Safeguarding the Smart Home: Heterogeneous Federated Deep Learning for Intrusion Defense

Mohammed Shalan
Department of Computer Science
North Dakota State University
Fargo, USA
mohammed.shalan@ndsu.edu

Juan Li
Department of Computer Science
North Dakota State University
Fargo, USA
j.li@ndsu.edu

Yan Bai
School of Engineering and Technology
University of Washington Tacoma
Tacoma, USA
yanb@uw.edu

***Abstract*— This study introduces an advanced federated learning framework tailored for smart home intrusion detection, incorporating knowledge distillation and transfer learning to tackle escalating threats to IoT devices. In light of the rapid expansion of IoT devices and their vulnerability to botnet incursions, our approach specifically addresses the challenges related to the privacy concerns of home device data, heterogeneity of devices, sparse intrusion data, and the dynamic nature of smart home settings. We adaptively select model architectures tailored to the computational capabilities of each device, ranging from simple Neural Networks (NNs) to more complex Convolutional Neural Networks (CNNs) and hybrid CNN-LSTM models, ensuring efficient local training without overburdening the devices. However, it can achieve good performance through collaborative learning, even for devices with lower capacity and sparse data. Our evaluation, conducted using the N-BaIoT dataset, demonstrates the effectiveness of our approach in detecting anomalies across a diverse set of IoT devices infected with real-world botnets such as Mirai and BASHLITE. The results highlight the potential of our framework to provide a robust, privacy-preserving, and adaptable solution for securing smart homes against emerging threats.**

Keywords—intrusion detection systems, smart home, federated learning, deep learning

I. INTRODUCTION

In the realm of modern home automation, smart home devices such as baby monitors, web cameras, and doorbells have become integral components. These devices not only offer convenience and efficiency but also play a crucial role in home security. However, with the increasing integration of these Internet of Things (IoT) devices into our daily lives, a significant concern arises regarding their vulnerability to cyber threats [1]. These vulnerabilities are not limited to the potential for hacking and data breaches but extend to more intrusive risks, such as unauthorized surveillance and access to personal information [2]. This aspect of security is particularly alarming given the personal nature of the data collected by these devices, which often include visual and audio recordings of private home environments.

Imagine a hacker intercepts the feed from your baby monitor, using it to spy on your most vulnerable moments. Or, an attacker disables your smart doorbell, granting them unimpeded entry into your home. These chilling scenarios, once relegated to the realm of science fiction, are becoming increasingly real in the era of smart homes. As an army of interconnected devices infiltrates our living spaces, from voice assistants to connected appliances, security and privacy concerns are escalating at an alarming rate.

The handling of user data collected by smart home devices has become a topic of paramount importance, raising both legal and ethical questions. Privacy concerns stem from how sensitive data that can reveal intimate details about a person's daily life are processed and stored. In light of regulations such as the General Data Protection Regulation (GDPR) [3], there is a growing demand for systems that can ensure the confidentiality and integrity of user data. Moreover, the way these privacy concerns are addressed significantly impacts user trust and the adoption rate of smart home technologies. Users are becoming increasingly aware of their digital footprint and are seeking assurances that their personal data is handled securely and responsibly.

Central to this security is the concept of intrusion detection, which plays a pivotal role in identifying and mitigating potential threats[4]. Intrusion detection systems (IDS) in IoT and smart home environments are designed to detect unauthorized access or anomalous behavior, thereby safeguarding networks and devices from various cyber threats. By continuously monitoring network traffic and device activities, these systems aid in the early detection of potential security breaches, contributing significantly to the overall security posture of smart homes.

Traditional Intrusion Detection Systems (IDS) face significant challenges in the context of IoT environments, such as those in smart homes. These challenges include serious privacy concerns due to the need to collect and analyze large volumes of potentially sensitive data. Typically, these data are processed in a centralized cloud-based system, increasing the risk of data exposure and misuse, particularly in the event of security breaches[5]. Moreover, this centralized approach, reliant on cloud computing, is increasingly viewed as problematic from a privacy standpoint. Additionally, the process of uploading vast amounts of traffic data to central servers can be inefficient, consuming excessive bandwidth and depleting device batteries.

This work was supported by the National Science Foundation (NSF) with award numbers: 2218046, 2334197, and 2334196.

XXX-X-XXXX-XXXX-X/XX/$XX.00 ©20XX IEEE

To address these privacy issues, federated learning presents a promising alternative [6]. This approach allows for the training of machine learning models directly on devices without the need to transfer raw data to a centralized cloud. In the context of smart homes, federated learning enables local processing and analysis of data, with only the model parameters (and not the actual data) being shared with a central server. This methodology significantly enhances privacy as sensitive data remain on the user's device.

Federated learning has been extensively researched across various sectors, but its application in smart home intrusion detection is still emerging. Despite its proven effectiveness in multiple contexts, the deployment within smart home ecosystems is complicated by several unique challenges. The heterogeneity of smart home devices, which varies significantly in bandwidth and computational power, can affect the performance of federated learning models. Additionally, the rarity of intrusion events means that some devices may not have sufficient local data for effective learning. The data generated by these devices are also diverse, reflecting the different environments and usage patterns, which leads to non-identically and independently distributed (non-IID) data that complicate the training of models. Moreover, smart homes are dynamic environments that constantly change with the addition or removal of devices and evolution of usage patterns. Models must navigate the delicate balance between personalizing for the unique needs of individual homes and generalizing to benefit from the shared learning across the network.

In light of these challenges and the potential of federated learning, this paper aims to bridge the gap in current research. We propose a novel framework that extends the principles of Federated Learning (FL) through Knowledge Distillation (KD) and Transfer Learning (TL) to address the unique challenges of smart home intrusion detection. Knowledge distillation is a technique that smaller models can learn from complex and larger models through knowledge transfer across the heterogeneous smart home devices. The goal is to achieve similar performance as larger model while being efficient in terms of computational resources. This distilled knowledge, in the form of class scores, can then be used to update and synchronize the models on individual devices, effectively transferring the collective insights without the need for raw data exchange. This approach is particularly effective given the heterogeneity of devices, the rarity of intrusion events, and the dynamic nature of smart home environments. The key contributions and novelties of our work include the following.

1. Heterogeneous model support: Unlike traditional FL, which requires a uniform model across devices, our framework allows for diverse models tailored to the specific capabilities and roles of each smart home device, thus enhancing model performance and efficiency.
2. Data scarcity solution: Recognizing the infrequency of intrusion events, we employ transfer learning from a large public dataset to pre-trained models, ensuring that even devices with limited exposure to intrusions can contribute to and benefit from the federated model.
3. Knowledge distillation for communication: To facilitate model collaboration without compromising data privacy, our framework uses knowledge distillation, allowing models to share insights via class scores derived from public data, thereby overcoming the challenge of direct data sharing.
4. Dynamic adaptation: Our approach was designed to adapt to the evolving landscape of smart homes, accommodate new devices, and change usage patterns without disrupting the learning process.
5. Personalization vs. generalization: By leveraging collective learning from the network while allowing for local model customization, our framework strikes a balance between personalizing intrusion detection for individual homes and harnessing broader insights from the federated network.

This innovative combination of techniques addresses the inherent challenges of applying FL to smart home intrusion detection, leading to a more robust, efficient, and adaptable solution.

II. Related Work

A diverse array of intrusion detection systems (IDS) has emerged to safeguard smart homes. Signature-based approaches identify malicious activities based on pre-defined attack patterns (e.g., Snort) [7]. However, their effectiveness dwindles against novel threats, requiring frequent signature updates. Anomaly-based methods [8], particularly those leveraging deep learning techniques, aim to automatically learn the complex patterns of "normal" behavior for each device and its environment. This could involve analyzing the power consumption, network traffic, audio/video streams, or sensor data. Techniques such as autoencoders [9], recurrent neural networks [10], and one-class support vector machines [11], [12] have shown promise in identifying subtle deviations from the expected behavior, and anomaly detection methods for smart home network intrusion detection systems using autoencoders potentially flagging novel or zero-day attacks[13] However, these approaches face several challenges. False positives can still arise due to inherent device variations, user activities, and environmental changes. For example, an unexpectedly high energy surge during cooking may be misclassified as intrusion [14]. Additionally, detecting rare attacks with limited training data can be difficult, leading to missed threats. Furthermore, the computational demands of deep learning models can be resource-intensive for smart home devices, particularly those with limited processing power and battery life [15]. Hybrid approaches combine signature and anomaly detection to offer broader coverage, but face similar limitations in terms of agility and accuracy [16].

Traditional centralized IDS collect and analyze data from all devices on a single server, providing centralized control and efficient model updates. However, this raises privacy concerns as sensitive data leaves the device, creating a central honeypot for attackers [17]. Additionally, network bottlenecks and high latency hinder responsiveness in geographically dispersed networks [18]. Decentralized approaches such as peer-to-peer IDS distribute intelligence among devices, fostering local autonomy and privacy [19]. Yet, they suffer from limited information-sharing and vulnerability to compromised peers [20]. Distributed multi-level IDS systems specifically for IoT networks have also been proposed to reduce the response time,

save IoT the energy and bandwidth of IoT devices. For example, in their work Roy et al. proposed a fog-cloud two layer hierarchical intrusion detection mechanism that can effectively detect intrusions in IoT networks while satisfying the IoT resource constraints [21]. However, these works cannot solve the privacy issues related to smart home requirements.

The limitations of traditional centralized and decentralized IDS approaches have paved the way for federated learning (FL) as a promising paradigm for intrusion detection[22]. FL offers a promising middle ground that enables collaborative model training without central data aggregation. Local models are trained on individual devices and then aggregated, preserving user privacy while leveraging collective knowledge. This fosters adaptive and robust intrusion detection models that can evolve to address novel threats [23].

Several recent studies showcase the potential of federated learning in anomaly detection and intrusion detection. For example, Mothukuri et al. proposed a federated-learning-based anomaly detection for IoT security attacks, achieving comparable accuracy to centralized approaches while protecting user data [24]. Rey et al. explore federated learning for enhancing malware detection across IoT devices, leveraging the N-BaIoT dataset, showcasing improved accuracy and adaptability compared to traditional solutions [25]. Ruzafa-Alcazar et al. delves into differential privacy techniques within federated learning for enhancing intrusion detection in industrial IoT environments, demonstrating its effectiveness in identifying network anomalies while preserving user privacy [26]. Roy et al. proposed an FL-based IDS for IoT devices. Their results demonstrated comparable performance to centralized learning on metrics including accuracy, precision, and recall, while addressing privacy and data leakage concerns [27]. Despite its advantages, FL faces challenges in the context of smart home security. Heterogeneity in device capabilities, communication bandwidth limitations, and potential security vulnerabilities within the FL framework require further research and development[28].

III. METHODOLOGY

Our methodology employs a system that integrates knowledge distillation and transfer learning adapted from the FedMD approach [29] to address the unique challenges of smart home intrusion detection. As illustrated in Figure 1, it begins with transfer learning initialization, where each device is trained on a broad dataset to grasp potential security threats, followed by local fine-tuning of private data to adapt to specific home environments. The core of our approach, knowledge distillation for federated learning [30], enables devices to share insights without raw data exchange, creating a consensus of learned features. This collective intelligence is then used to update and synchronize models across the network, enhancing detection capabilities even in devices with limited direct intrusion exposure. The framework was designed for continuous adaptation to the dynamic nature of smart homes, ensuring up-to-date defense mechanisms. Balancing model personalization and generalization ensures that while each system is tailored to specific home needs, it also benefits from the wider network's shared learning. Our approach aims to establish a robust, privacy-conscious, and flexible system that elevates security in the diverse ecosystem of smart home devices. The framework is illustrated in Fig. 1.

A. Federated Leaning Framework

1) Transfer Learning Initialization:

Transfer Learning Initialization: Each device d_i in the smart home network will initially train its intrusion detection model M_i using a publicly available dataset D_{public} relevant to general security threats. In this experiment, the public dataset D_{public} encompassed all classes and devices from N-BaIoT to ensure a comprehensive foundation for intrusion detection knowledge across all participating models. The objective is to optimize the initial model parameters $\theta_i^{(0)}$ by minimizing the loss function L_{public} of D_{public}:

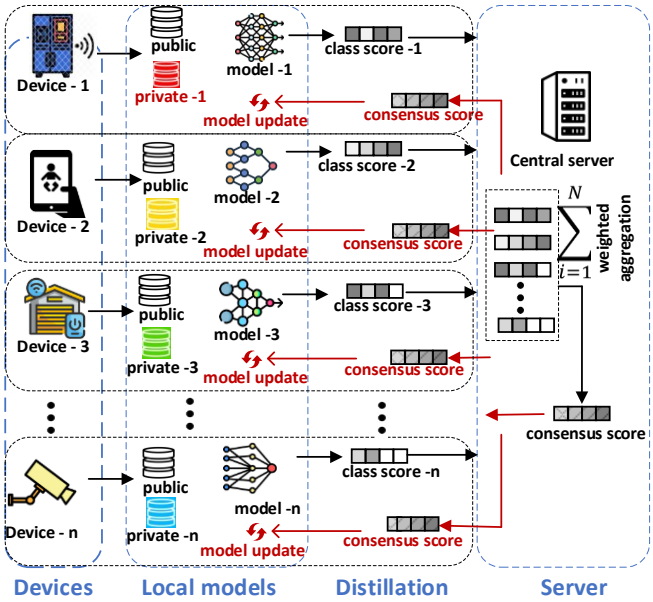

Fig. 1. System Framework

$$\theta_i^{(0)} = \underset{\theta}{\operatorname{argmin}}\, L_{public} M_i(D_{public}; \theta) \qquad (1)$$

This step ensures that every participating model has a foundational understanding of potential intrusions.

2) Local Fine-tuning:

Following the initial training, the devices fine-tune their models on their private, locally stored data $D_i^{private}$. Given the rarity of intrusion events, this step is crucial for models to learn from the limited but highly relevant examples of actual smart home security incidents. This step adapts the model to the specific security context of each smart home, optimizing the parameters $\theta_i^{(1)}$:

$$\theta_i^{(1)} = \underset{\theta}{\operatorname{argmin}}\, L_{private} M_i(D_i^{private}; \theta_i^{(0)}) \qquad (2)$$

3) Knowledge Distillation for Federated Learning:

The core of our approach involves smart home devices that share knowledge without exchanging raw data. Devices generate class scores $S_{i,j}$ for samples x_j from the public dataset D_{public} and share these scores with a central server. The server aggregates this information to create a distilled knowledge

dataset, representing a consensus $S_{consensus,j}$, of learned features from all participating devices.

$$S_{consensus,j} = \sum_{i=1}^{N} (S_{i,j}) * CVW_i \quad (3)$$

where N is the number of participating devices and CVW (Computational-Volume Weight) represents each device's impact in terms of its computational power and data volume.

Computational-Volume Weight (CVW): A metric determining a device's contribution to a machine learning model. CVW considers:

- Computational Power: More powerful devices (able to perform more calculations) have greater influence.
- Data Volume: Devices with larger datasets have a greater impact.

Therefore, CVW is calculated based on two parts:

- Complexity Assessment: The number of floating-point operations (FLOPs) a device's model performs determines its complexity [31].
- Data Volume: The amount of data a device contributes is measured.

Both complexity and data volume are normalized as shown Equation (4) and (5), ensuring each device's contribution is weighted fairly based on its capabilities

$$CVW_i = \frac{1}{2} (W_i^{flop} + W_i^{data}) \quad (4)$$

$$W_i^{flop} = \frac{F_i}{\sum_{j=1}^{N} F_i^j} \quad (5)$$

$$W_i^{data} = \frac{D_i}{\sum_{j=1}^{N} D_i^j} \quad (6)$$

4) Model Update and Synchronization:

With the distilled knowledge, each device updates its local model to align with the aggregated insights, optimizing the parameters $\theta_i^{(2)}$to minimize the difference between its class scores and the consensus:

$$\theta_i^{(2)} = \underset{\theta}{\text{argmin}}\, L_{distill} M_i(D_{public}; \theta_i^{(1)}, S_{consensus}) \quad (7)$$

thereby benefiting from the collective learning of the network. This step ensures that even devices with limited exposure to intrusion events can enhance their detection capabilities.

5) Continuous Adaptation:

The smart home environment is dynamic, with devices being added or removed and usage patterns evolving. Our framework accommodates these changes by periodically repeating the knowledge distillation and model update processes. This ensures that the models remain effective and up-to-date with the latest security threats.

6) Personalization vs. Generalization:

Our framework maintains a balance between personalizing models to the unique security needs of each smart home, and generalizing across the network to benefit from shared learning. This balance is crucial for maximizing the effectiveness of intrusion detection in diverse environments.

By customizing the FedMD approach for smart home intrusion detection, we aim to create a robust, privacy-preserving, and adaptable framework that enhances security across a network of diverse and dynamically changing smart home devices.

B. Local Training

In the Local Training phase of our distillation-based federated learning framework, selecting the appropriate model for each smart home device is critical, particularly given the diverse ecosystem of devices within a typical smart home. These devices range from high-capacity smart security systems to more constrained IoT devices, such as smart bulbs and sensors. The primary considerations in model selection are the computational capabilities, available memory, and energy constraints of each device, ensuring that the intrusion detection process is sustainable and does not impair the device's primary functions. The methodology for choosing the right model for a specific home device involves a multi-faceted approach:

- Device capability assessment: The first step is a thorough evaluation of each device's hardware specifications, including processing power, available RAM, and storage. This assessment helps in categorizing devices based on their computational capabilities.
- Energy consumption consideration: For battery-powered devices, energy efficiency becomes a pivotal factor. Models that require less computational power and, consequently, consume less energy are preferred to ensure that the device's primary functionalities are not compromised.
- Model complexity vs. performance trade-off: The trade-off between model complexity and intrusion detection performance was carefully analyzed. While simpler models are more resource-efficient, they might lack the sophistication needed for accurate intrusion detection. Conversely, more complex models, although potentially more accurate, may not be feasible for resource-constrained devices.
- Adaptive model architecture: The architecture of the models for local training is adaptively chosen based on each smart home device's computational capabilities and energy constraints. For example, in our experiments, our selection spans a range of complexities, from Neural Networks (NN) to more complex architectures such as Convolutional Neural Networks (CNN) with varying depths (e.g., CNN with two blocks for less capable devices and CNN with three blocks for more capable ones) and CNN-LSTM hybrids for devices that can afford additional computational overhead while benefiting from LSTM's ability to understand temporal patterns in data.

Algorithm1: Algorithm used to train heterogeneous models.

Input: Public dataset D_{public}, private datasets $D^{i}_{private}$ for each device d_i
Transfer Learning Initialization:
for each device *di*:
    $M_i \leftarrow$ model of each device
    Train intrusion detection model M_i on D_{public}
    $\theta_i^{(0)} = argmin_{\theta} L_{public} M_i(D_{public}; \theta)$// Optimize initial model parameters
end
Local Fine-tuning:

for each device d_i:

$\theta_i^{(1)} = \underset{\theta}{argmin}\, L_{private} M_i(D_i^{private}; \theta_i^{(0)})$ // Fine-tune M_i on $Di_{private}$

end

Knowledge Distillation:

sample$_{xj}$ ← in D_{public}:

for each device d_i:

Generate class scores S_{ij} for xj using M_i

end

Aggregate class scores:

$$S_{consensus,j} = \sum_{i=1}^{N} (S_{i,j}) * CVW_i$$

Models update:

for each device d_i:

$\theta_i^{(2)} = \underset{\theta}{argmin}\, L_{distill} M_i(D_{public}; \theta_i^{(1)}, S_{consensus})$ // Update M_i using $S_{consensus}$ by minimizing $L_{distill}$

end

Repeat

IV. EVALUATION

A. Dataset

For our evaluation, we utilized the N-BaIoT dataset, [32] chosen for its relevance to IoT security and the variety of IoT devices it encompasses, including nine commercial IoT devices infected with Mirai and BASHLITE botnets. The N-BaIoT dataset is particularly suited for studying IoT-based botnet attacks due to its real-world attack scenarios and diverse device behaviors. This dataset includes traffic data from devices such as security cameras, thermostats, and baby monitors, providing a comprehensive overview of typical smart home devices. Each device in the dataset exhibits unique features and behaviors,

which are essential for developing and testing intrusion detection models that can adapt to the heterogeneous nature of smart home environments. By using the N-BaIoT dataset, our evaluation aims to assess the effectiveness of our federated learning and knowledge distillation approach in detecting anomalies and potential security threats across a varied set of IoT devices, ensuring our methodology's applicability to real-world smart home settings.

B. Local Models

In our evaluation, the local model selection for each IoT device accommodated the distinct characteristics and limitations of the smart home devices. Considering the diverse nature of the devices and their computational constraints, we adopted various model architectures tailored to the specific needs and capabilities of each device.

For the Danmini Doorbell, a relatively simple Convolutional Neural Network (CNN) [33] with 2 Blocks (CNN_2Blocks) was chosen, reflecting the device's moderate computational resources. This model, comprising 81,952 parameters, strikes a balance between complexity and efficiency, and is suitable for a device such as a doorbell that requires real-time processing but does not require extensive data analysis. The Ecobee Thermostat and Provision PT-737E Security Camera utilized a simple Neural Network (NN) architecture consisting of 185,984 parameters. This choice was driven by the need for models that could efficiently process data without imposing significant computational loads, given the energy and processing constraints typical of such devices. The Philips B120N/10 Baby Monitor, which requires more nuanced data analysis due to its complex functionalities such as motion and sound detection, was assigned a CNN with three blocks (CNN_3Block), containing 318,752 parameters. However, it is important to note that this device, along with the Ennio Doorbell, was excluded from our final evaluation because of its limited class diversity, which did not align with the requirements of our model. For security cameras, which are pivotal in intrusion detection and require sophisticated analysis to identify anomalies in the video data, we employed more complex models. The Provision PT-838 and SimpleHome XCS7_1002_WHT Security Cameras were equipped with a CNN-LSTM hybrid model, blending the spatial feature extraction capabilities of CNNs with the temporal pattern recognition strength of LSTMs. This model architecture, with 261,344 parameters, is particularly well-suited for processing sequential data such as video streams. Lastly, the SimpleHome XCS7_1003_WHT Security Camera was fitted with a CNN model similar to that of the Danmini Doorbell, considering the similar operational and computational demands of these devices.

Table I illustrates each chosen model architecture and its parameters. The selection of different models for different devices underscores our methodology's emphasis on resource awareness and adaptability to the heterogeneous ecosystem of smart home devices.

C. Results

To model an idealized training scenario, we trained all models on all devices using a centralized approach. In this scenario, each

TABLE I. DEVICES AND THEIR MODELS

Device Category	Device	Model Architecture	Parameters	CVW
Doorbell	Danmini	CNN_2Blocks	81,952	0.16
Baby Monitor	Philips_B120N10	CNN_3Blocks	318,752	0.40
Thermostat	Ecobee	NN	185,984	0.06
Security Camera	Provision_PT_737E	NN	185,984	0.08
	Provision_PT_838	CNN+LSTM	261,344	0.07
	SimpleHome_XCS7_1002_WHT	CNN+LSTM	261,344	0.07
	SimpleHome_XCS7_1003_WHT	CNN	81,952	0.15

device has access to the entire dataset for training and testing. This approach is unlikely to be replicated in real-world deployments, where individual devices may have limited access to the full dataset due to storage or bandwidth constraints.

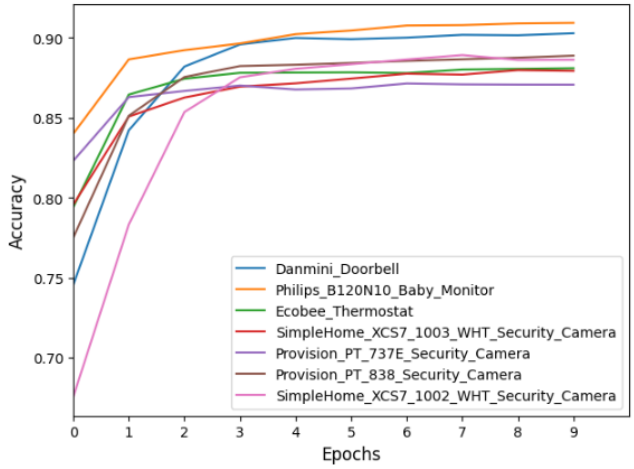

Fig. 2. Centralized accuracies

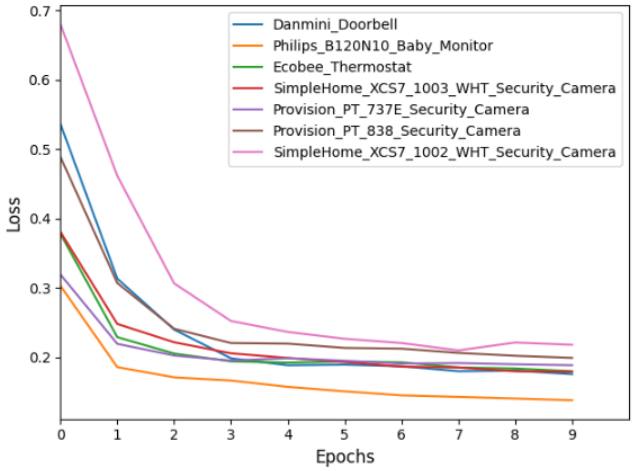

Fig. 3. Centralized losses

Figure 2 illustrates the training accuracy achieved by all models under this centralized approach. As shown, most devices reach an accuracy of around 88%. Notably, the baby monitor device, which utilizes the most complex model in our system, achieves an accuracy of 91%.

Figure 3 depicts the loss curves for all models during the centralized training process. The curves demonstrate a smooth convergence towards their minimum values, indicating effective learning. As expected, devices with more complex models, like the baby monitor (orange curve), achieve lower final loss compared to devices with simpler models, such as the Ecobee thermostat (shown here for comparison). This suggests that the increased complexity allows the model to better capture the underlying patterns within the data.

Figure 4 illustrates the accuracy improvement of all devices in our system throughout the training process (x-axis represents communication rounds). Each line depicts the performance of a single device. Devices begin with a pre-trained model (based on a public security threat dataset). This initial training provides a foundational level of accuracy, reflected in the starting points of the lines. As communication rounds progress, devices leverage

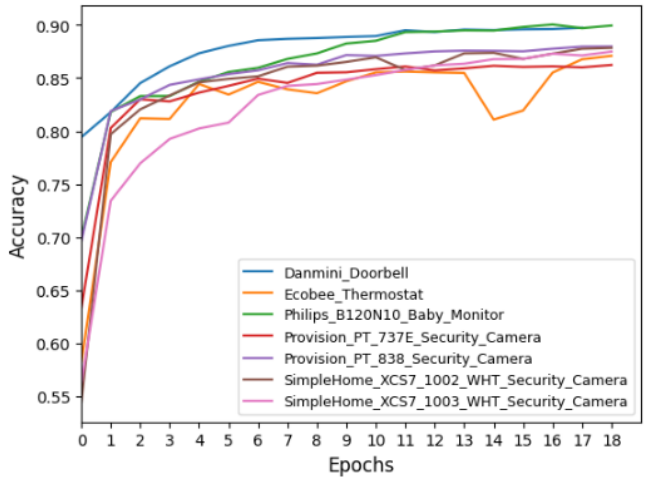

Fig. 4. Federated accuracies

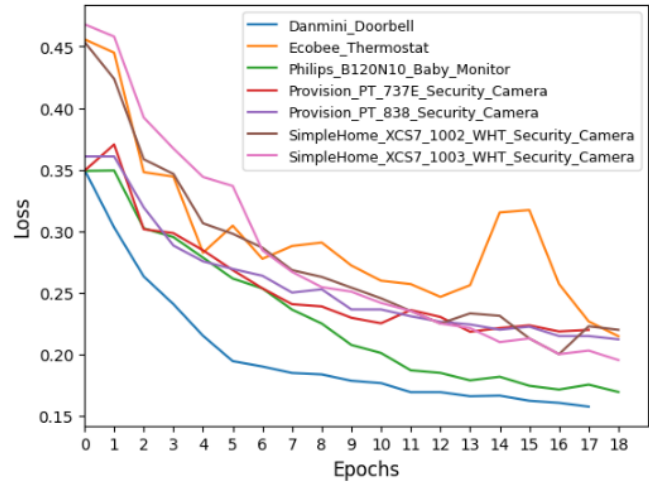

Fig. 5. Federated losses

a federated learning approach with knowledge distillation. This allows them to collaborate and share knowledge (without raw data exchange) to improve their threat detection capabilities.

We observe two key trends: First, devices with a higher contribution rate ,determined by CVW, exhibit a smoother and more consistent accuracy improvement over communication rounds. This is likely because they contribute more data and computation to the learning process. Moreover. even simpler models demonstrate accuracy improvement, albeit with some fluctuations during training. This can be attributed to the knowledge distillation step, where these models benefit from the knowledge transferred from more complex models in the system. For example, the Ecobee thermostat, equipped with a simple neural network (NN) model, shows improvement from 58% initial accuracy to 87% by the end of the training process.

TABLE II. PERFORMANCE OF FEDERATED MODELS

Device	Precision	Recall	Accuracy	F1-score
Danmini	0.85	0.90	0.90	0.90
Philips_B120N10	0.92	0.90	0.87	0.91
Ecobee	0.85	0.90	0.87	0.88
Provision_PT_737E	0.88	0.90	0.83	0.87
Provision_PT_838	0.89	0.89	0.86	0.89
SimpleHome_XCS 7_1002_WHT	0.83	0.86	0.84	0.87
SimpleHome_XCS 7_1003_WHT	0.86	0.87	0.85	0.87

TABLE III. PERFORMANCE OF INITIAL MODELS

Device	Precision	Recall	Accuracy
Danmini	0.54	0.60	0.52
Philips_B120N10	0.56	0.62	0.69
Ecobee	0.61	0.66	0.58
Provision_PT_737E	0.59	0.64	0.63
Provision_PT_838	0.71	0.71	0.69
SimpleHome_XCS 7_1002_WHT	0.57	0.66	0.54
SimpleHome_XCS 7_1003_WHT	0.58	0.70	0.56

Figure 5 illustrates the loss performance of devices during the federated learning process. Consistent with accuracy results, devices with more complex models (higher FLOPs weight) exhibit smoother convergence towards a minimum loss value. Simpler models, while showing some fluctuations, ultimately converge as well, demonstrating the benefits of federated learning for all devices.

To gain a deeper understanding of how well our federated learning system performs, we evaluate each device's model using three key metrics: precision, recall, and F1-score. These metrics provide a comprehensive picture of the model's ability to accurately detect threats. Table II presents the initial performance of each device's model after training on a public dataset. It serves as our baseline for comparison. Table III summarizes how the models perform across all devices after using our knowledge distillation-based federated learning system. Compared to their initial baseline performance in Table I, we see a dramatic improvement across simpler models. The Provision_PT_737E device, as a key example, demonstrates the most significant improvement. This highlights how our proposed system effectively boosts the performance of models with limited computational resources.

From the experiments, we can see that our proposed system effectively enhances the security detection capabilities of smart home devices, regardless of their model complexity, while ensuring user privacy. This collaborative learning approach, powered by knowledge distillation, shows particular promise for devices with limited computational resources.

V. CONCLUSIONS

In this research, we proposed a robust and adaptable federated learning framework specifically designed for intrusion detection within smart home environments. Our approach tackles several critical challenges inherent to this domain, including data privacy, device heterogeneity, sparse intrusion data, and the ever-changing nature of smart home settings. Key innovations of our framework include heterogeneous model support, where we strategically select diverse model architectures (such as NN, CNN, and CNN-LSTM) based on each device's computational capacity. This ensures that local training remains efficient while maximizing performance across the entire smart home network.

We prioritize privacy by using knowledge distillation. In this approach, devices share insights in the form of class scores derived from public datasets, preserving privacy while allowing devices to learn from each other's insights. Additionally, we address the issue of sparse intrusion data by pre-training all devices on a public dataset. This gives each device a baseline understanding of potential threats, ensuring robustness despite infrequent intrusion events on individual devices. Our framework is designed for continuous adaptation, seamlessly adjusting to changes within the smart home environment, such as added or removed devices, or evolving usage patterns, which keeps security models up-to-date. Finally, we strike a crucial balance between personalization and generalization. Local fine-tuning allows each system to tailor itself to the specific security needs of a smart home, while ongoing knowledge distillation ensures that all devices benefit from broader network insights.

Extensive evaluation using the N-BaIoT dataset demonstrates that our approach detects anomalies with high accuracy across a diverse set of IoT devices infected with real-world botnets like Mirai and BASHLITE. Our results show that even simpler devices benefit from knowledge distillation, experiencing significant performance gains when compared to their initial baselines. Notably, devices like the Provision_PT_737E exemplify the success of our system in helping those with limited resources.

Our work has demonstrated the effectiveness of a privacy-preserving, adaptable federated learning system for enhancing intrusion detection in heterogeneous smart home environments. This approach holds significant promise for the future of smart home security. In our ongoing research, we plan to investigate several avenues for further improvement. First, we intend to explore more complex distillation techniques beyond class scores. Distilling richer information such as feature maps could provide deeper insights for model collaboration, potentially leading to even more accurate intrusion detection. Additionally, we plan to incorporate adversarial learning techniques to enhance the robustness of our models. Adversarial learning involves training models to be resilient against attempts to

manipulate their behavior. By incorporating this approach, we can create models better equipped to handle new or unknown attack vectors, further strengthening the security posture of smart homes. Finally, to ensure broader real-world applicability, we plan to conduct large-scale evaluations across a wider range of smart home networks. This will provide a more comprehensive understanding of the system's performance and effectiveness in diverse real-world settings.

REFERENCES

[1] A. Mosenia and N. K. Jha, "A comprehensive study of security of internet-of-things," *IEEE Trans Emerg Top Comput*, vol. 5, no. 4, pp. 586–602, Oct. 2017, doi: 10.1109/TETC.2016.2606384.

[2] V. Hassija, V. Chamola, V. Saxena, D. Jain, P. Goyal, and B. Sikdar, "A Survey on IoT Security: Application Areas, Security Threats, and Solution Architectures," *IEEE Access*, vol. 7, pp. 82721–82743, 2019, doi: 10.1109/ACCESS.2019.2924045.

[3] S. Wachter, "Normative challenges of identification in the Internet of Things: Privacy, profiling, discrimination, and the GDPR," *Computer Law & Security Review*, vol. 34, no. 3, pp. 436–449, Jun. 2018, doi: 10.1016/J.CLSR.2018.02.002.

[4] E. Bertino and N. Islam, "Botnets and Internet of Things Security," *Computer (Long Beach Calif)*, vol. 50, no. 2, 2017, doi: 10.1109/MC.2017.62.

[5] R. Ben Chaabene, D. Ameyed, F. Jaafer, A. Roger, A. Esma, and M. Cheriet, "A Privacy-Preserving Federated Learning for IoT Intrusion Detection System," *9th 2023 International Conference on Control, Decision and Information Technologies, CoDIT 2023*, pp. 351–356, 2023, doi: 10.1109/CODIT58514.2023.10284221.

[6] D. C. Nguyen, M. Ding, P. N. Pathirana, A. Seneviratne, J. Li, and H. Vincent Poor, "Federated Learning for Internet of Things: A Comprehensive Survey," *IEEE Communications Surveys and Tutorials*, vol. 23, no. 3, pp. 1622–1658, Jul. 2021, doi: 10.1109/COMST.2021.3075439.

[7] V. Visoottiviseth, P. Sakarin, J. Thongwilai, and T. Choobanjong, "Signature-based and behavior-based attack detection with machine learning for home IoT devices," *IEEE Region 10 Annual International Conference, Proceedings/TENCON*, vol. 2020-November, pp. 829–834, Nov. 2020, doi: 10.1109/TENCON50793.2020.9293811.

[8] D. Kwon, H. Kim, J. Kim, S. C. Suh, I. Kim, and K. J. Kim, "A survey of deep learning-based network anomaly detection," *Cluster Comput*, vol. 22, no. 1, pp. 949–961, Jan. 2019, doi: 10.1007/S10586-017-1117-8/TABLES/2.

[9] F. Farahnakian and J. Heikkonen, "A deep auto-encoder based approach for intrusion detection system," *International Conference on Advanced Communication Technology, ICACT*, vol. 2018-February, pp. 178–183, Mar. 2018, doi: 10.23919/ICACT.2018.8323688.

[10] C. Yin, Y. Zhu, J. Fei, and X. He, "A Deep Learning Approach for Intrusion Detection Using Recurrent Neural Networks," *IEEE Access*, vol. 5, pp. 21954–21961, Oct. 2017, doi: 10.1109/ACCESS.2017.2762418.

[11] W. H. Chen, S. H. Hsu, and H. P. Shen, "Application of SVM and ANN for intrusion detection," *Comput Oper Res*, vol. 32, no. 10, pp. 2617–2634, Oct. 2005, doi: 10.1016/J.COR.2004.03.019.

[12] M. Al-Qatf, Y. Lasheng, M. Al-Habib, and K. Al-Sabahi, "Deep Learning Approach Combining Sparse Autoencoder with SVM for Network Intrusion Detection," *IEEE Access*, vol. 6, pp. 52843–52856, Sep. 2018, doi: 10.1109/ACCESS.2018.2869577.

[13] C. W. Tien, T. Y. Huang, P. C. Chen, and J. H. Wang, "Using Autoencoders for Anomaly Detection and Transfer Learning in IoT," *Computers 2021, Vol. 10, Page 88*, vol. 10, no. 7, p. 88, Jul. 2021, doi: 10.3390/COMPUTERS10070088.

[14] Z. Fang, D. Zhao, C. Chen, Y. Li, and Y. Tian, "Nonintrusive Appliance Identification with Appliance-Specific Networks," *IEEE Trans Ind Appl*, vol. 56, no. 4, pp. 3443–3452, Jul. 2020, doi: 10.1109/TIA.2020.2994279.

[15] N. D. Lane *et al.*, "DeepX: A Software Accelerator for Low-Power Deep Learning Inference on Mobile Devices," *2016 15th ACM/IEEE International Conference on Information Processing in Sensor Networks, IPSN 2016 - Proceedings*, Apr. 2016, doi: 10.1109/IPSN.2016.7460664.

[16] S. A. Raj, P. P. Amritha, Sethumadhavan, and S. Seshadhri, "Hybrid Intrusion Detection System for Industrial Control System," *Smart Innovation, Systems and Technologies*, vol. 379, pp. 573–583, 2024, doi: 10.1007/978-981-99-8612-5_47.

[17] T. Song, R. Li, B. Mei, J. Yu, X. Xing, and X. Cheng, "A Privacy Preserving Communication Protocol for IoT Applications in Smart Homes," *IEEE Internet Things J*, vol. 4, no. 6, pp. 1844–1852, Dec. 2017, doi: 10.1109/JIOT.2017.2707489.

[18] R. C. Green, C. Zhang, and R. Green, "Communication security in internet of thing: Preventive measure and avoid DDoS attack over IoT network," *SpringSim*, 2015, Accessed: Apr. 01, 2024. [Online]. Available: https://www.researchgate.net/publication/282375085

[19] C. V. Zhou, S. Karunasekera, and C. Leckie, "A peer-to-peer collaborative intrusion detection system," *2005 13th IEEE International Conference on Networks jointly held with the 2005 7th IEEE Malaysia International Conference on Communications, Proceedings*, vol. 1, pp. 118–123, 2005, doi: 10.1109/ICON.2005.1635451.

[20] R. Janakiraman, M. Waldvogel, and Q. Zhang, "Indra: A peer-to-peer approach to network intrusion detection and prevention," *Proceedings of the Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises, WETICE*, vol. 2003-January, pp. 226–231, 2003, doi: 10.1109/ENABL.2003.1231412.

[21] S. Roy, J. Li, and Y. Bai, "A Two-layer Fog-Cloud Intrusion Detection Model for IoT Networks," *Internet of Things*, vol. 19, p. 100557, Aug. 2022, doi: 10.1016/J.IOT.2022.100557.

[22] S. A. Rahman, H. Tout, C. Talhi, and A. Mourad, "Internet of Things intrusion Detection: Centralized, On-Device, or Federated Learning?," *IEEE Netw*, vol. 34, no. 6, pp. 310–317, Nov. 2020, doi: 10.1109/MNET.011.2000286.

[23] M. A. Ferrag, O. Friha, L. Maglaras, H. Janicke, and L. Shu, "Federated Deep Learning for Cyber Security in the Internet of Things: Concepts, Applications, and Experimental Analysis," *IEEE Access*, vol. 9, pp. 138509–138542, 2021, doi: 10.1109/ACCESS.2021.3118642.

[24] V. Mothukuri, P. Khare, R. M. Parizi, S. Pouriyeh, A. Dehghantanha, and G. Srivastava, "Federated-Learning-Based Anomaly Detection for IoT Security Attacks," *IEEE Internet Things J*, vol. 9, no. 4, pp. 2545–2554, Feb. 2022, doi: 10.1109/JIOT.2021.3077803.

[25] V. Rey, P. M. Sánchez Sánchez, A. Huertas Celdrán, and G. Bovet, "Federated learning for malware detection in IoT devices," *Computer Networks*, vol. 204, p. 108693, Feb. 2022, doi: 10.1016/J.COMNET.2021.108693.

[26] P. Ruzafa-Alcazar *et al.*, "Intrusion Detection Based on Privacy-Preserving Federated Learning for the Industrial IoT," *IEEE Trans Industr Inform*, vol. 19, no. 2, pp. 1145–1154, Feb. 2023, doi: 10.1109/TII.2021.3126728.

[27] S. Roy, J. Li, and Y. Bai, "Federated Learning-Based Intrusion Detection System for IoT Environments with Locally Adapted Model," *Proceedings - 2023 IEEE 10th International Conference on Cyber Security and Cloud Computing and 2023 IEEE 9th International Conference on Edge Computing and Scalable Cloud, CSCloud-EdgeCom 2023*, pp. 203–209, 2023, doi: 10.1109/CSCLOUD-EDGECOM58631.2023.00043.

[28] T. Li, A. K. Sahu, A. Talwalkar, and V. Smith, "Federated Learning: Challenges, Methods, and Future Directions," *IEEE Signal Process Mag*, vol. 37, no. 3, pp. 50–60, Aug. 2019, doi: 10.1109/MSP.2020.2975749.

[29] D. Li and J. Wang, "FedMD: Heterogenous Federated Learning via Model Distillation," Oct. 2019, Accessed: Apr. 01, 2024. [Online]. Available: https://arxiv.org/abs/1910.03581v1

[30] Z. Zhu, J. Hong, and J. Zhou, "Data-Free Knowledge Distillation for Heterogeneous Federated Learning," *Proc Mach Learn Res*, vol. 139, pp. 12878–12889, May 2021, Accessed: Apr. 02, 2024. [Online]. Available: https://arxiv.org/abs/2105.10056v2

[31] B. Noune, P. Jones, D. Justus, D. Masters, and C. Luschi, "8-bit Numerical Formats for Deep Neural Networks," Jun. 2022, Accessed: Apr. 02, 2024. [Online]. Available: https://arxiv.org/abs/2206.02915v1

[32] Y. Meidan *et al.*, "N-BaIoT: Network-based Detection of IoT Botnet Attacks Using Deep Autoencoders," *IEEE Pervasive Comput*, vol. 17, no. 3, pp. 12–22, May 2018, doi: 10.1109/MPRV.2018.03367731.

[33] O'Shea and R. Nash, "An Introduction to Convolutional Neural Networks," *Int J Res Appl Sci Eng Technol*, vol. 10, no. 12, pp. 943–947, Nov. 2015, doi: 10.22214/ijraset.2022.47789.