A Testbed to Evaluate Next-Generation Security Solutions in Cyber-Physical Systems using Hardware Acceleration

Jose Gomez*, Ali AlSabeh‡, Ali Mazloum†, Elie F. Kfoury†,
Ramy Harik††, Thorsten Wuest†, Jorge Crichigno†

*Katz School of Business, Fort Lewis College, Durango, CO, USA
‡College of Sciences and Engineering, University of South Carolina Aiken, Aiken, SC, USA
†Molinaroli College of Engineering and Computing, University of South Carolina, Columbia, SC, USA
††College of Engineering, Computing and Applied Sciences, Clemson University, Greenville, SC, USA

jagomez@fortlewis.edu, ali.alsabeh@usca.edu, amazloum@email.sc.edu,
ekfoury@email.sc.edu, harik@clemson.edu, twuest@sc.edu, jcrichigno@cec.sc.edu

***Abstract*—At the core of modern manufacturing systems lie Cyber-Physical Systems (CPS) that prioritize operational continuity over security, resulting in a rising number of cyberattacks targeting critical infrastructures. This paper presents a work-in-progress testbed that modernizes Smart Manufacturing Systems (SMS) by integrating Domain-Specific Accelerators (DSAs)—Data Processing Units (DPUs) and Programmable Data Plane (PDP) switches—to strengthen Operational Technology (OT) security without compromising availability or reliability. These accelerators provide fine-grained visibility, real-time anomaly detection, and efficient policy enforcement at line rate. Preliminary results show that accelerator-based applications outperform CPU-based implementations by several orders of magnitude. Demonstrated use cases include a DPU that performs memory inspection via Direct Memory Access (DMA) to detect injected anomalies and a PDP that implements inline detection using pre-trained Machine Learning (ML) models. With low processing overhead, the system also enables continuous telemetry collection for digital-twin generation without disrupting critical operations. The testbed, deployed on the South Carolina Cloud (SC Cloud), offers remote access for developing and evaluating next-generation CPS and OT security applications.**

***Index Terms*—Cyber Physical System (CPS), Smart Manufacturing System (SMS) testbed, Programmable Data Plane (PDP), Data Processing Unit (DPU), digital twins, Domain-Specific Accelerator (DSA).**

I. Introduction

Manufacturing has long been a foundation of industrial economies, yet growing demands for flexibility, customization, and efficiency have catalyzed the transition toward Smart Manufacturing Systems (SMS). This transformation, captured by the Industry 4.0 paradigm [1], integrates cyber, physical, and human systems into intelligent, interconnected, and adaptive production environments enabled by the Industrial Internet of Things (IIoT), edge computing, and real-time analytics [2].

At the core of these systems lie Cyber-Physical Systems (CPS) — tight integrations of computation, communication, and control that link digital intelligence with physical processes to enable autonomous, optimized decision-making [3]. Industrial Control Systems (ICS), including Supervisory Control and Data Acquisition (SCADA) architectures, represent the most widespread CPS deployments and form the operational backbone of manufacturing and critical infrastructure sectors. Traditionally, these systems operated within isolated Operational Technology (OT) networks, physically and logically separated from Information Technology (IT) environments to preserve safety, reliability, and deterministic behavior in control loops.

However, as Industry 4.0 advances, the demand for real-time data integration, predictive maintenance, and cross-layer optimization is accelerating the convergence of OT and IT domains. This convergence introduces substantial security challenges: legacy industrial protocols such as Modbus and DNP3—originally designed for trusted, isolated environments—are now exposed to IP-based networks and external interfaces. Unlike enterprise IT systems, where patching and endpoint protection are routine, OT networks depend on legacy controllers and proprietary stacks that prioritize availability and deterministic timing over security. Deploying Intrusion Detection and Prevention Systems (IDPS) based on general-purpose Central Processing Units (CPUs) can further disrupt time-sensitive operations, particularly for protocols like PROFINET [4], which support cycle times as low as 31.25 microseconds. As a result, OT infrastructures face a critical security gap requiring solutions that enhance protection without introducing disruptive or resource-intensive mechanisms.

Recent advancements in Domain-Specific Accelerators (DSAs) provide new opportunities to strengthen SMS security. DSAs offer (1) programmability to parse and analyze diverse industrial protocols; (2) statefulness to maintain contextual intelligence for threat detection; and (3) ultra-low latency at nanosecond-to-microsecond scales to meet time-critical operational constraints.

Jose Gomez and Ali AlSabeh contributed equally to this work.

This paper presents a work-in-progress testbed that integrates DSAs within an SMS architecture to address these challenges without compromising performance. The testbed, deployed on the South Carolina Cloud (SC Cloud), provides interactive access to hardware accelerators —Data Processing Units (DPUs) [5] and Programmable Data Planes (PDPs) [6] — through a web-based interface [7–9].

Preliminary results demonstrate that PDPs achieve line-rate packet processing with nanosecond-scale latency on custom OT protocols. Likewise, DPUs equipped with Regular Expression (RegEx) processors perform deep packet inspection and policy enforcement at sub-microsecond speeds, even when evaluating rule sets containing over one million entries. Cryptographic and compression accelerators within the DPU sustain high throughput while significantly reducing report generation time, thereby supporting large-scale digital twin deployments. The DPU also performs host memory inspection via Direct Memory Access (DMA) to identify injected binaries with minimal overhead, whereas the PDP dynamically loads pre-trained Random Forest (RF) models to enable inline anomaly detection at runtime.

The remainder of this paper is organized as follows: Section II provides background on hardware accelerators and describes the existing SMS testbed to be modernized with DSAs. Section III presents the proposed architecture, outlining data workflows between IT and OT networks and the integration of DSAs, as well as the SC Cloud platform used for remote access. Section IV discusses preliminary results related to latency-critical functions, network- and host-based defenses, and digital twin implementation within the proposed system. Finally, Section V concludes the paper and outlines directions for future work.

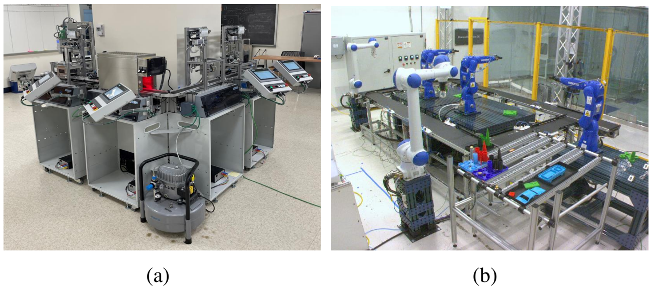
II. Background and Testbed Deployment

Hardware accelerators reduce the fetch, decode, and execute overheads of general-purpose CPUs by running specialized operations on optimized memory and data paths [10]. The result is lower latency and energy consumption per instruction, higher throughput for communication and analytics, and more headroom for control tasks running at the application layer. In the proposed testbed, hardware accelerators on DPUs and PDPs sustain line rate encryption, telemetry extraction, and signature or anomaly detection while preserving deterministic timing of critical operations.

A. Data Processing Units (DPUs)

DPUs are server coprocessors with dedicated hardware modules to implement specialized functions. DPUs provide onboard CPUs and engines for packet processing, encryption, compression, RegEx matching, and DMA. They run an independent operating system with hardware-enforced isolation from host applications. DPUs enable low-latency policy enforcement, telemetry collection, and Deep Packet Inspection (DPI) on commodity servers.

DPUs address the increasing data throughput and complexity of network traffic by moving compute-intensive and latency-sensitive tasks off the general-purpose CPU. Filtering, cryptography, telemetry export, and storage I/O are executed on the DPU, which reduces host interrupts and avoids full traversal of the protocol stack. DMA enables the collection of process lists and the inspection of memory regions for malware detection with minimal host load. This paradigm preserves deterministic behavior of supervisory tasks, enhances visibility, and frees CPU cycles for users' applications.


(a)

(b)

Fig. 1. Testbeds to be modernized with hardware acceleration. (a) Cyber-Physical Lab (CPLab) for assembling cellular phones, consisting of eight cells. Cells 1-4 are shown in the figure, each containing a Human-Machine Interface (HMI), a camera, and a PLC. Workpieces are conveyed between cells, orchestrated by a Manufacturing Execution System. (b) The Future Factories Lab (FFLab) is a testbed with five robotic arms for autonomous rocket assembly and disassembly, showcasing multi-robot coordination and conveyor-driven part handling in a closed-loop manufacturing system.

B. Programmable Data Planes (PDPs)

PDPs allow developers to determine how the switch pipeline will handle network packets [6]. This customization is achieved by defining how packets are parsed on ingress, matched against specified criteria, and processed through the predefined actions. Using languages such as Programming Protocol Independent Packet Processors (P4) [11], operators can deploy packet processing tasks at line rate, including flow classification, filtering, telemetry extraction, and policy enforcement [12]. PDPs provide flexibility parsing custom protocols (e.g., Modbus and PROFINET) and visibility while reducing processing overheads arising from centralized processing. This programmability allows networks to become more agile, responsive, and tailored to specific applications such as traffic classification [9], DDoS mitigation [13], granular performance monitoring [14], and real-time media processing [15].

Typical PDP switches implement programming models that include a programmable parser, a sequence of match action stages, and a programmable deparser. Incoming packets are parsed into headers, processed by tables that perform arithmetic operations and header updates, then reassembled for forwarding. The control plane manages table entries and runtime behavior, while the data plane executes decisions at line rate. This separation enables dynamic policy updates and maintains high performance within the hardware pipeline.

C. Testbed Deployment

In collaboration with smart manufacturing researchers at the University of South Carolina, two existing SMS testbeds —

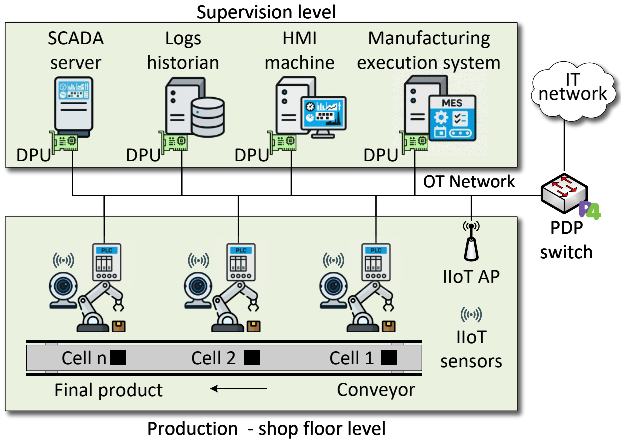


Fig. 2. Proposed DSA-enabled SMS. The production shop floor contains multiple cells connected by a final product conveyor and an OT network with Human-Machine Interfaces (HMIs) and a SCADA server, where each endpoint integrates a DPU for inline processing. DPUs provide offloading capabilities to the end hosts. The PDP switch inspects the traffic from and to the IT network.

namely, the Cyber Physical Lab (CPLab) [16] and the Future Factories Lab (FFLab) [17] — will serve as the foundation for the proposed DSA-enabled architecture, as illustrated in Fig. 1. These testbeds will be enhanced with DPUs and PDPs (Section III-A) to improve the visibility, resilience, and security of the OT layer (Section III-B). The CPLab focuses on cellular-phone assembly across eight interconnected cells integrating Human–Machine Interfaces (HMIs), cameras, Programmable Logic Controllers (PLCs), and industrial sensors, whereas the FFLab features coordinated robotic workcells for autonomous assembly and disassembly supported by conveyor-based part handling. To broaden accessibility for researchers, practitioners, and students, the entire ecosystem will also be made remotely available through the SC Cloud infrastructure (Section III-B).

III. Proposed System

A. DSA-Enabled SMS Testbed

Fig. 2 illustrates the high-level architecture of the proposed DSA-enabled SMS, modeling the core components of the CPLab and FFLab. The production layer comprises IoT sensors, actuators, and conveyor systems that perform assembly operations, while the supervision layer includes a SCADA server, logs database, HMI, and Manufacturing Execution System (MES). Each supervisory component integrates a DPU to offload security and monitoring tasks within the OT network. A PDP switch operates between the OT and IT domains to inspect network traffic at line rate, parse industrial protocols, and detect anomalies in real time. The proposed implementation employs NVIDIA BlueField-3 DPUs [18] and an Intel Tofino PDP [19]. Servers at the supervision layer are instantiated as Virtual Machines (VMs) running on a hypervisor, with DPUs interfaced via the Peripheral Component Interconnect (PCI) passthrough [20]. Together, the PDP and DPUs provide integrated visibility across network and host layers, achieving comprehensive situational awareness while adhering to the latency constraints of time-critical manufacturing processes.

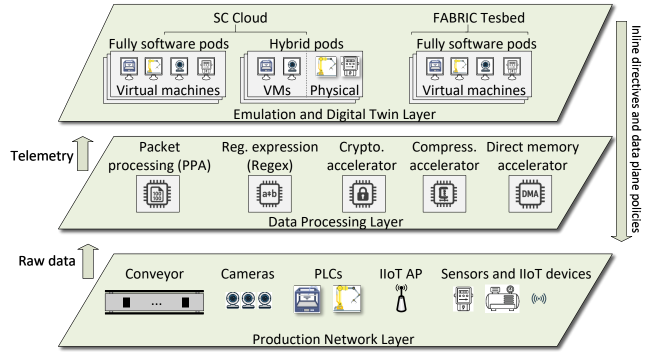


Fig. 3. System architecture. The Production Network Layer generates raw traffic. Hardware accelerators in the Data Processing Layer consume this traffic and produce telemetry reports. The Emulation and Digital Twin Layer performs high-dimensional data analytics. Inline policies and directives are applied to OT assets.

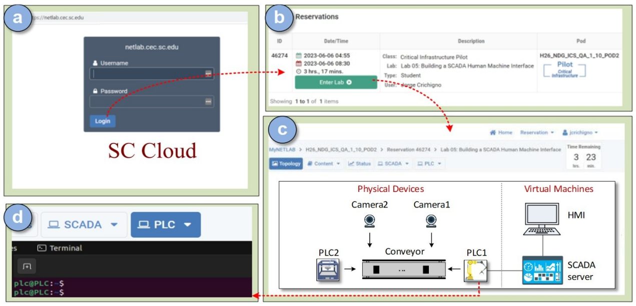
B. Data Workflow of within the DSA-Enabled SMS

Fig. 3 illustrates the workflow of the proposed DSA-enabled SMS, organized into three functional layers. The Production Network Layer represents the shop floor, where PLCs, IIoT sensors, Access Points (APs), cameras, and conveyor subsystems generate continuous operational data. The Data Processing Layer integrates DPUs and PDPs to accelerate and secure these data streams. Within this layer, a Packet Processing Accelerator (PPA) extracts fine-grained telemetry at line rate, a RegEx accelerator performs high-speed signature matching, a cryptographic accelerator secures telemetry exports, a compression accelerator reduces report size, and a direct memory accelerator transfers data directly to collector memory for analysis.

The Emulation and Digital Twin Layer hosts user interfaces and virtualized environments deployed across the SC Cloud and FABRIC—a national research infrastructure for experimentation at scale [21]. This layer supports fully virtual, physical, and hybrid pods to emulate end-to-end industrial processes. The architecture operates passively through traffic mirroring and telemetry collection, introducing no additional processing overhead to OT or IT systems. This design preserves deterministic, line-rate performance on the shop floor while enabling continuous monitoring and policy enforcement through advanced analytics.

C. SC Cloud Platform

Fig. 4 shows the interface that experimenters and learners use to access components in the proposed system [7, 8]. At the start of a reservation, the automated platform retrieves a pod with a preconfigured state for all devices in the pod. A pod is a collection of components, such as PLCs, DPUs, PDPs, and SCADA servers that are reserved and networked to reproduce a specific attack/defense scenario. Users receive exclusive access to these components and follow the steps outlined in a lab manual to conduct experiments. Provisioning



ID	Date/Time	Description	Pod
46274	2023-06-06 04:55 2023-06-06 08:30 3 hrs., 17 mins.	Class: Critical Infrastructure Pilot Lab: Lab 05: Building a SCADA Human Machine Interface Type: Student User: Jorge Crichigno	H26_NDG_ICS_QA_1_10_POD2

Showing 1 to 1 of 1 items

MyNETLAB > H26_NDG_ICS_QA_1_10_POD2 > Reservation 46274 > Lab 05: Building a SCADA Human Machine Interface

Time Remaining 3 hrs. 23 min.

```
plc@PLC:~$
plc@PLC:~$
```

Fig. 4. Overview of the SC Cloud web interface. (a) The user authenticates at the cloud portal. (b) Then, the user reserves a POD for a specific experiment. (c) The cloud recreates the scenario for the experiment. (d) The user then interacts with the devices by clicking on the corresponding tabs in the environment.

completes in seconds and requires no human intervention to reproduce scenarios. In the example, the SC Cloud emulates plant operation with a conveyor system, two PLCs, cameras, a SCADA server, and an HMI.

IV. Preliminary Results

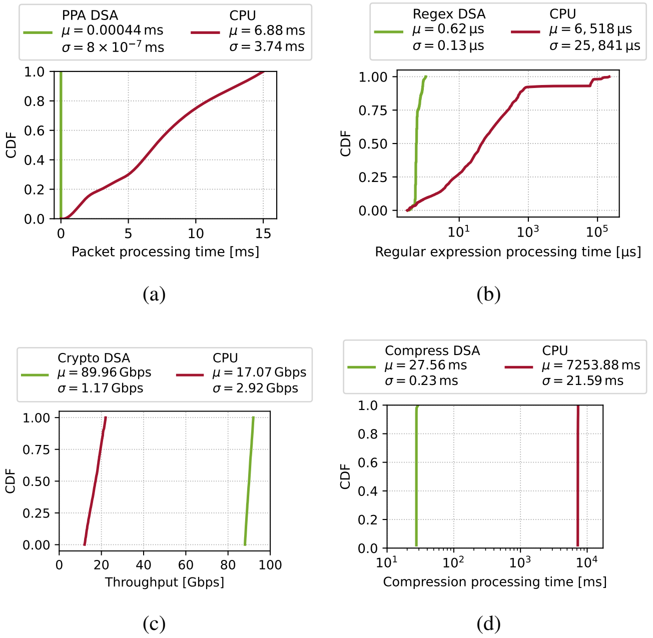
This section shows the preliminary results focusing on three areas of CPS security: anomaly detection in OT traffic to verify that the assets are not deviating from normal behavior, anomaly detection in the IT network, where a PDP is used to detect anomalies in lateral moves, and host memory inspection using a DPU with DMA collection and RegEx driven YARA matching to reveal malicious processes with minimal host overhead. The results also discuss how hardware accelerators can be used to collect shop floor metrics to implement digital twins.

A. Latency-critical Functionalities

OT systems have historically prioritized availability and reliability over security. Consequently, it is essential that newly introduced DSA components operate with minimal latency, remain non-disruptive, and provide sufficient adaptability for future modifications while sustaining high performance. Fig. 5 presents the performance evaluation of four key DSA accelerators — PPAs, RegEx engines, cryptographic accelerators, and compression accelerators — demonstrating their efficiency in maintaining deterministic, line-rate operation within the SMS environment.

PPAs. Traditional manufacturing networks rely on fixed-function switches that lack programmability and security awareness. PPAs embedded within programmable switches and DPUs enable parsing and monitoring of OT-specific protocols at line rate, supporting real-time anomaly detection and malware identification [22]. As shown in Fig. 5(a), the Tofino PDP achieves sub-microsecond per-packet processing, while CPU-based processing requires several milliseconds—an improvement of several orders of magnitude that ensures low jitter and preserves timing guarantees for critical control flows.

RegEx. Signature-based DPI remains a foundational layer in industrial cybersecurity, yet traditional CPU-based DPI engines struggle to maintain RegEx matching speed as rule sets increase in scale and complexity, leading to throughput degradation and delayed detection that disrupt real-time responsiveness. In contrast, the DPU-based RegEx accelerator (Fig. 5(b)) sustains sub-microsecond processing across a one-million-rule policy set, compared to millisecond-level delays on CPUs. This enables scalable, line-rate enforcement of complex detection policies without compromising deterministic communication.



(a)

(b)

(c)

(d)

Fig. 5. Performance of various Domain Specific Accelerator (DSAs) compared to CPUs: (a) Packet processing time. (b) Tegular Expression (RegEx) processing time. (c) Encryption throughput. (d) Compression processing time.

Crypto. Encryption throughput results, shown in Fig. 5(c), indicate that the DPU's cryptographic engine achieves approximately 90 Gbps compared to 17 Gbps on the CPU. This ensures that securing telemetry exports introduces negligible performance overhead, maintaining the integrity and confidentiality of data streams in real time.

Compress. Fig. 5(d) reports compression processing times, where the DPU completes report generation in tens of milliseconds versus several seconds on the CPU. Accelerated compression significantly reduces telemetry size, transfer latency, and storage pressure, enhancing scalability for continuous monitoring pipelines.

These results confirm that DPUs and PDPs can accelerate the entire monitoring pipeline—encompassing packet processing, signature matching, encryption, and compression—achieving orders-of-magnitude performance gains over CPU-based implementations while maintaining line-rate operation and non-disruptive integration with existing OT systems.

B. Network-Based Defenses

PDPs can be leveraged to detect network-level attacks in real time, enforcing learned security policies at line rate

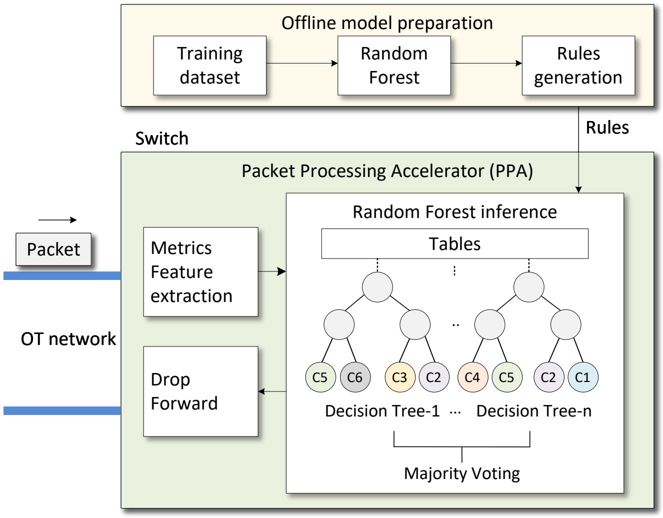


Fig. 6. Attack detection using an RF model implemented on a PPA inside a PDP switch. The RF is trained offline and then translated into rule-based trees, which are loaded into the PDP as tables, registers, and counters for line-rate inference.

while maintaining simple control logic. In the proposed approach, illustrated in Fig. 6, a Random Forest (RF) classifier is trained offline on curated traffic traces and subsequently compiled into rule tables that represent decision trees within the PPA of the switch. The RF model is trained externally on a general-purpose server due to the limited memory and computational resources available on PDP hardware. Once trained, the model is translated into compact match–action entries encoding decision nodes and leaves, which the control plane installs dynamically via standard APIs. This mechanism allows operators to refresh or replace models as new threats emerge without interrupting production traffic.

During operation, the PPA extracts per-packet and per-flow features (e.g., counts, rates, timing) and maps them to RF rule tables. Packets follow the corresponding decision paths, with majority voting implemented via table actions/counters; the resulting label determines forwarding or dropping. Implementing inference solely through match–action lookups preserves deterministic latency and line-rate throughput, and enables fast rollbacks and staged updates by swapping active table entries under control-plane supervision.

While PDPs enable high-speed in-network inference, they remain constrained by hardware memory, stateful depth, and the complexity of supported operations (e.g., multiplication and division are not supported). Therefore, certain anomaly detection tasks can be distributed collaboratively between PDPs and DPUs. In this configuration, the PDP performs initial packet parsing and metadata extraction, exporting structured flow records to the DPU for advanced machine learning analysis.This coordination combines line-rate feature collection by the PDP with high-fidelity ML inference on the DPU, enabling adaptive, multi-layered threat detection across the SMS architecture [23].

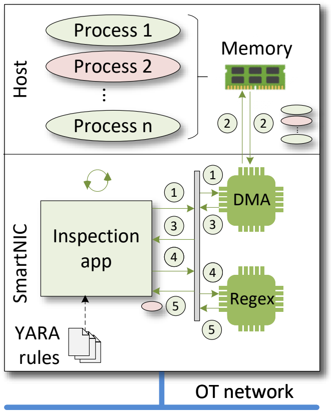


Fig. 7. Host memory inspection. An inspection application running on the DPU periodically collects host process lists and associated memory metadata using the DMA, then forwards this data to the RegEx engine to match against YARA rules.

C. Host-Based Defenses

While network-level monitoring enables real-time detection of traffic anomalies, certain attack vectors—such as fileless malware, process injection, or privilege escalation—cannot be identified through network traffic alone. Detecting these threats requires direct visibility into host memory and process activity. In CPS environments, critical hosts such as monitoring servers, log databases, and SCADA servers must remain secure and reliable while executing time-sensitive operations. Fig. 7 illustrates a DPU-assisted host defense approach in which the DPU performs isolated memory inspection to identify abnormal activity without affecting host performance.

The DPU runs a lightweight OS with dedicated accelerators and remains logically isolated from host applications, ensuring that inspection agents continue operating even if the host is compromised. When configured in restricted mode, it relies on an immutable runtime with no external interfaces and hardware-enforced isolation, reducing attack surfaces typical of general-purpose CPU systems. Detection on the DPU integrates rule-based methods with behavioral analysis to enable rapid identification of known threats and adaptive recognition of emerging anomalies.

The first detection strategy implements signature-based analysis using YARA rules—a rule-based language that specifies textual or binary patterns associated with known threats. As shown in Fig. 7, the DPU's inspection application periodically initiates a scan by invoking the DMA accelerator to retrieve process descriptors and selected memory regions from the host with minimal latency and no CPU overhead. The collected data are then forwarded to the RegEx accelerator, which performs pattern matching against the YARA rules and returns alerts for logging and response. This workflow prevents interruptions to the host, preserving deterministic timing for ongoing control and monitoring operations.

The second detection strategy applies anomaly-based analysis by establishing a baseline of legitimate system behavior through a Software Bill of Materials (SBOM) that catalogs software components, dependencies, and API usage. At configured intervals, the DPU captures memory snapshots and compares them against the baseline. Deviations—such as new binaries, unexpected modules, or modified code segments—are flagged as potential intrusions and reported to the supervision layer for further evaluation or mitigation according to defined policies.

DMA-based collection minimizes host overhead and enables frequent scanning without disrupting production workloads. Although higher throughput may imply greater energy usage, offloading inspection and pattern-matching operations to dedicated DPU accelerators improves overall energy efficiency compared to CPU-based inspection [24]. This architecture provides continuous visibility into critical hosts and preserves line-rate behavior on the production network.

D. Digital Twins Implementation

Digital twins rely on deterministic, continuous streams of telemetry from shop-floor assets to maintain synchronization between physical processes and their virtual counterparts. In CPS environments, this synchronization must occur in near real time without disrupting control loops or affecting production stability. The proposed system achieves this by leveraging hardware accelerators to collect telemetry with minimal processing overhead. Packet features and flow records are extracted directly within the programmable switch pipeline, while the operational state of OT devices is retrieved through Direct Memory Access (DMA) modules on the DPUs.

V. Conclusion and Future Work

This paper presents a work-in-progress testbed that revolutionizes the security of next-generation SMS. The testbed leverages specialized hardware accelerators, including DPUs and PDPs. Learners and experimenters access the platform through the SC Cloud web interface. The deployment will be integrated with two physical laboratories to monitor OT and IT assets and to enable digital twins. Preliminary results show that PDP- and DPU-based security applications achieve performance gains of several orders of magnitude over general-purpose CPU baselines. This capability supports line rate monitoring and anomaly detection across OT and IT domains. The system also loads pretrained ML models to extend detection to emerging threats. The proposed system will integrate with FABRIC [21] through its portal to broaden access, support, and federate resources for large-scale experimentation.

Acknowledgment

The authors acknowledge support from the National Science Foundation (NSF) under award 2400729. They also thank the FABRIC [21] team for granting access to the platform used in the preliminary experiments.

References

[1] H. Lasi, P. Fettke, H.-G. Kemper, T. Feld, and M. Hoffmann, "Industry 4.0," *Business & information systems engineering*, vol. 6, no. 4, pp. 239–242, 2014.

[2] F. Tao, Q. Qi, L. Wang, and A. Nee, "Digital twins and cyber–physical systems toward smart manufacturing and industry 4.0: Correlation and comparison," *Engineering*, vol. 5, no. 4, pp. 653–661, 2019.

[3] R. V. Arcot, "Cyber-Physical Systems: The Core of Industry 4.0." International Society of Automation (ISA) Smart Manufacturing & IIoT Division. [Online]. Available: https://blog.isa.org/cyber-physical-systems-the-core-of-industry-4.0.

[4] J. Feld, "Profinet-scalable factory communication for all applications," in *IEEE International Workshop on Factory Communication Systems, 2004. Proceedings.*, pp. 33–38, IEEE, 2004.

[5] E. Kfoury, S. Choueiri, A. Mazloum, A. AlSabeh, J. Gomez, and J. Crichigno, "A comprehensive survey on SmartNICs: Architectures, development models, applications, and research directions," *IEEE Access*, 2024.

[6] E. Kfoury, J. Crichigno, and E. Bou-Harb, "An exhaustive survey on P4 programmable data plane switches: Taxonomy, applications, challenges, and future trends," *IEEE Access*, 2021.

[7] Network Development Group (NDG), "NETLAB+." [Online]. Available: https://tinyurl.com/2yn88e85, Accessed on 06-18-2025.

[8] J. Gomez, E. Kfoury, J. Crichigno, and G. Srivastava, "A survey on network simulators, emulators, and testbeds used for research and education," *Computer Networks*, vol. 237, 2023.

[9] A. AlSabeh, A. Mazloum, E. Kfoury, J. Crichigno, and H. Berry, "Enabling line-rate TLS SNI inspection in P4 programmable data planes," in *IEEE Network Operations and Management Symposium (NOMS)*, 2025.

[10] W. Dally, Y. Turakhia, and S. Han, "Domain-specific hardware accelerators," *Communications of the ACM*, vol. 63, no. 7, 2020.

[11] P. Bosshart, D. Daly, G. Gibb, M. Izzard, N. McKeown, J. Rexford, C. Schlesinger, D. Talayco, A. Vahdat, and G. Varghese, "P4: Programming protocol-independent packet processors," *ACM SIGCOMM Computer Communication Review*, 2014.

[12] A. AlSabeh, J. Khoury, E. Kfoury, J. Crichigno, and E. Bou-Harb, "A survey on security applications of P4 programmable switches and a STRIDE-based vulnerability assessment," *Computer Networks*, 2022.

[13] A. Mazloum, A. AlSabeh, E. Kfoury, and J. Crichigno, "Domain name security inspection at line rate: TLS SNI extraction in the data plane using P4 and DPDK," *IEEE International Conference on Communications (ICC)*, 2025.

[14] A. Mazloum, J. Gomez, E. Kfoury, and J. Crichigno, "Enhancing perfSONAR measurement capabilities using P4 programmable data planes," in *Proceedings of the SC'23 Workshops of the International Conference on High Performance Computing, Network, Storage, and Analysis*, 2023.

[15] E. Kfoury, J. Crichigno, and E. Bou-Harb, "Offloading media traffic to programmable data plane switches," in *IEEE International Conference on Communications (ICC)*, 2020.

[16] J. Lenz, E. MacDonald, R. Harik, and T. Wuest, "Optimizing smart manufacturing systems by extending the smart products paradigm to the beginning of life," *Journal of Manufacturing Systems*, vol. 57, 2020.

[17] F. Kalach, M. Farahani, T. Wuest, and R. Harik, "Real-time defect detection and classification in robotic assembly lines: a machine learning framework," *Robotics and Computer-Integrated Manufacturing*, vol. 95, 2025.

[18] NVIDIA, "NVIDIA BlueField-2 DPU data center infrastructure on a chip." [Online]. Available: https://tinyurl.com/2tpkpb5u, Accessed on 10-03-2025.

[19] Intel, "Intel Tofino." [Online]. Available: https://tinyurl.com/mv273c9s, Accessed on 01-28-2025.

[20] Proxmox, "PCI(e) passthrough." [Online]. Available: https://tinyurl.com/yzbets2f, Accessed on 10-04-2025.

[21] I. Baldin, A. Nikolich, J. Griffioen, I. Monga, K. Wang, T. Lehman, and P. Ruth, "FABRIC: A national-scale programmable experimental network infrastructure," *IEEE Internet Computing*, vol. 23, 2019.

[22] A. AlSabeh, K. Friday, E. Kfoury, J. Crichigno, and E. Bou-Harb, "On DGA Detection and Classification Using P4 Programmable Switches," *Computers & Security*, vol. 145, p. 104007, 2024.

[23] A. Mazloum, E. Kfoury, A. AlSabeh, J. Gomez, and J. Crichigno, "Toward fingerprinting encrypted C2 traffic in the data plane," *IEEE Global Communications Conference (GLOBECOM)*, 2025.

[24] NVIDIA, "DPU power efficiency." [Online]. Available: https://tinyurl.com/36raaxec, Accessed on 10-03-2025.