Old MacDonald had a smart farm: Building a testbed to study cybersecurity in smart dairy farming

Sharad Agarwal
sharad.agarwal@bristol.ac.uk
University of Bristol
Bristol, UK

Awais Rashid
awais.rashid@bristol.ac.uk
University of Bristol
Bristol, UK

Joseph Gardiner
joe.gardiner@bristol.ac.uk
University of Bristol
Bristol, UK

ABSTRACT

With the advent of modern smart farming and agritech technology, farms are increasingly becoming an example of a cyber-physical system (CPS). For example, a modern dairy farm will feature internet-of-things (IoT) devices for monitoring animals and fully automated milking parlors. When considering the cyber security of CPS, we often talk about critical national infrastructure (CNI) with a focus on heavy industries such as energy generation, water treatment, and manufacturing, which all have a long history of digitization. Food supply is also considered part of CNI, so it is essential to consider it. A cyber attack on a farm can impact food supply, reduce revenue for farmers, and impact animal welfare. The security of smart farming has not been widely explored, and there is a lack of realistic testbeds that evaluate the security of agritech devices. This paper discusses the design of such a testbed, focusing on the dairy farming sector. We provide an overview of the testbed and discuss the challenges and lessons learned during the design and build process. We also present some early results from our analysis of the devices and software within the testbed and discuss future research directions.

CCS CONCEPTS

• **Computer systems organization** → *Sensors and actuators*; **Embedded systems**; Robotics; • **Networks** → Network reliability.

KEYWORDS

Cyber Security; Smart Farming; Testbed, IIoT; Internet of Things

ACM Reference Format:
Sharad Agarwal, Awais Rashid, and Joseph Gardiner. 2022. Old MacDonald had a smart farm: Building a testbed to study cybersecurity in smart dairy farming. In *Cyber Security Experimentation and Test Workshop (CSET 2022), August 8, 2022, Virtual, CA, USA.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3546096.3546097

1 INTRODUCTION

The Cybersecurity and Infrastructure Security Agency (CISA) classifies food and agriculture as critical infrastructure vital to the United States [8]. Increasingly, agriculture is using smart technologies and the Internet-of-Things (IoT) to increase efficiency and optimize production on farms, including the dairy farming sector. With the increase in smart, internet-connected technologies comes a vastly increased attack surface, which could allow malicious threat actors to cause widespread disruption to farming and food supplies. In 2019, information assurance firm NCC Group published a white paper describing some significant threat scenarios, including the loss of integrity of livestock sensors, unauthorized access to IP cameras, and loss of integrity and availability of automatic milking parlors and ventilation systems [7]. Later, in 2021 the Federal Bureau of Investigation warned that ransomware gangs are actively targeting and disrupting the operations of organizations in the food and agriculture sector [10].

Similar to any other sector, various categories of attackers can have a motivation to attack smart farms: *Organized cybercrime actors* have been known to conduct ransomware attacks against agritech [3, 4, 12]; *Nation-state actors* may aim to disrupt food supply throughout a target country or use it as a means to penetrate inside state networks [17]; *Activists* who might have a strong opinion about a particular farm or animal welfare activists [7] can aim to disrupt farming processes; and *Malicious competitors* may be motivated by cyber defamation or disrupting other competing farms for their financial gain.

As well as disrupting the food supply, cyber attacks on farms can cause significant monetary loss to farmers and also cause harm to animal welfare.

The white paper from NCC Group describes some significant threat scenarios, including the loss of integrity of livestock sensors, unauthorized access to IP cameras, and loss of integrity and availability of automatic milking parlors and ventilation systems [7]. Given our testbed's focus on smart dairy farming, we briefly define an attacker model impacting the farm and livestock:

- *Integrity attacks on actuators:* An attacker inside the network can perform various attacks to manipulate the physical processes on the farm, such as milking machines or cow sorting gates to cause them to function incorrectly, e.g., by mis-sorting cows to prevent them being milked or fed. Compromising the integrity of the processes causes financial loss to the farm and affects the cows' welfare. It might also cause significant issues wrt food safety standards and contaminate milk storage.
- *Tampering and eavesdropping attacks on sensors and data links:* The livestock sensors used in dairy farming can be attacked to disrupt their services or compromise their integrity. For instance, animal RFID tags may be cloned (as they do not use cryptographic protocols). Additionally, the attackers

CSET 2022, August 8, 2022, Virtual, CA, USA

ACM ISBN 978-1-4503-9684-4/22/08…$15.00
https://doi.org/10.1145/3546096.3546097

can eavesdrop on wireless channels and compromise the confidentiality of business-critical livestock data.

- *Availability attacks on physical processes:* An attacker can attack various physical processes and make them unavailable at any given time for a considerable duration (e.g., ransomware or denial of service attacks). If a milking machine were to be taken offline, it would be infeasible for a farmer to milk a large herd by hand, causing both a farm income loss and an animal welfare issue.

To date, the security of smart dairy farming is an under-researched area. This is in part due to the lack of suitable experimental infrastructures—as with any safety-critical cyber-physical system, it is difficult nigh impossible to test such things on a working farm due to the potential disruption to production, and in particular, the risk to the well-being of the animals, and humans, on the farm. We aim to address this by developing the first smart dairy farming testbed for security analyses of smart farming infrastructures as well as future evaluation of defensive security mechanisms in such a setting (including their impact on other properties such as safety and real-time needs).

However, building a smart dairy farming testbed is not a straightforward task. In order to build our testbed, we needed to understand the various physical processes that take place on a farm. The devices used in a smart dairy farming environment also differ from other industrial control systems (ICS) or IoT testbeds. The lack of technical details in the public domain further exacerbates the challenge of building such a testbed environment. There also exist various procurement challenges that we also discuss and document.

In summary, in this paper, we make the following contributions:

- We present the design of the first smart dairy farming testbed for cyber security research. Researchers can utilize the testbed to investigate security issues in smart farming and evaluate research prototypes. It also serves as a blueprint for others embarking on a similar endeavor – building on our experience rather than starting from scratch.
- We discuss lessons learned – good practices and pitfalls when others undertake the development of such a testbed.
- We present initial insights into the security state of smart dairy farms as revealed through our testbed.

2 RELATED WORK

Mekonnen et al. have built a small farm (horticulture) prototype at the FIU engineering campus using open-source hardware platforms, an Arduino-based micro-controller, and a ZigBee module to monitor and measure various environmental parameters, such as soil temperature and moisture, and real-time weather information [23]. Gokul and Tadepalli propose a new smart wearable device and sink node infrastructure that focuses on early detection of illness and deployment of various sensors in wearable [13].

In 2017, Saravanan and Saraniya also developed a cloud IoT-based livestock management system [26]. Similar UID systems have existed for a long time and are being used by farms globally. There exists much competition for smart wearables for cattle in the international market. Various farms are already using smart ventilation and smoke detection system across the globe.

Waraga et al. provide a literature survey of various IoT security testbeds [29]. Most of the testbeds consist of typical IoT devices like IP cameras and smart bulbs [19, 25, 28] which are not interconnected and are primarily independent IoT devices.

Schoofs et al. presented the implementation of an IP-based herd monitoring testbed to detect limping cows [27]. Their architecture seamlessly integrates IPv6-based sensor nodes, wireless mesh routers, and PCs. However, none of these testbeds focuses on precision agriculture or precision livestock farming security.

Gupta et al. provide an in-depth study on security and privacy in smart farming [18]. They provide insights into various possible attack scenarios through a roadmap of security and research challenges in the smart farming ecosystem.

Baker and Green provide insights into cybersecurity in UK agriculture [7]. They address the various possible threats to agritech, including the possibility of a high level of financial loss and suffering to livestock due to cyber attacks.

Iwasaki, Morita, and Nagata explain the requirements of different IoT sensors for smart livestock monitoring and discuss their advantages and disadvantages [22]. However, they do not discuss the security of these sensors.

Pan, Xu, Xi, and Hao compare the software web services for IoT livestock management services and introduce the prototype of a smart livestock farming IoT system based on Restful Web Services [24]. However, they mainly focus on efficiency and feasibility and do not evaluate the security of their prototype except by providing access control.

3 DESIGNING THE SMART DAIRY FARM

There is no proper reference architecture or network diagram available publicly that can be used to build a smart dairy farming testbed. The technical guides and detailed manuals are not readily available for devices used in such settings. Furthermore, most security researchers do not have deep insight into dairy farming. In order to understand how a smart dairy farm works, we used a multi-pronged approach. We visited a research farm using state-of-the-art technology for precision livestock farming and observed their operations and the equipment set up.

We developed a network of contacts in the industry through meetings with several manufacturers, sending hundreds of emails and several dozen calls to manufacturers and suppliers. We interviewed five farmers (ethics approval was provided by our Institutional Review Board) who own or work at farms with smart devices and learned from them about the working of a smart dairy farm. To understand the ecosystem, we also visited various events like dairy shows, where manufacturers put up stalls to advertise their products.

Through our close engagement with suppliers, manufacturers, and technicians, we were able to acquire the manuals and guides to acquaint ourselves with the relevant products and devices used in a smart dairy farm. Working with these suppliers, manufacturers, and technicians, we developed and refined the design of our testbed, subsequently leading to the final design of a table-top smart dairy farm, as shown in Fig 1. We discuss the various elements of our testbed next.

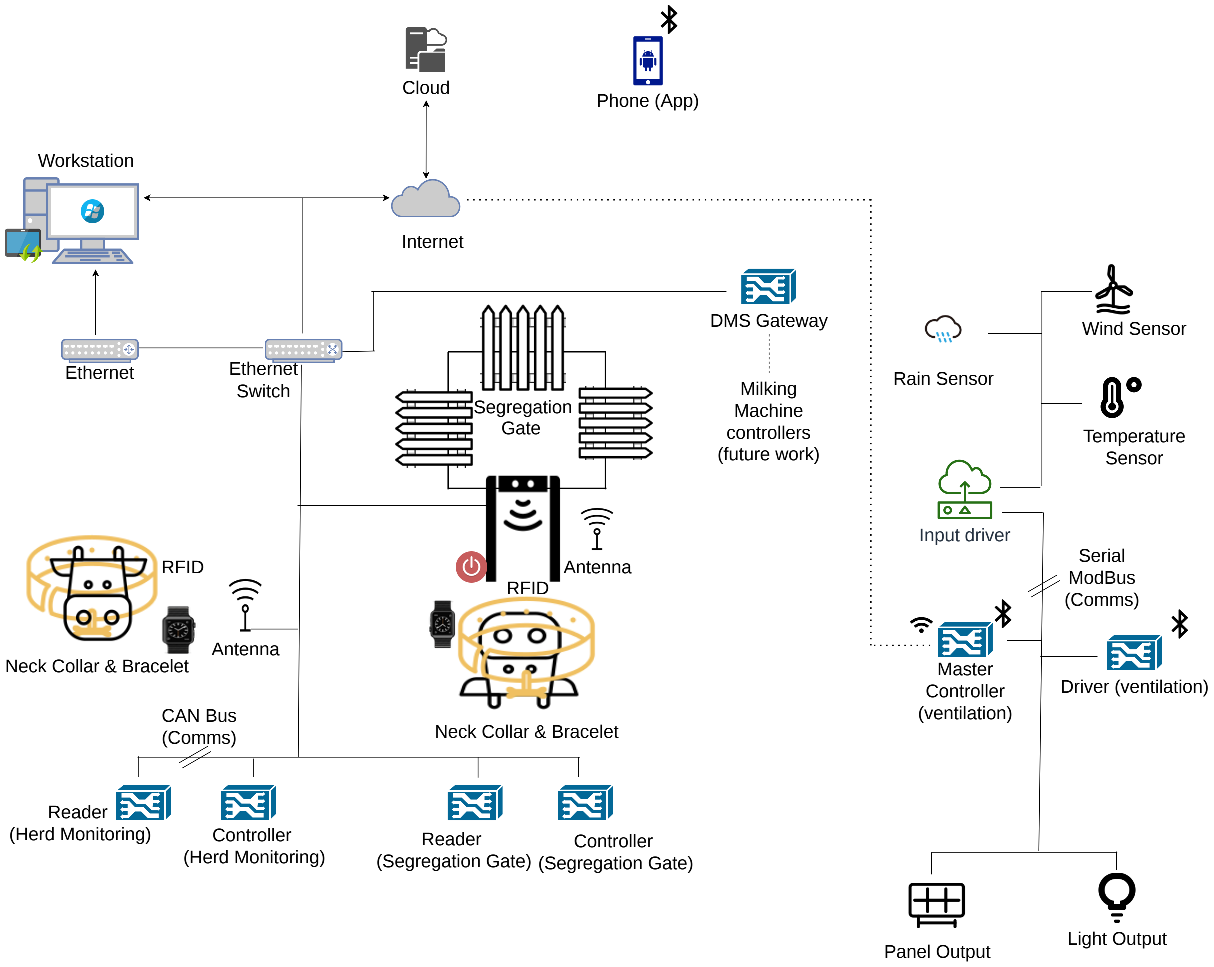


Figure 1: Diagram of the smart dairy farming testbed.

3.1 Physical processes in a dairy farm

There are a few critical processes that take place on a dairy farm. The main processes realised in our testbed are as follows:

3.1.1 Automatic segregation gate. In order to allow the free movement of cows on a dairy farm for their well-being and the farm's production efficiency, a segregation gate is used. When the cow reaches this gate, the readers installed at the gate read the cow ID through the cow's neck/leg/ear responder and, according to the statistics of the particular cow in the software, direct the cow to a particular direction – grazing field, resting area or the milking machine. Fig 2 shows the cow entering near the segregation gate: the backside is closed, and the cow is directed towards Pen A. We configured a two-way segregation gate in the testbed as it helps achieve enough understanding of the gate's working. One can also configure it to provide three different directions instead.

3.1.2 Herd monitoring: animal statistic collection. The cow neck and leg responders provide more functionalities than just an ID. They continuously monitor the cow's eating habits, lying time, stand-up counts, step counts, and temperature. These statistics are sent to the reader at regular 15 mins intervals and forwarded to the software to provide graphs.

3.1.3 Maintaining environmental conditions. The weather ventilation system monitors the temperature, moisture, wind speed and direction, and precipitation count. As per these conditions, the actuators provide the required conditions in the barn for the animal's well-being and better productivity.

3.2 Design Characteristics

Previous researchers have contributed many design considerations and lessons learned from developing various ICS/IoT testbeds [5, 6, 9, 11, 14–16, 29]. Taking all these into consideration, our design was driven by the need to capture the following characteristics within our testbed:

- *Diversity of devices:* The testbed consists of various devices used in a smart dairy farm (see Section 4.1).
- *Fidelity:* The testbed replicates a realistic dairy farm even though it does not includes any livestock.

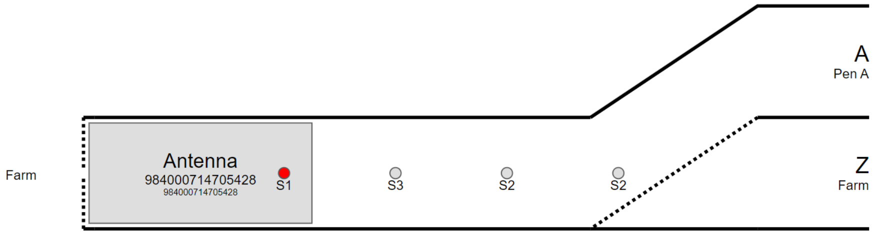


Figure 2: Working of a segregation gate; points the herd towards the direction as per the herd's statistics.

Limitations. As we do not have livestock, there is no continuous actual data to feed on devices like the cow collar. However, this does not affect the primary goal of the testbed, i.e., performing security analysis. Similarly, as the weather ventilation sensors are installed inside the lab, they are not exposed to real-world weather and do not run the actuators as a weather system would in the field. To overcome this, we interacted with the sensors regularly to generate the values manually (more details in section 4.3).

3.3 Procurement

One of the most difficult challenges we encountered was procuring devices, especially for research purposes. There are no off-the-shelf buying options in this sector where one can order equipment online or purchase it at a store. The manufacturers do not sell devices to customers directly.

Furthermore, the suppliers are not interested in selling one or two pieces of such devices to anyone except farms/farmers. One plausible reason is that most suppliers order directly from the manufacturer in bulk as per the requirement of the farm as devices are interconnected. For example, a cow neck collar/responder comes in a box of ten, and suppliers do not generally sell individual pieces, where each piece ranges from $130 - $200. Similarly, they sell wearable responders, readers, and controllers as a complete package, whereas the controllers and readers generally come as bundles designed for farms. Most suppliers sell wearable and automated robotic milking machines only, which can cost anywhere between $130k - $200k.

We reached out to farms and organizations, asking them for contacts for suppliers and manufacturers so we could discuss our requirements to buy devices. We engaged with various manufacturers, built links with suppliers, explained the benefits of the research, including conducting responsible disclosure, and built trust through this engagement. Subsequently, one manufacturer and one supplier – understanding the research value – agreed to willingly give us various pieces of equipment to *try before we buy*. This, in turn, enabled us only to purchase suitable products after some prototyping of the testbed.

3.4 Use Cases

Discussion with the stakeholders in agritech, especially precision livestock, has helped us identify the use cases needing further study through the testbed. The most critical use cases include:

Security analysis of devices. One of the main aims of the testbed is to perform security assessments and understand the current state of cybersecurity in this sector. The testbed enables the study of attacks on networking protocols, devices, and software and understanding of the implications of such attacks on the safety, integrity, and reliability of the physical processes.

Demonstration of attacks and their implications. An attack against the dairy farm can not be demonstrated readily on a live farm as it may disrupt the farm services and affect the livestock's well-being. The testbed acts as a vehicle to demonstrate cyber attacks to stakeholders, such as vendors, farmers, and policymakers, highlighting their implications in the farming ecosystem.

Testing of defense mechanisms. With the increasing number of cyberattacks being conducted to disrupt the agritech sector, the testbed allows researchers to design and evaluate defensive techniques to study their effectiveness. The testbed can also be used to *test* field-ready security mechanisms before their deployment on real-world farms.

Collaborative use. The testbed provides the basis for collaborative research between researchers and manufacturers/vendors. It also serves as a platform for security researchers to collaboratively develop and evaluate security mechanisms for smart farming environments. Furthermore, being the first testbed in this domain, other researchers can use this as a reference to design their own drawing upon our experience and insights.

4 TESTBED OVERVIEW

The physical testbed currently has two main systems: herd management and environment ventilation system. The testbed is also equipped with a gateway to support future expansion with milking parlor equipment. An overview of the testbed systems can be seen in Fig. 3. Fig. 4 shows the testbed and all installed components.

4.1 Devices

The devices installed in the herd management and environment ventilation systems are as follows (cf. Fig. 4); note that these replicate setups we gathered through our observations on farms and discussions with farmers, manufacturers, suppliers, and technicians.

4.1.1 *Herd Management.*

- *Cow neck and leg responder*: The responders work as an identification tag. They also collect cow movement, eating, and temperature data and send it to the software at regular intervals. The responders operate on the $134.2kHz$ full-duplex (FDx) low frequency [1] and on $430MHz$ high frequency to send the animal data to the reader [20, 21].

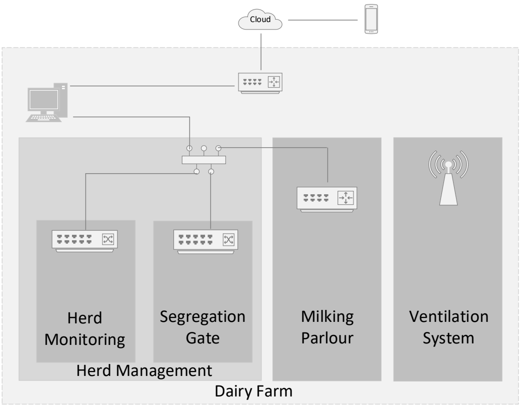


Figure 3: Overview of the testbed

- *Cow ear tag*: The ear tag does not collect any data but works as a small identification tag that the segregation gates and milking machines can use. It operates on the 134.2kHz half-duplex (HDx) low frequency [2, 20, 21].
- *Antennas*: The testbed has two antennas: 430MHz and 134.2kHz (supports both HDx and FDx) – the high-frequency one for reading the data at regular intervals (approx. 15 mins) from the neck and leg responders and the low-frequency one for the segregation gate identification using the neck, leg, and ear responder.
- *Readers*: The testbed consists of two readers. One is used for reading the animal health data, and the other for animal identification at the segregation gate. Both readers run the embOS with the FTP to communicate data and an HTTP server (this is the default set up on farms) to display info on the workstation. One of the readers that collect animal data communicates with the controller using CAN bus, whereas the other communicates with the controller using Ethernet.
- *Controllers*: The controllers are dedicated hardware devices manufactured and designed by specific companies. We expected to find controllers from manufacturers such as Siemens and Allen Bradley, but this was not the case – niche manufacturers developed for this market. In a herd management system, the controllers communicate over the Ethernet, except for one which communicates with the reader through the CAN bus.
- *Workstation*: 16 GB RAM, 1 TB HDD, Intel Core i7, 3.41GHz, Windows 10 Pro x64 with two network interfaces and a Bluetooth adapter.

4.1.2 Environment ventilation System.

- *Temperature, wind and rain sensors*: The ventilation system takes the following input: wind direction and speed, if it is raining or not, along with the measurement in mm, and the temperature with the wind, rain, and temperature sensors respectively.
- *Controllers*: Similar to herd management, the controllers used in the environment ventilation system are also dedicated hardware devices manufactured and designed by dedicated manufacturers. Controllers in the ventilation system communicate with each other via Modbus.
- *Ruuvitag temperature sensor*: Along with the temperature sensor, we also have a wireless temperature and humidity sensor from Ruuvitag[1].
- *Phone*: One Samsung and one Apple iPhone as the ventilation system has both Android and iOS apps.

We have been able to incorporate all necessary equipment listed above, except for specific equipment relating to the automated robotic milking parlor, due to procurement challenges (see Section 3.3). These will be attached to the testbed via the DMS gateway (already installed; see Fig. 1) when we acquire them.

4.2 Network

Technology in the area of agritech has recently evolved with a lot of new smart devices. However, installing and managing these smart devices is done either by the supplier or the farmer. No proper network topology is often designed while setting up a smart dairy farm. Farmers we interviewed were, understandably, not experts on computer networks and hence had little to no knowledge of network topology. They entirely depended on their suppliers to know everything about it.

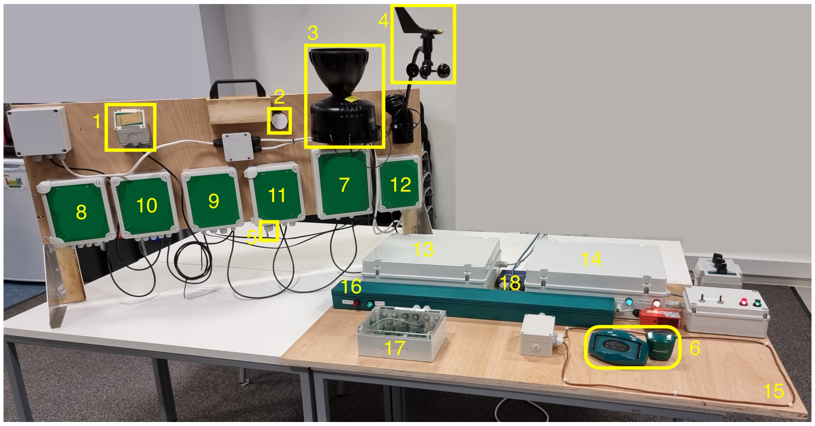
Dairy farms have a flatbed network where devices are installed directly into the network without any hierarchy or distributed system practices. There are no investments made to install firewalls or DMZs. However, this is not straightforward as expected for a flatbed network. There are various devices (see section 4.1) and multiple communication channels being used for communication (see section 4.5), increasing the complexity of a flatbed network. After understanding these various communication channels and their role in the management of the physical processes, we developed the functional network diagrams in Figs. 5 (Herd Management System) and 5 (Environment Ventilation System). The Herd Management system utilizes a local network without internet connectivity between the devices and the farm workstation. The herd monitoring controller also utilizes a connection to the internet (by directly connecting to an access point with a wired connection) to allow for remote monitoring. The workstation also has internet connectivity through the same access point.

The ventilation system master controller connects to the internet through the access point using WiFi. This allows control and monitoring of the ventilation through the smartphone application. The other controllers connect to the master using serial Modbus connectivity. All devices also feature Bluetooth for local access through the smartphone app when internet connectivity is unavailable.

4.3 Sensor Value Generation

Our lab is a secure closed area; therefore, the testbed does not have the exact environmental conditions. In order to provide input to the sensors, we use alternatives inside the lab. We use a table fan at various speeds to provide input to the wind sensor. We enable the rain sensor by touching it with our hands. We also do not have a cow, so we move the cow responders by hand to provide them with some input. One of the suppliers has also provided us with the data stored in the dairy management software.

[1] https://ruuvi.com/ruuvitag/



Key:

1	Rain detection sensor
2	Ruuvitag wireless sensor
3	Rain gauge sensor (mm/hr)
4	Wind speed and direction sensor
5	Temperature sensor
6	Neck and leg responders
7	Ventilation master controller
8,10,11	Ventilation system controllers
9	Ventilation system driver (sensor input)
12	Power supply
13	Herd monitoring controller and reader
14	Segregation gate controller and reader
15,16	Antennas
17	DMS gateway
18	Switch

Figure 4: Smart dairy farm testbed in our lab.

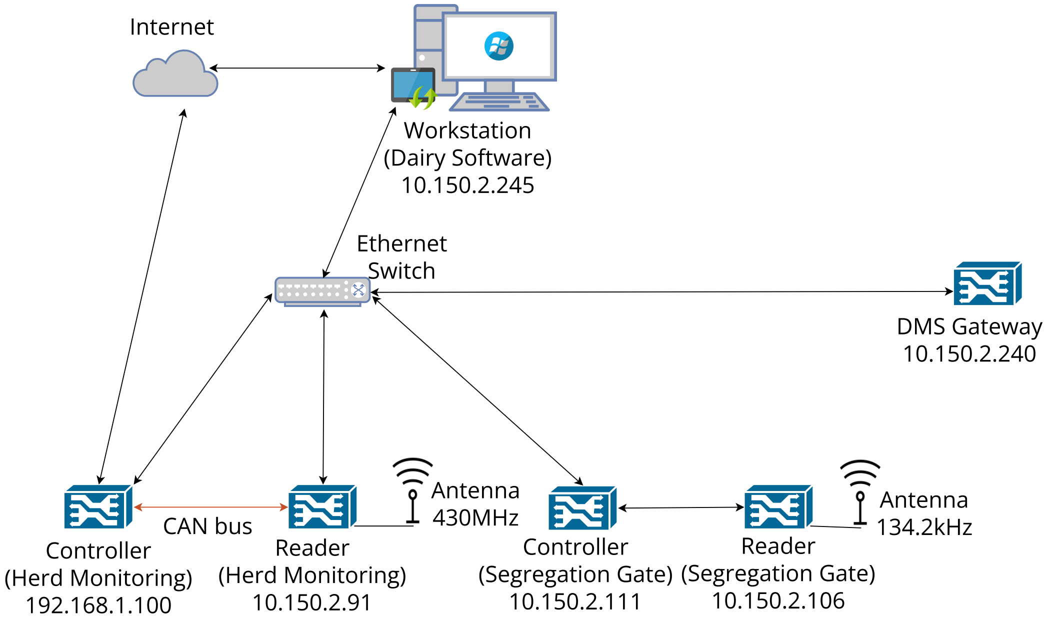


Figure 5: Network diagram of the herd management system.

4.4 Network Data Capture

The ventilation system's master controller sends information directly over the WiFi to the server and is used to plot graphs. In order to capture the network traffic, we use a raspberry pi network tap and connect it to the Ethernet port of the master controller. We build a bridge between two Ethernet connections that act as a network tap by running tcpdump[2] after connecting the tap to the device and the router, which is helpful to generate log files.

We are also running Wireshark[3]/ tcpdump on the dairy PC to capture the communication between the PC and the different readers and controllers installed in the herd management zones connected over the Ethernet using LAN.

4.5 Communication Channels

A smart dairy farm has multiple devices, as explained in section 4.1 and these communicate using a variety of communication channels.

- *WiFi*: The ventilation's master controller has WiFi to connect to the internet and send data over WiFi to the server.
- *RFID*: The cow's neck collar, leg bracelet, and ear tag use radio-frequency identification (RFID) to send signals and data to the reader via antennas.
- *Bluetooth*: The ventilation system's master controller has Bluetooth enabled to connect it to the mobile application.

[2] https://www.tcpdump.org
[3] https://www.wireshark.org

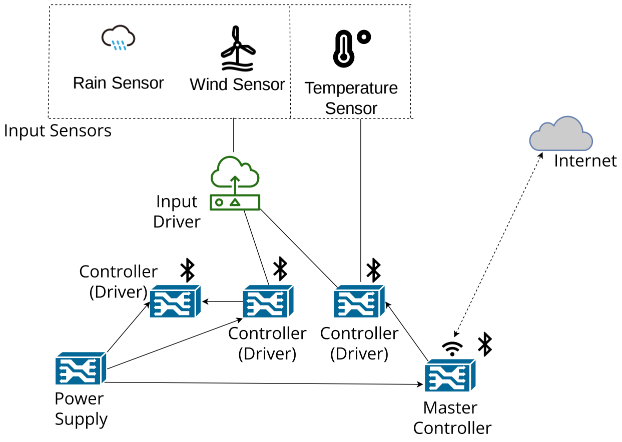


Figure 6: Network diagram of the environment ventilation system.

- *Ethernet*: The herd management system's controllers communicate with each other over Ethernet. It is also used to connect the central controller to the workstation, which runs the dairy software.
- *Modbus*: Serial Modbus is used by the ventilation system controllers to communicate with each other.
- *CAN bus*: The reader that collects data from the cow's neck/leg responder uses a Controller Area Network (CAN) bus to communicate information to the controller.

4.6 Software

We also deploy a dairy management software – the same software which is used by customers of the manufacturers from whom we procured the devices. It collects and manages the data sent from the animal responders through the reader. It also manages the IDs that try to pass through the segregation gate. If an unidentified animal tag is introduced, the software throws an alarm.

The herd management system (including the segregation gate) and the ventilation system also have web applications. The weather ventilation and the herd management web applications can be accessed remotely from the internet. The segregation gate web application can only be accessed locally through the farm's PC. The ventilation system also has a cross-platform mobile application.

5 LESSONS LEARNT

Other researchers have documented lessons from building testbeds. For instance, Green et al. provided ten lessons learned when building an industrial control system testbed [15]. Gardiner et al. revisited them to provide updates and added new lessons [11]. During the development of our testbed, we built on these lessons and found that several of them are equally valid and applicable in smart farming testbeds, e.g., comprehensively documenting as one builds, having a swappable capability, and considering maintenance from the design stage. However, our experiences also diverge on some of the lessons reported in prior research on testbeds:

Virtualization of processes is not possible. As most technical details are not publicly available, it is very challenging to virtualize the physical processes. It is easier to deploy a physical device and integrate it into the system compared to the use of virtualization.

Buying off-the-shelf is not supported in most cases. As discussed earlier, buying off-the-shelf is not an option. As the technical details are not publicly available, the DIY approach also does not work in this case. The only solution for building a dairy farm testbed is to build relationships with manufacturers, suppliers, and technicians to elicit details of technical setups and procure devices and software systems.

Setting up a flatbed network is not straightforward. Even though the dairy farm has a flatbed network, the internal communication protocols and channels are varied and underpin critical physical processes. This means carefully designing the flatbed network to reflect the topology of smart dairy farms in practice.

Inter connectivity of processes and testbed dependence on them is crucial. Contradictory to the ICS testbeds, process diversity is crucial in smart dairy farming. Without various processes, the testbed cannot replicate the real-world dairy farm and how they interact to deliver the production capacity required.

6 INSIGHTS FROM INITIAL SECURITY ANALYSIS BASED ON THE TESTBED

Using the testbed, we implemented and confirmed various possible attack classes (based on threats identified by researchers [7, 18]). Our initial analysis indicates that the current state of cybersecurity in smart dairy farming significantly lacks maturity, and significant work is needed to improve the security of devices, networking mechanisms, and deployment architectures. Table 1 provides an overview of the initial tests we conducted, the tools used, the outcomes in brief, and the potential impact of the attacks on the farm business and animal well-being.

- None of the web interfaces for the weather ventilation portals has an HTTPS/TLS certificate setup. This means that an attacker on the same network can easily listen to traffic connection requests and steal the data. Using Wireshark, we were able to see the username and password being sent in cleartext.
- We found that the SSH access to the master controller of the ventilation system has *no password* root access. We confirmed that an attacker could quickly get shell access to the device without entering a password. This is a significant issue as an attacker can remotely exploit this vulnerability (a controller being connected to the internet). While going through the shell, we also found the URLs to the APIs, one of which had access to a file containing a list of all farm IDs and the associated IP addresses. We also found a file with the username, password, and URL to the company database, which can be easily accessed by an attacker having remote access via SSH.
- We were able to get the ventilation system data of other farms through our customer account. This means any customer can access the data of other farms where the same company's devices are installed. This might allow competitors to use this data for their benefit.

Test Conducted	Description	Result	Effect
Scanning and Fingerprinting	Used Nmap to find open ports and services running on the devices	SSH port for one of the controllers had no password, FTP port for the readers has default password	Disrupt the weather ventilation system remotely, get access to the company's MongoDB database. Exploiting the FTP disrupts the readers that collects livestock information and reads ID for the segregation gate.
Web Application Testing	OWASP Top 10	No default credentials. Get other customer's farm data by URL manipulation due to weak API authorization	Data leakage of ventilation system but no known harm to livestock or farm.
Reading RFID Tags	Used Proxmark 3 RDV 4 to read the cow neck and collar responders	Get the animal ID of the tags	Clone the ID tags - introduce new cows with same IDs/ change the cows with the existing cow's ID/ change collars for existing cows with similar IDs; exploits the system and affects livestock health and farm production.
Reversing Android app	Used Apktool and dex2jar to get files from the android app's APK	Found API URLs	Beginning point for API pentesting. API exposes the list of user sites.
Checking known vulnerabilities	Parsed Exploit-DB and CVE lists to check existing security issues	Denial of service attack on embOS/IP FTP server (CVE-2018-7449)	Denial of Service on the readers that runs embOS/IP FTP server stops readers to read animal data and disrupts the farm process. The segregation gate needs the reader to read animal ID and send animal to milking machine or for feeding or in shelter as per animal statistics.
Check SSL certificate weakness	Used Wireshark to intercept network packets as no TLS/HTTPS present	Easily able to get username and password in clear text	Manipulate user's farm weather data and devices along with their actuators and calibration affecting animal well being.

Table 1: Summary of the initial security analysis conducted.

- While commissioning the devices, we were told that a customer does not have access to the admin control panel of the ventilation system using the user credentials. We could easily log in and get admin controls to our farm devices on the web app interface with our user credentials, indicating misconfiguration of access control to the admin control panel.

Implications: The above vulnerabilities allow the attacker to access the ventilation system and change the controls and threshold values affecting the environmental conditions in the barn. Extreme temperature conditions in the barn can, in turn, affect animal welfare and decrease farm production.

- The segregation gate needs an animal ID to send it in a particular direction. We could easily clone the cow collar and make a new low-frequency card with that animal ID. The cloned card could open the segregation gate and direct animals.
- The herd management system and the segregation gate are connected to a computer which is a *single point of failure.* If we switch off the PC or force it into hibernation mode, the Ethernet connection to the switch stops working, and the reader for the segregation gate and data collection gets disabled. This introduces new scenarios for denial of service on the farm.

Implications: The above allows the attacker to steal livestock, introduce a new less productive cow with the id of some other cow already on the farm or use the same id for more than one cow, which is going to impact the cow's health adversely, well-being and ultimately affect the farm's profits.

Responsible vulnerability disclosure and vendor's response. We undertook responsible vulnerability disclosure and provided reports to the manufacturer and vendor, providing a step-by-step guide to replicate the attacks and guidance to mitigate the issues. We received almost instant replies compared to the traditional bug bounty and vulnerability disclosures programs. The vulnerabilities were fixed quickly, and all devices deployed in the fields were patched.

7 DISCUSSION

Data generation challenges. The devices installed in the testbed were configured by the suppliers as they would on a real farm. Our testbed exists as a farm in their systems and can access such systems as a real farmer. The primary limiting factor of the testbed in its current form is the lack of animals, which reduces the amount of data generated by items such as the tracking collars. The vendor has supplied some actual tracking data from another farm; however, we are unable to generate new data.

Rapid responses to vulnerability disclosure. As shown in Section 6, several simple yet high severity vulnerabilities were discovered in the devices that were analyzed. The exciting aspect is that the vulnerabilities were patched in a concise time frame after disclosure. For example, the no password root access in the ventilation system was remotely patched by the vendor within three days of reporting across the entire install base. Compared to other CPS systems, including ICS, where patching is primarily up to operators and is rarely applied, the deployment of agritech is almost always under service contracts from the vendors or suppliers. So patching can be applied by the vendor in a brief time period.

Increased attack surface through vendors and suppliers. The devices in a dairy farm are interconnected to several systems, e.g., they collect the data and send it to the cloud. The vendors also have direct access to the controllers installed at the farms and collected data. Vendors and suppliers commonly have remote desktop access to the farm workstation. This increases the attack surface, allowing the attacker a pathway through vendor/supplier breaches.

Farm workstations as single points of failure. One thing that became apparent is that the farm workstation is critical to farm operations. For example, while using simple PLCs as control devices, the herd management system does not store any control information on the devices. Instead, all control information (which dictates where animals should be directed) is located on the workstation, with the controller communicating with the workstation every time an animal is seen. If the workstation goes offline for any reason, the automated gates do not function, making the workstation a single point of failure.

8 CONCLUSION AND FUTURE WORK

We designed and deployed the first smart dairy farm testbed for security research. This testbed is an essential tool to test the safety and security challenges and their implications without experimenting on live farms that might affect animal welfare. The paper explains the complete design process and describes the overview of the testbed, including the devices installed and the communication protocols they use. We explain the security vulnerabilities and misconfigurations found during the testbed deployment and the outcomes of the initial vulnerability penetration tests.

In order to understand the complete smart dairy farm ecosystem, our future work will focus on procuring automatic robotic milking parts which can be installed in the current testbed. We will also continue working on an in-depth security assessment and designing and evaluating security mechanisms suited to smart farming settings. Further aims include expanding the work to horticulture and procuring devices and sensors for precision agriculture.

ACKNOWLEDGMENTS

This research was supported in part by the PETRAS National Centre of Excellence for IoT Systems Security funded via the UK Research and Innovation grant EP/S035362/1.

REFERENCES

[1] n.d.. FDX-B Animal Identification Protocol description. *Electronics Design and Manufacture* (n.d.). https://www.priority1design.com.au/fdx-b_animal_identification_protocol.html.

[2] n.d.. HDX Animal Identification Protocol description. *Electronics Design and Manufacture* (n.d.). https://priority1design.com.au/hdx_animal_identification_protocol.html.

[3] L. Abrams. 2021. Pan-Asian retail giant Dairy Farm suffers REvil ransomware attack. *Bleeping Computer* (2021). https://www.bleepingcomputer.com/news/security/pan-asian-retail-giant-dairy-farm-suffers-revil-ransomware-attack/.

[4] L. Abrams. 2021. Second farming cooperative shut down by ransomware this week. *Bleeping Computer* (2021). https://www.bleepingcomputer.com/news/security/second-farming-cooperative-shut-down-by-ransomware-this-week/.

[5] U. Ani and J. Watson. 2021. What Makes an Industrial Control System Security Testbed Credible and Acceptable? Towards a Design Consideration Framework. In *Proceedings of the 11th International Conference on Simulation and Modeling Methodologies, Technologies and Applications - SIMULTECH,*. 181–190.

[6] U. Ani, J. Watson, B. Green, B. Craggs, and J. Nurse. 2019. Design Considerations for Building Credible Security Testbeds: A Systematic Study of Industrial Control System Use Cases. (2019). arXiv:1911.01471

[7] L. Baker and R. Green. 2019. Cybersecurity in UK Agriculture. *NCC Group Whitepaper* (2019).

[8] CISA. 2020. Critical Infrastructure Sectors. (2020). https://www.cisa.gov/critical-infrastructure-sectors.

[9] B. Craggs, A. Rashid, C. Hankin, R. Antrobus, O. Şerban, and N. Thapen. 2019. A reference architecture for IIoT and industrial control systems testbeds. In *Living in the Internet of Things (IoT 2019)*. 1–8.

[10] FBI. 2021. Cyber Criminal Actors Targeting the Food and Agriculture Sector with Ransomware Attacks. (2021). https://www.ic3.gov/Media/News/2021/210907.pdf.

[11] J. Gardiner, B. Craggs, B. Green, and A. Rashid. 2019. Oops I Did It Again: Further Adventures in the Land of ICS Security Testbeds. In *Proceedings of the ACM Workshop on Cyber-Physical Systems Security & Privacy (CPS-SPC'19)*. 75–86.

[12] S. Gatlan. 2021. Food giant JBS Foods shuts down production after cyberattack. *Bleeping Computer* (2021). https://www.bleepingcomputer.com/news/security/food-giant-jbs-foods-shuts-down-production-after-cyberattack/.

[13] V. Gokul and S. Tadepalli. 2017. Implementation of smart infrastructure and non-invasive wearable for real time tracking and early identification of diseases in cattle farming using IoT. In *2017 International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)*. 469–476.

[14] B. Green, S. Frey, A. Rashid, and D. Hutchison. 2016. Testbed diversity as a fundamental principle for effective ICS security research.

[15] B. Green, A. Lee, R. Antrobus, U. Roedig, D. Hutchison, and A. Rashid. 2017. Pains, Gains and PLCs: Ten Lessons from Building an Industrial Control Systems Testbed for Security Research. In *10th USENIX Workshop on Cyber Security Experimentation and Test (CSET 17)*.

[16] B. Green, B. Paske, D. Hutchison, and D. Prince. 2014. Design and construction of an industrial control system testbed. In *PG Net-The 15th Annual PostGraduate Symposium on the Convergence of Telecommunications, Networking and Broadcasting.*

[17] A. Greenberg. 2022. Chinese Spies Hacked a Livestock App to Breach US State Networks. (2022). https://www.wired.com/story/china-apt41-hacking-usaherds-log4j/.

[18] M. Gupta, M. Abdelsalam, S. Khorsandroo, and S. Mittal. 2020. Security and Privacy in Smart Farming: Challenges and Opportunities. *IEEE Access* 8 (2020), 34564–34584.

[19] M. Hale, K. Lotfy, R. Gamble, C. Walter, and J. Lin. 2019. Developing a platform to evaluate and assess the security of wearable devices. *Digital Communications and Networks* 5, 3 (2019), 147–159.

[20] ISO 11784:1996. *Radio frequency identification of animals — Code structure.*

[21] ISO 11785:1996. *Radio frequency identification of animals — Technical concept.*

[22] W. Iwasaki, N. Morita, and M. Nagata. 2019. Iot sensors for smart livestock management. In *Chemical, Gas, and Biosensors for Internet of Things and Related Applications*. 207–221.

[23] Y. Mekonnen, S. Namuduri, L. Burton, A. Sarwat, and S. Bhansali. 2019. Review—Machine Learning Techniques in Wireless Sensor Network Based Precision Agriculture. *Journal of The Electrochemical Society* 167, 3 (dec 2019), 037522.

[24] L. Pan, M. Xu, L. Xi, and Y. Hao. 2016. Research of livestock farming IoT system based on RESTful web services. In *2016 5th International Conference on Computer Science and Network Technology (ICCSNT)*. 113–116.

[25] V. Sachidananda, S. Siboni, A. Shabtai, J. Toh, S. Bhairav, and Y. Elovici. 2017. Let the cat out of the bag: A holistic approach towards security analysis of the internet of things. In *Proceedings of the 3rd ACM International Workshop on IoT Privacy, Trust, and Security*. 3–10.

[26] K. Saravanan and S. Saraniya. 2018. Cloud IOT based novel livestock monitoring and identification system using UID. *Sensor Review* (2018).

[27] A. Schoofs, C. Daymand, R. Sugar, U. Mueller, A. Lachenmann, M. Kamran, A. Gefflaut, L. Thiem, and M. Schuster. 2009. Poster abstract: IP-based testbed for herd monitoring. In *2009 International Conference on Information Processing in Sensor Networks*. 365–366.

[28] A. Tekeoglu and A. Tosun. 2016. A testbed for security and privacy analysis of IoT devices. In *2016 IEEE 13th International Conference on Mobile Ad Hoc and Sensor Systems (MASS)*. IEEE, 343–348.

[29] O. Waraga, M. Bettayeb, Q. Nasir, and M. Talib. 2020. Design and implementation of automated IoT security testbed. *Computers & Security* 88 (2020), 101648.