Graph-Based Simulation for Cyber-Physical Attacks on Smart Buildings

Rahul Agarwal[1], Na Meng[2], Xinghua Gao[3*], and Yuqing Liu[4]

[1] Computer Science, Virginia Polytechnic Institute and State University, Blacksburg, VA. E-mail: rahula@vt.edu
[2] Computer Science, Virginia Polytechnic Institute and State University, Blacksburg, VA. E-mail: nm8247@vt.edu
[3*] Myers-Lawson School of Construction, Virginia Polytechnic Institute and State University, Blacksburg, VA. E-mail: xinghua@vt.edu
[4] Computer Science, Virginia Polytechnic Institute and State University, Blacksburg, VA. E-mail: nap64@vt.edu

ABSTRACT

As buildings evolve towards the envisioned smart building paradigm, smart buildings' cybersecurity issues and physical security issues are mingling. Although research studies have been conducted to detect and prevent physical (or cyber) intrusions to smart building systems (SBS), it is still unknown (1) how one type of intrusion facilitates the other, and (2) how such synergic attacks compromise the security protection of whole systems. To investigate both research questions, the authors propose a graph-based testbed to simulate cyber-physical attacks on smart buildings. The testbed models both cyber and physical accesses of a smart building in an integrated graph and simulates diverse cyber-physical attacks to assess their synergic impacts on the building and its systems. In this paper, the authors present the testbed design and the developed prototype, SHSim. An experiment is conducted to simulate attacks on multiple smart home designs and to demonstrate the functions and feasibility of the proposed simulation system.

INTRODUCTION

Smart home automation is gaining popularity due to the convenience and facilities it provides to homeowners (Dong et al. 2019). Smart home devices connect into a common network that can be independently and remotely monitored. To provide such functionality, the devices store or transmit sensitive information, such as account information and live footage, which makes them cyber targets for attackers. On the other hand, smart home devices interact with the physical space as they can alter the lighting, configure air conditioner (AC) settings, monitor the space, unlock doors, etc. (Google, 2021). Breaching of physical space can lead to physical access to the devices and breaching of some devices can render the physical space insecure.

Unlike most cybersecurity threats, cyber-physical threats are of increasing concern. Gartner (2019) predicted that the financial impact of Cyber-Physical Systems (CPS) attacks resulting in fatal casualties will reach over $50 billion by 2023. Currently, there is no systematic approach to measure such attacks. To tackle this problem, we believe that the simulation of the entire system is desirable. Thus, in this paper, we present a simulation framework to investigate the security of a smart home environment and provide some recommendations on which smart home solution is more secure. With our tool, SHSim, researchers and customers can compare the

security levels of different smart home solutions. By analyzing the robustness of a given smart home solution, customers can make better decisions regarding which product they should purchase. Lastly, our study can help and motivate smart-home vendors to design a more secure suite of products.

SHSim includes a web interface to take user input of their smart home layout, a simulation engine that generates all possible attack paths, and a graph-based visualization of different attack models. In our solution, we first map the smart home layout to an integrated graph. Then, we traverse the graph with different starting points to generate all possible paths. We check the generated paths against various attack models and output all successful attack paths. In our experiments, we tested the vulnerability of two smart home ecosystems along with different configurations to check the robustness of the simulation engine. We found out that various cyber-physical attacks can be employed to breach the smart home ecosystem. The simulation engine exploits the vulnerability of popular communication protocols to generate all possible attack paths using which an attacker can gain access to critical smart home devices. To allow other researchers to use our tool, we have made SHSim publicly available at https://github.com/rahulaVT/SIM_app.

BACKGROUND

To simulate different attack models, we did a comprehensive literature review of cyber and physical attacks on IoT devices. Distributed Denial of Service (DDoS) (Zargar et al. 2013) is one of the most common attacks on IoT networks. It is an attempt to flood the server with internet traffic to shut down the service either completely or partially. The experiments conducted by Huraj et al. (Huraj et al. 2020) show how an attacker from outside can disturb the normal operation of smart home devices when they become a victim of a DDoS attack. Gómez et al. (Gómez et al. 2018) in their research work expose the vulnerability of ZigBee devices for wormhole attacks and packet injection. Chi et al. (Chi et al. 2020) present a variant of a jamming attack called concealed jamming attack that targets Zigbee devices, preventing them from receiving Zigbee packets that may contain important information.

Ciholas et al. (Ciholas et al. 2019) performed a systematic literature review of the security of Smart Buildings (SBs) wherein they investigated 90 relevant research papers. They found out that “very few papers consider the issue about physical attacks on sensors, actuators, and controllers in SB". In (Hager et al. 2012), the authors identified potential physical attacks such a s vandalism, burglary, and theft of SB devices. In (Mundt and Wickboldt 2016), the researchers exposed the vulnerability of KNX-based devices present at the field layer. Building Information Model (BIM) can be used to generate graphs, reflecting spatial relations between spaces within a building (Porter et al. 2014, Skandhakumar et al. 2016). Porter et al. (Porter et al. 2014) proposed a way to use the generated graph to simulate a physical attack on the building to see which rooms and obstacles (windows, doors, walls) are the most vulnerable.

Although the physical and cybersecurity of smart home environments have been studied independently, there lacks a tool that can simulate cyber-physical attacks together. Graph theory is a branch of discrete mathematics that has been proven to be effective in understanding complex

networks of interrelated entities (West 2001). Recent research studies have been implementing graph-based simulation and analysis methods in the research fields of building science (Hu et al. 2021) and Internet of Things (Mamonova & Maidaniuk 2020). This research presents a graph-based simulation technique to model both physical and cyber-attacks of a smart home environment.

METHOD

Figure 1 presents the system architecture of SHSim. Our approach primarily consists of five steps:

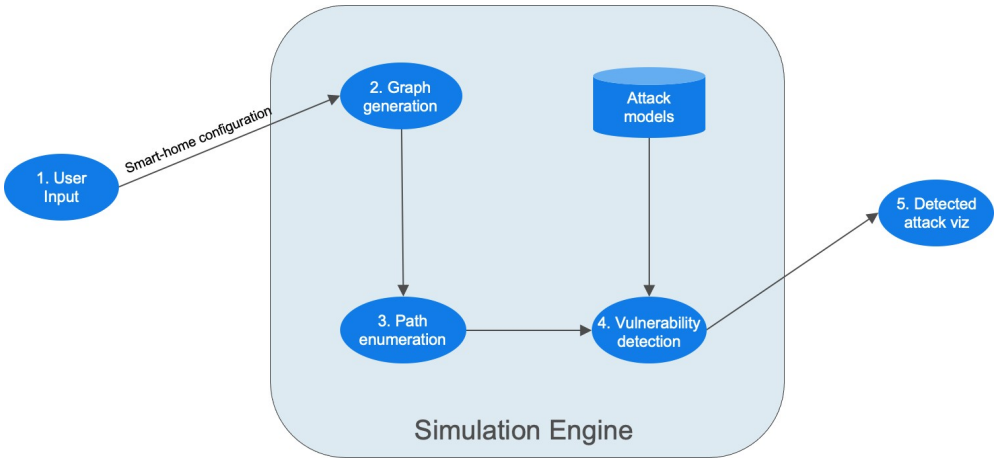

Figure 1. Architecture of SHSim

1. **User Input:** We have used Node.Js (Node.js 2021) & ReactJS (React 2021) for front-end and Python & Flask (Grinberg 2018) for running the back-end of our web application. The first step to running the simulation is to provide the smart-home configuration that includes details about the floor plan and the placement of smart home devices within the home.
2. **Graph Generation:** Graphs have been widely used in building network topology, social networks, biological networks, operational research, etc. (Trudeau 1993). A graph consists of vertices that are connected by edges. We have used graphs to model various smart home configurations, combining with agent-based simulation to analyze the cyber-physical security of smart homes.

   In order to model the smart-home system, we defined nodes and edges. We created *SpaceNode* to define the physical spaces such as hallway, kitchen, bedrooms, etc., and *DeviceNode* to define the smart home devices. Each *SpaceNode* holds a set of attributes such as area, floor level, window present (True/False), and occupied (True/False). *DeviceNode* has attributes such as type, placement, network, visibility, monitoring, etc. The edges connecting the nodes are of three types: physical connection, cyber connection, and containment connection, as shown in Figure 2. If there is a physical edge connecting two spaces, it means that the attacker can access and move between the spaces. The cyber connection shows connectivity between smart home devices. The placement of devices within spaces is defined by a containment edge.

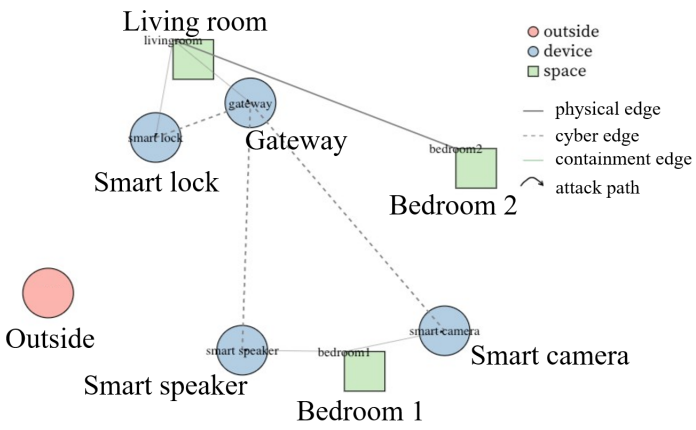

Figure 2. An example input graph of a smart home system

3. **Path Enumeration:** Once the graph is generated, different simulations can be performed on it, either locally or remotely in the cloud of the service provider. Our simulation engine works by considering certain assumptions about the attacker's capabilities. 1) The attacker must have a start position. For smart home simulation, a simple example can be that the attacker is located outside (represented as an orange circle in Figure 2) with access to the internet. 2) The attacker must also have access to some nodes to initiate attacks. 3) In order to initiate a cyber attack, we assume that the attacker can detect some wireless signals. 4) To simulate the attack method mentioned in (Sugawara et al. 2020), the visibility of the microphone must be modeled so the attacker can "see" the device and establish such an attack. 5) To initiate a physical attack, we assume that the attacker has tools to detect whether space is being monitored or not.

   We have used the Depth First Search (DFS) algorithm to traverse the generated graph and enumerate all possible paths between any two given nodes in the smart home system. The function "*generatePaths*" takes the start and end node as input parameters and returns all paths between the two nodes. The enumerated paths can be considered as candidate attack paths which will be used to find vulnerable attack paths.

4. **Vulnerability Detection:** We identified several attack vectors to model the cyber-physical security of the smart home system. Currently, we have modeled four different cyberattacks (as presented in Table 1), which can be combined with a physical attack if required. These attacks can target different layers of the Open Systems Interconnection (OSI) protocol stack. The modeled attacks are described as follows:

   Light Command. For this attack, the attacker targets the MEMS (microelectromechanical systems) microphone installed in smart devices (such as Home Assistant, smart speaker, etc.) to inject inaudible and invisible laser encoded commands. This enables the attacker to inject various commands such as "set volume to 0", "change temperature to 70F", "open garage door", etc.

   DDoS HTTP attack. HTTP Get Flood attack is one of the most common types of DDoS attacks of an application layer. During this attack, the attacker uses legitimate IP addresses which appear to be authentic, to overwhelm the webserver with multiple requests. Thus, new HTTP Get requests cannot be processed.

   Wormhole attack. This attack involves the introduction of a malicious node to form a tunnel between a source and a destination node to misguide network traffic or modify data packets. This attack typically occurs when the two nodes are far apart and there is a need to relay the information

through interim nodes.

Table 1. All attacks modeled in SHSim

Attack vector	Communication protocol	Constraint	Attacked device	Testing device	Difficulty
Light command (Sugawara et al. 2020)	Z-Wave, Zigbee	visibility	Smart speaker	Smart Lock	high
DDoS Attack HTTP (Huraj et al. 2020)	Z-Wave	none	Gateway	any Z-Wave sensor	medium
Wormhole attack (Gómez et al. 2018)	Zigbee	none	Gateway	any Zigbee smart device	medium
Reactive jamming (Wilhelm et al. 2011)	Zigbee	none	Gateway	any Zigbee smart device	medium
Window break-in	N/A	not monitored	NA	NA	easy

Reactive jamming attack. In this attack, the attacker pretends to be a legitimate WIFI device to prevent or delay the Zigbee device's communication to the gateway. It is an optimal jamming technique since it aims at destroying only selected data packets, which are in the air, by using a short jamming signal, thus minimizing its risk of being detected.

Window break-in. In this attack, the attacker can break into space that is not being monitored. Thus, the attacker can gain access to space as well as the devices in that space. This is the only physical attack that we have modeled to explore the synergy between cyber and physical attacks.

We consider an attack to be successful (or a candidate path to be vulnerable) if using one or more attack vectors, the attacker can enter the house. A resultant attack path can be as simple as having a single node (e.g., break-in through the window) or a much more complex attack that involves multiple nodes (e.g., performing laser-based voice command injection to unlock the door). For each resultant attack path, we estimate how difficult (low/medium/high) it is for the attacker to breach the smart home system. The "difficulty level" is a configurable parameter entered by the user to SHSim, which intends to reflect how easy/difficult it is for hackers to achieve the attack. We configured the parameters for individual attacks based on our comprehension of the attack methods described in related papers. Among the five attacks modeled, the "window break-in" attack is easy, as attackers do not need any specialized tool, device, or expertise. The "light command" attack is the most difficult one, as it requires the usage of specialized devices (i.e., a telephoto mounted on the geared tripod head for shooting laser to targets), and the domain knowledge of converting laser signals to meaningful audio commands. We believe the other three attacks to be easier than "light command" and harder than "window break-in". This is because the other attacks require hackers to bring their own devices (computers or WIFI devices) and send data to the Gateway or other nodes in the same network via specialized ways. These devices are easy to access, and the attacks intend to disrupt network routing or data transmission, instead of forging malicious instructions/commands. The implementation of this simulation is extendable to allow the addition of new attack vectors.

5. **Detected Attacks Visualization:** We have used *D3*'s force-directed graph to create interactive graphical visualizations of detected attacks on a smart-home environment. It uses a physics-based simulator, *forceLink()*, for positioning visual elements. We provide an attraction force between connected nodes such that our graph arranges itself according to our edges. In Figure 3, we can see an example of the output visualization.

EXPERIMENT

We performed a few experiments on two different smart home designs to cover divergent scenarios and test the validity of our simulation engine.

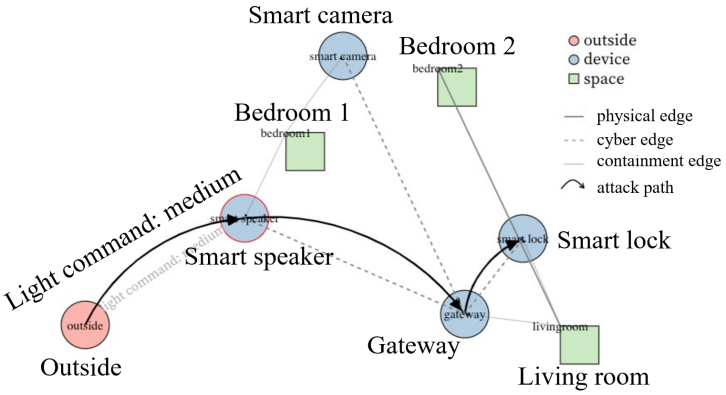

Figure 3. Light command attack on Smart home layout 1

Smart home layout 1:

As shown in Figure 4, we defined a smart home layout with three spaces (bedroom 1, bedroom 2, and a living room) and four smart home devices. Bedroom 1 (or the Master bedroom) contains a smart speaker which can also act as a voice-enabled assistant (e.g., Google home or Echo dot) and a smart camera that monitors the space. Bedroom 2 doesn't contain any smart device. The living room has a gateway that connects to other smart home devices and relays data packets. The living room also has a smart lock installed at the main door. Figure 2 presents the graph generated for the smart home layout. With the simulation, we obtained a list of attack paths with relevant information such as attack difficulty level, attack type, targeted node, etc.

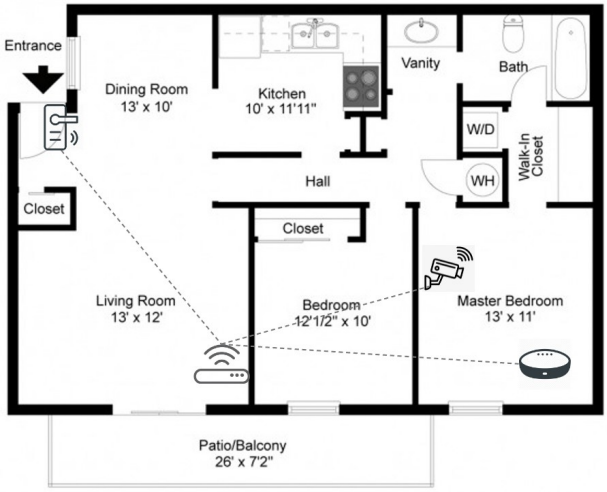

Figure 4. Smart home layout 1

Experiment 1a: Space monitored vs not monitored. For the given smart home layout, we experimented by first adding the smart camera to monitor bedroom 1 and then removing it to

observe the differences in output. We identified 9 attack paths when the space was being monitored and 12 attack paths when it was not being monitored. The difference lies in the fact that when space is not being monitored, the attacker can perform a physical attack on bedroom 1 window and enter the space. In Figure 3, we can see one of the resultant paths wherein the attacker uses the Light command (Sugawara et al. 2020) attack to target the smart speaker and unlocks the smart lock. The engine estimates that this attack has a “medium". Also, we eliminate some of the unknown parameters such as time taken to hack the passcode of the smart lock since it depends entirely on the kind of smart lock installed. We found that certain smart locks such as Nest X Yale locks are much more secure than August smart locks. Other possible attacks include a DDoS HTTP (Huraj et al. 2020) attack on the gateway which blocks its internet connectivity, thus restricting the homeowner from using commands which involve an internet connection.

Experiment 1b: Smart Speaker visibility. For the second part of our experiment, we removed the visibility of the smart speaker from the outside and ran the simulation again. We found out that the attacker was no longer able to target the smart speaker from outside since it was not in the line of sight. However, since bedroom1 was not being monitored, the attacker was able to physically enter and then perform the Light command attack, as now the smart speaker was in the line of sight.

Smart home layout 2

We took the design inspiration of the second smart home layout from Surreal System’s website (System 2021). On their homepage, they provide the floor plan (as shown in Figure 5) of a smart home which consists of a living room, a bedroom, a storage room, and a garage. The living room has a voice-enabled assistant and a smart lighting system. The bedroom and living room windows have motorized smart shades. The storage room contains the gateway and the garage is being monitored via a smart camera. The devices communicate with others over the Zigbee network and can be controlled via a switch, mobile app, or home assistant.

Experiment 2: No occupants. For this experiment, we configured the spaces with no occupants while running the simulation. We found the smart home design being quite vulnerable as the simulation provided us 31 different attack paths. The attacker was able to exploit the vulnerability of the Zigbee network and perform four different kinds of cyber-physical attacks:

Wormhole attack. The attacker inserts a malicious node to form a tunnel between the Home assistant and other smart devices such as smart light, smart shades, etc. By doing this, the attacker can misguide network traffic and even modify data packets.

Concealed jamming attack. (Chi et al. 2020): In this attack method, the attacker targets the gateway to jam or delay the data transmission between gateway and other smart devices such as Home assistant, smart lights, smart shades, smart camera, etc. In Fig 6, we can see one such attack where the attacker jams the network between smart shades and the gateway. Thus, if the resident wants to operate the smart shades using the Home assistant, they would fail. However, they can still operate it via a switch or mobile app.

Light command attack. The attacker targets the Home assistant device present in the bedroom to inject laser commands. Thus, the attacker can gain control over other controllable devices such as smart shades and smart lights.

Physical attack. The attacker has the option to break into the house through the living room and bedroom windows as the spaces are not being monitored or occupied. The attacker cannot enter through the garage as it is being monitored by a camera.

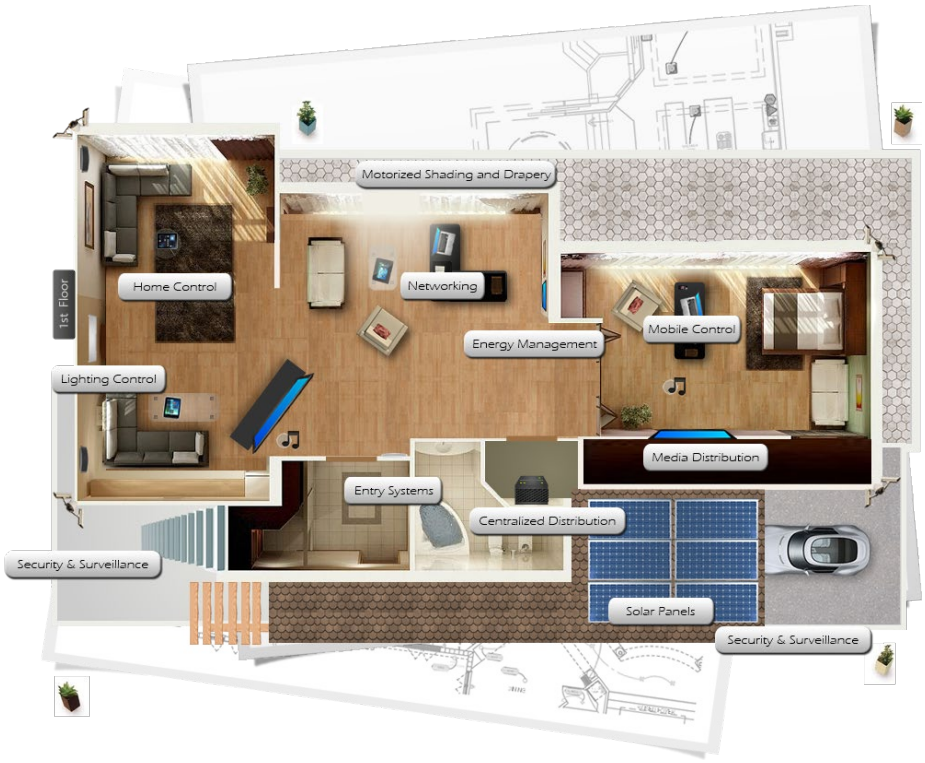

Figure 5. Smart home layout 2

The experiments conducted illustrate that the current generation of smart home systems are not always sufficiently secure. The attacker by sensing weak entry points can launch cyber-physical attacks to either enter the house or gain access to the network. We found that the Base stations or the gateways are the prime targets to various cyber attacks. Spaces which are not monitored or occupied can lead to physical attacks. The attacker on entering the house can sense neighboring spaces and plan his next attack accordingly.

DISCUSSION

SHSim can be used by researchers and customers to analyze the security of their smart home systems. Using the simulation engine, device manufacturers can find loopholes and try to fix them. Typically, a smart homeowner installs devices that are compatible with one common smart home solution (e.g., Google, Amazon, etc.) to avoid having multiple different gateways. Our simulation engine can handle multiple smart home solution ecosystems.

We have modeled four different cyberattacks in the current SHSim system prototype, and these attacks are applicable to the systems using Z-Wave and Zigbee as communication protocols. Zigbee and Z-Wave are two of the most commonly used communication protocols in the smart building field (Gao et al. 2019). We have modeled these example attacks to prove the concept. The SHSim system is further expandable, and we will incorporate more attack models applicable to protocols, such as BACnet, Modbus, and LonWorks, in the near future.

Even though we conducted several experiments to test the simulation engine and explore different attack paths, more study is needed to further improve the proposed system. More experiments using a real-life smart home setting can bring out interesting insights as well as test the robustness of the simulation engine to handle different circumstances.

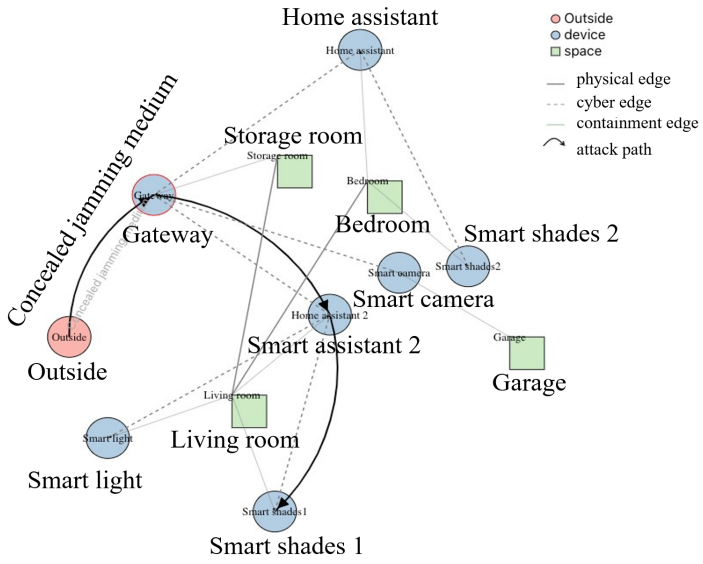

Figure 6. Concealed jamming attack on Smart home layout 3

We made certain assumptions for successfully running the simulation engine which can be taken as parameters in a future version of SHSim. For instance, we assume that the attacker from outside gets the network signal to access a few smart home devices. However, we do not consider signal information in our modeling. Thus, for future development, we can build a large database of smart home devices with their profile and provide a user the option to select a device from the list.

CONCLUSION

Cyber-physical security for smart-building is a relatively new topic and requires a scientific tool to help study potential cyber-physical attacks and impacts. This research has proposed a system, SHSim, that can simulate cyber-physical systems using graphs, and offered a prototype that is expandable and capable of analyzing the threat to the smart-home environment. We conducted several experiments to analyze the vulnerability of different smart home designs. The experiment results confirmed that the gateways and hubs are the weak links of the smart building systems and the primary targets of cyberattacks. Also, the simulation confirmed that the lack of a smart security system can provide an opportunity for a physical attack, which in turn can result in cyberattacks on smart devices. Future directions of improvements include more detailed modeling that better reflects reality, optimized algorithms for simulation, different algorithms to simulate a different aspect of the system, a standardized attack and impact library, and standardized modeling of devices and protocols.

REFERENCES

Chi, Z., Li, Y., Liu, X., Wang, W., Yao, Y., Zhu, T., and Zhang, Y. (2020). "Countering Cross-Technology Jamming Attack." *Proceedings of the 13th ACM Conference on Security and Privacy in Wireless and Mobile Networks*, WiSec '20, New York, NY, USA, Association for Computing Machinery, 99–110.

Ciholas, P., Lennie, A., Sadigova, P., and Such, J. M. (2019). "The security of smart buildings: a systematic literature review.

Dong, B., Prakash, V., Feng, F., & O'Neill, Z. (2019). A review of smart building sensing system

for better indoor environment control. *Energy and Buildings*, 199, 29-46.
Gao, X., Tang, S., Pishdad-Bozorgi, P., & Shelden, D. (2019). *Foundational research in integrated building Internet of Things (IoT) data standards*. Center for the Development and Application of Internet of Things Technologies.
Gartner (2019), *Predicts 2020: Security and Risk Management Programs*, <https://www.gartner.com/en/documents/3976275/predicts-2020-security-and-risk-management-programs>. [Accessed August 25st, 2021].
Google (2021), Google Nest, build your connected home, <https://store.google.com/us/category/connected_home?>. [Accessed August 25st, 2021].
Gómez, J. R., Vargas Montoya, H. F., and Henao, A. L. (2018). "Implementation of a wormhole attack on wireless sensor networks with xbee s2c devices." *Advances in Computing*, J. E. Serrano C. and J. C. Martínez-Santos, eds., Cham, Springer International Publishing, 98–112.
Grinberg, M. (2018). Flask web development: developing web applications with Python. O'Reilly Media, Inc.
Hager, M., Schellenberg, S., Seitz, J., Mann, S., and Schorcht, G. (2012). "Secure and qos-aware communications for smart home services." *2012 35th International Conference on Telecommu- nications and Signal Processing (TSP)*, 11–17.
Hu, Y., Castro-Lacouture, D., Eastman, C. M., & Navathe, S. B. (2021). Component Change List Prediction for BIM-Based Clash Resolution from a Graph Perspective. *Journal of Construction Engineering and Management*, 147(8), 04021085.
Huraj, L., Šimon, M., and Horák, T. (2020). "Resistance of iot sensors against ddos attack in smart home environment." *Sensors*, 20(18).
Mamonova, G., & Maidaniuk, N. (2020). Mathematical Tools for the Internet of Things Analysis. *Cybernetics and Systems Analysis*, 56(4), 621-627.
Mundt, T. and Wickboldt, P. (2016). "Security in building automation systems - a first analysis."
2016 International Conference On Cyber Security And Protection Of Digital Services (Cyber Security), 1–8.
Node.js. "About Node.js." *Node.js*, <https://nodejs.org/en/about/>. [Accessed April 21st, 2021].
Porter, S., Tan, T., Tan, T., and West, G. (2014). "Breaking into bim: Performing static and dynamic security analysis with the aid of bim." *Automation in Construction*, 40, 84–95.
React (2021). "About React.js, <https://reactjs.org/>. [Accessed April 20th, 2021].
Skandhakumar, N., Salim, F., Reid, J., Drogemuller, R., and Dawson, E. (2016). "Graph theory based representation of building information models for access control applications." *Automation in Construction*, 68, 44–51.
Sugawara, T., Cyr, B., Rampazzi, S., Genkin, D., and Fu, K. (2020). "Light commands: Laser-based audio injection attacks on voice-controllable systems.
System, S. (2021). "Smart Home Layout, <https://www.surrealsystems.com/solutions/home-solutions/smart-home-layout>. [Accessed April 18, 2021].
Trudeau, R. (1993). *Introduction to Graph Theory*. Dover Books on Mathematics.
West, D. B. (2001). *Introduction to graph theory* (Vol. 2). Upper Saddle River: Prentice hall.
Wilhelm, M., Martinovic, I., Schmitt, J., and Lenders, V. (2011). "Short Paper: ReactiveJamming in Wireless Networks—How Realistic is the Threat?." 47–52 (06).
Zargar, S. T., Joshi, J., and Tipper, D. (2013). "A survey of defense mechanisms against distributed denial of service (ddos) flooding attacks." *IEEE Communications Surveys Tutorials*, 15(4), 2046–2069.