Hierarchical Federated Transfer learning and Digital Twin Enhanced Secure Cooperative Smart Farming

Lopamudra Praharaj*, Maanak Gupta†, and ‡Deepti Gupta
*†Department of Computer Science, Tennessee Tech University, Cookeville, TN, USA
‡Texas A&M University - Central Texas, TX, USA
*lpraharaj42@tntech.edu, †mgupta@tntech.edu, ‡d.gupta@tamuct.edu

2023 IEEE International Conference on Big Data (BigData) | 979-8-3503-2445-7/23/$31.00 ©2023 IEEE | DOI: 10.1109/BigData59044.2023.10386345

***Abstract*—The agriculture industry is extensive utilizing AI and data-driven systems for efficiency and automation, with the goal to meet the rising food demand. Individual farm owners can leverage agricultural cooperatives to consolidate resources, exchange data, and share domain knowledge. These cooperatives can enable the generation of AI-supported insights for their member farmers. However, this collaborative approach has raised concerns among individual smart farm owners regarding cybersecurity threats, and privacy. A cybersecurity breach not only endangers the farm attacked but can also risks the entire network of smart farms members within the cooperative. In this research, we emphasize security challenges within cooperative smart farming and introduce a multi-layered architecture incorporating Digital Twins (DT). Further, we introduce a hierarchical federated transfer learning framework designed to address and mitigate the security threats in collaborative smart farming. Our approach leverages Federated Learning (FL) based Anomaly Detection (AD), which operate on edge servers, enabling the execution of AD models locally without exposing the farm's data. This localization also has excellent generalization ability, which can highly improve the detection of unknown cyber attacks. We employ a hierarchical FL structure that supports aggregation at various levels, fostering multi-party collaboration. Furthermore, we have devised an approach that integrates Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) models, complemented by transfer learning. The objective is to expedite training duration while upholding high accuracy levels. To illustrate the efficiency of our proposed architecture, we present a use case to demonstrate our model's capabilities. Furthermore, we also present a proof-of-concept implementation of our proposed architecture within Amazon Web Services (AWS) environment, reflecting real-world feasibility.**

***Index Terms*—Federated Learning, Anomaly Detection, Cooperative Smart Farming, Security, Privacy, CNN-LSTM, Transfer Learning, Digital Twin, Amazon Web Services (AWS)**

I. Introduction

According to the United Nations' Department of Economic Social Affairs, by year 2050, the global population is projected to reach approximately 9.1 billion, marking a substantial increase of around 34% compared to today's figures [1], [2]. The projected population growth is expected to lead to a simultaneous 70% increase in the global demand for food. In response to this challenge, precision agriculture, also referred to as digital agriculture, emerges as a critical solution to ensure food security at a global level. Precision agriculture encompasses the implementation of technology-driven, data-centric, and sustainable farm management systems. It requires adopting contemporary information technologies, software tools, and intelligent embedded devices to provide decision support for agricultural practices [3]–[6].

Cooperative (co-op) farming practice is extensively used where the farmers pool there resources and share them as per need from the members. These co-opss offer several benefits to its members including resource sharing, machine use and maintenance, hiring farm labour, specialized machine operators, coordinating market visits, estimating price/purchase data, etc. They [7], [8] also aid the member farms by alerting them to crop diseases, pest management, weather, changing labour costs, price fluctuations, etc. As the farming community adopts more precision agriculture practices, the working of co-ops have also evolved. Most recently, the concept of smart co-ops has been gaining momentum, which helps the member farmers to improve their productivity, sustainability, and profitability while providing valuable data-driven insights for better decision-making. However, as these farming co-ops become more connected and smart by aggregating shared data and resources from member farms, they are increasingly becoming a prime target for cyberattacks, with far-reaching consequences for rural communities' well-being and essential infrastructure like supply networks.

According to the Federal Bureau of Investigation (FBI) [9], most of these cyberattacks happen during planting and harvesting seasons, and lead to the theft of sensitive data and operational disruptions, potentially resulting in financial losses and food shortages. Notably, in 2021, a ransomware attack targeted meat producer JBS and two-grain purchasers in the United States during the harvest season. The cybersecurity of the farm and agribusiness sectors gained significant attention following these incidents. In September 2021, the BlackMatter ransomware struck Iowa's new co-op, demanding a ransom of $5.9 million. The company had to take vulnerable machines offline to prevent the ransomware from spreading further. Shortly after the new co-op incident, Crystal Valley co-op, a prominent agricultural co-op in Minnesota, fell victim to an as-yet-unidentified cyberattack strain [9]. This attack disrupted the company's ability to process major credit card transactions.

It should be noted that the implications of cyberattacks on a co-op extend beyond individual farms and can have a far-reaching impact on the entire co-op network. If a malicious actor manages to corrupt or manipulate data on one member farm, it can adversely affect all the member farms. For example, consider a scenario where *Farm A* deploys sensors to monitor soil moisture, temperature, and humidity and shares

the generated data through a central platform managed by co-op. The nearby farms rely on this shared data to make informed irrigation, fertilization, and pest management decisions. Let's consider that *Farm A* falls victim to a network cyberattack. The attacker gains unauthorized access to the network infrastructure and manipulates the transmitted sensor data. They may manipulate the soil moisture readings, transmitting false data that suggests the soil is adequately moist when, in reality, it's dry. As a result, the other farms within the co-op receive the manipulated data, assuming the soil moisture levels are correct. Relying on this deceptive information, farmers may postpone or reduce their irrigation efforts, resulting in insufficient water supply to their crops. The consequences for the affected farms may include crop stress, diminished yields, or in severe cases, crop failure due to the reliance on inaccurate shared data. Moreover, if the attacker continues to compromise the network, it can access other shared resources, such as cloud platforms or collaborative tools. This results in the disruption of shared services, compromised data integrity, spread of false information, loss of trust, and financial impact among the smart farms. In addition, the farmer will eventually refrain from joining the co-ops, which will impact adoption of precision agriculture approaches among agriculture.

To address this issue, we propose a hierarchical federated transfer learning approach in a multi-layered smart co-op architecture with DT to detect cyberattacks induced anomalies. The key contributions to this work are as follows:

1) *Problem Identification:* We identify and delineate the challenges associated with cooperative smart farming. We illustrate these issues through a practical use-case scenario to make them tangible.
2) *Hierarchical Federated Transfer Learning Model:* We introduce a novel hierarchical federated transfer learning model, which combines convolutional neural networks (CNN) and long short-term memory (LSTM) techniques for anomaly detection in cooperative smart farming.
3) *Practical Application:* We showcase the practical application of our proposed framework, and using a specific use case, demonstrate how our approach can effectively identify and flag cyberattacks induced anomalies.
4) *Implementation Framework:* We present an implementation framework in AWS that leverages DT and edge computing. This framework is designed to facilitate the seamless integration of our architecture for real-time anomaly detection in cooperative smart farming.

The remainder of this paper is organized as follows. Section II discusses relevant literature. Section III addresses challenges associated with co-op smart farming with a use-case scenario. Section VII introduces the DT-enhanced co-op farming architecture and highlights its need and limitations. Section V examines various threats encountered across multi-layered architecture. Section VII defines the building blocks of secure co-op farming. Section VII presents the hierarchical federated transfer learning based framework. Section VIII provides a proof-of-concept implementation within AWS, followed by Section IX discussing open challenges and conclusion in Section X.

II. Related Work and Background

A. Cooperative Smart farming

Cooperatives, often called co-ops, are structured as formal organizations owned and operated by their members. These co-ops unite individual farmers to amplify their business productivity and overall yields. The overarching objective is to improve farming practices, sustainability, and productivity, all while promoting collaboration, the exchange of knowledge, and the efficient utilization of resources.

In co-op, farmers can share sensor data, weather information, and insights with each other and experts, enabling collaborative decision-making and knowledge exchange. The authors [7] discussed various technical foundations and explore potential AI applications that can augment the co-op smart farming ecosystem. In the subsequent work, the authors [10] utilized co-op agriculture ontology to perform data transformation by adding white Gaussian noise to data generated by all individual smart farms. The authors [11] highlighted the issue of unfair collaboration within co-op smart farming, where some smart farms may generate low-quality data to develop machine-learning models and gain advantages over other farms with high-quality data.

B. Collaborative Intrusion Detection System (CIDS)

The concept of an Intrusion Detection System (IDS) in co-op smart farming takes on a collaborative nature. To comprehensively address intrusion detection in this environment, it is essential to explain the core concepts and necessary background information related to the research contribution in CIDS. This subsection defines the foundational principles of CIDS and how the integration of machine learning, federated learning, and blockchain principles are employed within CIDS. A CIDS can address the shortcomings of local IDSs [12]. A CIDS allows the sharing of information and detection of network attacks in a collaborative network. A classic CIDS unit comprises local monitoring, global monitoring, association and aggregation, and data-publishing components [13]. CIDSs are typically developed based on distributed Machine Learning (DML) for detecting known and unknown attacks with some generalization capability [14]. The authors [15] introduced a Privacy-Preserving Machine Learning-Based CIDS designed for Vehicular Ad Hoc Networks (VANETs). The initial step involves utilizing the Alternating Direction Method of Multipliers (ADMM) to establish a decentralized approach for solving the Distributed Empirical Risk Minimization (ERM) problem within a VANET.

FL model guarantees the privacy of general DML algorithms [16]. The authors [12] proposed a novel software-defined VANET IDS, referred to as SDVN, which blends FL and Software-Defined Networking (SDN) for training detection models. In this proposed CIDS, various SDN clients train models within their sub-networks and upload them to a centralized cloud server for model aggregation.

Blockchain solutions can be used to enhance trust within CIDS in network and cloud environments. To illustrate, the authors [17] conducted a survey exploring methods to integrate CIDS with blockchain technology. They introduced the idea of leveraging blockchain techniques to enhance the

reliability of CIDS, emphasizing that blockchain's characteristics can promote trust between different IDS and provide accountability. The authors [18] conducted a survey to offer a comprehensive overview of cutting-edge methodologies in predicting cybercrime, leveraging Machine Learning (ML), Deep Learning (DL), and Transfer Learning (TL). In [19], the authors employed statistical analysis methods and machine learning models for predicting different types of crimes in New York City, based on 2018 crime datasets. The integration of Hierarchical FL with DT technology is presented in [20], [21].

Our research distinguishes itself from previous studies within co-op smart farming scenarios in several notable ways. We leverage DT technology to facilitate continuous real-time simulation and monitor traffic data. DT also helps with vulnerability assessment and simulation of attack scenarios within smart farms. Our investigation identified research gaps linked to DT-enabled smart farming and illustrated these gaps through diverse use cases. Moreover, we utilize federated transfer learning, a resilient solution designed to address data distribution challenges effectively across various farms in co-op smart farming environments. FL addresses privacy and security concerns, enables Early Threat Identification, and safeguards against zero-day attacks on individual farms. Additionally, our approach incorporates the CNN-LSTM hybrid model, ensuring the precision of AD model. Our methodology also embraces a hierarchical approach that distinguishes between farms at both local and regional levels. This design ensures that DT data for individual farms remains localized on the edge server, with only the updated model weights being transmitted to the cloud server. This innovative approach is a practical solution to address the privacy and security challenges associated with AD in co-op smart farming systems.

III. Cooperative Smart Farming

Cooperative smart farming (CSF), characterized by collaborative data sharing and resource pooling, introduces innovative opportunities for optimizing agricultural practices. However, the interconnected nature of these collaborative environments also raises significant security concerns. This section presents a CSF use case that is aligned with a real-world scenario.

A. Cooperative Smart Farming Use Case

As shown in Figure 1, in this use case, we examine two smart farms referred to as *Smart Farm1* and *Smart Farm2*. *Smart Farm1* uses three sensors: Temp sensor, Humidity sensor, and Ultraviolet sensor. The utilization of temperature and humidity sensors is instrumental in assessing the weather conditions on a rice farm, enabling farmers to make informed decisions about the most suitable crops for cultivation. Furthermore, these sensors play a crucial role in the event of unexpected weather occurrences, such as abrupt temperature fluctuations or excessive rainfall. The Ultraviolet sensors can measure the intensity of UV radiation, which is a component of sunlight. Excessive UV exposure can harm crops, causing sunburn and damage to leaves. By monitoring UV levels, farmers can take preventive measures like shading or adjusting planting times to protect crops from UV stress. These sensor data help *Smart Farm1*'s farmer to improve crop quality, reduce environmental impact, and improve farm management.

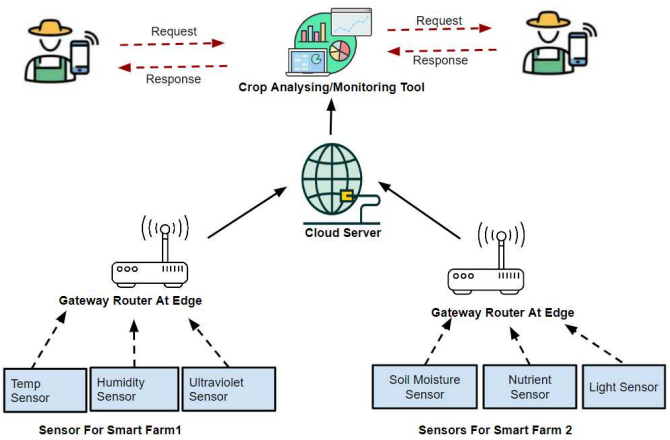

Fig. 1: Cooperative Smart Farming Use-Case

Similarly, *Smart Farm2* uses three sensors: a Soil Moisture sensor, a Nutrient sensor, and a Light sensor. Soil moisture sensors measure the moisture content of the soil, which is a critical parameter for crop health and irrigation management. By monitoring soil moisture sensors, the farmers can avoid under and over-water irrigation, which is crucial for growth and yield optimization. Nutrient sensors measure the levels of essential nutrients like nitrogen, phosphorus, and potassium in the soil. These data help farmers apply fertilizers more accurately, reducing excess fertilization. Light sensors are used to monitor light levels in fields. The light sensor data also help to determine if crops receive sufficient sunlight for photosynthesis and growth. The sensors employed in *Smart Farm1* and *Smart Farm2* aid farmers in gaining valuable insights and enhancing their decision-making for their respective agricultural endeavours. However, *Smart Farm2* may require temperature and humidity data to safeguard against unforeseen weather conditions in its fields. The farm two may not install two additional sensors due to budget constraints. Consequently, a collaborative agreement has been reached between *Smart Farm1* and *Smart Farm2*. They have come together to exchange data, with *Smart Farm1* benefiting from *Smart Farm2* NPK sensor data to improve their crop predictions for the upcoming season. *Smart Farm2* is also beneficial with the temperature and humidity data. This co-op arrangement allows both parties to leverage the available resources more effectively for mutual benefit.

As illustrated in Figure 1, each smart farm is equipped with sensors that generate a vast volume of data. This sensor data is transmitted through a Gateway router, edge to a cloud platform, such as Azure FarmBeats [22]. To access this data, farmers can request sensor data from particular fields through a crop monitoring tool hosted on a cloud server. In response, the requested data is sent via a mobile or desktop application. The privacy and security of collected data heavily rely on the security measures implemented by the manufacturers of the smart devices.

In a broader context, an anomaly encompasses any unexpected or irregular behaviours or events within the infrastructure connecting the smart farms, central server, and mobile/desktop application. These anomalies can potentially

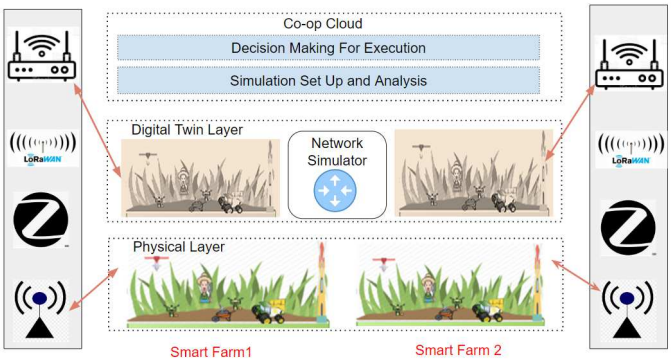

Fig. 2: Digital Twin Enhanced CSF

signify issues or security threats. The following use case outlines instances of unusual behaviour and potential threat scenarios within this CSF environment:

- The soil moisture data transmitted by *Smart Farm 2* is tampered with or altered during the transmission, that is sending false information indicating that the soil is adequately moist when dry. As a result, *Smart Farm 1* receive this manipulated data, assuming the soil moisture levels are suitable. Based on this false information, they may delay or reduce their irrigation practices, leading to insufficient watering of crops.
- Malicious actors flood the co-op farming network with excessive data requests, causing network congestion. *Smart Farm 1* and *Smart Farm 2* may experience disruptions in data sharing and collaboration, leading to delays in decision-making and farming activities.
- A cybercriminal gains unauthorized access to the central data repository through an exploited vulnerability in Farm1's network or server. Once inside the system, the attacker ex-filtrates sensitive data from the central repository. The malicious actor can use this stolen data for various purposes, such as selling sensitive agricultural information to competitors, extorting the co-op, or leveraging the data for financial gain.

IV. Digital Twin Enhanced Co-op Smart Farming

Digital Twin (DT) is a simulation model representing a physical entity in the past, present, and future. The physical entity could be a sensor, device, system, or process [23], [24]. For the past and the present, a DT in the virtual space mirrors the behaviour of an entity in the physical space. Regarding the future, the DT accurately predicts the entity's behaviour, which is essential for the control process. The DTs look for data discrepancies between the physical and virtual entities by collecting massive amounts of data from all phases of the product life-cycle and provide simulation data to the physical entity so that it may improve its calibration and testing procedures [25]. Such recurrent processes improve DT models and their physical equivalents, allowing for more accurate estimate prediction.

In the farmland DT framework context, the object modeled consist of sensor nodes and the gateways responsible for transmitting field data to a cloud platform for analysis [26], [24], [27], [28]. This DT system offers a precise, real-time portrayal of the farm field's condition. It achieves this through visual representations, drone-captured imagery, and data on essential soil parameters like pH, salinity, nitrogen, phosphorus, potassium levels, temperature, and humidity.

DTs offer several advantages when integrated into network anomaly detection in a CSF system (Figure 2). DTs create a virtual replica of the entire network and its components of each smart farm, enabling continuous real-time simulation and monitoring. This means that network activities and anomalies of the individual smart farm can be analyzed as they happen before sending the data to the central server, allowing for immediate threat detection and response [29], [30], [31]. For example, DoS attacks often flood the network with excessive traffic, causing congestion and disruptions. The DT layer can monitor network traffic in real-time and detect unusual traffic patterns for that farm, such as a sudden increase in traffic volume to specific devices or services. Such anomalies can trigger alerts for further investigation. DTs allow for the creation of isolated testing environments where security updates, intrusion detection algorithms, and other changes can be tested without affecting the other smart farm's network [32]. [33]. This reduces the risk of unintended disruptions in the other smart farm's network data. DTs can also model vulnerabilities within the Smart Farm's network. By simulating attack scenarios, farms can identify weak points in their infrastructure that could be exploited in a network attack. This information allows for proactive vulnerability remediation.

Nevertheless, while utilizing DTs for network anomaly detection in CSF has several benefits, it also has inherent limitations. As shown in Figure 2, this model consolidates all network data from sensors, devices, and individual farms within a central repository. The central server is susceptible to single point of failure. The entire network's anomaly detection capabilities may be compromised if the server experiences technical issues or a security breach. As the Co-op grows more prominent, more smart farms add to infrastructure. The central server may be incapable of handling the increased data volume and processing demands, leading to scalability issues. Also, analyzing data in a centralized manner can introduce delays, which could impact farm operations negatively. Individual farms may also have reservations about sharing their network traffic data with others, which poses challenges when designing network anomaly detection systems for CSF. Moreover, when the DT of a system is created, the potential attack surface effectively doubles—adversaries can go after the physical system or attack the DT of that system.

V. Threat Model

In this section, we describe the possible threat situation and motivated our research and led to the development of hierarchical federated transfer learning.

A. Attacks on Physical Layer

The physical layer consists of sensors and gateway devices spread across agriculture farms. These devices include drones flying in the air, autonomous tractors, sensors embedded in

livestock, or hub devices installed to communicate among the DT. Attackers may physically tamper with these sensors, actuators, or other devices in the farming environment. For instance, the attackers may alter sensor readings or sabotage machinery to disrupt farm operations. They could also gain unauthorized control over farm machinery, such as tractors or irrigation systems, causing damage or disrupting farming processes. Jamming devices can disrupt wireless communication used by IoT devices in the field. This interference can cause data loss or affect automated farming processes. They may deploy rogue IoT devices, which could collect data or interfere with operations.

B. Attacks on Network layer

A DT (resides in the edge cloud) continuously receives data from its physical counterpart to provide an up-to-date virtual model, and the virtual model can also provide feedback to the physical world through the same communication channel. The data transmission from or to the cloud may face serious threats. Threats on data communication may be divided into five main types such as: Man in the Middle Attack, Denial of Service or Distributed Denial of Service, Eavesdropping, Spoofing, and Replay Attack.

In a Man-in-the-Middle (MITM) attack, the attacker can insert malicious code between communicating nodes or eavesdrop on the ongoing conversation between any two communicating nodes [34]. MITM vulnerabilities extend across various protocols and technologies inherent in smart farming systems. For instance, security vulnerabilities within the Wi-Fi standard widely used in Smart farming are the primary cause of the wireless re-installation attack, which can expose DT data. Smart farming systems utilizing Bluetooth and the ZigBee Protocol are also susceptible to exploitation through MITM attacks [35], [36].

Denial of Service (DoS) [37] and Distributed Denial of Service (DDoS) [38] attacks endanger the availability and accessibility of DTs in Smart farming environments. These attacks happen by overloading the farming network with excessive traffic. By this attack, a malicious actor can disrupt operations within the deployed DT in Smart farming, potentially rendering essential services inaccessible to farmers. The unavailability of these services can potentially disturb the functionality of the DT in the Smart farming system.

In an eavesdropping attack, the network traffic flowing from sensors to controllers is susceptible to interception [39]. This passive attack allows the attacker to gain insights into the communication between sensors and controllers.

In a spoofing attack, the attacker masks their identification to engage in malicious and deceitful actions [39]. The attacker manipulates a node within the wireless sensor network to achieve network access or reroute network traffic. Spoofing has the potential to corrupt signals or messages transmitted from sensors to the controller.

A Replay Attack is a type of post-attack that relies on a prior preparatory phase. During this preliminary stage, the attacker observes, captures, and stores a specific data set to resend it later [40], [41]. Consequently, when this re-transmitted data or signal is employed, it can potentially trigger harmful actions on sensors or controllers within the system.

C. Attacks on Virtual Layer

In anomaly-based intrusion detection tools, farmers (both benign and malicious) can continuously monitor the ongoing process to observe the excepted behaviour of farming objects. The smart farming system uses the DT concept, connecting real-time data (dynamic variables) with historical data (virtual process) to prevent rule violations related to safety and security. For example, by analyzing the relationship between the dynamic variable (pest activity) and the historical variable (past pesticide applications and weather conditions), the smart farming system can detect a potential S&S rule violation. The benign user (Farmer) uses the information to spot deviations from a defined or learned baseline and alert security analysts. However, malicious users can exploit the data and the co-relation of variables to disrupt the DT's behaviour such that twins do not follow the expected misbehaviour. If the attacker successfully attacks DT, no anomaly can be detected; thus, it is difficult to identify long-term deviations in the network.

To replicate the simulation of a DT network, a network simulation tool proves invaluable. A network DT serves as a computer-based model that encompasses the communication network, its operational surroundings, and the application traffic it carries. This network DT proves highly useful for studying the behavior of its physical counterpart across various operating scenarios, even including cyberattacks, all within a low-cost, zero-risk environment.

However, it's important to recognize that if an attacker manages to intercept and gain insights into the traffic generated by the network simulation tool, they could potentially manipulate not only the DT but also its physical counterpart. This opens the door for the attacker to execute various attacks as outlined in section V-B.

D. Attacks on cloud Storage

Most data storage activities in DT applications are conducted within cloud computing environments. Numerous trust-related concerns and privacy issues are associated with data storage, mainly when it involves public cloud services where the service provider company holds complete control.

VI. Building Blocks For Secure Co-op Smart Farming

This section describes the fundamental components of our proposed model and underscores their significance within our cooperative farming system. It provides a concise overview of FL, Transfer learning and federated transfer learning in our proposed approach.

A. Federated Learning (FL)

FL is a communication-efficient process for training neural networks on decentralized data. FL process comprises a central server and a group of clients, each equipped with a predefined local dataset. Such a process consists of several rounds of FL in which the server selects a random number of clients and sends them to the neural network model for local training. The selected client trains the model with the local data and sends it back to the server, which integrates all the updates with the global model. This process is iterated several times until the test accuracy is reached. The central concept of this

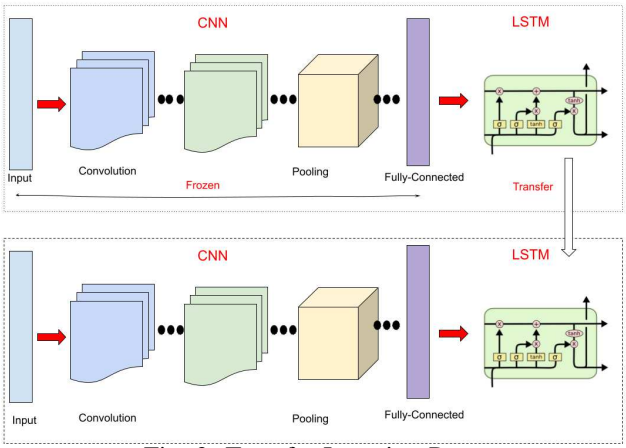

Fig. 3: Transfer Learning Process

process is the aggregation of local updates and the amount of computation performed at each round [42], [43].

In our proposed model, FL is used to share attack and anomaly profiles with other smart farms, enhancing the development of a comprehensive and versatile model capable of recognizing various attack patterns and behaviors without sharing the raw data itself. FL enables an anomaly detection system to adapt quickly to new and evolving cyberthreats across smart farms in the co-op. Imagine that one of the Smart Farms faces a novel malware strain, specifically targeting agricultural automation systems. This malware was previously unidentified during the initial model creation. Through FL, the central aggregator combines this update with other farms to create an improved model that now includes detection capabilities for this new threat. As a result, all farms in the co-op benefit from the rapid adaptation of the anomaly detection system to this emerging cyberthreat. The collaborative nature of FL also enables farms to collectively identify cybersecurity threats and zero-day attacks early, providing more time to respond and mitigate potential damage. If one of Smart Farms identifies an unusual sensor data pattern indicating possible tampering with irrigation controls. Through FL, all farms in the network can now detect this new threat pattern, even if it is a zero-day attack that has not been seen before in the broader cybersecurity community. This rapid sharing and adaptation help in early threat identification and response across the co-op smart farming network.

B. CNN-LSTM Model with Transfer Learning

A Convolutional Neural Network (CNN) typically comprises several key layers: an input layer, a convolutional layer, a pooling layer, a fully connected layer, and an output layer. The input time series data is processed through convolutional kernels. The pooling layer is placed after the convolutional layer, and this pooling operation helps in reducing the number of connections between the convolutional layers while also aiding in downscaling the time series data. Subsequently, a fully connected layer summarizes the local features extracted by all the convolutional units. CNNs can automatically learn features from the data and feature local connectivity, weight sharing, pooling operations, and multi-layer structures. These characteristics help reduce complexity, mitigating overfitting and improving the model's generalisation ability.

The fundamental component of an LSTM network comprises three essential elements: forgetting gates, input gates, and output gates (Figure 3). These elements play distinct roles in processing input data and managing the network's memory.

1) Forgetting Gates: The input values, denoted as x_i, are integrated into the forgetting gate alongside the previous state memory unit s_{t-1} and the intermediate output h_{t-1}. They collectively contribute to forgetting part of the state memory unit.
2) Input Gates: The input values x_i transform the sigmoid and tanh functions within the input gate. These transformations jointly determine the retention vector within the state memory cell. This retention vector decides which information should be stored or updated in the cell state.
3) Intermediate Output: The intermediate output h_t is calculated based on the updated state memory S_t, in combination with the output O_t. Calculating the output O_t follows a specific procedure outlined in the Equation below.

$$
\begin{aligned}
f_t &= \sigma\left(W_{fx}x_t + W_{fh}h_{t-1} + b_f\right) \\
i_t &= \sigma\left(W_{ix}x_t + W_{ih}h_{t-1} + b_i\right) \\
o_t &= \sigma\left(W_{ox}x_t + W_{oh}h_{t-1} + b_o\right) \\
g_t &= \tau\left(W_{gx}x_t + W_{gh}h_{t-1} + b_g\right) \\
S_t &= g_t.i_t + S_{t-1} \cdot f_t \\
h_t &= \tau\left(S_t\right).o_t
\end{aligned}
$$

Where $f_t, i_t, o_t, g_t, h_t, S_t$ in Eq. describes the states of oblivion gate, input gate, output gate, input node, intermediate output and state unit. $W_{fx}, W_{fh}, W_{ix}, W_{ih}, W_{ox}, W_{oh}$ denote the matrix weights of input x_t multiplied by the intermediate output h_{t-1} in the corresponding gate. b_f, b_i, b_o, b_g denote the bias in the corresponding gate. σ, τ represent the sigmoid and tanh activation functions. . represents the dot product of matrix elements. LSTM network utilizes these components to control the flow of information and manage memory, allowing it to effectively capture and retain relevant patterns and dependencies in sequential data.

In the depicted transfer learning process (as shown in Figure 3), the proposed hybrid CNN-LSTM model comprises two components: the CNN and the LSTM. The CNN handles feature extraction, while the LSTM focuses on network anomaly classification. Specifically, the CNN model excels at extracting valuable features from network traffic data, emphasizing its ability to capture essential patterns. In contrast, the LSTM network's primary function is analyzing historical time series relationships among quality indicators. To optimize this model for smart farming applications, we maintain the Convolution layer in a frozen state and adjust the parameters of the LSTM layer, a customization that aligns with individual preferences within the smart farming framework. This strategic approach significantly reduces training time by transferring trainable parameters into non-trainable ones. The proposed technique streamlines the model's training process, allowing for efficient customization of the LSTM component while leveraging the pre-learned features from the CNN for enhanced performance in smart farming scenarios.

C. Federated Transfer Learning

The Federated learning process solves data privacy and collaborative learning, but another crucial issue is data heterogeneity. Suppose we directly apply the server model to the client. In that case, it still performs poorly due to greater data distribution between different farms' networks and the unique characteristics of each IoT network within the co-op. Moreover, the server model typically learns coarse features from a large dataset of traditional network traffic but cannot capture the finer details specific to individual IoT networks within the co-op smart farming environment. Therefore, after obtaining the server model, the individual farm can perform transfer learning to get a personalized client model.

In our model, Federated transfer learning is used to minimize the training time while maintaining high accuracy. The model training for federated transfer learning mainly includes six steps, as discussed below.

1) First, the server model is trained according to the public network traffic dataset and distributed to all client farms.
2) Then each local farm can train its model on its own network traffic dataset. In this step, the data distribution between the server and the client farm is different. Transfer learning is performed to reduce the training time for each client.
3) Each client model computes the logits based on the public dataset as the input. The logits are the intermediate values in a CNN's output layer before they are transformed into probabilities for classification. It represents how confident the network is about each class before making the final decision.
4) Each client farm uploads the logits to the server.
5) The server integrates them and transmits the new logit to the farm clients.
6) Each client trains their model on the public dataset to make its logit approach to new logits. After that, each of the farm client models trains again on a private dataset for a few epochs to get a personalized client model.

The step 3 to 6 are repeated throughout the training process till the desired accuracy has been achieved. After the training process, the personalized network intrusion model generated in the final transfer learning process is used to detect network intrusion. The detailed algorithm for federated transfer learning is provided in Algorithm 1.

Algorithm 1: Federated Transfer Learning Algorithm for Cooperative Smart Farming

Data: Private Datset D_{PR}, Public Dataset D_{PL}
Result: Trained Model f_k , k= 1,2,3..n
Initialization: Train the CNN-LSTM Model f_s with public Datset D_{PL} on Cloud;
`// The CNN-lSTM is used for federated transfer learning`
Distribution: The server model f_s is distributed to all the client farms;
`//` f_s `represent the sever model`
Transfer learning: Each client farm trains their client model f_k on public and private dataset D_{PL} and D_{PR} using(4);
Federated Process:
for *Each round 1, 2..r* **do**
  Each client farm calculate their logits l_k on public dateset D_{PL} and upload it to cloud;
  The cloud aggregates the logits l_k of all the client farm and calculate the average logits l_{avg} ;
  The cloud sends l_{avg} to all the client farms;
  Each client farm trains their client model f_k on the public dataset D_{PL} to make its logit close to l_{avg} ;
  Transfer learning: Each client model trains client model f_k on private dataset D_{PR} again based on fine tune;
end for;

VII. Proposed Hierarchical Federated Transfer Learning Framework

The proposed architecture demonstrates a real-world scenario of CSF where multiple farms signed a cooperative agreement between them. We consider that there are four smart farms named as *Smart Farm 1*, *Smart Farm 2*, *Smart Farm 3* and *Smart Farm 4*. Figure 4 illustrates the use case according to our proposed system model.

Integrating IoT technologies and implementing smart farming practices in CSF ecosystem are governed by shared agricultural policies and regulations, ensuring consistency and standardization across different agricultural regions within the smart farming network. Each agricultural region within the smart farming network may have multiple smart farms, contributing to the overall intelligent agriculture framework. Illustrating this scenario, consider two smart farms, *Smart Farm 1* and *Smart Farm 2*, operating within Region 1 of the CSF system and *Smart Farm 3* and *Smart Farm 4* operating within region 2. Each Smart farm allows for diverse Smart farming technologies and practices with unique features, capabilities, and innovations. These solutions may encompass advanced sensors, communication systems, and intelligent algorithms to optimize agricultural operations and align with the overarching goals of smart farming.

Various Smart Farms require distinct computing capabilities, leading to establishment of a localized computing resource known as "Edge Server". The Edge server also helps to facilitate efficient data processing and AD within each smart farm. Each smart farm within the region can host and deploy the DT and network simulation tool. The DT can help in continuous real-time simulation and monitor traffic data described in section . A network simulation tool represents the network topology, traffic load, and the benign and malicious traffic flowing in their network. Information is collected from DT, and the network simulation tool at this juncture aids in examining and detecting malicious activities within the smart farming network.

A Packet auditor tool is used, which facilitates network traffic data synchronization between the DT and IoT devices. The PA tool verifies the authenticity and integrity of incoming

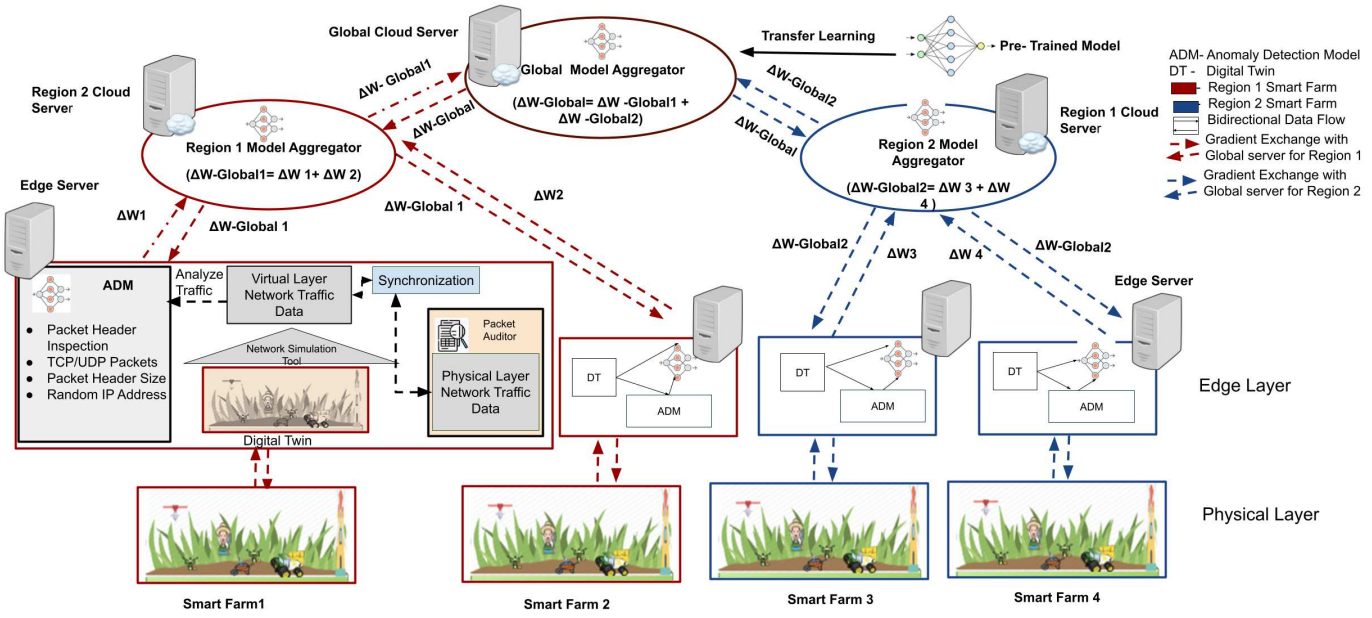

Fig. 4: Proposed Framework for Secure Cooperative Smart Farming

packets. It checks that the data packets have not been altered or tampered with during transmission, ensuring the accuracy of the information used for anomaly detection. Providing the data's security prevents potential attacks like man-in-the-middle attacks, which could compromise the quality and reliability of the data used to detect anomalies. Features are collected from the DT and network DT that include TCP and UDP packets, packet time to arrive, open or closed connections, IP addresses, and sensor data are used for analysis.

The network traffic data and sensor data associated with network DT and DT train an AD model for Smart Farms within Region 1. Similarly, Smart Farm 3 and Smart Farm 4, operated within Region 2, can have their AD Models. There's a strong emphasis on fostering collaboration among these models to reinforce the efficiency of this anomaly detection model. In this scenario, the anomaly detection models deployed within each smart farm region possess the capacity to engage in collaborative learning facilitated by FL described in section . FL empowers these models to exchange knowledge and insights while preserving the region's privacy and security of agricultural data. A more resilient and precise anomaly detection model can be constructed by consolidating the acquired expertise from the individual models through techniques like weighted aggregation and CNN-LSTM-based transfer learning described in section . This collective intelligence enhances Smart Farm's overall anomaly detection capabilities for improved farm management and productivity.

In CSF, collaboration goes beyond individual farming regions. For example, the anomaly detection model in Region 1 can work together with the one in Region 2 using a concept called Hierarchical Federated Learning (HFL) [44]. This collaboration happens by exchanging model information, represented as gradients, on a shared multi-cloud server. These gradients represent shared knowledge that can boost the performance of both anomaly detection models. Using the collaborative power of FL and sharing these gradients on the multi-cloud server, the anomaly detection models in different farming regions can learn from each other's experiences and insights. In Figure 4, *Smart Farm 1* and *Smart Farm 2* collaborate together by exchanging their local gradients (Δw1, Δw2) with Region server-1 to build robust anomaly detection model. Similarly, *Smart Farm 3* and *Smart Farm 4* exchange their local gradients (Δw3, Δw4) with region server-2 using hierarchical federated approach. The smart farm's global gradients (Δw-Global1, Δw-Global2) aggregate at the Federated multi-cloud server to build a global model Δw Global at the top layer. This cross-regional collaboration improves anomaly detection across the entire smart farming network by combining knowledge from different areas and equipment behaviours, making the system more effective.

VIII. Proof of Concept in AWS

In this section, we discuss the prototype implementation, which has the potential for further expansion to accommodate a substantial number of sensor devices and has the capabilities to suit a wide array of farms. Our approach involves FL, which mandates that the training data remains on the edge devices. Edge computing also guarantees swift data retrieval and accelerated data processing. AWS offers an array of services designed to facilitate data processing, analysis, and storage in proximity to your endpoints. This setup enables the deployment of APIs and tools in locations beyond AWS data centres, fostering the creation of high-performance applications that can process and store data near its source [45].

We have opted for AWS Green Grass Group [46] deployment at the edge to meet these requirements. As depicted in Figure 5, this framework encompasses an array of components, including sensor devices, Docker containers, Lambda functions, Raspberry Pi, and Streamline Databases. We deploy a diverse set of sensor devices, such as Temperature Sensors, Humidity Sensors, Soil Moisture Sensors, Barometric Sensors, and Light Sensors. Specifically, we designate these sensors as

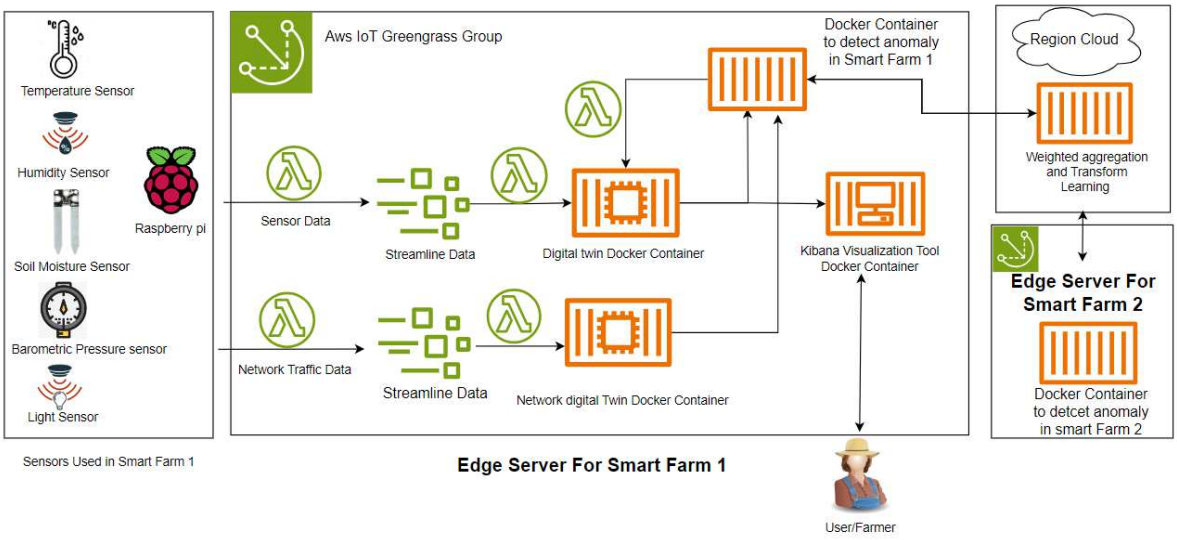

Fig. 5: Proof of Concept Implementation of Secure Cooperative Smart Farming in AWS

follows: DHT11 Temperature Humidity Sensor, AITRIP Capacitive Soil Moisture Sensor, Adafruit BMP390 - Barometric Pressure Sensor, and GL5528 Light Sensor. Within this setup, a user-defined Lambda function operates on the Greengrass core, ingesting time-series data and subsequently storing it within the Streamline data storage. Simultaneously, another Lambda function operates in parallel to transmit network traffic data to the Streamline data storage.

We employ a container-based application, deploye using the Docker application deployment connector, to execute the DT. This functionality is facilitated through the utilization of a Docker Compose file. Container images, whether stored in public or private repositories like Amazon Elastic Container Registry (Amazon ECR) or Docker Hub, can be referenced for this purpose [47]. Notably, we adopt distinct containers for the DT and the network analysis tool. Additionally, we have incorporated Lambda functions to guarantee no packet loss or data loss throughout the establishment of the network DT or DT module. The data generated by both containers is directed to the docker container for anomaly detection. Subsequently, users or farmers can access visualizations through a tool like Kibana [48] (Figure 5). To enhance the accuracy of local AD models, the local parameters of the anomaly detection model are shared with a cloud server through a grouping approach. Our proposed model effectively identifies anomalies and triggers alerts to the farm's DT via Lambda functions, as depicted in Figure 5.

IX. Open Challenges

A. Robustness to Environmental Factors

"Robustness to Environmental Factors" involves improving anomaly detection models to effectively distinguish between real anomalies and typical, natural variations in the farming environment. The goal should be to reduce the number of false alarms triggered by the system. Environmental conditions in farming are inherently subject to regular fluctuations. For instance, temperature, humidity, and light levels fluctuate throughout the day and with changing seasons. Soil moisture levels can vary due to factors like rainfall or irrigation schedule. Without robustness to environmental factors, the anomaly detection system may generate numerous false alarms when faced with typical, natural environmental variations.

B. Adversarial Attack

In our approach, we have used hierarchical federated transfer learning for anomaly detection, employing machine learning to identify anomalies within each farm. However, it's essential to acknowledge that machine learning systems are susceptible to various attacks. These attacks can occur during both the training and testing/inferring phases and potentially undermine the performance of the machine learning system, leading to sub-optimal decision-making. These attacks fall into several categories, including Poisoning Attacks, Impersonation Attacks, Evasion Attacks, and Inversion Attacks [49].

C. Explainable AI

Explainable AI is an approach in artificial intelligence that focuses on making machine learning and AI models more transparent and understandable. When applied to anomaly detection in co-op smart farming, it entails developing a model beyond anomaly detection. These models should be designed to explain the classification of a specific data point or event as an anomaly. This explanation is essential because it helps farmers to understand and trust the system's decisions.

X. Conclusion

CSF is emerging as a critical driver in the agricultural sector, with a strong potential to support precision agriculture practices and contribute to a country's GDP. As more farmers adopt these smart co-ops, it is essential to make them secure and trustworthy. In this research paper, we explore cooperative smart farming comprehensively and identify key challenges to make this ecosystem secure. First, we propose a DT-enhanced cooperative smart farming multi-layered architecture. Next, we identify potential cyberthreats across layers and explain conceivable attack scenarios. To counter these threats, we propose a hierarchical federated transfer learning approach and present a proof-of-concept implementation highlighting this

architecture integration in both edge and AWS environments. We conclude by discussing open challenges. The solutions put forth in our framework hold significant promise for the adoption of smart farming technology. For future work, we will transition from a prototype to real-world deployment, thus facilitating further exploration and investigation in this evolving field.

ACKNOWLEDGEMENT

This work is partially supported by the NSF grants 2230609 and 2226612.

REFERENCES

[1] M. Roser, "Future population growth," *Our world in data*, 2013.
[2] H. C. J. Godfray *et al.*, "Food security: the challenge of feeding 9 billion people," *science*, 2010.
[3] N. Zhang *et al.*, "Precision agriculture—a worldwide overview," *Computers and electronics in agriculture*.
[4] V. A. Hakkim *et al.*, "Precision farming: the future of indian agriculture," *Journal of Applied Biology and Biotechnology*.
[5] A. Dobermann and K. Cassman, "Plant nutrient management for enhanced productivity in intensive grain production systems of the united states and asia," *Plant and soil*.
[6] M. Gupta *et al.*, "Security and privacy in smart farming: Challenges and opportunities," *IEEE Access*, vol. 8, pp. 34 564–34 584, 2020.
[7] S. S. L. Chukkapalli *et al.*, "Ontologies and artificial intelligence systems for the cooperative smart farming ecosystem," *IEEE Access*, 2020.
[8] *Threats to precision agriculture*. [Online]. Available: https://static1.squarespace.com/static/55e8e9ece4b09a2da6c9b923/t/5bce1e7ee2c48363bc5d728f/1540234885121/2018+CHS+Threats+to+Precision+Agriculture.pdf/
[9] *Cyber Attacks On the Rise in the Agriculture Industry*. [Online]. Available: "https://edgelabs.ai/blog/cyber-attacks-on-the-rise-in-the-agriculture-industry/"
[10] S. S. L. Chukkapalli *et al.*, "A privacy preserving anomaly detection framework for cooperative smart farming ecosystem," in *IEEE International Conference on Trust, Privacy and Security in Intelligent Systems and Applications*.
[11] D. Gupta, P. Bhatt, and S. Bhatt, "A game theoretic analysis for cooperative smart farming," in *2020 IEEE International Conference on Big Data (Big Data)*. IEEE, 2020, pp. 2350–2359.
[12] J. Cui *et al.*, "Collaborative intrusion detection system for sdvn: A fairness federated deep learning approach," *IEEE Transactions on Parallel and Distributed Systems*, 2023.
[13] E. Vasilomanolakis *et al.*, "Taxonomy and survey of collaborative intrusion detection," *ACM Computing Surveys*.
[14] H. Huang *et al.*, "Distributed machine learning on smart-gateway network toward real-time smart-grid energy management with behavior cognition," *ACM Transactions on Design Automation of Electronic Systems (TODAES)*.
[15] T. Zhang and Q. Zhu, "Distributed privacy-preserving collaborative intrusion detection systems for vanets," *IEEE Transactions on Signal and Information Processing over Networks*.
[16] Y. Qiang *et al.*, "Federated learning–synthesis lectures on artificial intelligence and machine learning," *Morgan & Claypool Publishers*, 2019.
[17] N. Alexopoulos *et al.*, "Towards blockchain-based collaborative intrusion detection systems," in *Critical Information Infrastructures Security: 12th International Conference, CRITIS 2017, Lucca, Italy, October 8-13, 2017, Revised Selected Papers 12*. Springer.
[18] V. Mandalapu *et al.*, "Crime prediction using machine learning and deep learning: A systematic review and future directions," *IEEE Access*, 2023.
[19] L. Elluri *et al.*, "Developing machine learning based predictive models for smart policing," in *2019 IEEE International Conference on Smart Computing (SMARTCOMP)*. IEEE, 2019, pp. 198–204.
[20] D. Gupta *et al.*, "Hierarchical federated learning based anomaly detection using digital twins for smart healthcare," in *2021 IEEE 7th International Conference on Collaboration and Internet Computing (CIC)*. IEEE, 2021, pp. 16–25.
[21] D. Gupta, S. S. Moni, and A. S. Tosun, "Integration of digital twin and federated learning for securing vehicular internet of things," in *Proceedings of the 2023 International Conference on Research in Adaptive and Convergent Systems*, 2023, pp. 1–8.
[22] *Overview of Azure FarmBeats*. [Online]. Available: https://learn.microsoft.com/en-us/azure/industry/agriculture/overview-azure-farmbeats
[23] H. Xu, J. Wu, Q. Pan, X. Guan, and oth, "A survey on digital twin for industrial internet of things: Applications, technologies and tools," *IEEE Communications Surveys & Tutorials*.
[24] A. Nasirahmadi and O. Hensel, "Toward the next generation of digitalization in agriculture based on digital twin paradigm," *Sensors*.
[25] A.-R. Al-Ali, R. Gupta, T. Zaman Batool, T. Landolsi, F. Aloul, and A. Al Nabulsi, "Digital twin conceptual model within the context of internet of things," *Future Internet*.
[26] P. Angin *et al.*, "Agrilora: a digital twin framework for smart agriculture." *J. Wirel. Mob. Networks Ubiquitous Comput. Dependable Appl.*
[27] R. G. Alves *et al.*, "A digital twin for smart farming," in *2019 IEEE Global Humanitarian Technology Conference (GHTC)*. IEEE.
[28] W. Purcell and T. Neubauer, "Digital twins in agriculture: A state-of-the-art review," *Smart Agricultural Technology*.
[29] S. Vakaruk *et al.*, "A digital twin network for security training in 5g industrial environments," in *2021 IEEE 1st International Conference on Digital Twins and Parallel Intelligence (DTPI)*. IEEE.
[30] Y. Wu, K. Zhang, and Y. Zhang, "Digital twin networks: A survey," *IEEE Internet of Things Journal*.
[31] S. A. Varghese, A. D. Ghadim, A. Balador *et al.*, "Digital twin-based intrusion detection for industrial control systems," in *2022 IEEE International Conference on Pervasive Computing and Communications Workshops and other Affiliated Events (PerCom Workshops)*. IEEE.
[32] Q. Xu, S. Ali, and T. Yue, "Digital twin-based anomaly detection in cyber-physical systems," in *2021 14th IEEE Conference on Software Testing, Verification and Validation (ICST)*. IEEE.
[33] P. Calvo-Bascones, A. Voisin, P. Do, and M. A. Sanz-Bobi, "A collaborative network of digital twins for anomaly detection applications of complex systems. snitch digital twin concept," *Computers in Industry*.
[34] H. F. Atlam and G. B. Wills, "Iot security, privacy, safety and ethics," *Digital twin technologies and smart cities*.
[35] O. Olawumi *et al.*, "Three practical attacks against zigbee security: Attack scenario definitions, practical experiments, countermeasures, and lessons learned," in *2014 14th International Conference on Hybrid Intelligent Systems*. IEEE.
[36] T. Melamed, "An active man-in-the-middle attack on bluetooth smart devices," *Safety and Security Studies*.
[37] S. Biffl *et al.*, *Security and quality in cyber-physical systems engineering*. Springer.
[38] M. De Donno *et al.*, "Ddos-capable iot malwares: Comparative analysis and mirai investigation," *Security and Communication Networks*.
[39] K. Wang, L. Yuan *et al.*, "Jamming and eavesdropping defense in green cyber–physical transportation systems using a stackelberg game," *IEEE Transactions on Industrial Informatics*.
[40] M. Hosseinzadeh, B. Sinopoli, and E. Garone, "Feasibility and detection of replay attack in networked constrained cyber-physical systems," in *2019 57th annual allerton conference on communication, control, and computing (Allerton)*. IEEE, 2019.
[41] C. Fang, Y. Qi, P. Cheng, and W. X. Zheng, "Cost-effective watermark based detector for replay attacks on cyber-physical systems," in *2017 11th Asian Control Conference (ASCC)*. IEEE, 2017.
[42] B. Ghimire and D. B. Rawat, "Recent advances on federated learning for cybersecurity and cybersecurity for federated learning for internet of things," *IEEE Internet of Things Journal*.
[43] M. Alazab *et al.*, "Federated learning for cybersecurity: Concepts, challenges, and future directions," *IEEE Transactions on Industrial Informatics*.
[44] A. Wainakh *et al.*, "Enhancing privacy via hierarchical federated learning," in *2020 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW)*. IEEE.
[45] *AWS for the Edge*. [Online]. Available: https://aws.amazon.com/edge/services/
[46] *AWS IoT Greengrass Build intelligent IoT devices faster*. [Online]. Available: https://aws.amazon.com/greengrass/
[47] *Managing Docker container lifecycle with AWS IoT Greengrass*. [Online]. Available: https://aws.amazon.com/blogs/iot/managing-docker-container-lifecycle-with-aws-iot-greengrass/
[48] *Install Kibana with Docker*. [Online]. Available: https://www.elastic.co/guide/en/kibana/current/docker.html
[49] Q. Liu *et al.*, "A survey on security threats and defensive techniques of machine learning: A data driven view," *IEEE access*, 2018.