

OBJECTIVE

Create a physical and virtual model of an Industrial Control System to demonstrate the effects of cyberattacks in a controlled environment.

VALUE PROPOSITION

- Facilitate cybersecurity development and education of the critical systems that provide resources such as water and gas
- Create an accessible and affordable small-scale model of an industrial control system to perform cybersecurity-related research

BACKGROUND

- Industrial Control System (ICS): Complex system employed to control a specific process using specialized components
- Human Machine Interface (HMI): User interface for an ICS
- Programmable Logic Controller (PLC): Devices commonly leveraged by ICS to control the state of the system

REQUIREMENTS

- Physical model of an ICS including an HMI that can simulate a real-world process
- Ability to run cyberattacks to disrupt the regular operation of the ICS via an attacker machine
- Ability to observe all network traffic and logs from an observer machine that cannot be affected by the cyberattacks
- Recreation of the model in a virtual environment

CONCEPTUAL DEVELOPMENT

Implement physical process as a Ferris wheel—a process with clear visual indications of normal versus attacker manipulated behaviour.

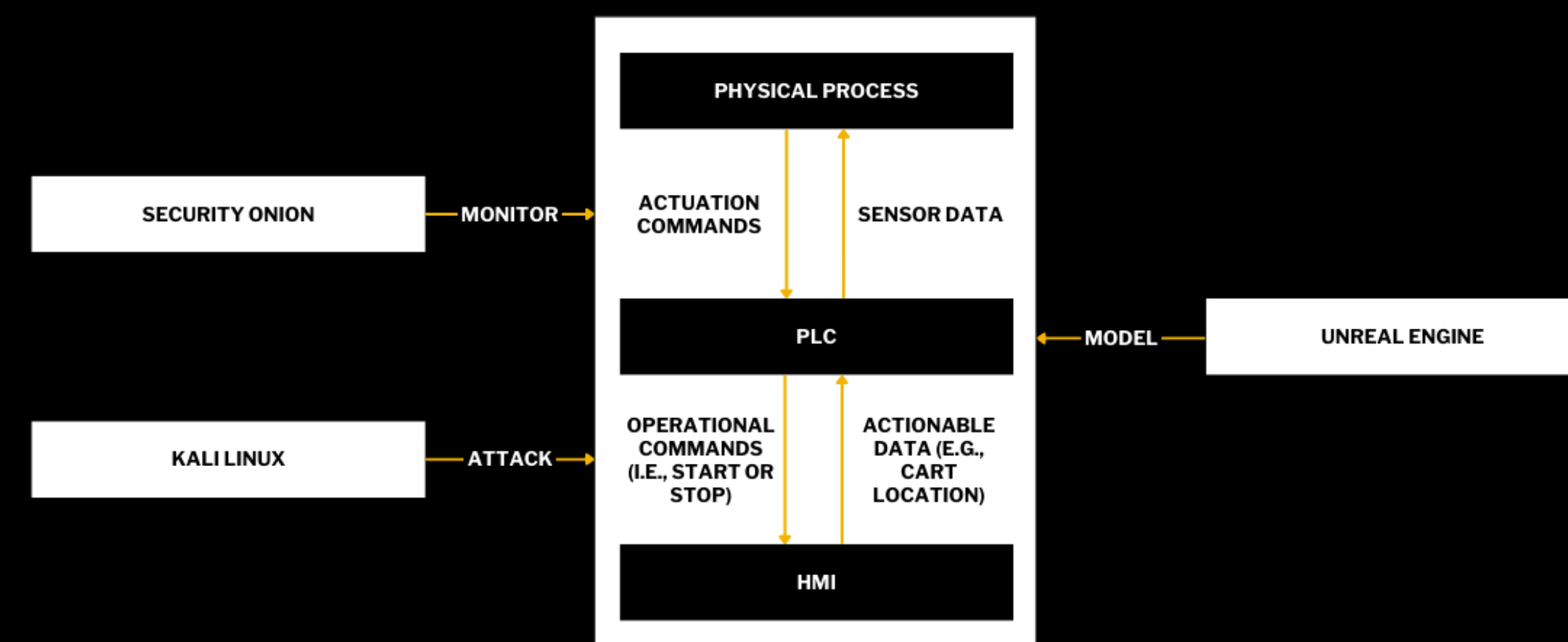


Figure 1. Dataflow Diagram of Testbed

FINAL DESIGN

- Libre PC Le Potato mini-pc running OpenPLC Runtime gathers sensor data and controls the Ferris wheel motor
- Virtual machine (VM) running SCADA-LTS acts as the HMI, displaying valuable statuses about the physical process
- Network of VMs hosting various aspects of Security Onion monitor and log the activity of the testbed
- VM running Kali Linux employs Python scripts—incorporating libraries such as scapy and Ettercap—to attack the testbed with Denial-of-Service (DoS) and Man-in-the-Middle (MitM) attacks

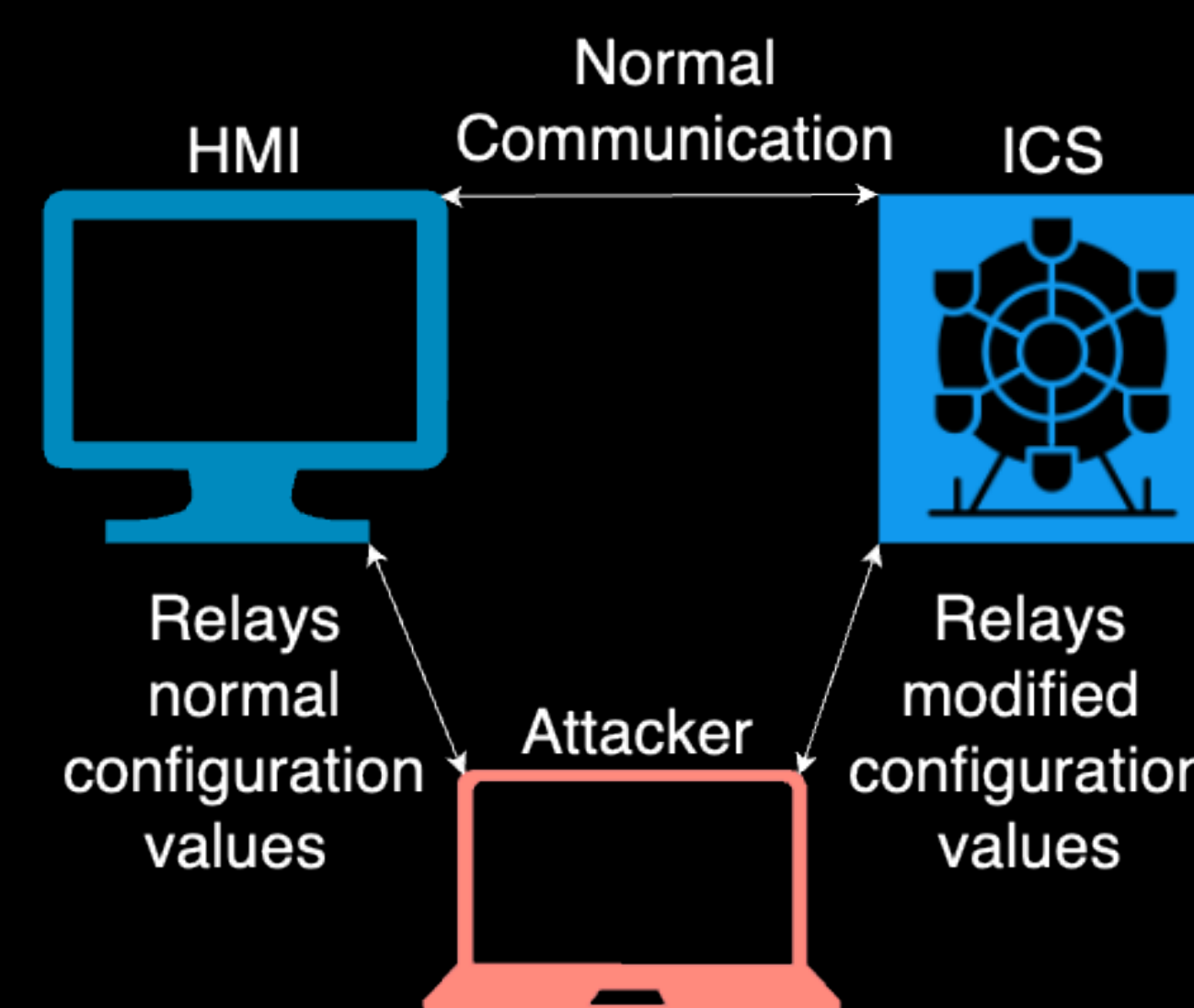


Figure 2. MitM Cyberattack Diagram

- Instance of Unreal Engine hosts the digital model of the physical testbed elements

VALIDATION

- Ferris wheel successfully simulates a physical process (e.g., rotation)—controllable via an HMI
- DoS and MitM attacks successfully interrupt the normal functioning of the system by preventing or hijacking rotation control

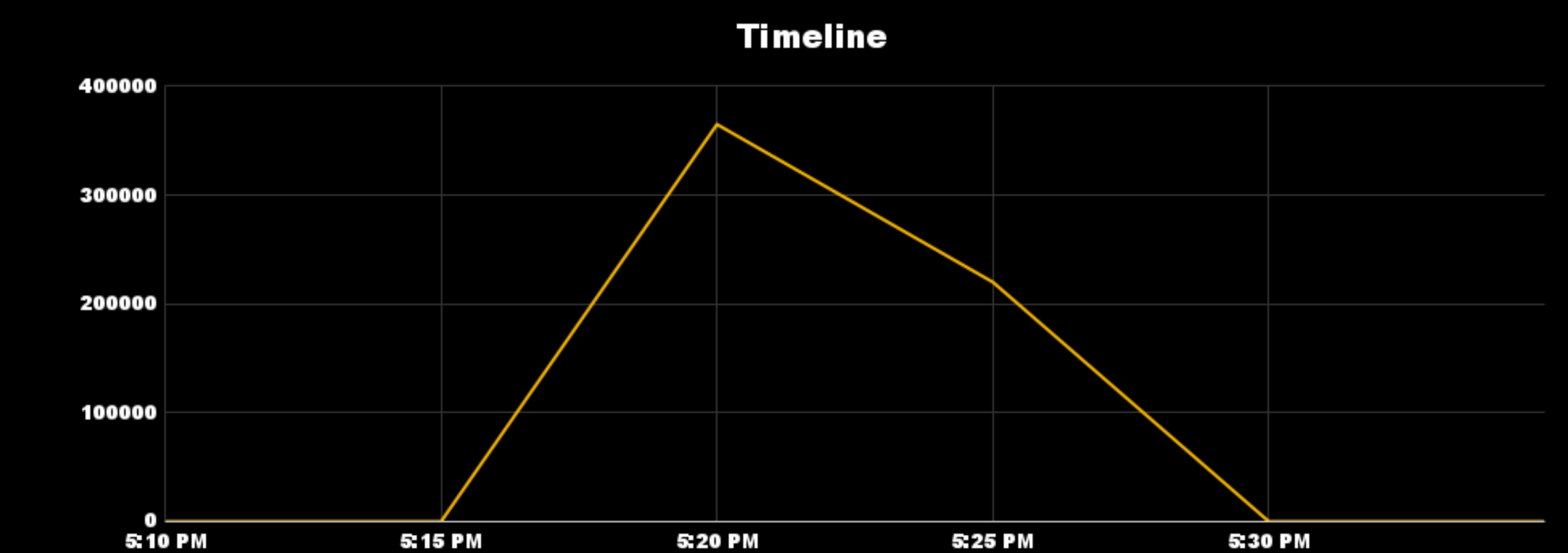


Figure 3. Testbed Traffic During DoS Attack

- Security Onion successfully logs all network traffic events, enabling the observation of cyberattacks
- Virtual model mirror the motion of the physical wheel, responding to commands from the PLC

CONTINUATIONS

- Model status modifications to virtual model as physical modifications (bidirectional modelling)
- Expend cyberattacks to include additional techniques
- Implement rotation speed control