SINGAPORE UNIVERSITY OF TECHNOLOGY AND DESIGN

CPS-SPC 16 @ Vienna AU

Towards High-Interaction Virtual ICS Honeypots-in-a-Box

DANIELE ANTONIOLI ANAND AGRAWAL N. O. TIPPENHAUER

Overview

SINGAPORE UNIVERSITY OF TECHNOLOGY AND DESIGN

In this work we:

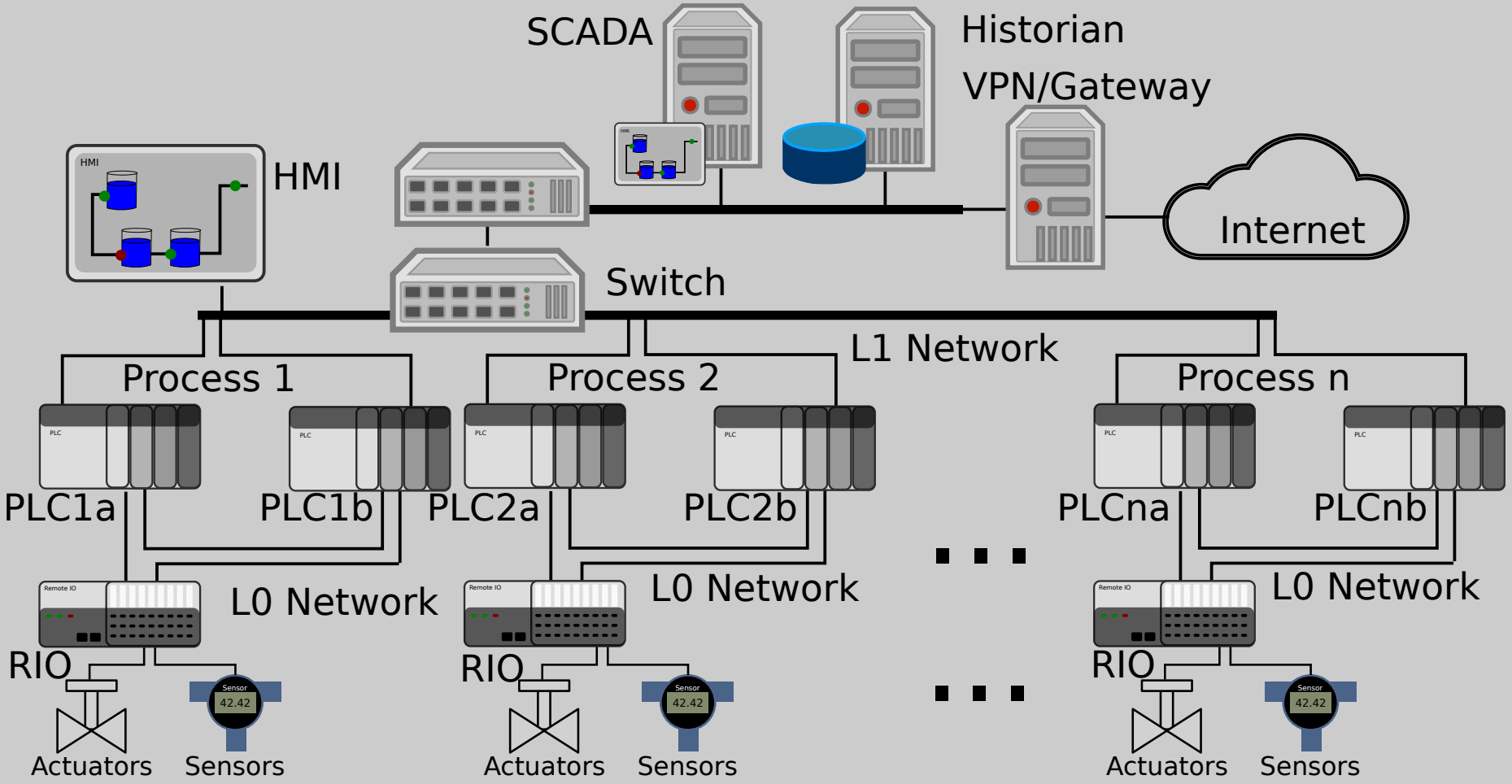
- Present the design of a realistic ICS honeypot
  - Satisfying traditional, and ICS requirements
  - That is high-interaction, virtualized and low-cost
- Show an implementation of such a design
  - Targeting ICS based on Ethernet/IP
  - High-interaction without full virtualization
  - Compatible with Software-Defined Networking
- Discuss its evaluation
  - S3's Capture-The-Flag (CTF) for ICS

Industrial Control Systems (ICS)

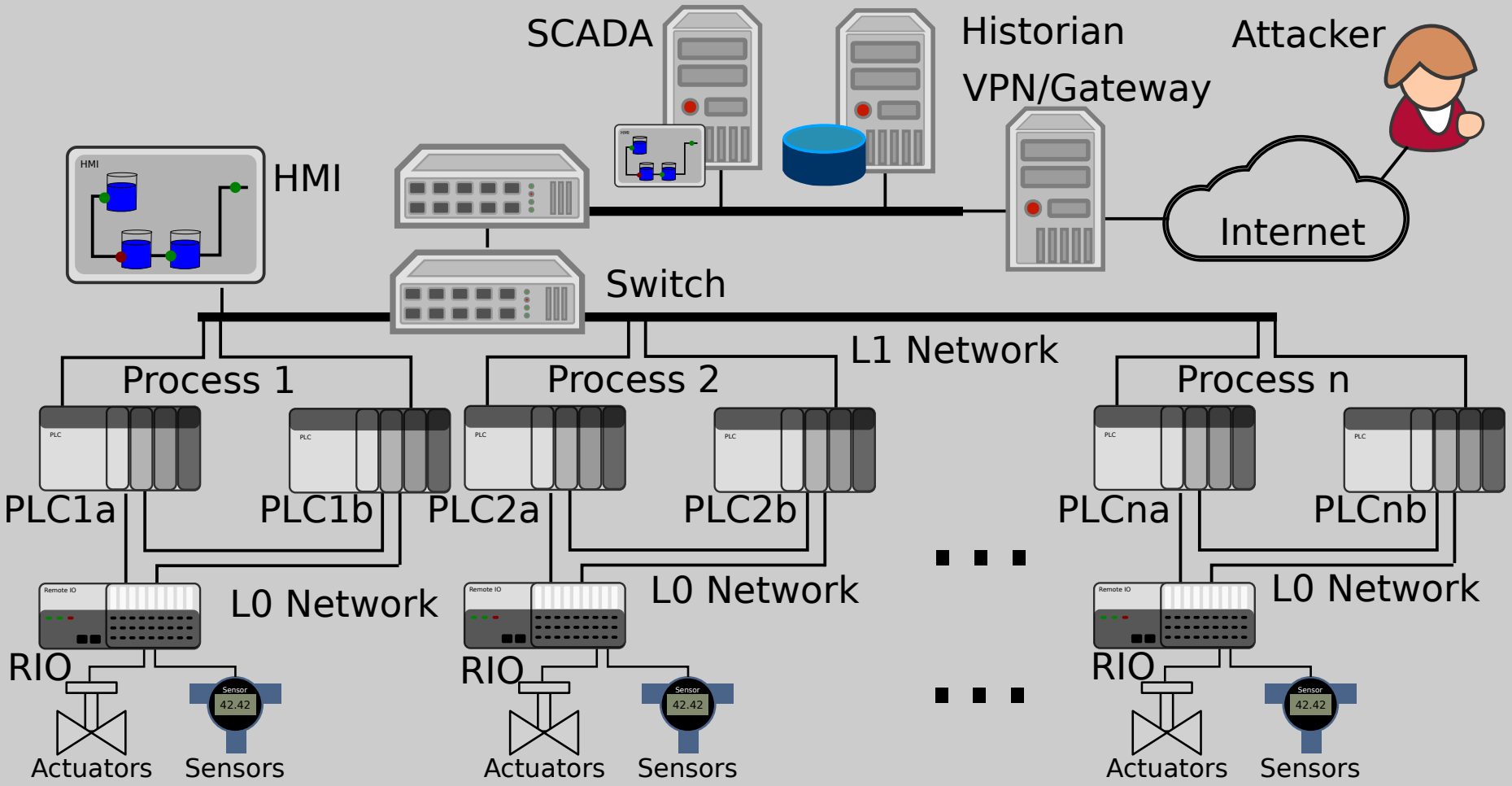
SINGAPORE UNIVERSITY OF TECHNOLOGY AND DESIGN

- Industrial Control Systems (ICS)
  - Connected devices, managing an industrial process
  - Control and monitor: PLC, SCADA, HMI
  - Physical: sensors, actuators
  - Cyber: switches, routers, gateways
- ICS security is a major challenge
  - Internet-facing control networks
  - Cyber *and* physical attacker surface
  - Legacy-code, uncertified devices

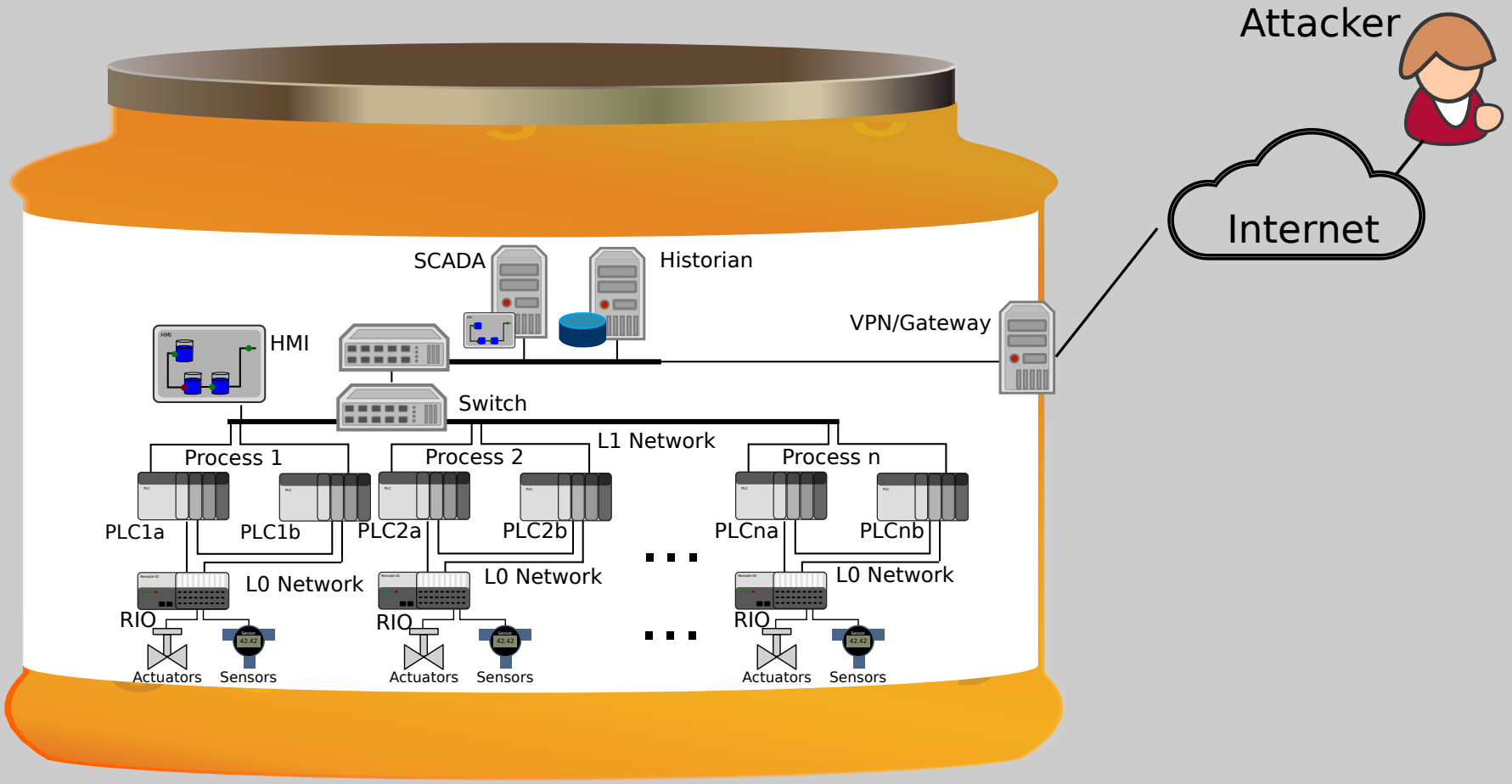
Real Water Treatment ICS



Real Water Treatment ICS



Our Idea: ICS Honeypots



ICS Honeypots: Introduction

SINGAPORE UNIVERSITY OF TECHNOLOGY AND DESIGN

- Systems *intended* be probed, attacked, and compromised
  - Lures the attacker impersonating an ICS
  - Stop, or slow-down the attack
  - Study attacker's behaviours
- Classifications
  - Infrastructure: real vs. virtual (vs. hybrid)
  - Realism: low-interaction vs. high-interaction
  - Role: client vs. server
  - Usage: research vs. production

Our Honeypot: Attacker Model

SINGAPORE UNIVERSITY OF TECHNOLOGY AND DESIGN

- Assumptions
  - Honeypot reached over the Internet
  - Vulnerable interface determines the attacker surface
- Capabilities
  - Fingerprinting: addresses, ports, protocol
  - Protocols: knowledge of all protocols used in system
  - Physical system: limited knowledge of process and devices
- Interactions
  - Denial-of-Service: flood the network
  - Man-in-the-Middle: passive and active
  - Device impersonation: valid and malformed packets
  - Sabotage: trigger actions through malicious commands

Our Honeypot: Requirements

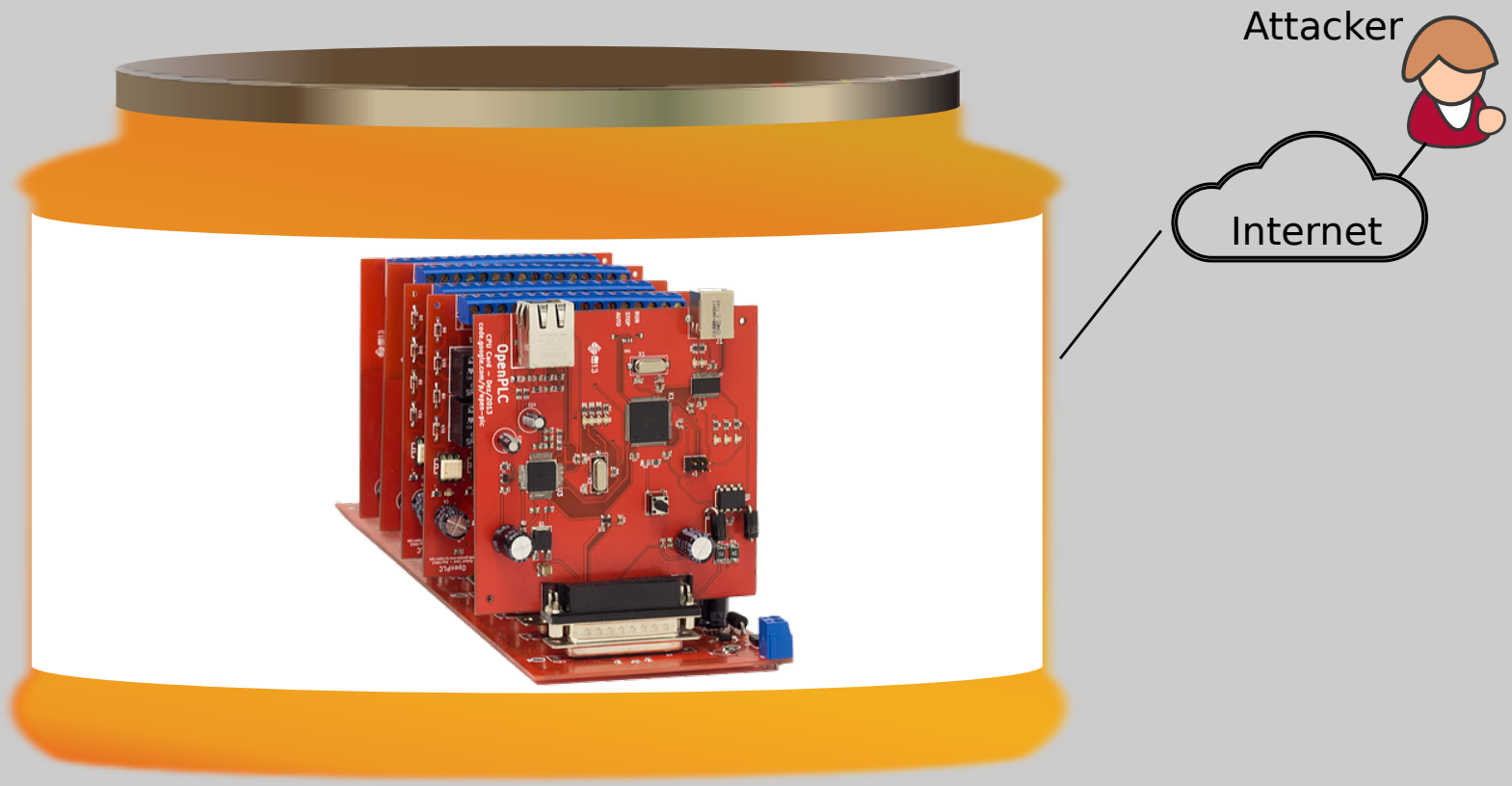
SINGAPORE UNIVERSITY OF TECHNOLOGY AND DESIGN

- High-interaction ICS honeypot
  - Simulate the physical process
  - Simulate the ICS devices: control logic, services
  - Emulate the network infrastructure
- Low-cost
  - Reconfigurable
  - Scales
- ICS requirements
  - *Time*: completion of tasks, and delivery of packets
  - *Determinism*: schedule of tasks, and order of packets

Simple Design Approach

SINGAPORE UNIVERSITY OF TECHNOLOGY AND DESIGN

- How about an OpenPLC[1] indexed on `shodan.io`?
  - Classification: real, low-interaction, server
  - Pros: low-cost, configuration
  - Cons: realism, scale

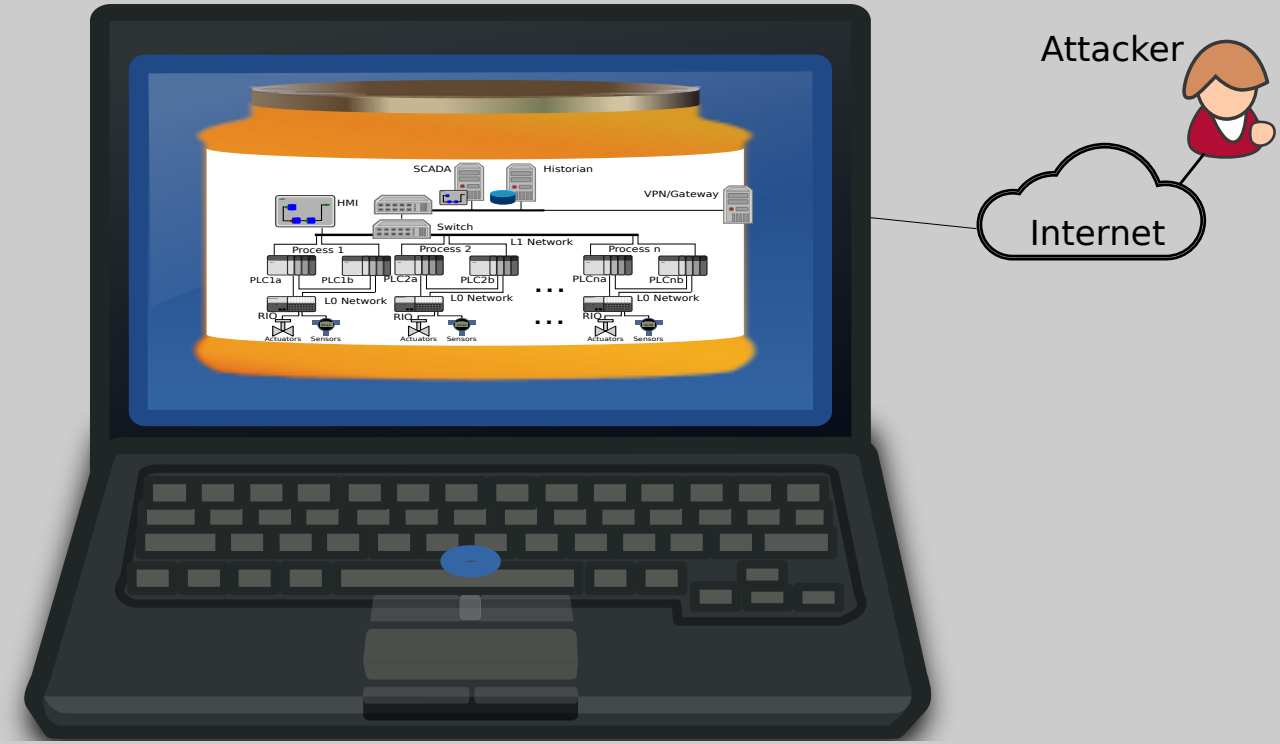


[1] http://www.openplcproject.com/

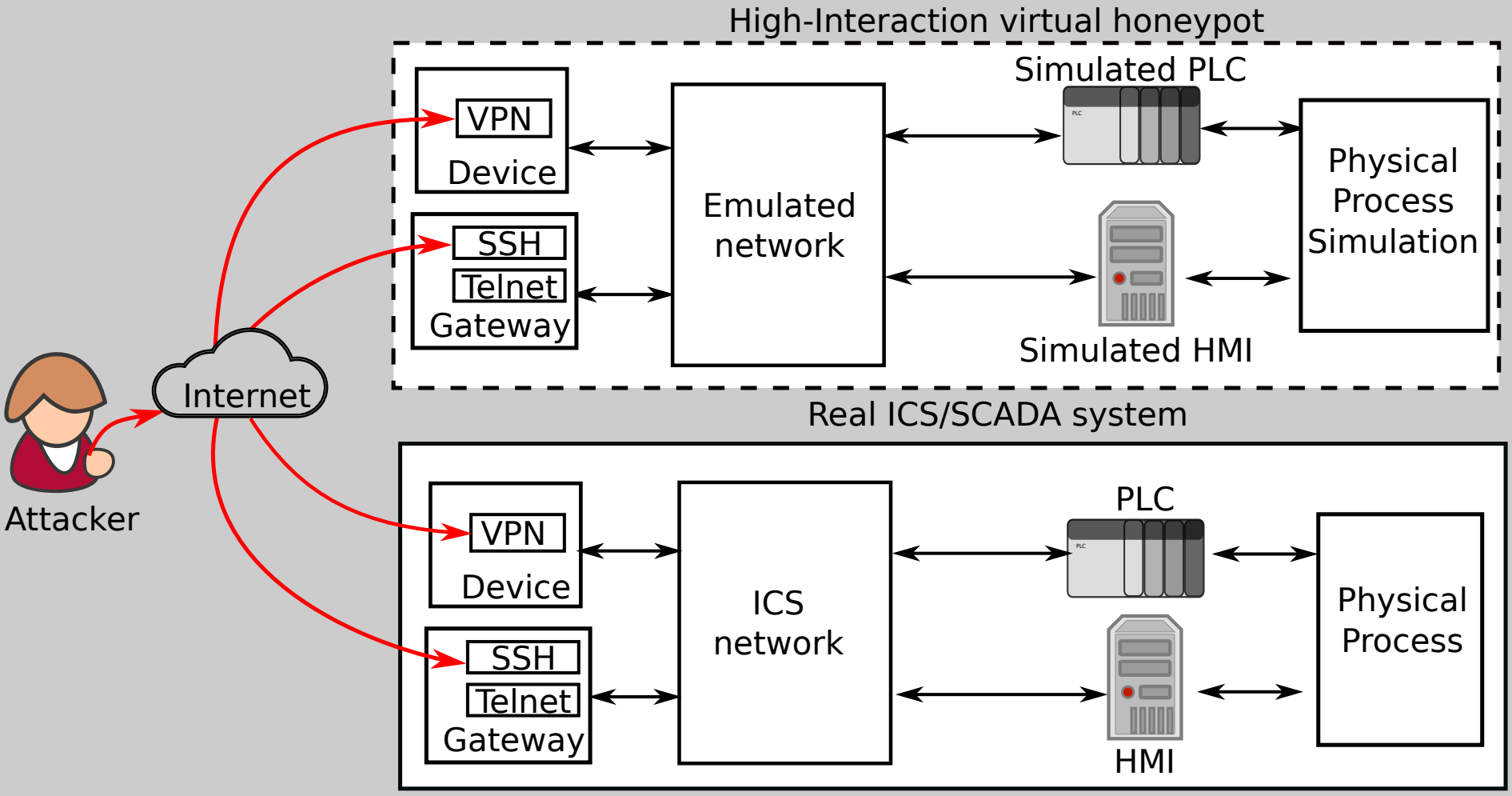
Our Honeypot: Design Choices

SINGAPORE UNIVERSITY OF TECHNOLOGY AND DESIGN

- **Virtual** and **high-interaction**:
  - Simulation of physical process and ICS devices
  - Lightweight network emulation
  - Runs **in-a-Box** (with SDN support)
- ICS requirements
  - Time: real-time emulation, and simulation
  - Determinism: scriptable environment

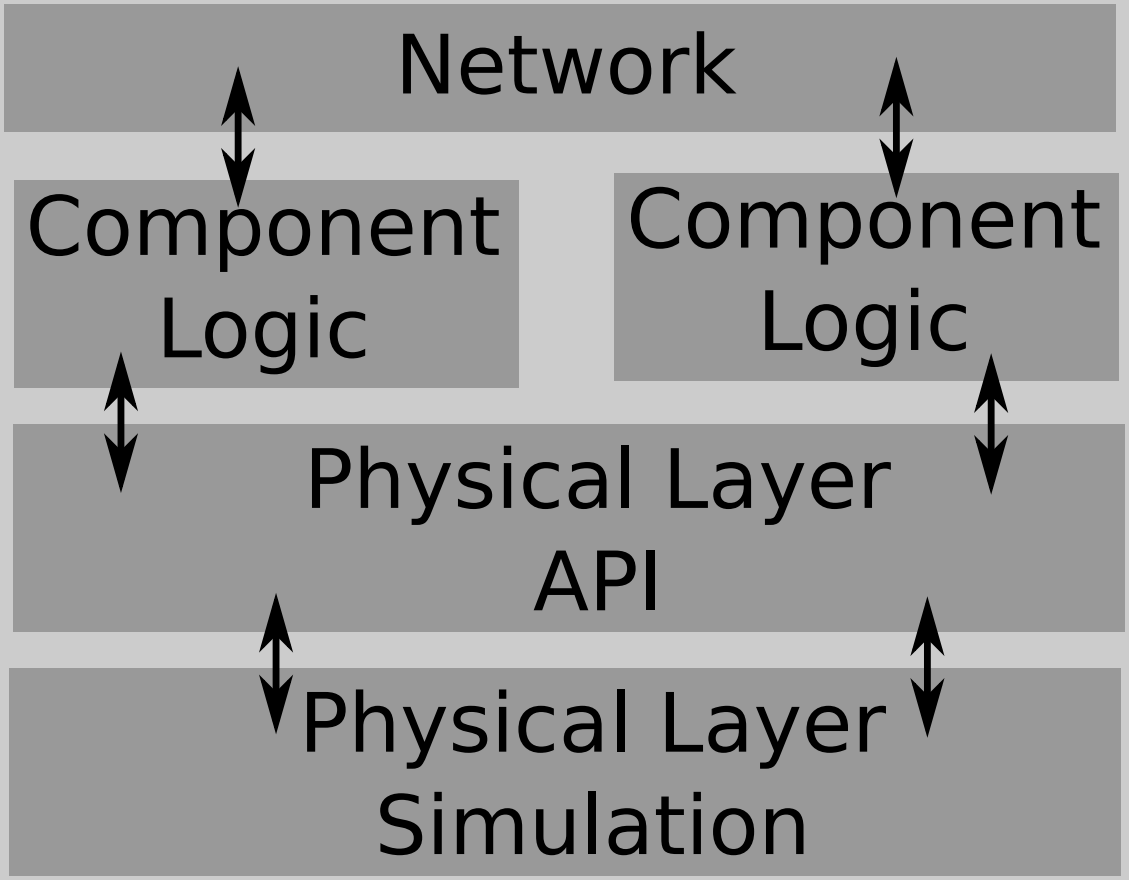


Our Honeypot: Architecture



Proposed Honeypot (top) vs. Real ICS (bottom).

MiniCPS Framework [CPS-SPC 15]

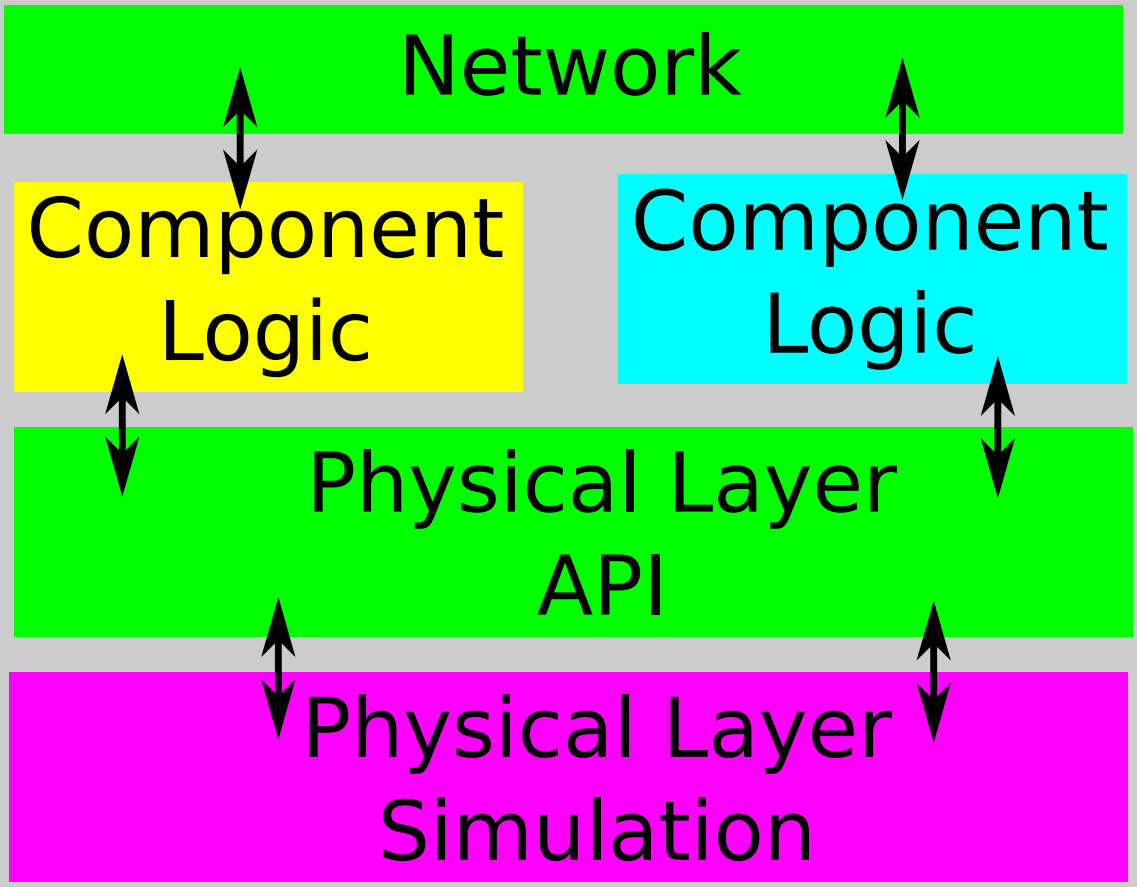


"MiniCPS: A toolkit for security research on CPS Networks."

https://github.com/scy-phy/minicps

(C)yber → Network Emulator
(P)hysical → Physical Layer Simulation and API
(S)ystem → Devices Simulation

MiniCPS Framework [CPS-SPC 15]

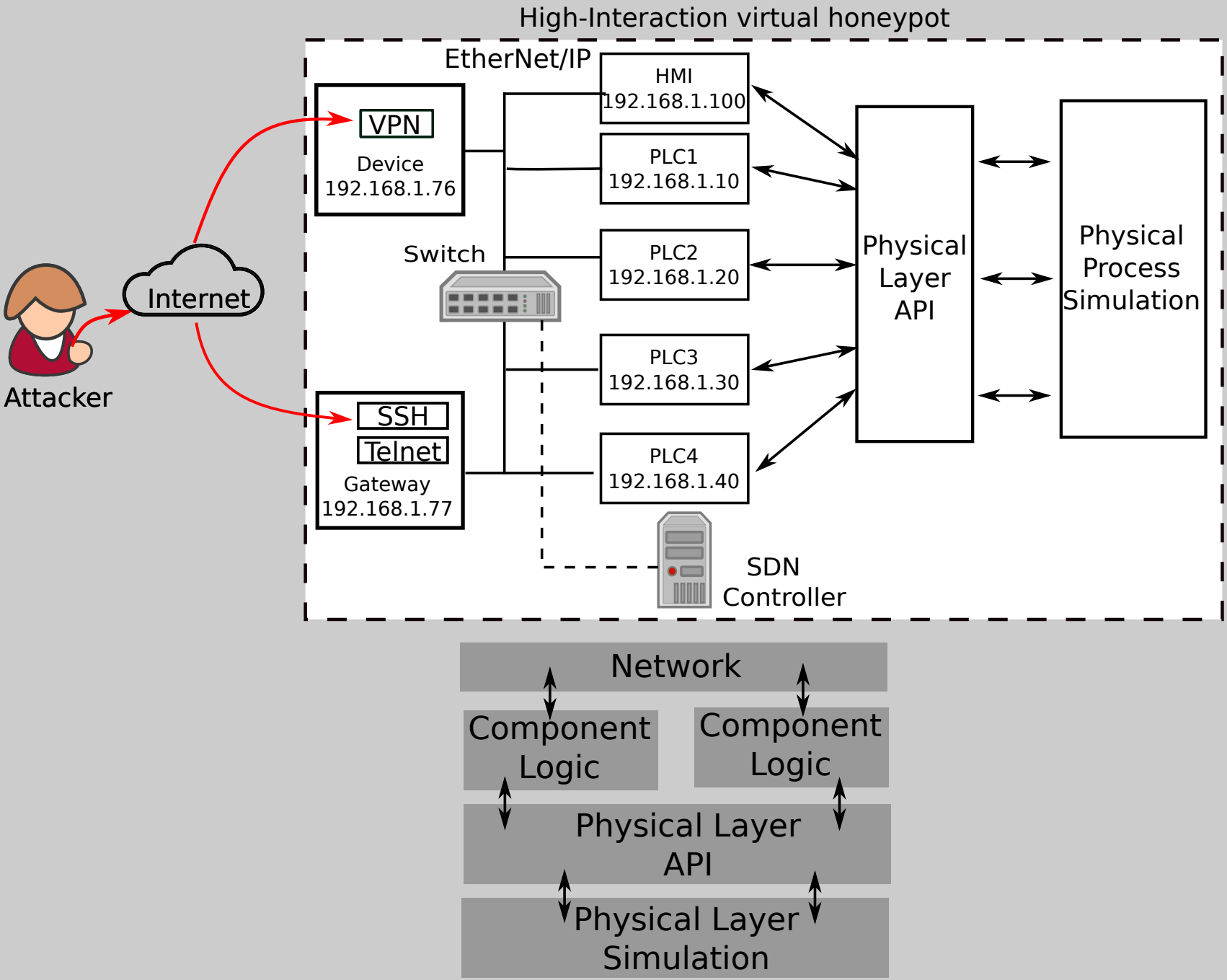


"MiniCPS: A toolkit for security research on CPS Networks."

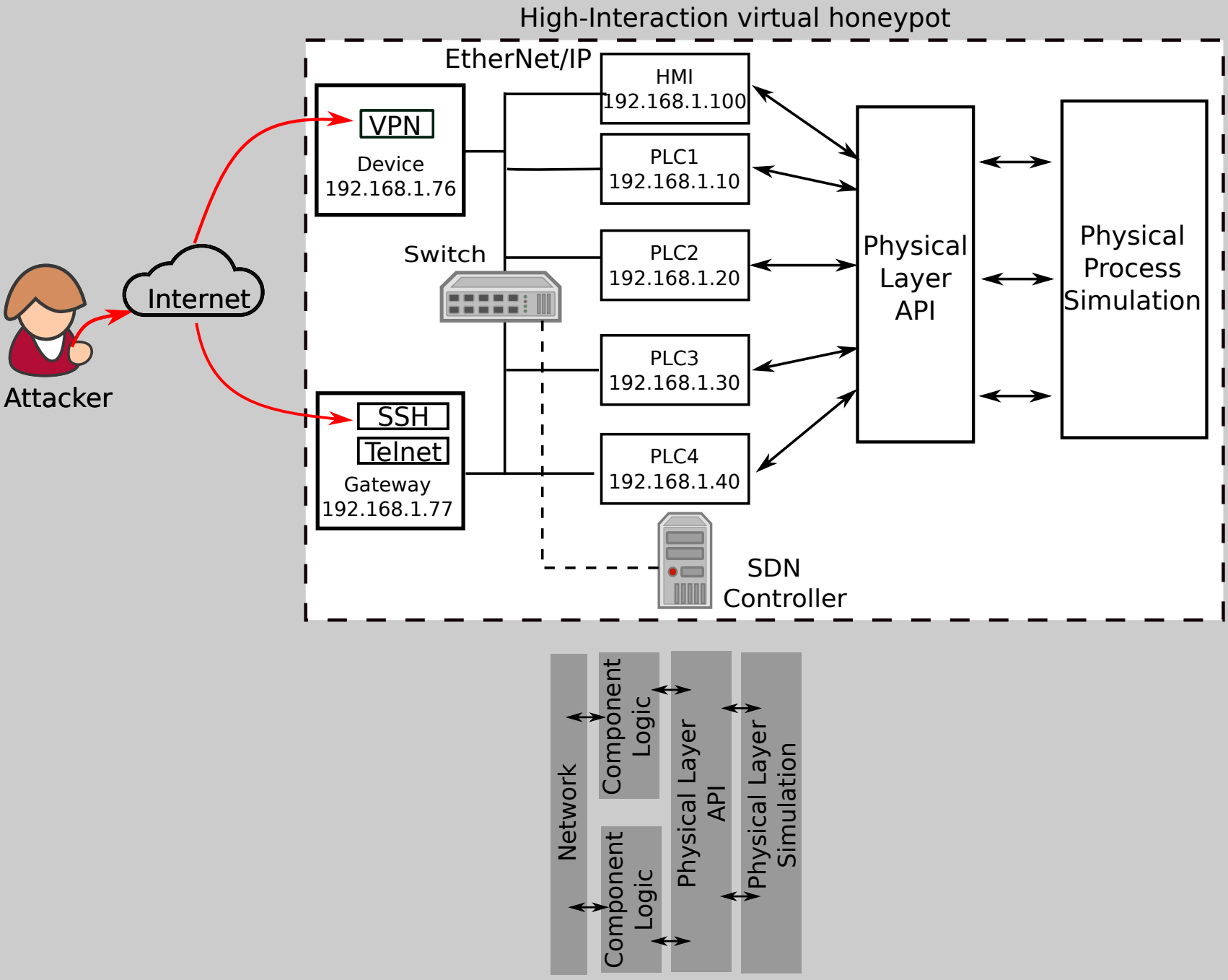
https://github.com/scy-phy/minicps

(C)yber → Network Emulator
(P)hysical → Physical Layer Simulation and API
(S)ystem → Devices Simulation

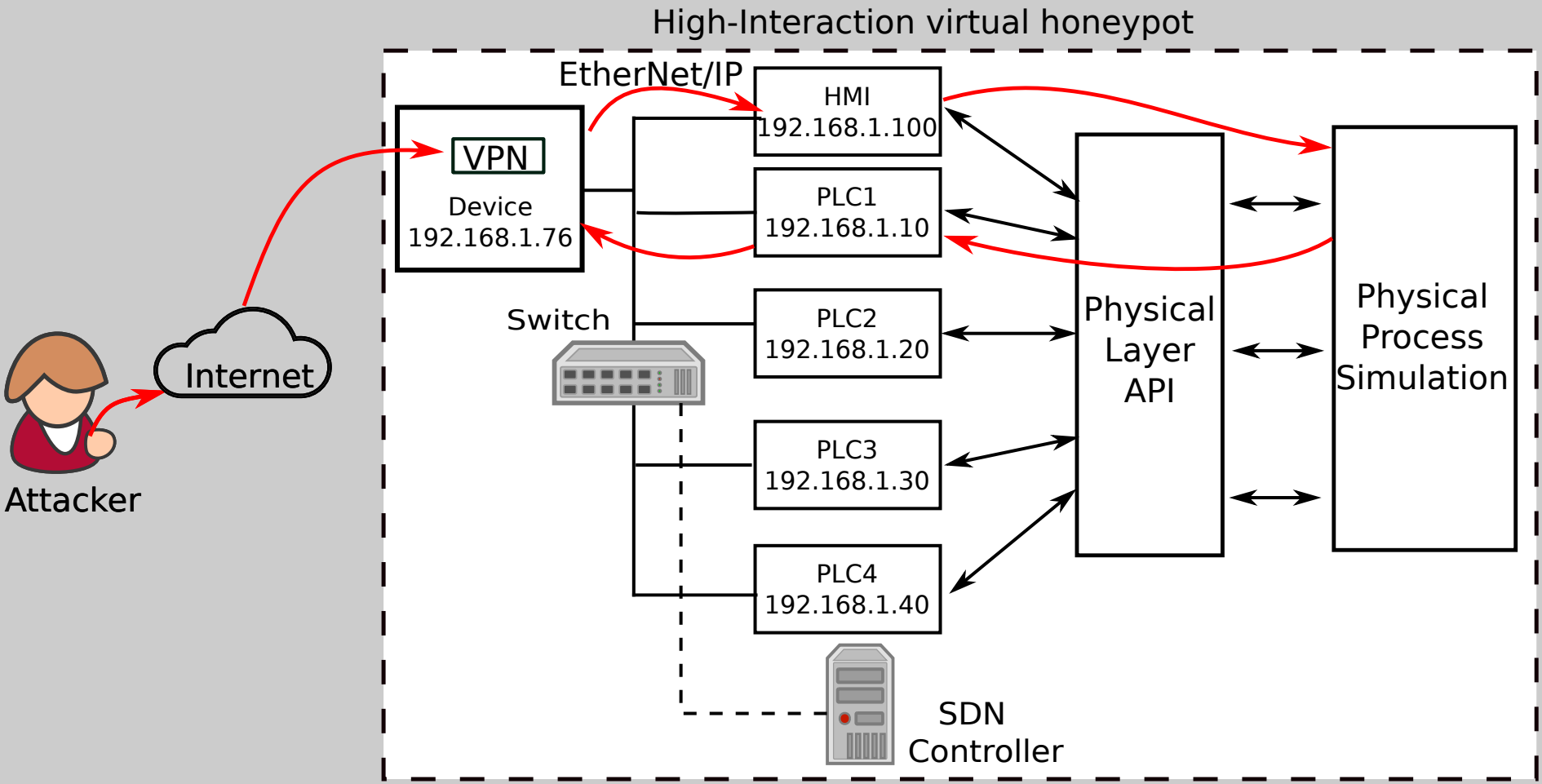
Honeypot Implementation



Honeypot Implementation



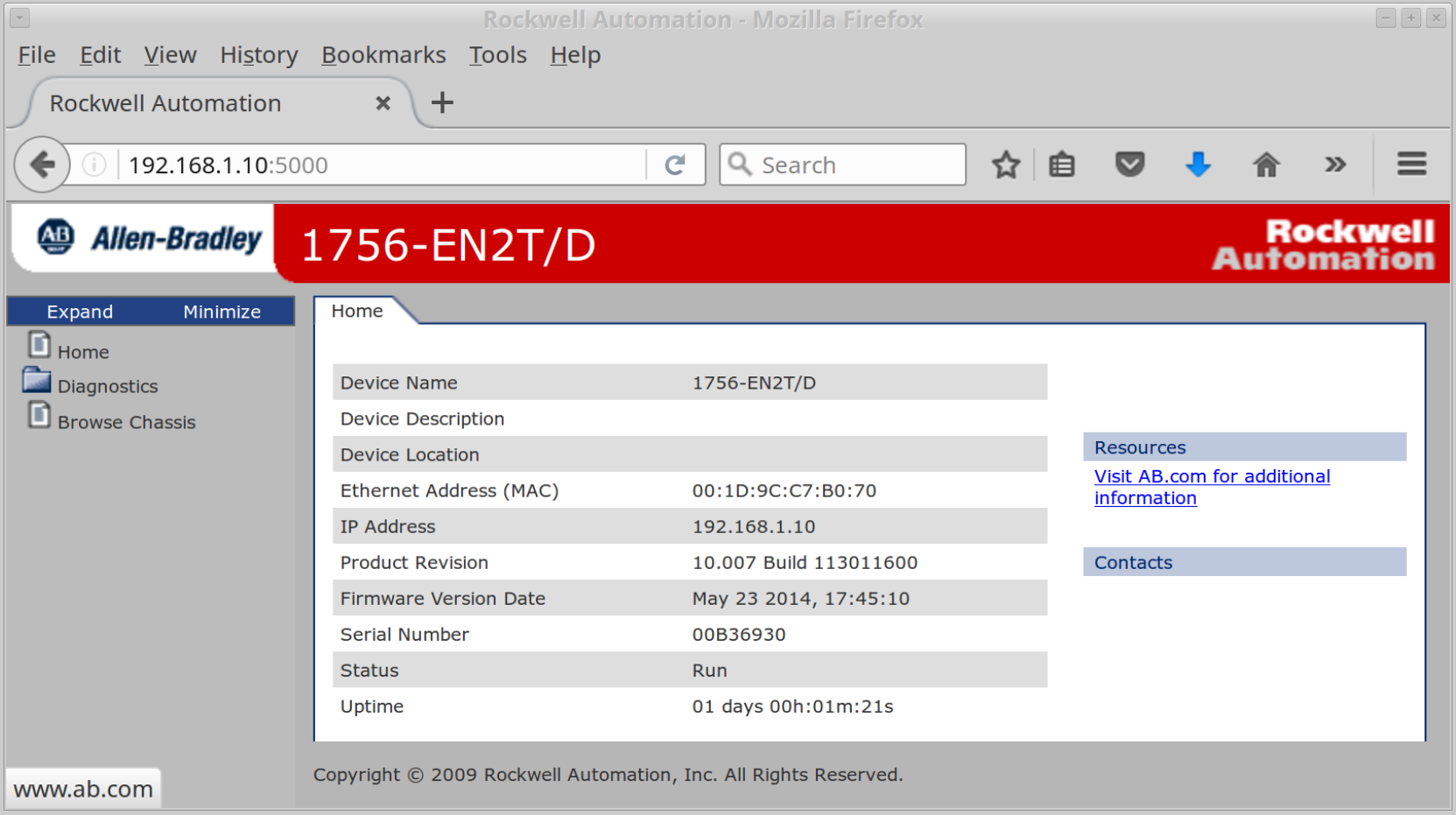
Realistic Attack Propagation



Attack propagates over the simulated components

PLC Implementation

- Allen-Bradley ControlLogix
  - Same IP, MAC, and netmask
  - Simulated control logic (modifiable in real-time)
  - Ethernet/IP server on port 44818, and client
  - Same monitoring Webserver



Network Gateway Device Implementation

SINGAPORE UNIVERSITY OF TECHNOLOGY AND DESIGN

- Moxa OnCell IP gateway
  - Eg: provide IP over 3G connection
  - SSH server with default credentials
  - Telnet server with default credentials (plaintext authentication)
- Virtual implementation
  - Same IP, MAC, and netmask
  - `sshd` on port 22 with default credentials
  - `telnetd` on port 23 with default credentials
  - Attacker gets a (chrooted) shell

Evaluation: S3 Capture-The-Flag (CTF)

SINGAPORE UNIVERSITY OF TECHNOLOGY AND DESIGN

- Capture-The-Flag (CTF)
  - Cybersecurity competition (online and offline)
  - Two types: attack-defense, and jeopardy-style
- S3 CTF was *online* and *jeopardy-style*
  - Tasks divided into categories (cyber, physical)
  - A task has a description, some clues, and reward points
  - A task is solved finding and submitting the correct flag
  - Team that captures most flags (scores most points) wins

Evaluation: S3 CTF Honeypot Setup

SINGAPORE UNIVERSITY OF TECHNOLOGY AND DESIGN

- Honeypots running on AWS EC2 instances[2]
  - Linux, m3-medium: 1 vCPU, 3.75 GB RAM, 1 GB SSD
  - Set up a single instance (tricky)
  - Replicate it (easy, press a button)
- Vulnerable gateway interface
  - SSH's credentials given (CTF)
  - Attacker has a (chrooted) shell
- Replicated part of a water treatment ICS
  - Two tanks, sensors, and actuators
  - Four PLCs and a HMI
  - Ethernet/IP protocol, star topology

[2] https://aws.amazon.com/ec2/

Evaluation: S3 CTF Challenges

SINGAPORE UNIVERSITY OF TECHNOLOGY AND DESIGN

1. Network warm up
   - Task: eavesdrop what PLC2 sends to PLC3
   - Required: testbed's topology, MitM attack skills
   - Solution: passive MitM attack between PLC2 and PLC3
2. Ethernet/IP warm up
   - Task: can you use `cpppo`[3] to access `README:2` tag?
   - Required: Ethernet/IP industrial protocol
   - Solution: Ethernet/IP request (read)
3. Overflow the Raw water tank
   - Task: overflow the Raw water tank controlled by PLC1
   - Required: physical process setup
   - Solution: Ethernet/IP packets to overflow the tank

[3] https://github.com/pjkundert/cpppo

Evaluation: S3 CTF Challenges II

SINGAPORE UNIVERSITY OF TECHNOLOGY AND DESIGN

4 Denial of Service HMI

- Task: change the keep alive value sent from the HMI to PLC3?
- Required: active MitM brute-force attacks
- Solution: active MitM with packet dropping

5 Overflow the Ultra-filtration tank

- Task: control PLC4 to overflow the Ultra-filtration tank
- Required: all the previous challenges
- Solution: active MitM with selective filter

Evaluation: S3 CTF Results

SINGAPORE UNIVERSITY OF TECHNOLOGY AND DESIGN

Table 1: CTF Results Summary.

Teams	# Captured Flags	# Distinct Cmds	# Executed LOC	# Recon Tools	# Attack Tools	Most Used Tools*
Team 1	2	20	1074	3	1	{1, 2, 6, 8}
Team 2	5	30	2488	6	2	{1, 2, 3, 4, 5, 6, 7, 8}
Team 3	3	23	2045	5	2	{1, 2, 3, 4, 6, 7, 8}
Team 4	4	27	963	5	2	{1, 2, 3, 4, 6, 7, 8}
Team 5	1	3	52	1	0	{1}

: Number Of, LOC : Lines Of Code

*{1: `ettercap`, 2: `nmap`, 3: `netstat`, 4: `tcpdump`, 5: `tshark` 6: `ifconfig`, 7: `cpppo`, 8: `ping`}

Conclusions

SINGAPORE UNIVERSITY OF TECHNOLOGY AND DESIGN

In this work, we:

- Address the problem of designing a realistic honeypot for ICS
- Present the design of an *high-interaction*, *virtual*, low-cost ICS honeypot that runs *in-a-Box*
- Show an implementation of such a design based on the MiniCPS framework [CPS-SPC15]
- Discuss its evaluation in the context of an ICS CTF [paper draft]

Acknowledgments: Anand, Nils, and S3 participants'.

Thank you for your time!