HDMI-Watch: Smart Intrusion Detection System Against HDMI Attacks

Luis Puche Rondon, Leonardo Babun, Kemal Akkaya, *Member, IEEE*, and A. Selcuk Uluagac, *Member, IEEE*

***Abstract*—The High Definition Multimedia Interface (HDMI) is the backbone and the de-facto standard for Audio/Video connections between video-enabled devices. Today, nearly ten billion HDMI devices are used to distribute A/V signals in homes, offices, concert halls, and sporting events. An important component in HDMI is the Consumer Electronics Control (CEC) protocol, which allows HDMI devices to share an HDMI distribution to communicate and interact with each other. In this work, we identify security and privacy issues in HDMI networks by taping into CEC protocol vulnerabilities, using them to implement realistic proof-of-work attacks on HDMI distribution networks. We study how current insecure CEC protocol practices and carelessly implemented HDMI distributions may grant an adversary a novel attack surface for HDMI devices, otherwise thought to be unreachable through traditional network means. We first present HDMI-WALK, a novel attack vector, that can be used by an attacker to gain arbitrary control of HDMI devices and perform malicious analysis of devices, eavesdropping, Denial-of-Service attacks, targeted device attacks, and even facilitate other well-known existing attacks through HDMI. To defend against these new HDMI-based threats on a smart network system, we further propose HDMI-Watch, a novel intrusion detection system to detect unexpected CEC-based activity within an HDMI distribution. HDMI-WATCH operates as a standalone smart intrusion detection framework within an HDMI distribution, passively monitoring and thus imposing no additional overhead to CEC communication. To test HDMI-WATCH's performance, we evaluated our system in a realistic HDMI testbed with a variety of consumer HDMI-enabled devices. Our extensive evaluation results show that the proposed system achieves an average 98% accuracy in classifying unexpected CEC behavior and identifies attacks occurring without any form of modification required to existing devices in an HDMI distribution.**

***Index Terms*—Attacks, HDMI, CEC, intrusion detection system, consumer electronics control, CEC testbed.**

AUDIO/VIDEO(A/V) devices have always witnessed a wide range of adoption as consumer electronics. The High Definition Multimedia Interface (HDMI) is used primarily for the distribution of A/V signals and has become the de-facto standard for this purpose [1]. For instance, in many applications such as concert halls or sporting events, large displays are connected together via HDMI to show concert images and gameplay, creating HDMI distributions. Figure 1 shows possible use-cases of HDMI distributions. Indeed, as of this writing, there have been close to 10 billion HDMI devices distributed worldwide [2]. With the requirement to merge control and communication over a single connection, the HDMI Consumer Electronics Control (CEC) protocol was specified with the release of the HDMI v1.2a [3]. CEC provides control and communication between HDMI devices through HDMI cabling, providing audio and video capabilities to smart network systems. This has led many vendors to implement CEC features on their devices under different trade names, including: Anynet+ (Samsung), Aquos Link (Sharp), BRAVIA Link/Sync (Sony), CEC (Hitachi), CE-Link and Regza Link (Toshiba), SimpLink (LG), VIERA Link (Panasonic), EasyLink (Philips), Realink (Mitsubishi) [4]. The adoption of CEC has become a means of control for well-known household devices (e.g., Google Chromecast, Apple TV, Sony A/V Receivers, Televisions). This rapid adoption has made CEC into an ubiquitous protocol in many A/V installations and the adoption of CEC enabled devices in conference rooms, homes, offices, and secure facilities. Previous research works have shown that attackers are in search of new threat vectors against resource-limited smart devices and have been working on defense mechanisms against these threats [5]–[13]. Therefore, given the popularity and the proliferation of HDMI-based devices, their security is of utmost importance.

As HDMI distributions are non-traditional components of smart network systems, current security mechanisms do not offer any protection against to HDMI-based attacks. Thus, CEC remains as a widely-available, unprotected, and unexplored attack surface without mainstream user awareness. To demonstrate the viability of HDMI-based threats, we introduce HDMI-WALK, a novel attack vector that allows attackers to leverage insecure CEC protocol design and HDMI distribution networks to attack HDMI devices thought to be unreachable before through traditional means. Specifically, HDMI-WALK demonstrates that an attacker can use CEC communication to perform topology inference, Denial-of-Service (DoS) attacks, eavesdropping, targeted device attacks, and facilitate existing attacks even without traditional network access. To evaluate HDMI-WALK, we implemented and executed our attacks in a testbed containing commodity HDMI devices. Our results demonstrate that an attacker can use CEC as a threat vector to achieve arbitrary control of multiple devices.

Manuscript received April 29, 2020; revised July 19, 2020; accepted August 13, 2020. Date of publication August 28, 2020; date of current version September 16, 2021. This work was supported by the US National Science Foundation Awards: NSF-CAREER-CNS-1453647 and NSF-1663051. Recommended for acceptance by Dr. Enrico Natalizio. *(Corresponding author: Luis Puche Rondon.)*

The authors are with the Department of Electrical, Computer Engineering, Cyber Physical Systems Security Lab, Florida International University, Miami, FL 33199 USA (e-mail: lpuch002@fiu.edu; lbabu002@fiu.edu; kakkaya@fiu.edu; suluagac@fiu.edu).

Digital Object Identifier 10.1109/TNSE.2020.3020084

(a) Conference room with multiple displays and points of HDMI connection.

(b) Airport information kiosks with multiple displays and visible HDMI connections.

(c) Concert displays, where used may be HDMI for multiple displays.

(d) Sports bars where multiple displays present the same image from a single cable box.

Fig. 1. Possible examples of HDMI distribution use cases where HDMI-WALK could present a novel threat [14].

Moreover, to defend against these threats, we propose HDMI-WATCH; a novel passive smart intrusion detection system that protects HDMI distributions against CEC-based attacks. HDMI-WATCHoperates as a standalone framework in HDMI distributions, passively monitoring CEC traffic for CEC malicious behavior. HDMI-WATCH leverages CEC command types and machine learning techniques to detect unexpected activities in CEC communication. Additionally, HDMI-WATCH accounts for expected command lengths, associating CEC command types to their acceptable message lengths to improve detection. To test HDMI-WATCH performance, we performed an extensive set of evaluations in a realistic HDMI testbed with a variety of consumer HDMI-capable devices and against HDMI-WALK attacks. Our results show that HDMI-WATCH performance achieves an average accuracy and precision of 98%, detecting unexpected activities without any form of operational overhead or modification to HDMI devices.

Summary of Contributions: The contributions of this work are as follows:

- We introduce HDMI-WALK, a novel attack vector against HDMI distributions to demonstrate that arbitrary control of CEC devices is feasible for an attacker using this method.
- We implemented five unique attacks to HDMI distributions. Specifically, we performed topology inference, Denial-of-Service (DoS) attacks, eavesdropping, targeted device attacks, and facilitate existing attacks.
- We propose HDMI-WATCH, a novel intrusion detection system that protects HDMI distributions against CEC-based threats in HDMI distributions. HDMI-WATCH monitors CEC communication and detects unexpected CEC behavior occurring in an HDMI distribution.
- We evaluate HDMI-WATCH in a realistic HDMI testbed with a variety of consumer devices (e.g., Google Chromecast and Sharp Smart TV) achieving an average accuracy and precision of 98%.

Organization: The rest of the paper is organized as follows: Section I presents background information on HDMI device distributions and the CEC protocol. Section II covers the related work. Section III presents the assumptions, and HDMI threat model in our paper. In Section IV, we cover the architecture, attack implementations, and evaluate our findings for the novel HDMI-WALK attacks. In Section V, definitions and the HDMI-WATCH Architecture are presented. Section VI covers the HDMI-WATCH implementation. In Section VII, we evaluate the effectiveness of HDMI-WATCH detection against HDMI-WALK attacks and discuss our findings. Finally, we conclude this manuscript in Section VIII.

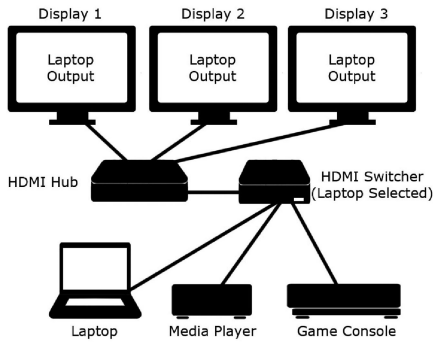

Fig. 2. Example HDMI device distribution network including three displays sharing the same source image (Laptop). Usually, in bars and conference rooms, displays are chained via the HDMI cables (Figure 1).

I. BACKGROUND

In this section, we present some necessary concepts about the Consumer Electronics Control (CEC) protocol and distributed HDMI-based device setups.

A. HDMI Distribution Networks

HDMI deployments are not limited to one-to-one connections. Similar to Ethernet networks, there are many devices which control the HDMI signal flow and distribute signal in a controlled and organized manner. For instance, in Figure 2, the user maintains the same visual image over three displays and switches between three source devices. This figure also shows the laptop selected as the active source over multiple displays. Depending on the device setup, there is a distribution of CEC through the same connection. We note the following components in an HDMI distribution and will refer to them during our work.

Displays: Any device with a primary purpose of being an end-display such as a television or a projector.

Hubs/Splitters: Any device which primarily allows multiple video signals to be split to various displays from a single video input without switching.

Switches: Any device with a primary purpose of allowing various source device inputs to one or more display device

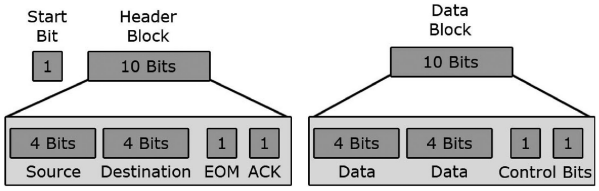

Fig. 3. The CEC stack and structure as used in HDMI.

outputs. They also perform switching between these sources to a different output(s).

Source Devices: Any device which is primarily an HDMI output-only devices such as a Chromecast or a laptop.

B. The Consumer Electronics Control (CEC) Protocol

CEC was developed to enable interoperability between HDMI devices, with full the specification in 2005 [3]. CEC signals are carried through Pin 13 as part of the HDMI interface [15]. The communications in CEC are divided into 10-bit blocks that include a header, opcode, and data blocks. The flow of information is dictated by the header, the first eight bits note the source and destination. Message Destination may refer to a specific device by logical address or broadcast. This broadcast functionality is especially exploited by HDMI-WALKattacks. Figure 3 shows how the CEC header allows for 16 unique IDs (4 bits). IDs 0-E specify device addresses while the last logical address (F) is reserved for broadcast within the HDMI distribution. This logical address assignment usually follows certain device-type guidelines. For example, displays are usually assigned to the logical address (0), and additional displays self assign to "free use" (E).

II. Related Work

There has been some research in compromising A/V devices through a variety of attacks. Work from Zhang *et al.* presented a security overview on connected devices and noted common vulnerabilities such as weak authentication, overprivilege, and implementation flaws in connected devices [16]. Within the scope of Smart TVs, Oren & Keromytis describe a method of compromising connected Smart TVs through Hybrid Broadcast-Broadband Television (HbbTV) and web-based code injections [17]. Related work from Niemietz *et al.* on Smart TVs explores the attacks on Smart TVs through app-based approach [18]. This work centers on TV embedded applications, and the security flaws which may come from vendor-specific apps. On the other hand, research related to HDMI systems and their security issues has remained a relatively uninvestigated avenue or not systematically investigated by the research community. The most relevant work in HDMI systems is a 2012 work published by NCC Group, which focused on vulnerabilities with fuzzing [19]. Similarly, Smith presented CEC as an avenue of attacks through fuzzing [20]. And, further work presented CECSTeR as a fuzzing tool [21].

Our work differs from these works as follows: We introduce a novel attack method called HDMI-WALK *to HDMI devices. Our scope is entirely through CEC as the main vector of attack and does not rely on any custom applications, software vulnerabilities, fuzzing, buffer-overflows, vendor-specific attacks, or traditional network connectivity. We focus on the exploitation of the CEC protocol in both local and remote attacks. We demonstrate proof-of-concept implementations of five different types of attacks; specifically, (1) malicious device Scanning, (2) eavesdropping, (3) facilitation of attacks(e.g,. WPA Handshake theft), (4) information theft, and (5) denial of service through HDMI.*

III. Problem, Assumptions, and Threat Model

In this section, we provide a summary of assumptions, definitions, and the threat model for HDMI-WALK-based attacks.

A. Problem Scope

This work denotes an HDMI distribution network within a conference room which may be used for confidential presentations. The topology of this distribution network includes common HDMI distribution equipment such as switches and hubs as well as HDMI devices such as displays and sources. The attacker is an invited guest presenter Mallory, who has a small amount of time to prepare in the conference room without any supervision. Mallory either compromises an existing HDMI device through malware or hides a malicious HDMI-capable device within the distribution (e.g., connected behind a television). Mallory connects her own laptop to auxiliary ports on the podium prior and during the presentation and perpetrates the HDMI-WALK attacks. After presenting, Mallory leaves. Sometime after her departure, further security policies are enacted, and unsupervised access to the conference room is disallowed to visitors. Mallory's only avenue of attack is to access her hidden device indirectly, locally or remotely.

Attack Mode 1 (Local Communication): Mallory only has local access when connecting directly to the HDMI distribution network as a presenter. This case is independent of any form of network access; it relies on Mallory's ability to connect to the auxiliary connection on the conference room podium. Local communication from her laptop through the HDMI distribution with HDMI-WALK and to the hidden or compromised device.

Attack Mode 2 (Remote Communication): In this case, Mallory has found an open guest network connection during her first visit or later gained unauthorized Internet access. This allows Mallory to enable remote access to her hidden device. Furthermore, this allows Mallory to perform specific attacks.

B. Assumptions

To perform the HDMI-WALK attacks, we have the following assumptions.

CEC Propagation: This work assumes full CEC protocol propagation over the distribution of HDMI devices. Some devices tested had no function to disable CEC propagation, even if CEC control was disabled. In testing performed on

devices with multiple HDMI ports, we found 80% of devices provided some form of propagation.

CEC Control: We assume CEC control is active on connected devices in the distribution. This is a realistic scenario, as we found that in all CEC-capable devices tested, CEC functionality was enabled by default. We also observed that many devices revert to default settings after a firmware update.

Access to HDMI Components: We also assume that Mallory has access to some HDMI components (or endpoints) in the distribution. This is a realistic assumption as A/V components are often not as secure as networking components. Display inputs and outputs are often visible and available to presenters. Presenters are often given enough time to prepare and free access to A/V equipment in a conference room without supervision or suspicion. In some cases, we have found displays outside conference rooms (connected to the main distribution) which could act as an access point to HDMI equipment inside a conference room. The location of these components (e.g., wall mounts, cabinets, podiums, etc.) also makes it trivial for an attacker to connect and hide their own physical devices to exposed HDMI components. Additionally, exposed, unsupervised HDMI-capable devices allow an attacker to install a malicious app (e.g., Android-based) or a storage device (e.g., USB storage, SD card) with altered firmware and compromise the integrity of the software of this device.

C. Threat Model

HDMI-WALK assumes the following five threats as part of the threat model and the feasibility of remote attacks.

Threat 1: Malicious CEC Scanning: This threat considers the malicious use of *scanning* features through CEC and exposed HDMI ports to gather information about the connected devices. For instance, Mallory can create a topology of available HDMI devices to control and use this information to perform further attacks.

Threat 2: Eavesdropping: In this threat, Mallory is not present but actively eavesdrops on CEC communication through an implanted device.

Threat 3: Facilitation of attacks: This threat eliminates time and physical access limitations in network attacks. HDMI-WALKfacilitates many of these attacks so that they become more viable or more difficult to detect. For example, Mallory installs a device to passively capture WPA handshakes, avoid detection, and control through CEC remotely.

Threat 4: Information Theft: This threat considers information theft as a form of data transfer that Mallory may find valuable. For example, information about available HDMI devices or wireless handshake data which would enable future attacks.

Threat 5: Denial of Service: This threat considers Denial-of-Service attacks where Mallory disrupts the availability of a system through an HDMI connection. These attacks may be targeted to a specific device or broadcast to multiple devices. For example, Mallory prevents the use of a television through the repeated broadcast of HDMI control commands.

Feasibility of Remote Attacks: An attacker may perform remote attacks on HDMI distributions in several different ways. First, Internet-connected devices with an HDMI connection may act on an attacker's behalf to execute HDMI-WALKattacks. This can be accomplished through the use of malicious applications (e.g., Android-based apps, drivers), or a compromised operating system on the device. For instance, privileged malware applications in an Android-based A/V device could make use of the `HdmiControlManager` and `HdmiControlService` to transmit and receive arbitrary messages [22]. Additionally, without compromising devices, guest wireless networks are readily available in many locations. As such, it may be trivial for an attacker to find a guest network and connect an external malicious device. Further, cellular interfaces are also available and where a guest network is not available, an attacker may use cellular to enable remote portions of HDMI-WALK attacks. With Internet access from guest networks or cellular, the attacker can now remotely access this device and perform the HDMI-WALK attacks.

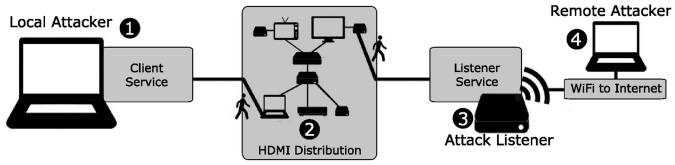

Fig. 4. General end-to-end implementation for HDMI-WALK-based attacks.

Note that this work does not consider attacks that focus entirely on IP networks; data injection attacks through CEC such as buffer overflows over CEC or setting manipulation attacks. Similarly, other protocols, such as USB or Bluetooth are entirely outside the scope of this paper.

IV. HDMI-WALK

In this section, we present an overview of the HDMI-WALK-based attacks. The complete details of HDMI-walk can be found in a previously published work [14]. Figure 4 depicts the general architecture of HDMI-WALK with four main components: *local attacker, HDMI Distribution, attack listener, and remote attacker.*

A. HDMI-Walk End-to-End Implementation

The first component of HDMI-Walk is the Local Attacker which runs the Client Service in their local machine. This local hardware is temporarily connected to the HDMI distribution. The client service contains any required modules for communication to the listener and facilitates the attacks through HDMI-WALK (①). The physical communication from the local attacker to the HDMI distribution allows for attack mode 1. The second part is the HDMI Distribution, which is the core of our attacks and allows for end-to-end communication between devices through HDMI as a medium. The user may scan the distribution for addressed CEC devices, as well as communicate bidirectionally with other devices (②). The third part of the architecture involves the Attack Listener. The attack listener is the physical attacker device and hosts the Listener Service. The listener service includes all the required modules for HDMI-WALK communication and listener-run

TABLE I
HARDWARE AND SOFTWARE USAGE

Hardware	Software
Sharp Smart TV.	Pulse Eight LibCEC 4.0.2
Samsung UN26EH4000F	Python 3.6.1
Monoprice Blackbird 3x1 HDMI Switch	Aircrack-ng 1.2-rc4
Wyrestorm - 1x4 HDMI 1.3b Splitter	Eclipse IDE
Chromecast NC2-6A5	PyAudio v0.2.11
Sony STR-ZA2100ES	Jersey JAX-RS
Raspberry 3 Model B x2	Raspbian Version 9
TP-Link TL-WN722N V1 Adapter	Swagger.io
Motorola G5 Plus Phone	Java 1.8
TP-Link TL-WR841N Router	AWS Elastic Beanstalk

TABLE II
ATTACK 1–INFORMATION GATHERED VIA HDMI-WALK

Info	Addr 00	Addr 01	Addr 02	Addr 04	Addr 05
P. Addr	0.0.0.0	f.f.f.f	4.0.0.0	3.0.0.0	1.0.0.0
Active	No	Yes	No	No	No
Vendor	Unk	Unk	Pulse-Eight	Google	Sony
OSD Str	TV	RPI	CECTestr	Chromecast	STR-ZA2100
CEC Ver	1.4	1.3a	1.4	1.4	1.4
Pow Status	ON	ON	ON	ON	Standby
Language	Eng.	Eng.	Eng.	Unk	Unk

attacks. This service also includes a remote access module to enable communication to the remote client if a connection is available (③). Finally, we have the Remote Attacker, which communicates directly through a remote connection to the attack listener if available for attack Mode 2 (④).

B. The Implementation of HDMI-WALK

In order to ensure the attacks are implemented in a realistic HDMI environment, we created a CEC capable testbed with standard and widely available commodity HDMI devices presented in Table I. Here we included two displays, an HDMI switcher, an HDMI hub, a source and the attacker devices. We utilized LibCEC, an open-source CEC implementation [23]. This library provides Python modules which we used to create both the client and the listener services. Due to readily available CEC support in Raspberry Pi v3 devices, we used two Pis, one as the listener and one as the local client to perform the attacks and evaluations. To test WiFi (handshake) and remote attacks, we created a wireless network.

C. Attacks

In this subsection, we realized HDMI-WALK attacks and discuss their implications.

Attack 1: Topology Inference Attack (Local and Remote) This attack is a demonstration of Threat 1 (Malicious CEC Scanning) possible through CEC in online and offline scenarios. We use the HDMI-Walk architecture to move through the distribution and gather information about every device available with malicious intent. This attack can be executed through the local or remote client. As seen in Table II, we gather information such as the device logical/physical address, active source state, Vendor name, CEC Version, OSD Name, and power status. With this information, an attacker may as well infer usage from the power state of the equipment.

Attack 2: CEC-Based Eavesdropping (Local) We perform this attack to demonstrate Threat 2 (Eavesdropping) and Threat 4 (Information Theft). In this local attack, an attacker has access only to the HDMI port for communication with the listener device. The attacker walks the HDMI distribution and forwards messages to the listener to activate and record audio. This audio data is stored locally in the listener device. The audio data is then transferred to the client at a later date. This further opens the possibility to a listener which could await keywords such as "password" passively or use voice-to-text technology to transfer days of conversations to an adversary.

Attack 3: WPA/WPA2 Handshake Theft (Local) This attack was specified in order to demonstrate the concepts of Threat 3 (Facilitation of Attacks) and Threat 4 (Information Theft). In this local attack, the attacker uses HDMI-WALK to facilitate WPA/WPA2 handshake capture and prevent detection by a security system in place. In traditional handshake theft attacks, an attacker has to wait for a handshake to occur, this can take an indefinite amount of time as the WPA handshake is only transferred in specific cases [24]. This raises the issue that forced de-authentication may be detected through a network scanner such as Wireshark or through more complex IDS [25]. In this attack, we facilitate such a threat through the removal of time constraints.

Attack 4: Targeted Device Attack (Local and Remote) This attack was developed to demonstrate Threat 5 (Denial of Service) through arbitrary sniffing and control of a device. In this attack, the attacker uses functionality from the Python-based listener service to target a specific device in the HDMI distribution. No matter which method of powering on, the attack could not be avoided as the targeted display would shut off before it fully powered up. Additionally, this attack may prove difficult to detect as it may be mistaken for a malfunctioning display.

Attack 5: Display Broadcast DoS (Local and Remote) We developed this attack to demonstrate Threat 5 (Denial of Service) through broadcast functionality. This attack abuses the broadcast function in CEC to cause a DoS condition in any display within a given HDMI distribution. This attack targets displays by producing standard CEC commands for source and input control. This attack first powered on the display if it was powered off then began rapid input change over all inputs on the display, rendering the display unusable.

Summary and Findings: During testing of HDMI-WALK attacks, we identified a vendor-specific vulnerability. HDMI-WALKcan identify specific device information to develop further attacks. We have proven arbitrary control over HDMI devices which could be used to an attacker's advantage. Also, we enabled control of the TV volume and Amplifier volume with devices in our testbed. This control is completely feasible in an HDMI distribution with the concepts of HDMI-WALK. We find these attacks critical as they occur over a medium without any form of security mechanisms or existing techniques for mitigation. Via Attack 4, we found that the input change control could become a viable form of a visual attack. With these functions, display input changes could be used to trigger seizures (e.g., television epilepsy) with the rapid

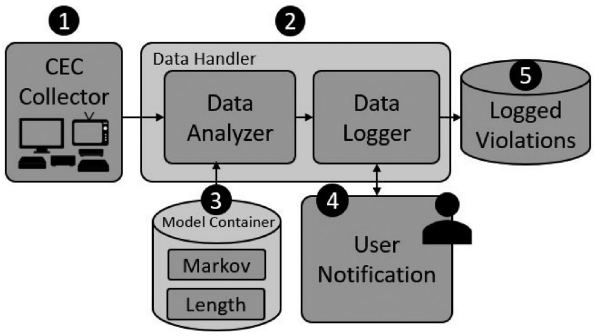

Fig. 5. Architecture of HDMI-WATCH. Each module numbered.

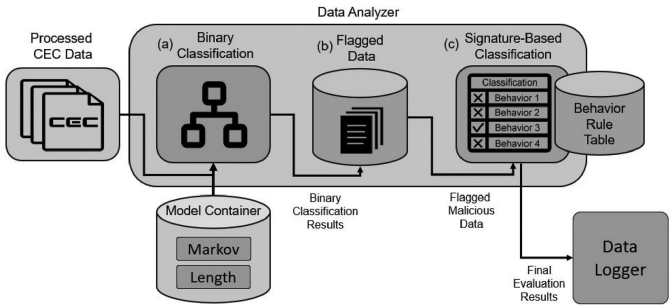

Fig. 6. HDMI-WATCH classification process.

flickering of a display switching between inputs [26]. We also consider volume control to an Amplifier device. A remote attacker with the control of a distribution can easily adjust the volume of devices with CEC commands. Extended playback at high volumes is known to damage sound equipment [27]. An implementation of Attack 1 would first allow an attacker to infer room occupancy via power state. Combining this with Attack 4, the attacker could peak the volume output in a room when nobody is present and cause gradual damage to a sound system that supports HDMI, which cause a notable financial cost to the user.

V. HDMI-WATCH ARCHITECTURE

To address CEC-based threats, we introduce HDMI-WATCH, a passive, easily configured intrusion detection system. In this section, we detail the different modules of the HDMI-WATCHarchitecture.

A. *HDMI-WATCH Overview*

The proposed architecture of HDMI-WATCH is divided into five different modules, as seen in Figure 5. The first module is the *CEC collector* which captures CEC packets from an HDMI distribution and supplies them to the data handler ①. The data handler evaluates and logs the incoming CEC traffic utilizing the two sub-modules: the *data analyzer* and the *data logger* ②. The data analyzer is a sub-module used for classification, applying a machine learning model over incoming CEC traffic to perform both binary (malicious or benign) and signature-based (scanning, data transfer, power change, or input control abuse) classification. The *data logger* is then used to forward this processed data (classification results, and violations) to both the logged violations and the user notification module. The *model container* stores a machine learning model of expected communication behavior for CEC-enabled devices, which is used by the data analyzer sub-module to evaluate CEC data ③. Any incoming CEC data flagged as a violation by the data analyzer is forwarded to the data logger. The data logger sends the flagged violations to the user notification module which notifies the user on unexpected CEC activity which occurs over the distribution ④. Finally, the logged violations module stores all the flagged violations and relevant data found by HDMI-WATCH ⑤. The logged information may be queried later for reference, or further analysis.

B. *CEC Collector*

The CEC collector provides HDMI-WATCH the CEC traffic necessary to operate over an HDMI distribution. Due to the design of CEC as a bus architecture, a single point of connection allows HDMI-WATCH to monitor all active CEC communication from devices within the same HDMI distribution. Additionally, the CEC collector parses the raw data received into a format that other modules of the HDMI-WATCH architecture can interpret. Formatted messages out of the collector include all information necessary for evaluation: timestamp, CEC command type, and CEC packet length. If needed for logging purposes, the entire CEC packet is also included. The command type is the main feature and later used by the Markov model, while the length model uses the packet length during the binary classification stage.

C. *Data Handler*

The data handler acts as the evaluation stage for HDMI-WATCH. We divide its functionality into two main sub-modules; the Data Analyzer and the Data Logger.

1) Data Analyzer: The data analyzer is the core of HDMI-WATCH, making the distinction on whether incoming CEC data is from malicious or benign activities. Additionally, the data analyzer performs signature-based classification of malicious activity into different types of attack behaviors (see Section IV). We refer to Figure 6 for the classification process performed by the data analyzer. The first step in HDMI-WATCH classification is binary classification, which attempts to classify CEC activity as benign or malicious. To perform binary classification, the data analyzer refers to the model container which contains the Markov and length model used for binary classification. Any violation found from the incoming data by the models is cached locally by the data analyzer into lists as flagged data. The data analyzer can determine the number of violations for the number of messages received. If the number of flagged violations exceeds the detection threshold, the activity is deemed malicious. Once a malicious activity has been identified, HDMI-WATCH begins signature-based classification, inferring the specific type of activity in the flagged data (scanning, data transfer, power control, or input control abuse). To perform signature-based classification, HDMI-WATCH uses a behavior rule table where command types associated with different types of activities. For instance, malicious activity found during the binary classification stage

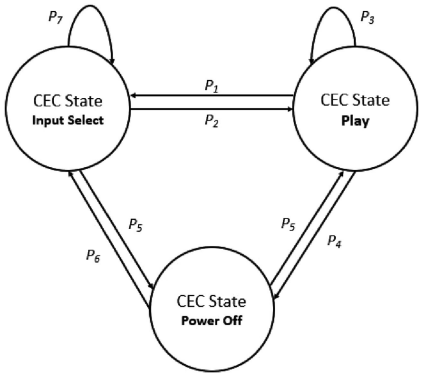

Fig. 7. Three command types are shown as states in a CEC Markov Model. Probabilities given (P_1 to P_x) as the possibility one command type following another command type.

will be labeled as power change abuse if most of the violations in the flagged data are power control commands. The resulting evaluations from both binary and signature-based classification along with the flagged data are then passed to the data logger module for logging and to the user notification module.

Binary Classification. The data analyzer uses binary classification in HDMI-WATCH to infer if activities within the HDMI distribution are benign or malicious. HDMI-WATCH is a flexible system with adjustments to improve classification accuracy and better fit the likely heterogeneous HDMI distributions where HDMI-WATCH is deployed. Thus, HDMI-WATCH employs a configurable *violation threshold* as the acceptable number of violations for a number of received CEC messages. In addition, the sample of received CEC messages may also be adjusted to improve the quality of classification. Binary classification in HDMI-WATCH classifies sets of violations as malicious CEC activity if the number of violations exceeds the violation threshold for a number of received messages.

Signature-Based Classification. HDMI-WATCH uses the data analyzer to perform signature-based classification of malicious CEC behavior. We create a set of rules for each behavior (scanning, data transfer, power control, or input control abuse) and classify violation sets as a second step to the initial binary classification. HDMI-WATCH uses a predefined ruleset to infer the type of behavior occurring in a malicious set of data. These behaviors were selected as they are related to HDMI-WALK attack behaviors. It is possible to configure HDMI-WATCH to classify other behaviors with additional rules. We narrow down to four different unauthorized behaviors from five attacks using command types:

- Scanning. Scanning is heavily associated with Attack 1. Scanning is required for an attacker to gather information about a CEC distribution and devices.
- Data Transfer. Attempting file transfer over a CEC bus is not standard CEC operation and is strongly associated with Attack 2 and 3. To the best of our knowledge, data transfer capabilities through CEC are not commonly used by any manufacturer.
- Power Control. While power control commands may be issued by devices in a benign manner. We can associate the unexpected use of power control to Attack 4.
- Input Control. Input change may be issued by devices in standard operation. However, abuse of these commands is associated with Attack 5.

2) Data Logger: The data logger module receives evaluation results, violations, and relevant data found during the data analyzer stage. The data logger serves a storage endpoint to process these results into a database-compatible format. Additionally to formatting, the logger is responsible for storing this data into the logged violations database. This module, essentially allows for the users of HDMI-WATCH to refer to past events and view logs on activity which may have been deemed suspicious.

D. Model Container

The model container stores the HDMI-WATCH models used to evaluate CEC traffic in an HDMI distribution. This module uses the command type of CEC packets as its main feature, with length as a secondary attribute and is used by the data analyzer module to predict unexpected behaviors and also specific types of malicious activities. Specifically, the model container is divided into two parts, the *Markov model* and the *length model*. HDMI-WATCHu ses the Markov model to determine if a command type in CEC traffic is expected after the last message received. Additionally, HDMI-WATCH uses the length model to determine if the length of a CEC packet matches expected lengths for the command type. Packets which violate these models are flagged as *violations*.

1) Markov Model: The Markov model is the core module of the model container and is used by the data analyzer for CEC data evaluation. Since vendors and attackers have the ability to create CEC packets of any command type, HDMI-WATCHmust consider all possible command types as states. In effect, this yields to a total of 256 (00-FF) states for the HDMI-WATCH machine learning model. The data analyzer refers to this model to analyze CEC traffic and infer if received command type follows expected behavior. Any CEC packet which does not follow expected behavior is marked as a *violation* and acts against the *detection threshold*, causing CEC activity to be deemed malicious after being reached.

Mathematical Foundations: We build a Markov-Chain-based model to perform binary classification of the CEC behavior within the HDMI distribution. With the Markov chain model, we evaluate the probability of changes in CEC command behavior over time. The Markov chain model serves as the core classification mechanism for HDMI-WATCH.

We represent the probabilistic condition of CEC state changes in Equation 1 where X_t denotes the CEC command as a Markov chain state at time *t*. Figure 7 illustrates a simplified version of a CEC behavior with three command types defining Markov Chain states and probabilities (P_1,...,P_7) of transitions between these states. For instance, P_5 being the probability of a CEC "Power Off" command being sent after an "Input Select" message.

$$\begin{aligned} P(X_{t+1} = x | X_1 = x_1, X_2 = x_2 \ldots, X_t = x_t) \\ = P(X_{t+1} = x | X_t = X_t) \\ when, P(X_1 = x_1 . X_2 = x_2 \ldots, X_t = x_t) > 0. \end{aligned} \quad (1)$$

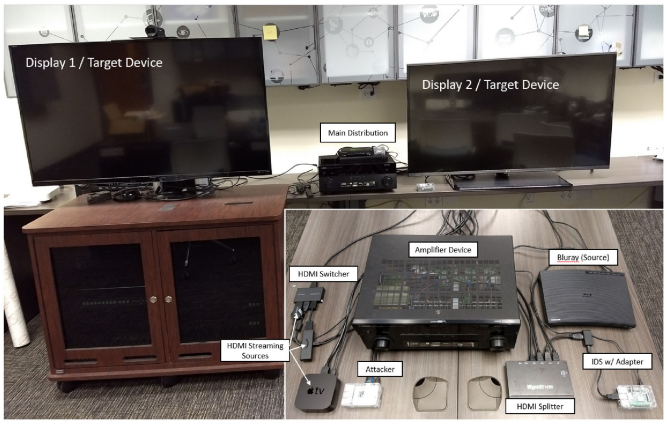

Fig. 8. HDMI testbed, including two targeted displays, an attacker device, and the HDMI-WATCH device within the same distribution.

In HDMI-WATCH, we observe the commands transmitted by a set of devices over time. Let us assume that C denotes a set which represents a set which contains all transmitted command types over a CEC bus, such that $C = \{C_1, C_2, C_3,...,C_n\}$, where $C_1, C_2, C_3,...,C_n$ = any CEC standard command value type (e.g., 36, 8f, 00). For the function of time, t, we consider the command during time t as the state in our model. If we consider the number of total number of unique CEC commands, there are a total of 256 possible commands which must be considered.

If we assume that the distribution's states are $X_0, X_1,..., X_T$. at a given time of $t = 0, 1,..., T$. We can then represent the transition probability P_{ij} as shown below:

$$P_{ij} = N_{ij}/N_i,$$

where N_{ij} is the number of transitions from X_t to X_{t+1}, with $X_t = i$ and $X_{t+1} = j$; N_i being the total number of transitions from state i. Initial probability distribution of this Markov Chain is represented as follows:

$$Q = [q_1, q_2, q_3, \ldots, q_m],$$

this model denotes q_m as the probability that the model is in state m at time 0. The probability of observing a sequence of states (X_T) at a given time T, can be computed as shown:

$$P(X_1, X_2, \ldots, X_t) = q_{x1} \prod_{2}^{T} P_{X_{t-1}X_t}.$$

For HDMI-WATCH, instead of predicting future states, we determine the probability of a transition between states at a given time. We train our Markov Model with a dataset collected from active CEC devices and create a prediction model to calculate the probability accordingly. Any packet with a probability of zero from a previous state is deemed unexpected and therefore a violation of the Markov model.

2) Length Model: HDMI-WATCH uses a length model to determine if a command received is of the expected length. This model contains mappings of associated lengths to command types found during the training phase of HDMI-WATCH, taking advantage of the association between length and type of CEC packets in CEC communication. For instance, messages with the command type "36" shut off should not contain additional data and the length model serves as a method to detect discrepancies in cases such as these. The data analyzer refers to the length model in the binary classification stage to determine if CEC packets are *violations* of the expected length model. Any CEC packet which is deemed a violation of the length model acts against the *detection threshold* and may cause incoming data to be classified as malicious.

E. User Notification

The user notification module is the primary form of notification to a network administrator using HDMI-WATCH. After CEC traffic is analyzed, the user notification module details any violations to an administrator. This administrator is shown the complete set of packets including packet source, packet destination, command type, data, length, timestamp, violation type, and possible attack type. Ideally, an administrator receives a text message or an email when a violation or a set of violations occur. This information is also archived by logged violations module of HDMI-WATCH.

F. Logged Violations

The logged violations module acts as a storage database for any violations found during HDMI-WATCH monitoring. The administrator queries this module to view a history of violations and react accordingly to any threat. This enables proper mitigation by the administrator and allows for simple tracking of past suspicious behavior. Logged violations only include commands deemed malicious, as well as evaluations of what type of action is occurring with a set of these violations based on HDMI-WALK attacks. This acts as the final stage of HDMI-WATCHand as a point of reference for any network administrator authorized to view HDMI-WATCH logging.

VI. IMPLEMENTATION OF HDMI-WATCH

To implement HDMI-WATCH's necessary modules, we used a modified version of Pulse-Eight's LibCEC library [23] and Python extensions [28]. All the software used is open source and freely available online. We refer to Figure 8 as the testing environment for HDMI-WATCH. Our testing environment uses several commodity HDMI devices (A/V receiver, switcher, source devices, displays), an attacker client, the attack listener, and a device hosting HDMI-WATCH. We assume the attacker executes all the HDMI-WALK attacks as covered in Section IV of this manuscript, receiving execution commands from an attacker with a connection to the attacking device.

A. CEC Collector

Implementing the CEC collector required modification to the original LibCEC library. The original LibCEC code filters some incoming CEC packets, which was not desirable for HDMI-WATCH. For the CEC collector, we removed all forms of filtering from the original LibCEC source code. The removal of such allowed HDMI-WATCH to monitor all ongoing

TABLE III
SIGNATURE-BASED EVALUATION BY BEHAVIOR TYPE RULE SET. MALICIOUS ACTIVITY IS TAGGED AS THE CASE WITH THE MOST OCCURRENCES BY COMMAND TYPE

Command	Behavior	Characteristics
83	Scanning	Scanning, eavesdropping behavior
00	Data Transfer	Multiple occurrences data transfer
36	Power Control	Possible abuse in power control
20	Input Control	Possible input control abuse

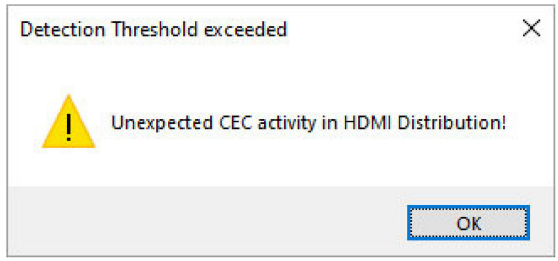

Fig. 9. User notification shown when detection threshold is exceeded in HDMI-WATCH.

CEC communication in an HDMI distribution, no matter the source or destination. Live CEC data was received using our modified LibCEC library. The received data was then pre-processed with Python into a comma-delimited string that included the CEC packet's timestamp, command type, length, and the CEC packet itself. Once data was formatted, data was passed to the data handler module.

B. Data Handler

Both sub-modules for the data handler were instrumented using Python libraries.

1) Data Analyzer: The data analyzer was implemented as a Python-based sub-module operating within HDMI-WATCH. Internally, the data analyzer fetches each attribute from the received data (timestamp, packet, and length). The data analyzer takes note of the command type of each incoming packet. For instance, the CEC packet "02:89:01" was marked as command type "89" and length of "8". This sub-module then queries the model container in search of violations in incoming data. Incoming CEC packets deemed suspicious are flagged and logged. This flagged data was then used to infer the type of CEC activity occurring in an HDMI distribution given the specific rules from Table III.

2) Data Logger: The data logger uses standard Python I/O libraries to convert data into a comma-delimited format. This information was then exported as an external document containing all the flagged data, classification results, and relevant information.

C. Model Container

The model container was comprised of the two models used by HDMI-WATCH for classification purposes, the *Markov model* and the *length model*. In this subsection, we overview the Markov model, the length model, and the data process to implement these models.

1) Markov Model: The Markov model was stored as a map of key pairs and probabilities created from the training data. This model was a comma-delimited document that was imported into the HDMI-WATCH at runtime. For instance, an association of command 9D to 90 was stored as follows "9D-90,0.5". The first value expresses a command type pair (90 received after 9D). The second value was the probability of the command type pair occurs in this trained model.

2) Length Model: The length of each packet was associated with the command type gathered during the training phase. The length model was stored as a serialized Python dictionary derived from the training data and functions as a complementary feature to the Markov model. The length model stored the associated lengths in key-list pairs to their respective command type. For instance, if the command type "87" has the expected length as 11 and 14, this was stored as "'87': [14, 11]" in the length model.

3) Data Collection and Training: To train the HDMI-WATCHclassifier, we collected daily data usage from an HDMI environment. We performed normal operation of the HDMI equipment as defined in Section V. Normal operations involved using the environment for music, movies, videos, and any manner consistent with an HDMI distribution system. Data from the environment was collected over the span of two weeks, where users performed normal operations using the testbed. This collected data was then used as the training data for the model container module using a python-based software designed for HDMI-WATCH. In total, we collected 61,765 benign CEC communication packets during the standard operations of the HDMI testbed as the training data. To train the model, we followed an unsupervised learning approach that did not require labeled data for training. We found that the unsupervised learning approach with the Markov model provided more flexibility for HDMI-WATCH and required fewer system resources during the learning process.

D. User Notification

The user notification module in HDMI-WATCH was implemented to notify the user when the detection threshold was exceeded and thus, malicious activity was detected. The current implementation functioned as a popup notification on the local machine using the Python module Tkinter [28]. Figure 9 shows the notification which occurs when violations exceed the detection threshold. Implementations of email, texting and other notifications are a very straightforward task which can be very easily implemented.

E. Logged Violations

Violations are logged directly into a comma-delimited document stored in the device hosting the HDMI-WATCH system. This file is created using standard Python I/O libraries, appending messages to the end of the output document. This comma-delimited document contains all relevant information to the violation: timestamp, data, packet length, violation type, and possible attack-type of each violation.

VII. Performance Evaluation

In this section, we evaluate our solution against HDMI-Walkattacks. Specifically, we aim to answer the following research questions:

RQ1: Threshold Evaluation. How does HDMI-Watch's detection threshold affect the binary classification results in HDMI-Watch?

(Section VII-C1)

RQ2: Malicious Activity Type. How effective is HDMI-Watchin classifying between different types of malicious behaviors? (Section VII-C2)

A. Attack Implementation

Based on previously mentioned HDMI-Walk attacks, we perform the attacks as per Section IV specifications. We assume that the attacker perpetrates the attacks following a Normal distribution. As such, we assume there is an equal probability that any attack will occur at any given time and a valid model to emulate a single attack executed at the time, which we consider a realistic approach to emulate HDMI-Walkattacks. Considering *t=[0,T]* as the timeframe of the attack, we present the vector of the attacks as follows, with five types of attacks:

$$AT_i = [AT_1, AT_2, AT3, AT_4, AT_5], \tag{2}$$

such that the probability of having an attack occurring:

$$P(AT) = \frac{1}{\sigma\sqrt{2\pi}} e^{-(AT-\mu)^2/2\sigma^2}, \tag{3}$$

where μ is the mean number of attacks per given timeframe and σ being the standard deviation of attacks occurring within a given timeframe.

B. Attack Data Collection.

To evaluate the data classification capabilities of HDMI-Watch, we collected data from the interaction between all CEC devices in the distribution. The activity collected for the evaluation included regular CEC activity as defined in Section V and malicious activities from HDMI-Walk attacks. During the malicious data collection, we executed each attack 30 times within the HDMI test environment. The collection resulted in a total of 150 datasets of attack data, 30 for each attack. Additionally, we recorded 30 datasets of CEC traffic from the expected operations of the HDMI testbed as defined in Section V. All of the attacks were executed as noted in Section IV. For AT_1, our attacker device performed CEC-based scans over the distribution. For AT_2 and AT_3, we performed file transfer of the captured data to the attacker device. For AT_4, we activated the attack and attempted to power on the display under attack conditions, recording the attacker device shutting off the display. In AT_5, we attacked a display in the distribution via rapid input change.

TABLE IV
Binary Classification Performance Evaluation of HDMI-Walkon Varied Violation Thresholds *T*

T	Binary	TPR	TNR	FPR	FNR	ACC	PREC	REC	F1
1	Benign	0.87	1.0	0.0	0.13	0.87	1.0	0.87	0.93
	Malicious	1.0	0.87	0.13	0.0	0.97	0.97	1.0	0.99
2	Benign	0.90	1.0	0.0	0.1	0.9	1.0	0.9	0.947
	Malicious	1.0	0.9	0.1	0.0	0.98	0.98	1.0	0.99
3	Benign	0.97	0.99	0.01	0.03	0.94	0.97	0.97	0.97
	Malicious	0.99	0.97	0.03	0.01	0.99	0.99	0.99	0.99
4	Benign	0.97	0.96	0.04	0.03	0.81	0.82	0.97	0.89
	Malicious	0.96	0.97	0.03	0.04	0.95	0.99	0.96	0.98
5	Benign	0.96	0.93	0.07	0.03	0.73	0.74	0.97	0.84
	Malicious	0.93	0.97	0.03	0.07	0.92	0.99	0.93	0.96
6	Benign	0.96	0.91	0.10	0.03	0.66	0.67	0.97	0.79
	Malicious	0.91	0.97	0.03	0.10	0.90	0.99	0.91	0.95

C. Performance Metrics

For performance metrics, we utilized the standard parameters: accuracy, True Positive Rate (TPR), True Negative Rate (TNR), False Positive Rate (FPR), False Negative Rate (FNR), recall, precision, and F-score. For instance, in the case of benign evaluation True Positive Rate (TPR) denotes the total number of correctly identified benign CEC activity within the test environment. True Negative Rate (TNR) denotes the total number of correctly identified malicious CEC activity within the test environment. False Positive Rate (FPR) denotes the total number of cases where malicious CEC activity was mistaken as being benign. False Negative Rate (FNR) denotes the total number of cases where benign CEC activity is mistaken as malicious.

$$RecallRate = \frac{TNR}{TNR + FPR}, \tag{4}$$

$$PrecisionRate = \frac{TPR}{TPR + FPR}, \tag{5}$$

$$Accuracy = \frac{TPR + TNR}{TPR + TNR + FPR + FNR}, \tag{6}$$

$$F1 = \frac{2 * RecallRate * PrecisionRate}{RecallRate + PrecisionRate}. \tag{7}$$

1) Performance of HDMI-Watch Classification for Different Violation Thresholds (RQ1): As part of *RQ1*, we evaluate binary classification and the effects of the violation threshold as defined in Section VI, we processed a total classification of 180 datasets (30 malicious for each attack and 30 for benign cases) with HDMI-Watch using different threshold values. As highlighted in Section V, part of HDMI-Watch classification involves the violation threshold per number of packets received. We refer to the classification results in Table IV for different detection threshold values (per 400 packets received). In these results, we show how different threshold values affect the classification results in terms of accuracy and precision.

Table V presents the binary classification results for T = 2 against malicious and benign behavior with accuracy, precision, recall, and F1 metrics for each case. We use this value for T as it presents no false negatives for malicious test cases. For this case, we obtain an accuracy of 90% and precision of 100% for benign detection, with only three benign cases

TABLE V
BINARY PERFORMANCE EVALUATION FOR HDMI-WATCH. CLASSIFYING EXPECTED VS UNEXPECTED BEHAVIOR FOR T=2

Binary	TPR	TNR	FPR	FNR	ACC	PREC	REC	F1
Benign	0.90	1.0	0.0	0.1	0.9	1.0	0.9	0.947
Malicious	1.0	0.9	0.1	0.0	0.98	0.98	1.0	0.99

misclassified as malicious out of 30 cases. In the case of malicious activity, we achieve an overall accuracy and precision of 98%.

With the configurable design of HDMI-WATCH, evaluating the behavior on different thresholds is important and yields some interesting results dependent on attack behavior. We observed that the violation threshold is particularly important for attacks with fewer violations (scanning and power control). The proposed scanning and power control attacks require less CEC packets to execute than attacks involving rapid input switching or data transfer. As a result, the number of violations is much less for scanning and power control behaviors, making such attacks less noticeable. Behaviors such as data transfer in CEC were more easily detectable (higher number of violations) than power control, as an attack involving data transfer or input change spam involves many more packets than abusing shut-off commands. Therefore, if the threshold is too high then attacks with fewer violations may be missed by HDMI-WATCH. Inversely, a threshold that is too low, benign behavior may be improperly classified as malicious, reducing the accuracy of HDMI-WATCH. During testing, A more interesting case was a benign case with 16 violations. This case occurred during a manual power cycling of an AppleTV in the distribution. Media centers such as these are "always-on" devices thus power cycling was benign, but unexpected operation detected by HDMI-WATCH.

2) Classification of Malicious CEC Behavior (RQ2): As part of *RQ2*, we refer to Table VI for HDMI-WATCH's effectiveness in classifying different malicious CEC-based behaviors. For HDMI-WATCH testing, we classified malicious activities into four different types of behaviors (scanning, file transfer, input control, or power control) based on the command type of the violation. Our results show that HDMI-WATCHachieves an average accuracy of 98% and an average precision of 99% for signature-based classification. Additionally, our results show that transfer and input change behavior detection achieved perfect classification with HDMI-WATCH. In the case of power command abuse, HDMI-WATCH mislabeled one case as scanning behavior, impacting the individual accuracy for scanning and power behavior classification.

One of the observations made is that some attacks were more easily distinguishable than others. We highlight that data transfer and input behaviors were classified with better accuracy as they involve many CEC packets from the attacker. For instance, in the case of file transfer, the serialization of packets and then transmission of those packets yields to many violations of command type "00". This command type is matched to values on the table and the activities identified as data transfer. Similarly, rapid input requests require the use of CEC command "20," allowing HDMI-WATCH to easily distinguish the type of behavior being executed. In contrast, other

TABLE VI
SIGNATURE-BASED CLASSIFICATION PERFORMANCE EVALUATION OF THE PROPOSED IDS. DETECTING BEHAVIOR BY TYPE

Behavior	TPR	TNR	FPR	FNR	ACC	PREC	REC	F1
Scan	1.0	1.0	0.008	0.0	0.968	0.968	1.0	0.984
Transfer	1.0	1.0	0.0	0.0	1.0	1.0	1.0	1.0
Input	1.0	1.0	0.0	0.0	1.0	1.0	1.0	1.0
Power	0.967	1.0	0.0	0.033	0.967	1.0	0.967	0.983

behaviors with a smaller footprint (less CEC packets required) such as scanning and power control do not rely on a large number of CEC commands to execute. In effect, these behaviors were more difficult to detect by HDMI-WATCH compared to other types of behaviors.

3) Benefits and Discussion: There are notable benefits to using HDMI-WATCH.

Complete Passive Monitoring. HDMI-WATCH is based on complete, passive monitoring. This has two main advantages. First, HDMI-WATCH does not affect CEC performance in an HDMI distribution. Second, our method does not require any changes to the overall protocol or to existing devices in a distribution.

Complete Black-box integration. HDMI-WATCH resolves one of the biggest issues of HDMI-based devices, the lack of technical documentation. HDMI-WATCH does not require knowledge of source code or operation of any device. HDMI-WATCHmay learn from live analysis during the training phase of a system.

Flexible Design. It is possible to adjust HDMI-WATCH to specific deployments. For instance, the threshold used to classify if an activity is malicious or benign can be adjusted. In addition to the threshold, the number of packets on which the threshold is applied to may also be easily configured to fine-tune HDMI-WATCH classification. Additionally, the signature-based classification ruleset may be altered to include different types of behaviors not covered under this work.

Privacy. HDMI-WATCH only requires command type and length to operate. With the ability to strip communication data from packets HDMI-WATCH requires no sensitive information to operate. HDMI-WATCH can improve the privacy of an HDMI system by detecting the insertion of new devices and preventing malicious CEC scanning. Even though inferring sensitive information within an HDMI distribution from malicious scanning or passive sniffing is possible, a more formal privacy analysis on HDMI is necessary to fully evaluate CEC information leakage. As CEC behavior varies from different devices and device-to-device interaction, a future study on privacy issues from HDMI behavior is needed.

Detection Time. HDMI-WATCH attack detection is based on how quickly CEC packets are issued during the attack, most attacks were detected before the attacks had finished. As such, attacks that issue more packets in a shorter span of time are going to be detected quicker. The time of detection is also difficult to guarantee due to several factors affecting the attack behavior. For instance, an attacker may send CEC packets at a slower rate to accomplish the same attack over a longer span of time. Additionally, data-transfer behaviors may vary

depending on the amount of data transmitted and the rate of transmission. As attacks are detected through the detection threshold, the threshold can be adjusted to detect slower attacks by reducing the detection threshold. However, while this may raise the number of false positives, attacks would also be slower and more ineffective.

Scalability. As with many machine learning systems, data must be gathered on every testbed and on the addition of every new device. This is not unreasonable, as new devices might not be added often within HDMI distributions. As such, the design of HDMI-WATCH allows to save and load models. It may be possible to create models of combinations of devices, and just load the right model into HDMI-WATCH when a new device is added. As devices of the same make and model should have similar behavior, models may be reused in multiple distributions. Additionally, it is possible for HDMI-WATCHto ignore certain devices temporarily while a new model is built. Such actions could reduce false positives for devices which were not in the testbed before. Another solution is to add a CEC-less adapters to new devices, this would keep them from entering the CEC bus while allowing all other HDMI functionality.

VIII. CONCLUSION

Today there are close to 10 billion High Definition Multimedia Interface (HDMI) devices in the world and HDMI has become the de-facto standard for the distribution of A/V signals in smart homes, office spaces, sports events, etc. A component of this widely-deployed interface is the CEC protocol which is used to control devices using the HDMI interface. With no currently known security solutions in place or security implementations in the CEC protocol design, CEC opens a realm of possibilities to attackers. In this work, we highlighted HDMI-WALK, a novel attack surface against HDMI distribution networks and presented five different attacks using this vector. We studied how current insecure CEC protocol practices and HDMI distributions may grant an adversary a viable attack surface against HDMI-enabled devices. Using HDMI-WALK, we analyzed the CEC propagation and implemented a series of local and remote CEC based attacks as a proof-of-concept design. Specifically, we used HDMI-WALKto perform malicious analysis of devices, eavesdropping, Denial-of-Service attacks, targeted device attacks, and facilitate existing attacks through HDMI. As current network security mechanisms only protect traditional networks and components, CEC-based threats are outside of their scope. To defend against these threats, this manuscript proposed HDMI-WATCH, a novel easily-configured security mechanism tailored specifically for the classification of CEC-based abnormal behavior. HDMI-WATCH operates as a passive, standalone framework in an HDMI distribution and provides no additional overhead to CEC communication. Finally, We evaluated the performance of HDMI-WATCH in an HDMI testbed under realistic conditions. HDMI-WATCH evaluation results showed levels of over 90% in accuracy and precision for binary and signature-based classification.

ACKNOWLEDGMENT

This work was partially supported by the US National Science Foundation Awards: NSF-CAREER-CNS-1453647 and NSF-1663051. The views expressed are those of the authors only, not of the funding agencies. We would also like to express sincere appreciation to Amit Kumar Sikder for his contributions to this work.

REFERENCES

[1] A. Tsutsui, "Latest trends in home networking technologies," *IEICE Trans. Commun.*, vol. 91, no. 8, pp. 2470–2476, 2008.
[2] D. Wright, "Shipments of products with HDMI interface nears 900 million devices in 2017; Total installed base approaches seven billion," HDMI Licensing LLC, Jan. 2018. [Online]. Available: https://www.hdmi.org/press/press_release.aspx?prid=153
[3] K. Holman, "HDMI licensing LLC announces availability of HDMI 1.2a specification," Dec. 2005. [Online]. Available: https://www.hdmi.org/press/pr/pr_20051227.aspx
[4] Google, "What is CEC?" 2018. [Online]. Available: https://support.google.com/chromecast/answer/7199917?hl=en
[5] Z. B. Celik *et al.*, "Sensitive information tracking in commodity IoT," in *Proc. 27th USENIX Secur. Symp.*, 2018, pp. 1687–1704.
[6] L. Babun, Z. B. Celik, P. McDaniel, and A. S. Uluagac, "Real-time analysis of privacy-(un)aware IoT applications," 2019, *arXiv:1911.10461*.
[7] J. Lopez, L. Babun, H. Aksu, and A. S. Uluagac, "A survey on function and system call hooking approaches," *J. Hardware Syst. Secur.*, vol. 1, no. 2, pp. 114–136, 2017.
[8] C. Kaygusuz, L. Babun, H. Aksu, and A. S. Uluagac, "Detection of compromised smart grid devices with machine learning and convolution techniques," in *Proc. IEEE Int. Conf. Commun.*, 2018, pp. 1–6.
[9] L. Babun, H. Aksu, and A. S. Uluagac, "A system-level behavioral detection framework for compromised CPS devices: Smart-grid case," *ACM Trans. Cyber-Phys. Syst.*, vol. 4, no. 2, pp. 16.1–16.28, Nov. 2019. doi: 10.1145/3355300.
[10] L. Babun, H. Aksu, and S. A. Uluagac, "Detection of counterfeit and compromised devices using system and function call tracing techniques," U.S. Patent 10 027 697, Jul. 2018.
[11] L. Babun, H. Aksu, and S. A. Uluagac, "Method of resource-limited device and device class identification using system and function call tracing techniques, performance, and statistical analysis," U.S. Patent 10 242 193, Mar. 2019.
[12] A. K. Sikder, L. Babun, H. Aksu, and A. S. Uluagac, "Aegis: A context-aware security framework for smart home systems," in *Proc. 35th Annu. Comput. Secur. Appl. Conf.*, 2019, pp. 28–41.
[13] K. Denney, E. Erdin, L. Babun, M. Vai, and S. Uluagac, "USB-Watch: A dynamic hardware-assisted USB threat detection framework," in *Proc. Int. Conf. Secur. Privacy Commun. Syst.*, 2019, pp. 126–146.
[14] L. P. Rondon, L. Babun, K. Akkaya, and A. S. Uluagac, "HDMI-walk: Attacking HDMI distribution networks via consumer electronic control protocol," in *Proc. 35th Annu. Comput. Secur. Appl. Conf.*, 2019, pp. 650–659.
[15] HDMI Licensing LLC, "Inside an HDMI Cable," 2018. *https://www.hdmi.org/installers/ insidehdmicable.aspx*
[16] N. Zhang *et al.*, "Understanding IoT security through the data crystal ball: Where we are now and where we are going to be," *CoRR*, vol. abs/1703.09809, 2017.
[17] Y. Oren and A. D. Keromytis, "From the aether to the ethernet—Attacking the internet using broadcast digital television," in *Proc. 23 rd USENIX Secur. Symp. (USENIX Secur. 14)*, 2014, pp. 353–368.
[18] M. Niemietz, J. Somorovsky, C. Mainka, and J. Schwenk, "Not so smart: On smart tv apps," in *Proc. Int. Workshop Secure Internet Things*, Sep. 2015, pp. 72–81.
[19] A. Davis, "What the HEC? Security implications of HDMI Ethernet Channel and other related protocols," Aug. 2013. [Online]. Available: https://www.nccgroup.trust/globalassets/our-research/uk/whitepapers/2013/44con_hdmi_ethernet_channel_andy_davis _ncc_group_wp.pdf
[20] J. Smith, "High-def fuzzing : Exploring vulnerabilities in HDMI-CEC," Nov. 2015. [Online]. Available: https://media.defcon.org/
[21] A. Davis, "HDMI : Hacking displays made interesting," Mar. 2012. [Online]. Available: https://media.blackhat.com/bh-eu-12/Davis/bh-eu-12-Davis-HDMI-Slides.pdf

[22] "Hdmi-cec control service," May 2019. [Online]. Available: https://source.android.com/devices/tv/hdmi-cec
[23] Pulse-Eight, "USB-CEC adapter communication library," 2018. [Online]. Available: https://github.com/Pulse-Eight/libcec/
[24] A. H. Lashkari, M. M. S. Danesh, and B. Samadi, "A survey on wireless security protocols (wep, wpa and wpa2/802.11i)," in *Proc. 2nd IEEE Int. Conf. Comput. Sci. Inf. Technol.*, 2009, pp. 48–52.
[25] N. Baharudin, F. H. M. Ali, M. Y. Darus, and N. Awang, "Wireless intruder detection system (wids) in detecting de-authentication and disassociation attacks in IEEE 802.11," in *Proc. 5th Int. Conf. IT Convergence Secur.*, 2015, pp. 1–5.
[26] M. J. I. Sirven, "Photosensitivity and seizures," Nov. 2013. [Online]. Available: https://www.epilepsy.com/learn/triggers-seizures/photosensitivity-and-seizures
[27] C. Heyne, "AV Tip: How to avoid blowing out your speakers," Jan. 2013. [Online]. Available: https://www.audioholics.com/home-theater-connection/avoid-blowing-speakers
[28] Python.org, "tkinter – Python interface to Tcl/Tk," Accessed: Dec. 20, 2019. [Online]. Available: https://docs.python.org/3/library/tkinter.html

Luis Puche Rondon received the bachelor's degree in computer science and the master's degree in cybersecurity from Florida International University, Miami, FL, USA, in 2016 and 2017, respectively. He is a member of the Cyber-Physical Systems Security Lab (CSL), Florida International University, Department of Electrical and Computer Engineering and the CyberCorps Scholarship for Service (SFS) fellow. He has over a decade of experience in Audio/video, smart homes, professional smart systems, and CCTV installations. His research interests focus on the security of high-end smart systems, IoT, and the security implications of threats under-researched threat vectors.

Leonardo Babun received the master's degree in electrical engineering from Florida International University, Miami, FL, USA, in 2015, and the master's degree in computer engineering and the Doctoral degree in electrical and computer engineering from the Department of Electrical and Computer Engineering, Florida International University, Miami, FL, USA, in 2019 and 2020, respectively. He is currently a member of the Cyber-Physical Systems Security Lab (CSL) and a CyberCorps Scholarship for Service Alumni with the Department of Electrical and Computer Engineering, Florida International University. His research interests are focused on the security and privacy of cyber-physical systems (CPS) and the Internet of Things (IoT).

Kemal Akkaya (Member, IEEE) the Advanced Wireless and Security Lab (ADWISE), FIU. His research areas span various challenges of mobile and wireless networks, Internet of Things and cyber-physical systems, such as security, privacy, quality of service, topology control, and mobility management. He is the Area Editor of Elsevier *Ad Hoc Networks* and serves on the editorial board for the IEEE COMMUNICATION SURVEYS AND TUTORIALS. He has been a Guest Editor for various journals and serves in the organizing committees of leading IEEE communication conferences, such as IEEE LCN, ICC, Globecom, WCNC, and SmartGridComm. He is a member of IEEE Computer Society. He is also a Member in IEEE Technical Committees on Communication, Cybersecurity, Smart Cities, and Online Social Networks.

A. Selcuk Uluagac (Member, IEEE) received the M.Sc. degree in information security from the School of Computer Science, Georgia Tech, Atlanta, GA, USA and the M.Sc. degree in electrical and computer engineering (ECE) from Carnegie Mellon University, Pittsburgh, PA, USA, in 2009 and 2002, respectively, and the Ph.D. degree with a concentration in information security and networking from the School of ECE, Georgia Tech, Atlanta, GA, USA, in 2010. He is currently an Associate Professor with the Department of ECE, Florida International University (FIU), Miami, FL, USA. Before joining FIU, he was a Senior Research Engineer with the School of ECE, Georgia Institute of Technology. Prior to Georgia Tech, he was a Senior Research Engineer with Symantec. The focus of his research is on cyber security topics with an emphasis on its practical and applied aspects. He is interested in and currently working on problems pertinent to the security of cyber-physical systems and Internet of Things. In 2015, he received a Faculty Early Career Development (CAREER) Award from the US National Science Foundation (NSF). In 2015, he was awarded the US Air Force Office of Sponsored Research (AFOSR)'s 2015 Summer Faculty Fellowship. In 2016, he received the Summer Faculty Fellowship from the University of Padova, Italy. He is also an active Member of IEEE (senior grade), ACM, and ASEE and a regular contributor to national panels and leading journals and conferences in the field. He is currently the Area Editor for the IEEE TRANSACTIONS ON MOBILE COMPUTING, Elsevier *Computer Networks*, and serves on the editorial board for the IEEE COMMUNICATION SURVEYS AND TUTORIALS as the Network Security Track lead.