802.11s WLAN mesh standard. At the application layer we make exemplary use of the Constrained Application Protocol (CoAP). Finally, we experimentally evaluate the device acquisition and selection based on our network partitioning algorithm.**

I. Introduction

During the evolution of building automation Systems (BAS) hard wired sensors and actuators became replaced by embedded devices that are connected via a network to a central control system. Through the usage of the IEEE 802.11s WLAN mesh standard, large areas can be covered without installing additional WLAN repeaters or access points. WLAN mesh peers act as infrastructure nodes by routing frames at the MAC layer. Loosely coupled devices that have network connectivity within the local area network can be linked to each other to create applications. E.g., sensors and actuators form a closed control loop for heating and ventilation. Modern BAS often provide Internet accessibility by various interfaces so that users, operators and technicians can interact with the system. The isolation of the internal network of the building was seen as a countermeasure against security threats from attacker operating from the Internet. In case a single device within the BAS has a vulnerability, it can be used to run attacks against other devices within the local network. Next to Virtual Private Networks (VPN), MAC-layer separation is an option to isolate devices locally [1]. So far, there is no algorithm to create and distribute MAC-layer groups of devices that trust each other (Trust Zones), leveraging the potential of using devices' meta data in combination with Building Information Modeling (BIM). The firmware version, the application class (door locking system, lighting, ventilation) and the installation location (public accessibility) of an end device (ED) are part of the ground truth so that a security controller (SC) performs a partitioning algorithm to place every ED into a Trust Zone. In this paper we explain in section II the openBIM standard and its valuable information. The characteristics of a modern BAS and the security threats that we defend against are presented in section III. Subsequently, we explain our approach how devices are bootstrapped into a configuration network, how key material is negotiated between SC and EDs and the ground truth is verified in section IV. Furthermore, we demonstrate the creation of applications, including their formal verification regarding logical errors. We explain our novel approach how BIM can be used to calculate Trust Zones for MAC-layer isolation of traffic and devices. Then, we show in section V the low implementation effort that vendors have to raise during development of an ED. We also evaluate a prototypical implementation of our location-based Trust Zone formation algorithm. Our network partitioning approach is compared against other solutions in section VI. Finally, in section VII we draw our conclusions.

The main contributions of our BIM-based network partitioning algorithm are:

- Creation of a ground truth data base including verified installation locations based on BIM
- Policy-driven computation of Trust Zones
- Example policies that can be extended depending on requirements of the BAS operator
- Assuring network connectivity within IEEE 802.11s WLAN mesh networks through the utilization of valuable information provided by BIM
- Guaranteeing the absence of logical flaws within decentrally executed application logic
- Deriving a small set of features to be implemented by device vendors
- Evaluating the execution time for the SC during network partitioning

II. Building Information Modeling

During the life-cycle of a building, the *Building Information Modeling* (BIM) is used to model buildings digitally. Next to 3D models it consists of valuable information of material properties and the relationships of buildings. All information can be exchanged in an unified format overarching all phases from planning to destruction. There are two options to use BIM: One the one hand it is possible to use prorietary tools and data formats, on the other hand there is the openBIM approach that relies on open standards, which enables to enhance the interoperability between participant. Due to these benefits, we make use of the open [1] *Industry Foundation Classes* (IFC) data model, that is standardized as ISO 16739. The first IFC data model versions already provided devices information. Starting with version 4 of IFC in 2013 innovative use cases could

[1] https://technical.buildingsmart.org/standards/ifc/ifc-schema-specifications/

develop due to a rich set of metadata of the network and devices. Specific data schemes like the "IfcBuildingControlsDomain", "IfcElectricalDomain", and "IfcHvacDomain" consist of modeling classes for actuators, sensors and controller. Furthermore, they describe circuits for *Heating Ventilation and Air-Conditioning* (HVAC) including control panels for human-machine interaction. An machine readable format for data links between devices exists. It contains information of network infrastructure (like switches and routers), cabling and end devices. Currently, the full potential of using these information is not elaborated yet. Therefore, we developed an approach to create a security planning algorithm that uses knowledge of, amongst others, devices, network infrastructure, cabling, physical accessibility, installation location and logical links between devices. The IFC model also supports time as a dimension to track changes in the system (building, network infrastructure and devices). E.g., it can be modeled that devices are installed and removed. In [2] we present a survey on many other projects that also model device related data. Due to the fact that these models are not compatible with each other nor standardized they are out of our scope of investigation. Nevertheless, they can describe additional modeling details (especially networked devices, IT and security relevant properties) in the future. IFC and other approaches are typically, represented by ontotogies. If the data of the models is automatically reasoned, new knowledge can be derived.

III. Building Automation System Architecture

The first generation of Building Automation Systems consists of sensor and actuator devices that are connected to a control unit. Every device is connected via a dedicated wired link to the control unit that reads analog sensors and controls actuators via power supply. The next generation of BAS is characterized by a separate power supply and proprietary data connections between control unit and devices. Current BAS introduce IP-based communication through open web standards for embedded devices to increase interoperability between control units and sensors/actuators. Next to the widespread use of BACnet IP [3] there are service-oriented architectures based on DPWS (Devices Profile for Web Services) or RESTful API based interfaces using HTTP or CoAP [4]. The installation process of devices becomes easier because only a power supply needs to be provided. The control unit will read sensors and control actuators via an open web protocol standard. Especially, when web services are offered within a wireless network there is a decreased installation effort. Because a control unit is a single point of failure, systems can be more robust by using a decentralized control structure. In [5] such an approach is proposed. Every device in the network may offer a so-called "Configuration Service" to receive user defined if-this-then-that (ITTT) rules. These ITTT rules describe the control logic that a central control unit would have executed to read and evaluate sensor data to control actuators. Our proposed algorithm is based on a BAS network that consists of sensors and actuators that offer their funcionalities via RESTful APIs. These devices are connected via Ethernet or IEEE 802.11s WLAN mesh [6] to achieve wired as well as wireless connectivity with a large range. Due to the fact that WLAN mesh devices route frames at the MAC-Layer, they are considered as network infrastructure. In figure 1 of [7] we proposed a security concept that introduces a central security controller with a global knowledge base of the network infrastructure, application and the BIM to group devices into so-called Trust Zones. These Trust Zones are isolated MAC-layer

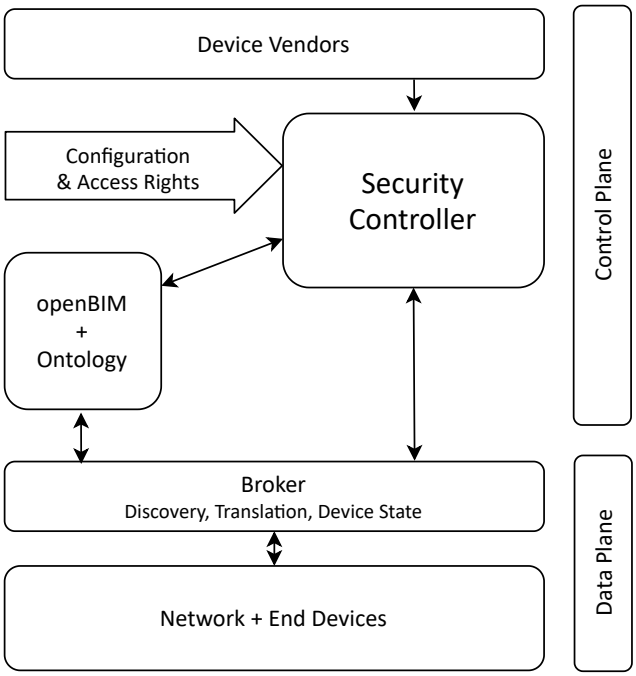

Fig. 1. Underlying Security Architecture [7]

secured networks containing devices using the same pre-shared key (PSK). The PSK is sent by the SC to each End Device (ED) individually. So far, we described the overall architecture in [7] and how devices/applications are separated on MAC-layer [1], [8]. So far, there is no algorithm that determines the association of each ED to an isolated MAC-layer Trust Zone. In this paper, we present our algorithm to assign every ED into a Trust Zone while using the BIM and user verified ED metadata as ground truth. We introduce a heuristic approach to calculate possible network configurations and eliminate solutions that violate security policies. Furthermore, we define a format of universal security policies that are applied within our security risk analysis. Using the BIM as part of the ground truth, we can make use of the installation location of each ED. Thus, we can consider security risks which are induced by location aspects. If an ED is installed in a publicly accessible location and it is connected to an ED with a high criticality like a door locking system, the system can be exploited by an attacker having physical access (e.g., JTAG interface, power supply or serial interface).

IV. Concept

Our concept consists of five phases that are listed in table I. First, a technician bootstraps EDs to a configuration network, then our SC collects ED's metadata and network information to map applications of sensors and actuators to EDs and put them into isolated MAC-layer Trust Zones following universal security policies. Table I gives an overview of the phases. In the following sections, all phases are explained in detail.

A. Phase 1 - Commissioning of new ED

During the first phase the SC executes our algorithm shown in figure 3. The message flow between the SC and the ED with a user as broker for bootstrapping the ED to the network is visualized by figure 2. Only authorized personnel

TABLE I. PHASES OF THE NETWORK AND APPLICATION PARTITIONING ALGORITHM

Phase	Meaning
1) Commissioning of new ED	A device will be connected to a configuration network to be able to receive data from the SC. The message flow between ED and SC with an user as broker is visualized in figure 2. During the first phase key material to create a public key infrastructure with the SC as the root node is negotiated. EDs metadata and the installation location become part of the ground truth and global network knowledge. This sequence (figure 3) is executed until all devices are commissioned.
2) Information Gathering of Network Plane	During this phase the SC collects information of how devices are connected. These information contain the link type (wired Ethernet, WLAN) and the communication peers. Ethernet devices links are assigned to a switch or router with additional information of the physical accessibility within the building. Peer links of all WLAN mesh devices are gathered and stored into a database. The algorithm is depicted in figure 5.
3) Creation and Verification of Application Plane	During the installation phase of EDs applications were defined following ITTT rules. E.g., temperature sensors, window controls, ventilation and air conditioners (actuators) were linked together in form of a graph. The absence of logical errors between graphs, that control the same actuators in a different manner, are detected using BDDs in this phase.
4) Application Mapping on Devices	During this phase, all nodes of the application graph are mapped to EDs. The required data of device type and installation location are taken from the ground truth database.
5) Policy-based Formation of Trust Zones	EDs of the same application are put into a Trust Zone (figure 4). If EDs of an application graph lack of network connectivity, possible infrastructure nodes (WLAN mesh peers) are selected depending on their physical installation location. During a second mesh peer selection, only devices that fulfill all security policies are selected. There are multiple policies: strict separation of application domains (e.g.: door locking and lighting domain), isolating independent application graphs and isolating devices of the same physical accessibility.

is allowed to bootstrap devices. Therefore, technicians have to authenticate themselves using a smart phone which is later used to act as a broker for an authenticated Diffie-Hellman key negotiation between SC and ED following the implementation of [9]. After that process key material is obtained to assure end-to-end encryption e.g., at transport layer (TLS 1.3 or DTLS 1.3) or application layer (OSCORE). Furthermore, the ED receives credentials via smart phone for accessing a configuration network (blue edges in figure 6). During the next steps the SC requests EDs' metadata via RESTful API (see table II) using previously exchanged key material (more precise: symmetric 256-bit key as pre-shared key) for CoAP over DTLS or OSCORE over UDP. The metadata contains information of e.g., device type, vendor, firmware version and supported protocol stacks. Additionally, the technician sets the installation location manually after the BIM naming scheme. ED metadata and the installation location become part of the

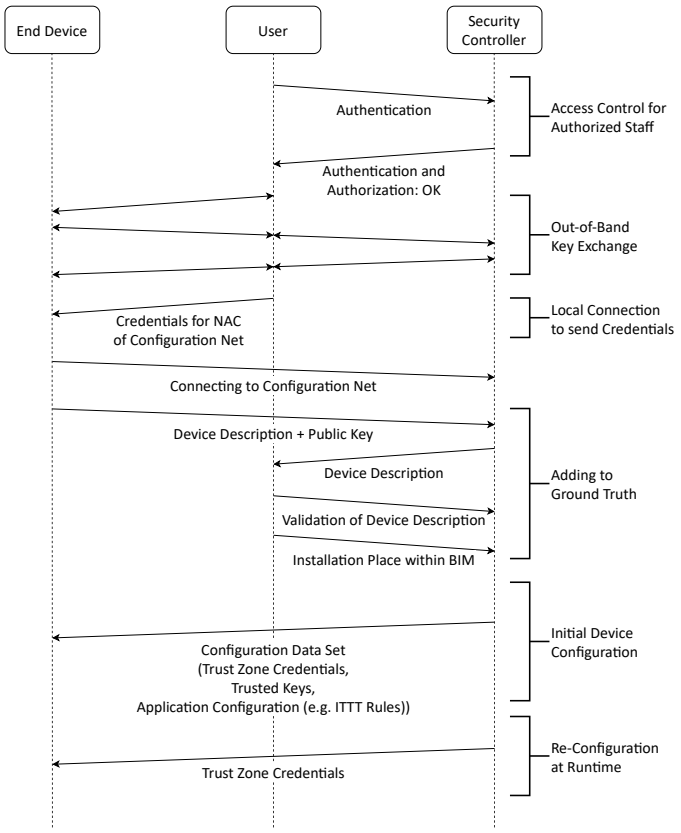

Fig. 2. Process of Single Device Commissioning in Phase 1

global knowledge base or ground truth of the SC. When all EDs are bootstrapped to the configuration network and their meta data and installation location are part of the ground truth, the network partitioning including application setup will be executed.

B. Phase 2 - Information Gathering of Network Plane

To create MAC-layer partitions, aka. Trust Zones, the SC collects information of the network plane. Within the WLAN mesh-based configuration network, we make use of SNMP (Simple Network Management Protocol) to collect status information of every ED. There is an SNMP server on each ED to offer data in form of status objects. These objects contain a list of all peer links. Because our network according to the IEEE 802.11s mesh standard is based on a distance vector routing protocol, every ED stores a routing table containing the next hop MAC address for every destination. When all routing tables are combined by the SC, a network graph can be derived. The same principle was proven by us in [10] for a different application where next to the network graph additional distance metrics were collected. The second group of devices are connected via wired Ethernet. There are so-called Mesh Portals in our network infrastructure that realize connectivity between switched Ethernet and WLAN mesh (see figure 6). The network topology including the physical location of switches and wires is defined during commissioning phase. If an application consists of WLAN mesh and Ethernet EDs, a Mesh Portal has to be in reach to forward frames.

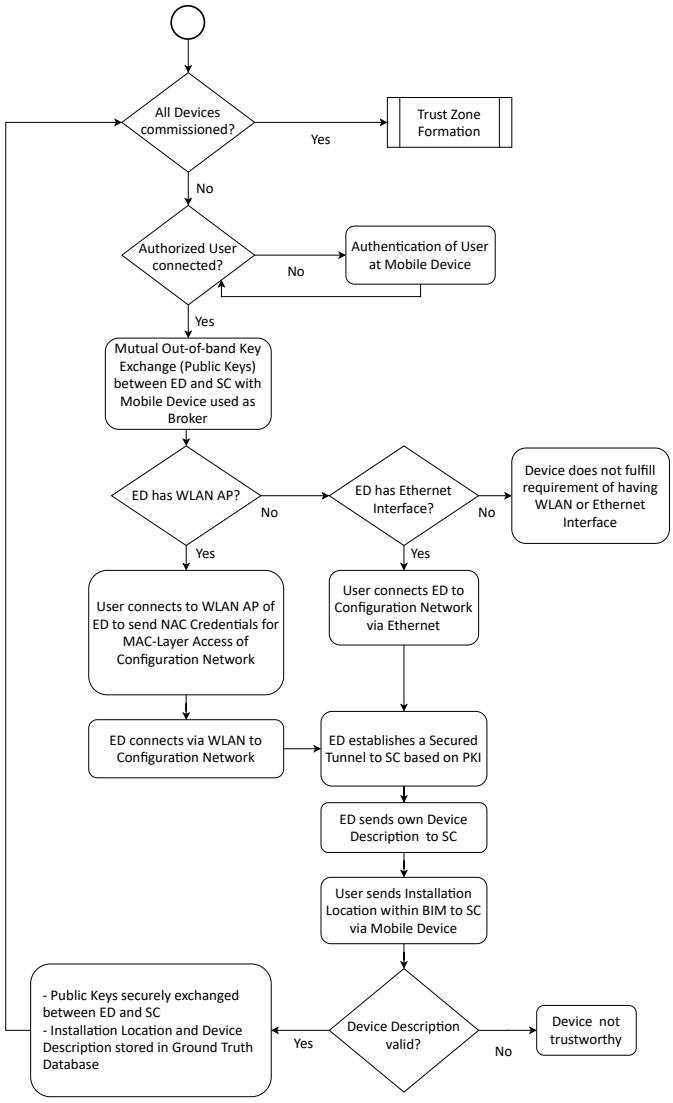

Fig. 3. Commissioning executed by SC in Phase 1

C. Phase 3 - Creation and Verification of Application Plane

During the next phase, logical links between devices are planned (figure 4). The administrator connects sensors and actuators, which results in multiple directed a-cyclic graphs. To every application graph an ITTT rule is assigned (e.g.: if desired indoor temperature is smaller than the actual indoor temperature, then control air conditioner and air flow). It is possible that more than one rule controls the same actuator. Therefore, we verify the absence of logical errors by modeling the rules in form of binary decision trees (BDD). The nodes of the BDD represent the terms of the rules (3-tuple consisting of [sensor value],[comparison operator],[threshold]). If the sequence of variables of all BDDs is consistent, then two BDDs can be XOR combined to search for an input vector that triggers different output behavior. During that verification phase the administrator is assisted by our algorithm to identify logical flaws, which could result in an unwanted behavior (safety requirements).

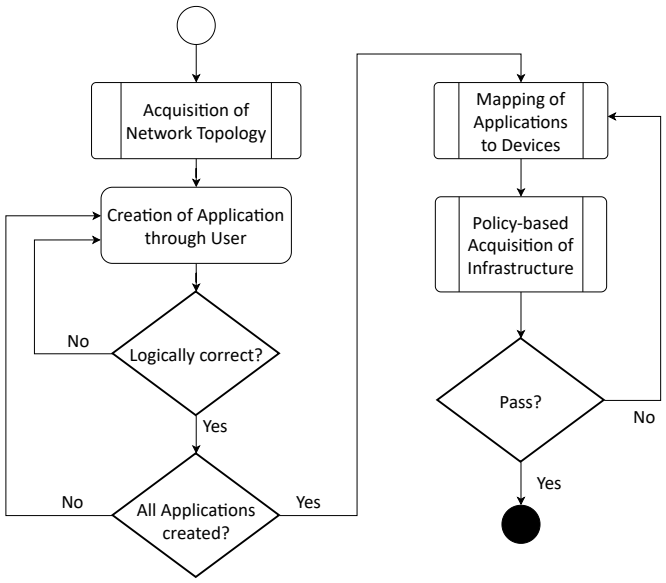

Fig. 4. Formation of Trust Zones in Phase 5

D. Phase 4 - Application Mapping on Devices

After all application graphs were formally verified, the mapping sequence begins (figure 4 "Mapping of Applications to Devices"). Because the SC has global network knowledge including self-descriptions (e.g.: via RESTful API exposed by a CoAP or http server) and installation locations, sensors and actuators can be mapped to physical devices. The resulting graph of connected devices has a single hop from sensor(s) to actuator or a two-hop distance from sensor(s) via an actuator to multiple/single actuator(s).

E. Phase 5 - Policy-based Formation of Trust Zones

Applications that do not share the same physical devices will be put into separate Trust Zones (figure 4 "Policy-based Acquisition of Infrastructure"). The application graphs are independent of the underlying network infrastructure. Next to wired Ethernet links our system makes use of the IEEE 802.11s WLAN mesh specification. In case a hop in the application graph cannot be realized using a single hop at the MAC layer, additional peers have to be put into the Trust Zone to assure connectivity. WLAN mesh devices that have the same firmware and physical accessibility like the devices of the application graph are candidates to serve as infrastructure peers (figure 5 "Filter EDs after Policy"). Because at this stage the Trust Zone is only virtually planned within the SC, there is no real connectivity test (application layer ping) available. Due to the fact that the SC has knowledge of WLAN channel characteristics and the BIM including physical characteristics of the building (signal attenuation caused by absorption of walls) the algorithm can simulate the individual visibility of all application nodes. The algorithm to estimate MAC-layer connectivity makes use of WLAN settings that enable maximum peer link lengths. Applications that get sensor values to control other actuators are able to operate at the modulation and coding scheme 0 (MCS 0) according to the IEEE 802.11n/ac standard. The corresponding data rate of 6.5 Mbit/s fulfills the requirements of such an application scenario. Based on the real-world experimental outcome of [11] and using the ITU attenuation model (4), the maximum peer link distance

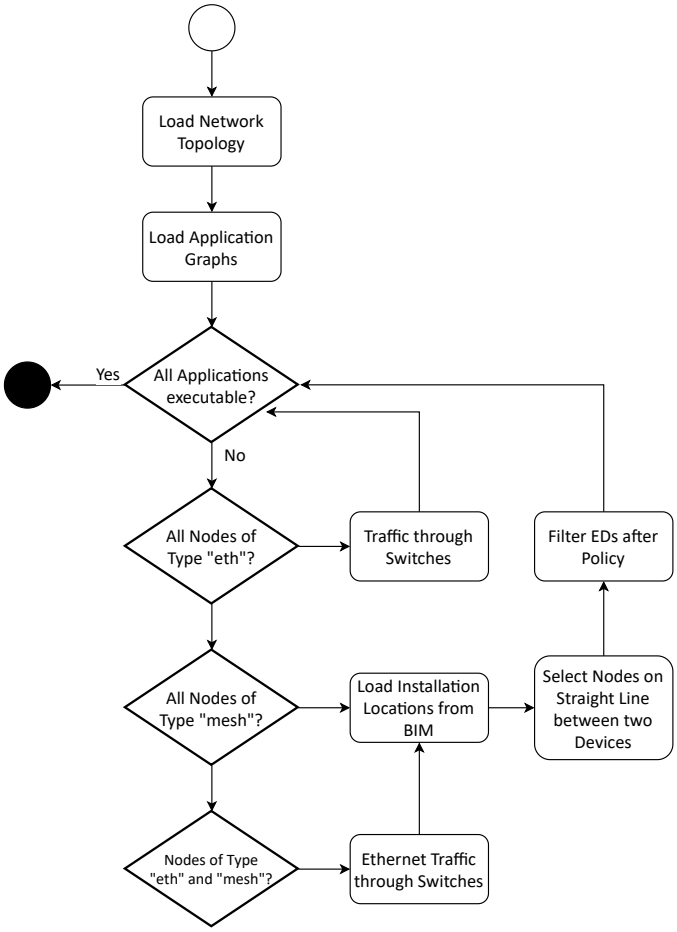

Fig. 5. Policy-based extension of Trust Zones performed in Phase 5

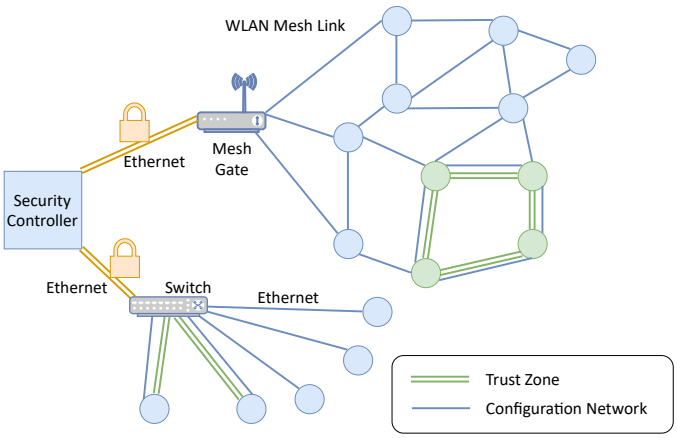

Fig. 6. Final Solution of Network Setup with a Trust Zone

in line-of-sight can be assumed to be at 227 m at 20 dBm transmit power, -93 dBm receive sensitivity for MCS 0, 3 dBi antenna gain, channel 1 (2412 MHz) with 20 MHz channel width and the path loss exponent p = 2,18. Because estimation results can differ from real-world setups, a connectivity test with an ICMP ping will be performed at a later stage. If nodes do not have MAC-layer connectivity this step of the partitioning algorithm will be executed again. A final risk analysis ensures that the calculated solution does not violate any security policies. These are: for applications between EDs there must be at least one protocol layer assuring the security requirements authenticity, integrity and confidentiality. If a device does not support transport layer or application layer security, all linked EDs have to be isolated within the same Trust Zone. Only devices with the same firmware version are allowed to act as infrastructure peers within the Trust Zone. The minimal amount of peers is allowed to join the Trust Zone for network infrastructure. The structured network is shown in figure 6.

V. Implementation & Evaluation

Our concept proposes the usage of a configuration service that is offered by EDs. In [5] we present an implementation of a configuration service that is based on the Devices Profile for Web Services (DPWS). A device which is offering the configuration service can be discovered using an Android application. The device is connected to other supported devices like sensors and actuators. The required metadata is gathered from a WSDL (Web Service Description Language) file, that is exchanged during the Web Service Discovery procedure executed by the Android client. Because the Web Service approach for embedded systems became replaced by RESTful APIs used by HTTP and CoAP, we selected CoAP as the state-of-the-art machine-to-machine protocol. We implemented a configuration service using jCoAP [12], a Java implementation of CoAP. Because CoAP do not support a self-description that contains information such as installation location, firmware version, device type or offered and supported interfaces, we created a data model. Every ED offers this information in form of resources of a RESTful API (table II). Another significant difference to our work [5] is, that devices aren't linked to each other using an Android application. Our SC replaces the Android Application by sending end-to-end encrypted configurations to EDs. The configurations in form of ITTT rules from [5] are extended by providing authenticated key material. In case devices support end-to-end security at the application layer like OSCORE or at the transport layer like DTLS, the SC provides trusted public keys to EDs. If a device only supports MAC-layer security (which is a mandatory requirement) the traffic will be isolated using Trust Zones. Next to the RESTful API shown in table II every ED executes a state machine presented in figure 7. First an ED is unconnected after first boot. Following the scheme in figure 2 an authorized technician negotiates key material using an authenticated diffie-hellman key exchange after [9]. Furthermore, every ED will be provided with credentials allowing MAC-layer network access control. In case of a wired Ethernet link, IEEE 802.1AE MACsec is used. It is implemented by a kernel module since Linux 4.5 in 2016, but must be enabled manually. When the ED is based on a IEEE 802.11 WLAN interface, AuthSAE implemented by a mesh daemon [13] is used within our proof-of-concept test setup. After the ED is connected to the configuration network, which uses MAC-layer network access control, the device's installation located is annotated to the BIM. Furthermore, the SC reads and stores the self-description of the ED including all offered resources of the RESTful API (GET request of mandatory /.well-known/core resource) and metadata offered by resources shown in table II. When all devices are commissioned and the planning algorithm is successfully executed (a solution without any policy violations is found), every device gets individual MAC-layer credentials so that is enters the next state of the state machine (figure 7).

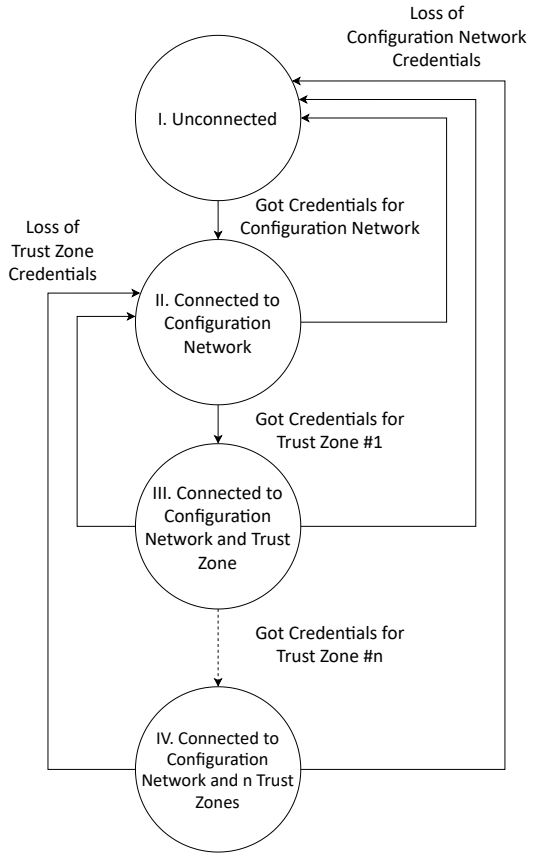

Fig. 7. State Machine of an ED

TABLE II. RESTful API offered by EDs

Ressource	**Example Data**
/device/Vendor	Example Company
/device/ModelName	Lightbulb E27
/device/FirmwareRevision	V 1.0
/device/Location	First Floor, Room 001
/security/MACSecurity	SAE, WPA2-PSK
/security/MACCredentials	WLAN1, MySSID, MyPwd
/security/joinTrustZone	disconnected
/security/trustedPublicKeys	[Public key(s)]
/security/ProtocolStack	WLAN/IPv4/UDP/OSCORE
/security/StatusConfigNetwork	Disconnected
/security/ANTsInterfaceIDs	WLAN1, WLAN2
/config/getSupportedInterfaces	Motion Detector
/config/ConnectDevice	null
/config/ConfigurationDone	false
/config/getConnectedDevices	Motion Detector [ID]
/config/SetRule	null
/config/getRules	null

We implemented our algorithm to extend Trust Zones with additional network infrastructure (figure 5) in Java. A result for a 2 D scenario with 100 randomly, evenly distributed WLAN mesh EDs is depicted in figure 8. We choose a total number of 100 EDs to visualize to location-based node selection in a 1000 x 1000 point grid. Three EDs, that are marked in green, form an application consisting of a sensor, a control unit and an actuator. These three EDs are put into the same Trust Zone. Due to the distance between them a single hop communication is not possible between two EDs of the graph. Therefore, a first naive algorithm to acquire additional network infrastructure selects peers that are located along the connecting lines. The selected EDs must be within a circle with the center at the half of each application link (green circles in figure 8). The radius of the circle must be at the maximum peer link distance. The EDs that are potential Trust Zone members are highlighted in red. After an evaluation EDs that do not fulfill our policy are eliminated. Only a minimal subset of EDs joins the Trust Zone to assure connectivity. We measured the execution time of the algorithm on the SC, depending of the number of randomly positioned nodes. Each experiment with a new random placement was executed 1000 times on a Desktop PC (Intel Core i7 5600U @ 2.6 GHz) and a Raspberry Pi 3. The average results including the standard deviation are depicted in figure 9. Because the acquisition algorithm evaluates the physical position of every ED relatively to the applications line-of-sight separately, the algorithm has a linear timing complexity. The quality metric R^2 larger than 0,998 indicates a high accuracy of the linear regression. The elimination of EDs that violate the policies has the same linear timing complexity. In a real-world scenario the SC would be implemented using an embedded device of the same class as the Raspberry Pi 3. We tested our partitioning algorithm up to 10.000 nodes in steps of 1000 nodes, because a BAS will always consist of less devices. The absolute average execution time for 10.000 EDs is at 201 ms. Thus, we can validate the practicability of our algorithm regarding the execution time. During our experiments, the Trust Zone formation algorithm eliminated 80 % on average of EDs for the uniformly distributed nodes scenario that are too far away to act as infrastructure peers. 20 % of the EDs are candidates to act as WLAN mesh peers for an application. During a second elimination step, EDs that violate the security policy are removed. The result is a minimal subset of EDs that is included into the Trust Zone to serve as infrastructure peers for an application.

VI. Related Work

There are other approaches that follow the software-defined security principle that a control instance orchestrates the network. In [14] there is a general architecture for software-defined buildings, but only small security considerations are done. They identify amongst other protocols WLAN as a communication technology in general, but do not go into details of WLAN mesh. The authors of [15] make use of an OpenFlow infrastructure to control traffic between an application logic controller and end devices (sensors and actuators). The threat to smart homes or buildings caused by compromised devices is the subject of [16]. They also isolate traffic within the local network. Their partitioning algorithm is, in contrast to our approach, based on a global blockchain to exchange anomaly behavior of devices and compute a trust score. The authors of [17] present a network-centric approach consisting of a so-called Smart IoT Gateway to control devices. There are neither BIM-based policies applied, nor a fine grained approach, how to create partitions within a communication technology (e.g., Ethernet and WLAN mesh). To sum up, there is no approach that leverages the potential of BIM to create network partitions for decentrally organized applications. Most security architectures have a general character, because implementations

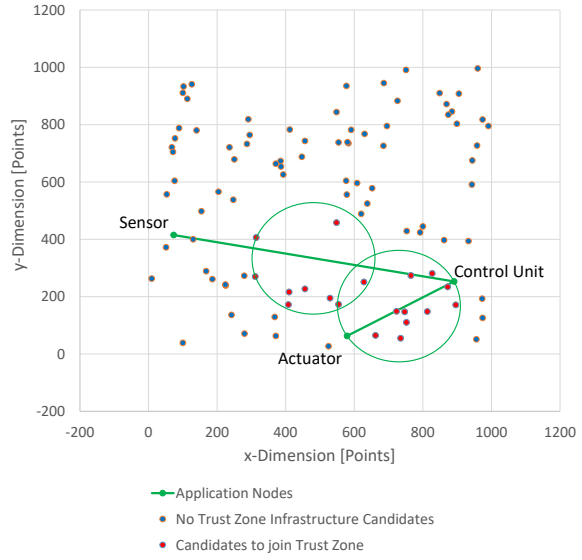

Fig. 8. Uniformly distributed EDs with an 3-node application Trust Zone (sensor, control unit and actuator) that is extended with infrastructure peers

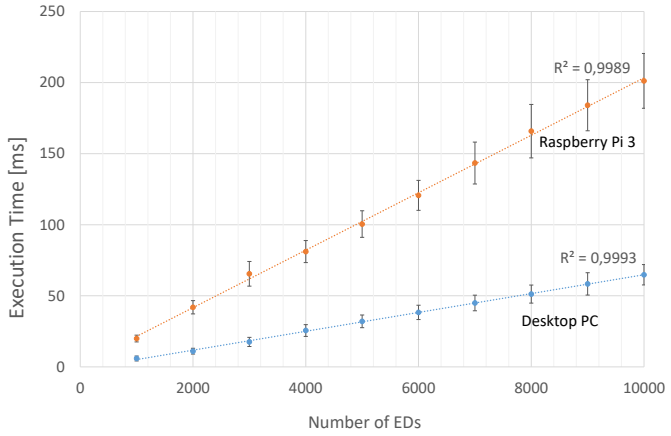

Fig. 9. Execution time of location-based selection of infrastructure EDs

and partitioning results for a network are missing. None of the related works consider a WLAN mesh network, that is configured by a SC.

VII. Conclusion

We proposed an algorithm for this modern building automation system that consists of embedded devices which are offering their sensor and actuator functionality in form of a CoAP-based RESTful API within a wired Ethernet and IEEE 802.11s WLAN mesh network. We show how a PKI is established based on proven concepts [9] to guarantee end-to-end security in terms of authenticity, integrity and confidentiality between all devices. The network topology in combination with the ground truth, which contains devices' metadata and the individual installation location as part of the BIM, are the input for our partitioning algorithm. We have shown how the member size of Trust Zones for MAC-layer isolation of devices is minimized by leveraging valuable information of the BIM. We explained and visualized the internal logic which is executed by the SC. The core component of infrastructure acquisition and minimal subset selection was implemented and evaluated regarding execution time. Furthermore, we implemented an example device with the proposed state machine and RESTful API. The partitioning of the network at the MAC-layer and configuration of applications can be automated by a single device in compliance with security policies. Vendors of EDs only have to implement a small set of functionalities, so that the devices can be part of a BIM-based and secured building automation system.

Acknowledgement

We would like to thank the German Federal Institute for Research on Building, Urban Affairs and Spatial Development (BBSR) within the Federal Office for Building and Regional Planning for their support in this project. This work is partially granted by BBSR within the scope of project "Zukunft Bau".

References

[1] A. Wall, H. Raddatz, M. Rethfeldt, P. Danielis, and D. Timmermann, "ANTs: Application-Driven Network Trust Zones on MAC Layer in Smart Buildings," in *Proc. of CCNC '18*.

[2] B. Butzin, F. Golatowski, and D. Timmermann, "A survey on information modeling and ontologies in building automation," in *Proc. of IECON '17*.

[3] S. Liaisons, R. Hall, M. Modera, C. Neilson, B. Isler, M. Osborne, D. Alexander, C. Brumley, C. Copass, S. Dinges *et al.*, "BACnet-A Data Communication Protocol for Building Automation and Control Networks," *ANSI/ASHRAE Standard*, vol. 135, 2012.

[4] Z. Shelby, K. Hartke, and C. Bormann, "The Constrained Application Protocol (CoAP)," RFC 7252, Jun. 2014. [Online]. Available: https://rfc-editor.org/rfc/rfc7252

[5] A. Wall, V. Altmann, J. Müller, H. Raddatz, and D. Timmermann, "Decentralized configuration of embedded web services for smart home applications," in *Proc. of SysCon '17*.

[6] IEEE, "IEEE Standard for Local and metropolitan area networks, Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications," *IEEE Std 802.11-2012*.

[7] A. Wall, B. Butzin, F. Golatowski, M. Rethfeldt, and D. Timmermann, "Software-defined security architecture for smart buildings using the building information model," in *2019 IEEE Global Conference on Internet of Things (GCIoT)*, 2019, pp. 1–5.

[8] A. Wall, H. Raddatz, M. Rethfeldt, P. Danielis, and D. Timmermann, "Performance evaluation of MAC-layer trust zones over virtual network interfaces," in *Proc. of MobiSecServ '18*.

[9] S. Unger and D. Timmermann, "Bridging the UI gap for authentication in smart environments," in *Proc. of ISCC '14*.

[10] M. Rethfeldt, A. Wall, P. Danielis, B. Konieczek, and D. Timmermann, "AKadeMesh: Software-defined overlay adaptation for the management of IEEE 802.11s networks," in *2016 13th IEEE Annual Consumer Communications Networking Conference (CCNC)*, 2016, pp. 477–482.

[11] M. Rethfeldt, B. Beichler, H. Raddatz, F. Uster, P. Danielis, C. Haubelt, and D. Timmermann, "Mini-Mesh: Practical assessment of a miniaturized IEEE 802.11n/s mesh testbed," in *2018 IEEE Wireless Communications and Networking Conference (WCNC)*, 2018, pp. 1–6.

[12] W. Group, "Git Repository: jCoAP," 2016. [Online]. Available: https://gitlab.amd.e-technik.uni-rostock.de/ws4d/jcoap

[13] cozybit, "Git Repository: cozybit/authsae," 2017. [Online]. Available: https://github.com/cozybit/authsae

[14] M. Mazzara, I. Afanasyev, S. R. Sarangi, S. Distefano, V. Kumar, and M. Ahmad, "A Reference Architecture for Smart and Software-Defined Buildings," in *Proc. of SMARTCOMP '19*, 2019, pp. 167–172.

[15] N. Xue, X. Huang, and J. Zhang, "S2Net: A Security Framework for Software Defined Intelligent Building Networks," in *Proc. of IEEE Trustcom/BigDataSE/ISPA '16*, 2016, pp. 654–661.

[16] M. Boussard, S. Papillon, P. Peloso, M. Signorini, and E. Waisbard, "STewARD:SDN and blockchain-based Trust evaluation for Automated Risk management on IoT Devices," in *Proc. of INFOCOM '19*, 2019, pp. 841–846.

[17] H. Iqbal, J. Ma, Q. Mu, V. Ramaswamy, G. Raymond, D. Vivanco, and J. Zuena, "Augmenting Security of Internet-of-Things Using Programmable Network-Centric Approaches: A Position Paper," in *Proc. of ICCCN '17*, 2017, pp. 1–6.