A Voice Assistant for IoT Cybersecurity

Jeffrey S. Chavis*, Malcom DosterΨ, Michelle Fengλ, Syeda Zeeshanϕ, Samantha Fu†, Elizabeth Aguirre†, Antonio Davila§, Kofi Nyarko‡, Aaron Kunz*, Tracy Herriotts*, Daniel Syed*, Lanier Watkins*, Anna Buczak*, and Aviel Rubin†
* *The Johns Hopkins University Applied Physics Laboratory, Laurel, MD*
Ψ Charles Herbert Flowers High School, Prince George's County, MD
λ Bryn Mawr High School, Baltimore MD
ϕ Atholton High School, Howard County, MD
† The Johns Hopkins University, Baltimore, MD
§ The American University, Washington, DC
‡ Morgan State University, Baltimore MD

Abstract—The Internet of Things (IoT) is becoming more pervasive in the home, office, hospital, and many other user-facing environments (UFEs) as more devices are networked to improve functionality. However, this explosion of networked devices in UFEs necessitates that security systems become easier to help users remain aware of the security of the devices on their network. Users may not have the skills or the time needed to continuously monitor networks of increasing complexity using common open-source tools. Specifically, they are not likely to fully comprehend the data that those tools present, nor are they likely to have a working knowledge of the tools needed to monitor and protect their IoT-enabled network environments. This paper explores development of a system that uses ambient computing to facilitate network security monitoring and administration. Our system is designed to combine machine-learning–enriched device awareness and dynamic visualization of IoT networks with a natural language query interface enabled by voice assistants to greatly simplify the process of providing awareness of the security state of the network. The voice assistant integrates knowledge of devices on the network to communicate status and concerns in a manner that is easily comprehensible. These capabilities will help to improve the security of UFEs while lowering the associated cognitive load on the users. This paper outlines continued work in progress toward building this capability as well as initial results on the efficacy of the system.

Keywords: Ambient Computing, Cybersecurity, Internet of Things, Machine Learning, Network Visualization, Voice Assistant

I. Introduction

The Internet of Things (IoT) is becoming pervasive in the home, business, and mission-critical environments as more consumer, business, and industrial control devices are networked and IoT enabled, thus improving functionality in user-facing environments (UFE). UFEs are smart environments that have deep technological foundations, but strive to only present a portion of the technology to the user. This explosion of networked devices exposes users to many security vulnerabilities and thus necessitates that smart-system owners become more aware of the activity on and security of the devices on their network.

System owners likely do not have the skills necessary nor the time needed to continuously monitor their network using common open-source tools. Furthermore, network defenders are often inundated by the sheer number and diversity of devices and associated traffic and alerts. No one individual is likely to be able to fully use and comprehend the data that those tools present. We propose a system that uses ambient computing to facilitate network security monitoring and administration for smart and connected environments. Ambient computing refers to technologies that allow people to use a computer without realizing they are doing it [1]. This work is an extension of earlier work on the Connected Home Automated Security Monitor (CHASM) [2] [3]. In this paper, we combine dynamic visualization of IoT networks with a natural language query interface enabled by voice assistants to greatly simplify the process of providing information about the security state of the network to the casual user as well as the more seasoned network defender.

This paper (1) demonstrates a voice assistant capability for IoT security applications that lowers the cognitive load on the end user, independent of their security and network skill level, (2) integrates trigger-based packet capture (PCAP) capability for deep packet inspections and IoT ML development, (3) integrates an ML-based discovery capability to characterize devices on a network, and (4) introduces a graph analytics capability for visualizing network connectivity and situational awareness.

The remainder of this paper is organized as follows: Section II discusses voice assistant cyber scenarios and use cases. Section III describes voice assistant applications and related works on which our research builds. Section IV presents an overview of the system architecture. Section V overviews voice assistant interface design and associated commands. Section VI describes the embedded underlying capability. Section VII details the experimental evaluation we conducted. Section VIII explores voice assistant security concerns. Lastly, Section IX presents conclusions.

II. Voice Assistant Cyber Scenarios and Use Cases

Provided next are examples where this voice assistant interface could improve current user experiences.

A. Use Case #1: Low cyber skilled (average smart home, IoT devices owner)

A homeowner with limited understanding of computing and the Internet installs a personal cyber assistant to help protect their devices and personal information. The personal assistant connects with the main router in the house; the home owner provides the serial number. Once installed, the personal assistant detects and inventories all the devices connected to the associated Internet Protocol (IP) address. The personal assistant provides that inventory to the home owner, who then provides confirmation for each device. Whenever a new device is

detected for the first time, the personal assistant sends an alert and requests confirmation that the device belongs on the network.

B. *Use Case #2: Traveler's Assistant (high-end IoT application owner with minimal networking)*

A tech-savvy business person spends a large amount of time traveling and is highly dependent on Internet and cellular. Their business requires the secure transmission of large amounts of proprietary and otherwise sensitive data. As a result, they are not comfortable transmitting that data from or receiving it at a hotel room, local coffee shop, or the airport.

The personal assistant can provide real-time assessments of the security of local networks and access information about potentially more secure networks nearby. In addition, the personal assistant can help the business person set up a virtual private network (VPN) or other security structures to ensure secure transmission of information.

III. Voice Assistant Applications and Related Works

The voice assistant capabilities can be catalogued into two areas: (1) network monitoring smart assistance and (2) general voice user interface (VUI) applications.

A. *Network Monitoring Smart Assistants*

The growth of common IoT devices in UFEs and their respective networks opens up considerable risk possibilities. To combat this issue, network monitoring is useful for alerting and guiding network users through the security statuses of their devices. Narayanan et al. [4] developed systems that use smart assistance technology to regularly scan networks and users in search of needed backups, notifying a previously designated client in the event of positive detection. Without regular backups, systems become vulnerable to permanent data loss from a data breach or attack. With the combination of technology and human-driven backup systems, users can eliminate this vulnerability through double-layered authentication and the removal of sole human or computer error.

Additionally, with the constantly changing landscape of cybersecurity threats, early detection of such attacks has become more challenging. Even with advanced monitoring protocols, hackers can be present on a system for more than 100 days before being detected. Bassett et al. [5] developed a cybersecurity system that intakes data from a number of sources to have multiple collaborative smart agents complete a collection of network security tasks. This results in more data-informed decisions for security administrators, while simultaneously lowering their cognitive load and the potential for human error.

B. *General VUI Applications*

The diverse applications of VUI-enabled smart home devices can be further divided into two categories: convenience and practical.

The majority of convenience in both home and professional UFEs involves the replacement of traditional appliances with smart devices for easier use. Smart home assistants can control an interconnected network of WiFi-connected light bulbs, alarms, thermostats, and other instruments without any practical need for automation. [6] For instance, the Amazon Alexa smart home interacts with users through a VUI, and is able to control any device connected to WiFi through either their own application or WiFi-connected smart plugs. Although mainly implemented through VUI, Amazon Alexa's nature allows communication through e-text on mobile devices and on Amazon-provided screen displays. Other examples of the simplified lifestyle effects these VUI assistants provide include starting workouts, ordering online purchases, controlling smart homes, scheduling routines, and performing calculations.

The variety of practical applications includes uses with real demand and necessity in home and office use. An IoT-based fall-detection system proposed by members of the Department of Electrical and Computer Engineering at the University of Kentucky uses cameras and motion sensors to identify potentially life-threatening falls in homes with vulnerable occupants. [7] Once activated, the connected smart assistant launches a VUI-based dialog, prompting endangered users to notify the police or caregivers. Such application of the smart assistant relies heavily on the vocal component of the device because the use of a touch-based system may not be possible in a risk situation. This allows for more independent living, thereby saving money and time.

IV. System Architecture

The ambient computing capabilities presented in this paper leverage both internally facing and cloud-based capabilities. This section provides an overview of the computing and network infrastructure and describes an end-to-end flow of a user query and its associated response through the system.

The voice interface provides access to a set of underlying cyber-focused capabilities exposed as edge-based RESTful services running on Raspberry Pi devices. We used two Raspberry Pi 3B devices, each having a four-core, 1.2-GHz Broadcom, 64-bit ARMv7 CPU, and 1 GB of RAM. The REST services return results via JavaScript Object Notation (JSON) to an ESXi server running Linux virtual machines (VMs). The Pi edge devices monitored collection of more than 70 IoT network devices connected to the same network, and a span port on the network switch forwarded all the network traffic from each device to the Raspberry Pi (Fig. 1).

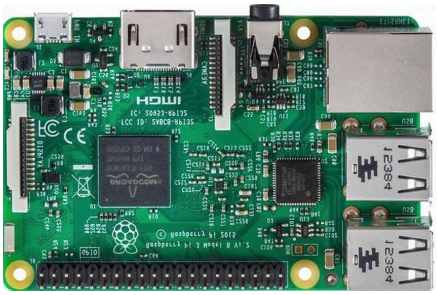

Fig. 1. Raspberry Pi

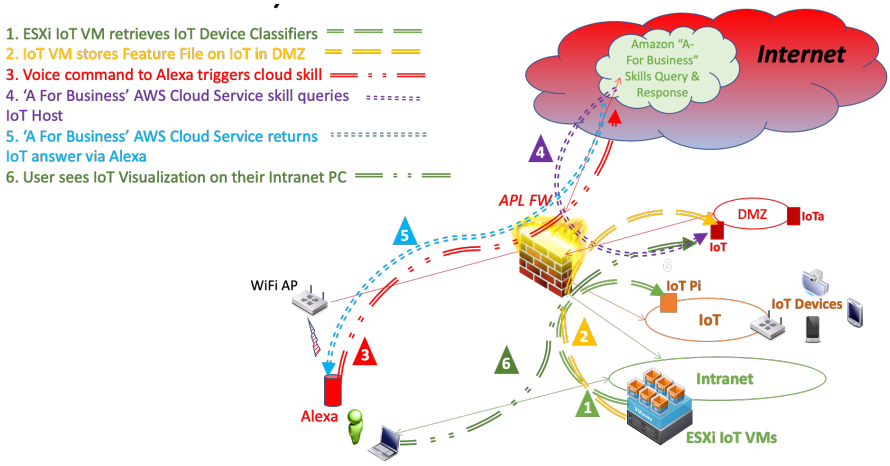

Fig. 2. Alexa IoT Device and Network Status

We used Amazon Alexa for business (A4B) to create the user interface. Any Alexa user registered on our network can connect via WiFi and access A4B services in the cloud. Upon joining, the user is authenticated and their voice commands go to the A4B cloud account and they can access related backend services in the IoT testbed. (See Section VI for an overview of the backend embedded cyber services.) Fig. 2 illustrates the data and information flowing though the network.

An ESXi IoT VM retrieves IoT device classification from an edge-deployed Raspberry Pi (IoT Pi) running ML-based device classifiers. The IoT Pi monitoring the collection of Pi sends JSON messages via a standard syslog port (UDP port 514).

The ESXi IoT VM stores the Feature File on the IoT Pi in the DMZ. All data collected here are processed and stored on a DMZ that can be reached by the Internet.

The user's voice commands are streamed by Alexa, using TLS 1.2, to an A4B account for processing by Amazon Voice Services (AVS). Amazon's Automatic Speech Recognition (ASR) processes the stream into text strings that are then forwarded to Amazon's Natural Language Understanding (NLU) system. NLU interprets the result and produces an Intent. The service then routes the Intent to one of our custom Skills.

- The Skill retrieves the IoT network data from the IoT host located in our DMZ.
- The Skill formulates the raw data into Simple Speech Markup Language (SSML) text .The response system then takes the SSML and uses text-to-speech to generate an audio speech file. The resulting audio is then streamed back to Alexa.
- The user sees a visualization of their IoT network from the IoT DMZ host.

V. Voice Assistant Interface Design

Although there are many existing frameworks for the creation of voice assistant capabilities, our cyber assistant was specifically developed as an Amazon Alexa skill. Amazon allows for the creation of custom cyber-assistant capabilities through the creation of Alexa skills, which can be published and then added to a specific Alexa device. This framework was chosen specifically because Amazon provides a robust Software Development Kit (SDK) for Alexa skill development and handles any necessary speech-to-text and natural language processing, allowing us to focus on the development of the VUI and underlying logic. Internally, an Alexa skill splits an individual piece of functionality into Intents, each having frontend and backend pieces. A specific intent's frontend consists of the voice user interface, also known as "utterances" or invocation phrases that trigger this specific piece of functionality. The backend consists of an IntentHandler, and deals with the actual logic of the triggered intent [8]. The development of the Alexa skill thus began with the design of the voice interface and the intended functionality, with a focus on user experience.

Because the Alexa skill's goal is to provide an easy-to-use, easy-to-understand, natural cyber voice assistant, we paid particular care to the multiple ways in which a user could interact with the skill, as well as the types of functionality provided by the skill; for instance, the voice interface needed to be able to handle cases where users may phrase the same request differently. Alexa specifically handled this by allowing for multiple utterances to be tied to each intent. The specific intents and commands supported by the cyber assistant were also chosen by considering the kinds of information both technical and nontechnical users may want to know about their network.

The different commands supported by the Alexa skill can be divided into three categories. The first includes administrative commands that control the overall skill and the underlying ML capability. For instance, a user can start and stop the Alexa skill, change certain settings including the setting of certain triggers and alerts, and ask for help. On the backend side, a user will also be able to start, stop, and reboot the ML pipeline as well as check its status.

The second category of commands forms the bulk of the assistant's core functionality. These are the commands that query the ML pipeline for specific information about the user's network, including the number of devices, categories of the

devices, newest device, etc. The goal of these commands is to give a user increased access to information about their network in a way that is user friendly and non-technical so that any network owner may use this capability.

The final category of commands includes those aimed toward users with more advanced knowledge of cybersecurity and networking to support a range of users. These commands will give a network owner access to more technical information regarding their network and allow for the configuration of combinations of event triggers. This category also includes the commands that integrate other cyber capabilities that the skill supports, including the directed PCAP and network visualization.

This effort was focused on implementing this functionality as an AWS Lambda-hosted Alexa skill, with corresponding utterances and intents. Table I briefly summarizes the commands that the skill currently supports, including sample invocation phrases and intended behavior.

VI. Embedded Underlying Capability

The voice interface provides an interface to a set of underlying cyber-focused capabilities. This section describes the collection of applications exposed to the user.

A. IoT Device Discovery and Classification

IoT discovery is foundational to providing good cybersecurity because a user needs to have a good accounting and understanding of the devices on a network to be able to protect and secure them. Therefore, the focus of this work is to explore discovery, profiling, and verification of IoT devices solely based on their network behavior or other information contained in individual or constrained groups of packets [2].

To support this goal, an ML model was trained to analyze packet sequences and predict what type of IoT device each packet sequence came from [3]. To construct the training data set, traffic from more than 60 IoT devices was collected, grouped by MAC address, and arranged in time order. The data from each device was then transformed into sequences of 20 packets each. The values in these sequences were then normalized, one-hot encoded, and labeled according to device category (see Table II), and an ML model was trained on that data. The result was a model that can take a 20-packet sequence of network data from a device and provide a prediction as to that device's category.

The model was deployed to run in streaming mode on a Raspberry Pi, resulting in a stream of output predictions. A dashboard was created to aggregate and display these predictions (see Fig. 3).

B. Integrated PCAP

The key element to providing cyber protections to a system is to have timely relevant data with labels related to the time, event, and situation that generated the event of interest. Monahan et al. states that organizations using PCAP as part of their normal toolsets were more confident in the information they received about their environments and therefore were better prepared to protect them. Specifically, they had:

- Shorter breach detection and response time
- More confidence in their workflows and processes

TABLE I. Core Functionality

Intent Request Name	Purpose	Sample Utterances	Slots (optional)	Returns
GetNetworkSummaryIntent	Provide an overview of connected devices and status of network.	"Tell me about my network." "Summarize my network." "Tell me a summary."		Number of total connected devices, number of devices active in the last 24 hours, and last device added
GetDevicesSummaryIntent	Provide a summary about the connected devices and their categories.	"How many devices do I have?" "What devices are on my network." "Tell me about the devices on my network."		Number of connected devices and number of connected devices in each category
GetNewestDeviceIntent	Provide an overview of the newest device on the network.	"What's the newest device?" "Get the newest device." "Tell me about the newest device."		The newest device added to the network, the time it was added, and the type of device it is
GetLatestDevicesIntent	Provide a list of devices added after a certain specified time. If no time is specified, default to last time the skill was used.	"Any new devices since yesterday?" "Any new devices?" "How many new devices do I have since last week?"	Amazon.DATE Amazon.TIME	A list of devices added to the network after the specified time and the category breakdowns
GetNetworkMapIntent	Provide to the user a graphical representation of the network and devices connected to it.	"Show me my network map."		A user-viewable network map with labeled nodes and edges representative of the network and activity on the network
GetPacketDataIntent	Provide the user PCAP data collected from the network.	"Capture data from the network for me."		PCAP data from *tcpdump*
HelpIntent	Suggest commands to ask Alexa about your network.	"Help." "What can I ask?" "What can you tell me about my network?"		A list of commands available to ask the Alexa skill

TABLE II. SUPPORTING METADATA

Metadata Tag	Description
“predicted_category”: “television” # This line and the next two are a summary of the category_scores	Category with the highest evaluated score
“confidence_level”: “Low”,	Confidence level of the prediction
“confidence_percent”: 49,	Score of the prediction
“device_id”: “b8:27:eb:3d:c2:a2”, #	Unique identifier for tracking devices inside the ML pipeline
“first_seen_ts”: #	Unix timestamp in ms (This is the time that the device was first detected on the network.)
“first_seen_utc”: “2020-06-05 14:33:52.013460+00:00”, #	Human-readable version of first_seen_ts
“last_seen_ts”: 1591887289012.231, #	Unit timestamp in ms (This is the most recent time that the device was seen on the network.)
“last_seen_utc”: “2020-06-11 14:54:49.012231+00:00”, #	Human-readable version of last_seen_ts
“ground_truth”: {	Ground truth for the MAC address (if known)
“iot_testbed_alias”: ““,	Testbed alias for a device (if available)
“iot_testbed_category”: ““	Category with the highest evaluated score
“ip_addresses_used”:	IP addresses that the device was seen using
“mac_address”: “b8:27:eb:3d:c2:a2”, # MAC address of the device	MAC address of the device
“mac_manufacturer”: “Raspberry Pi Foundation”, #	Possible manufacturer of the device, based on first three octets of MAC address

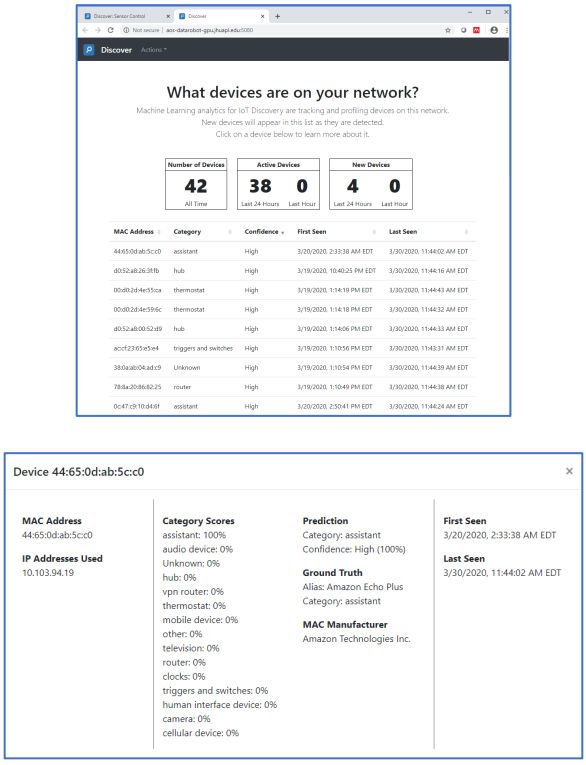

Fig. 3. Dashboard to Aggregate and Display Predictions

Therefore, the use of PCAP is key to providing useful cybersecurity in a network [9], [10]. However, gathering these data is hard because the storage capacity for full PCAP is expensive. In addition, being able to capture data coincident with a specific incident can be difficult because of the unpredictability of cyber events. To meet this challenge, we integrated a trigger-based PCAP capability that captures data based on passed-in configuration parameters. This gives the users the ability to perform offline deep packet inspection on data that are coincident with an event, and have relevant, insightful metadata stored with the data itself. To provide PCAP, we integrated *tcpdump* [11], a software program that allows the user to capture network packets being transmitted or received over a network to which a device is attached.

The PCAP capability is composed of two parts: an Edge Sensor Service that runs on each network sensor and a Sensor Management Service that runs in the cloud.

C. Network Mapping

To gain a sense of network situational awareness, an additional processing layer is needed that is responsible for network visualization, analytics, and general user interaction. This layer, termed IoTA for IoT Analytics, primarily taps into the pipeline after feature extraction is complete; however, it was designed to fuse information across several points in the pipeline (e.g., in the model training and execution phases).

The frontend renderer runs in a standard web browser for maximum flexibility. This enables practically any network-enabled computing device to interface with the processing pipeline across the enterprise firewall. Because of the limited computational resources available at the frontend, the backend processor essentially compresses the available data through context-based slicing and aggregation operations. To minimize operator cognitive load, the interface of the renderer is intentionally minimalistic because research has shown that cluttered interfaces greatly increase operator distraction and negatively impact their focus on other tasks [12].The renderer is divided into four main sections in which content changes are context sensitive. In this way, only the information pertinent to the task at hand is displayed. Fig. 4 shows the primary interface for IoTA, where the display is segmented into (1) data transformations and rendering controls, (2) network device listing with filtering, (3) micro network rendering, and (4) macro network rendering.

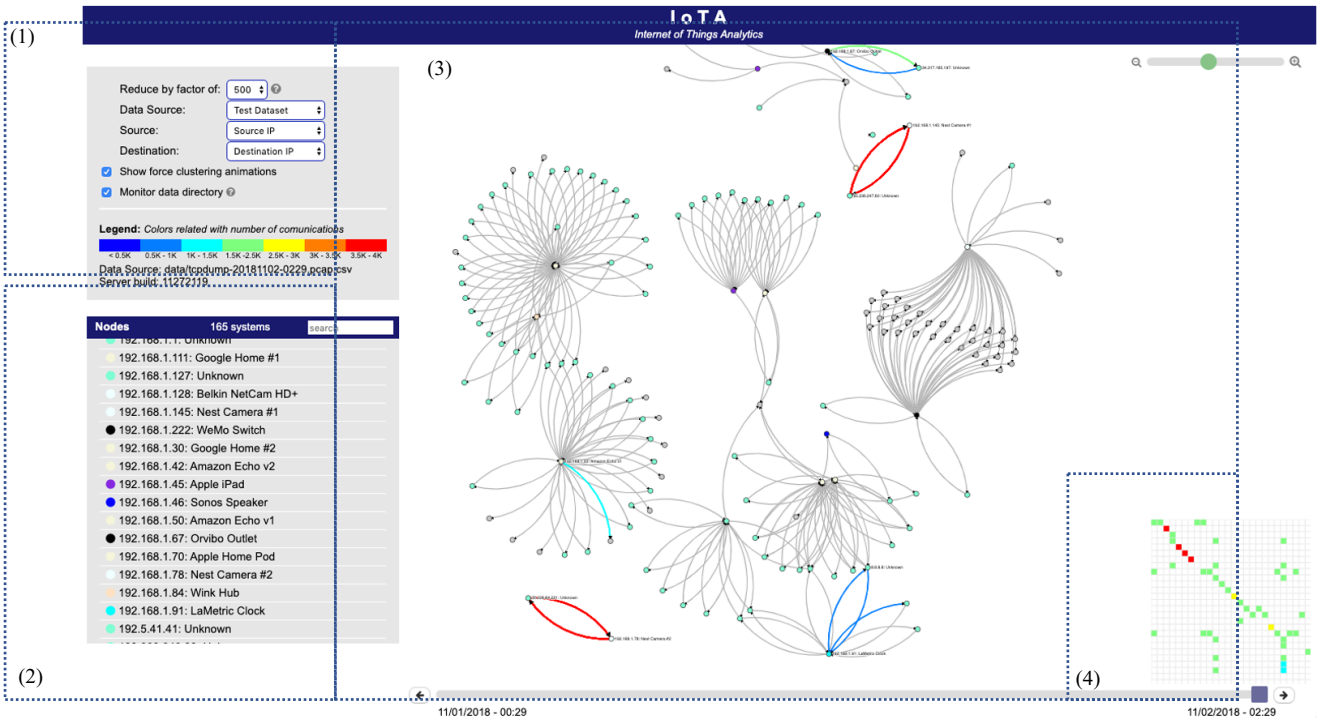

Fig. 4. IoTA Front-End Renderer with Fixed Context Areas: (1) data transformation and rendering, (2) network device listing and filtering, (3) micro network rendering, and (4) macro network rendering

TABLE III. PRELIMINARY EVALUATION AND COMPARISON OF INTENTS

Intent Request Name	Returns	Tools Needed To Perform Intent	Level of Difficulty To Perform the Intent		Time (voice assistant)
			Manual (1–5)	Voice (1–5)	
GetNetworkSummaryIntent	Number of total connected devices, number of devices active in the last 24 hours, and last device added	Wireshark	4	1	15 seconds
GetDevicesSummaryIntent	Number of connected devices and number of connected devices in each category	Wireshark	4	1	22 seconds
GetNewestDeviceIntent	The newest device added to the network, the time it was added, and the type of device it is	Wireshark	3	2	20 seconds
GetLatestDevicesIntent	List of devices added to the network after the specified time and the category breakdowns	Wireshark, router admin console	3	2	25 seconds
GetNetworkMapIntent	User-viewable network map with labeled nodes and edges representative of the network and activity on the network	*tcpdump*, python, tableau	5	3	40 seconds
GetPacketDataIntent	PCAP data from *tcpdump*	*tcpdump*	3	4	Variable based on *tcpdump* query

The primary visualization method used by IoTA is force-directed clustering with semi-radial layouts that minimize link crossings. After careful consideration and experimentation, this method proved to be the most robust in presenting device network information to highlight relationships between devices while minimizing clutter and occlusion. Fig. 5 presents a screenshot of this visual layout. The display supports zoom operations and an infinite canvas to scale well with large network collections, while using minimal resources on the client device. All node positions can be altered by simply clicking and dragging nodes across the canvas. During this operation, the layout manager continuously updates the force-directed layout through smooth animations. This aspect is important for preserving operating context as the network analyst interacts with the visualization [13]. Furthermore, node positions can be correlated to physical spatial coordinates and overlaid on spatial maps that can represent building schematics, geographic maps, or any other spatially relevant maps as illustrated in Fig. 5.

VII. Experimental Evaluation

Because the goal of this work was to reduce the cognitive load on the user and make the system owner/user more effective, we conducted a limited study that compared aspects of preforming commands associated with each utterance to evaluate the cognitive load and relative difference in difficulty performing a command manually versus using the voice assistant. Table III compares the cognitive load needed for a skill and a comparative measure of reduction of the cognitive load achieved by leveraging the voice assistant.

To formulate Table III, we asked a cyber subject matter expert and a typical home IoT device owner to quantitatively evaluate the level of difficulty to perform each of the skills manually and with the voice assistant using a five-level Likert scale. We also measured the time needed to perform the skill using the voice assistant. This timeframe was measured from the moment the user began speaking to invoke the Alexa skill to the moment when Alexa stopped speaking her response to the user. The measurement for each intent also included the time it took for the intent to reach out to the REST API to receive a network summary, although the skill caches that data within sessions for increased efficiency when multiple intents are invoked per session.

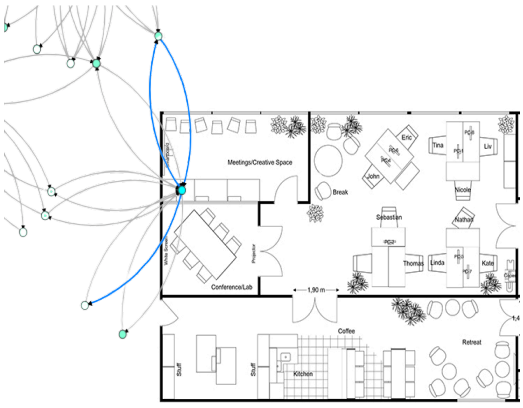

Fig. 5. Node Position Correlation with Spatial Mapping

From these preliminary results we see that the voice can lower the relative level of difficulty to perform a command. Specifically, intents that provide a summary of the networks were most impacted. As we continue to add skill and intents, we will expand the testing to more completely evaluate the effectiveness of the voice assistant.

VIII. Security Concerns

Smart Personal Assistants (SPAs) will change the way users interact with UFEs; however, they also present significant security and privacy issues. Keeping these applications safe without sacrificing the benefits of efficiency is key. Some security concerns of SPAs include synthesized speech, voice squatting, weak authentication, and profiling [14].

Smart Assistant technology lacks the ability to recognize whether the user is the legitimate owner of the device or an illegitimate user making a request to the device. This weakness makes SPAs unable to detect whether inaudible sounds or signals are requests to the device. As a result, radio signals or other sound waves from other technology can interfere with SPAs and cause the Smart Assistant technology to approve illegitimate requests.

Voice squatting is a method wherein a threat actor takes advantage of or abuses the way a skill or action is invoked. This threat can be activated when a user prompts a request, but receives a response that can be a potential threat to the owner. Weak Authentication

To ensure the safety of users and their networks, proper implementation of authentication controls is necessary for the usability of SPAs. Vulnerabilities regarding weak authentication include inadequate lockout implementations and the inability to determine one authenticated user from another. An example of weak authentication is when an SPA confuses an unauthenticated user as the owner of the device. For example, the daughter of an owner prompts an SPA to order a doll house; as a result, the SPA takes her order as a legitimate request to the device and orders the dollhouse without authenticating the identity of the user.

Beyond the issue of authorization, SPA users face the threat of profiling. Profiling is when data are collected about the user's personal information such as their interests, behaviors, and preferences. There are three main types of profiling: en-route profiling, profiling by third-party developers, and profiling by SPA providers. Attackers leverage *en-route profiling* to determine a user's presence during traffic analysis. The attacker can then use these techniques to conduct more serious threats. *Profiling by the third-party developers* involves the sharing of valuable personal and network information, resulting in malicious apps that combine various data the user has shared, thereby creating a complete profile of the user that may compromise the privacy of the SPA owner. In the *profiling by SPA providers* threat, the SPA makes compromises to uphold the user's privacy by collecting sensitive data such as the user's conversations, online search habits, and other information stored on the SPA. The SPA then may have access to personal data of the SPA user, which poses a security and privacy risk to the user [14].

Addressing security concerns is beyond this specific effort and therefore an area of focus for future work.

IX. Conclusions

This paper summarizes ongoing work toward integration of voice-assistant technology to lower the cognitive load placed on a user to monitor and maintain their smart environments by providing access to complex capabilities in a natural way. We developed a proof-of-concept cyber voice assistant as an Alexa Skill, which implements a set of Intents that allow users to access information on their networks and the associated devices. Together, these intents make up the most basic user commands, which provide a network owner awareness of their connected devices. We evaluated the effectiveness of the voice assistant by capturing qualitative perceptions on the ease of use of an intent and quantitatively capturing the time to perform an intent from a limited user base.

The advantage of integrating voice capability in a cyber personal assistant is that it can guide the user to ask the right questions the right way despite limited expertise. We understand that the ability to use natural language processing to support dialog using a specialized language in which the machine may be more proficient than the user will be a real challenge to achieve. Nonetheless, even with limited voice, the analytics that enable even rudimentary monitoring of home networks integrating voice is a potentially large step in securing the IoT as a whole.

References

[1] "What Is Ambient Computing, and How Will It Change Our Lives?" https://www.howtogeek.com/547655/what-is-ambient-computing-and-how-will-it-change-our-lives/ (accessed 16 October 2020).

[2] J. S. Chavis, L. A. Watkins, A. L. Buczak, and A. Rubin, "Connected Home Automated Security Monitor (CHASM): Protecting IoT Through Application of Machine Learning," in *10th Annual Computing and Communications Workshop and Conference (IEEE CCWC 2020)*, 2020, p. 6.

[3] [J. S. Chavis, A. Kunz, L. A. Watkins, A. Rubin, and A. L. Buczak, "A Capability for Autonomous IoT System Security : Pushing IoT Assurance to the Edge," in *2nd Annual Workshop on Assured Autonomous Systems (IEEE WAAS 2020)*, 2020, p. 6.

[4] S. N. Narayanan, A. Ganesan, K. Joshi, T. Oates, A. Joshi, and T. Finin, "Early detection of cybersecurity threats using collaborative cognition," in *Proceedings - 4th IEEE International Conference on Collaboration and Internet Computing, CIC 2018*, November 2018, pp. 354–363, doi: 10.1109/CIC.2018.00054.

[5] G. Bassett, "System and Method for Cyber Security Analysis and Human Behavior Prediction," US 2016/0205122 A1, 2016.

[6] R. K. Kodali, V. Jain, S. Bose, and L. Boppana, "IoT based smart security and home automation system," in *Proceeding - IEEE International Conference on Computing, Communication and Automation, ICCCA 2016*, January 2017, pp. 1286–1289, doi: 10.1109/CCAA.2016.7813916.

[7] S. Greene, H. Thapliyal, and D. Carpenter, "IoT-Based fall detection for smart home environments," in *Proceedings - 2016 IEEE International Symposium on Nanoelectronic and Information Systems, iNIS 2016*, January 2017, pp. 23–28, doi: 10.1109/iNIS.2016.017.

[8] "Technical Documentation | Amazon Developer Portal." https://developer.amazon.com/documentation (accessed 21 October 2020).

[9] "How important is PCAP for cyber defense? - Help Net Security." https://www.helpnetsecurity.com/2019/09/23/packet-capture/ (accessed 6 September 2020).

[10] D. Monahan, "Report Summary : Unlocking High Fidelity Security 2019," 2019. [Online]. Available: https://www.endace.com/ema-2019-research-report-download.pdf.

[11] "TCPDUMP/LIBPCAP public repository." https://www.tcpdump.org/ (accessed 15 October 2020).

[12] G. Ellis and A. Dix, "A taxonomy of clutter reduction for information visualisation," *IEEE Trans. Vis. Comput. Graph.*, vol. 13, no. 6, pp. 1216–1223, Nov. 2007, doi: 10.1109/TVCG.2007.70535.

[13] M. Steinberger, M. Waldner, M. Streit, A. Lex, and D. Schmalstieg, "Context-preserving visual links," *IEEE Trans. Vis. Comput. Graph.*, vol. 17, no. 12, pp. 2249–2258, 2011, doi: 10.1109/TVCG.2011.183.

[14] J. S. Edu, J. M. Such, and G. Suarez-Tangil, "Smart Home Personal Assistants: A Security and Privacy Review," March 2019 (accessed 16 October 2020) [Online]. Available: http://arxiv.org/abs/1903.05593.