Volume 1 (2026), Issue 2
pp. 31–54

TSARKA SCIENCE

eISSN: 3134-6057
pISSN: 3134-6049

DOI: 10.66571/tsarka-3134-6057-06

ARCHITECTURE OF DATA PLANE DEVELOPMENT KIT (DPDK) FOR ACCELERATING PACKET PROCESSING AND TRAFFIC PROTECTION AT 100 GBPS AND BEYOND

A. Kabdylkak[1]*, M. Zhumabek[1]
[1]TSARKA GROUP, Astana, Kazakhstan
*Corresponding author: kabdylkakarnur@gmail.com

Abstract

This article presents a systematized analytical framework for evaluating Data Plane Development Kit (DPDK) -based architectures for the acceleration and protection of network traffic at 100 Gbps and beyond, with explicit consideration of Critical Information and Communication Infrastructure (CICI) protection requirements established by the Law of the Republic of Kazakhstan No. 418-V “On Informatization”. The methodology combines structured literature review of peer-reviewed publications from 2021–2026, quantitative comparison of architectural alternatives based on published performance data, and mapping of technical capabilities to regulatory requirements. Four evaluation criteria are defined: per-core throughput, per-packet latency, cryptographic overhead, and scalability. The analysis demonstrates that DPDK closes a fourfold per-core efficiency gap relative to the Linux kernel networking stack on identical hardware at 100 Gigabit Ethernet (GbE) rates, that only Network Interface Card (NIC) inline cryptographic offload satisfies the throughput-degradation threshold under mandatory encryption, and that tiered Deep Packet Inspection (DPI) architectures combining hardware pre-classification, Hyperscan-based pattern matching, and Field-Programmable Gate Array (FPGA) acceleration are required to meet incident response monitoring obligations at 100 Gbps. A conceptual FPGA-based switch-encryptor architecture integrated with DPDK is proposed, synthesizing recent SmartNIC research (FpgaNIC, SuperNIC, reconfigurable pipelines on Xilinx Alveo U200) into a deployment-oriented design suitable for CICI inter-facility links. Three integrated findings establish direct correspondence between architectural choices and Articles 7-1, 51, and 55 of Law No. 418-V, providing a compliance-traceable foundation for high-speed network protection in the Republic of Kazakhstan.

TSARKA SCIENCE

Volume 1 (2026), Issue 2
pp. 31–54

eISSN: 3134-6057
pISSN: 3134-6049

Keywords: *DPDK, high-performance packet processing, 100G Ethernet, kernel bypass, traffic protection, FPGA-based networking, inline cryptographic offload, Critical Information and Communication Infrastructure, deep packet inspection.*

1. Introduction

The exponential growth of network traffic driven by 5G/6G deployments, cloud computing, artificial intelligence workloads, and Internet of Things (IoT) proliferation has created unprecedented demands for packet processing performance in modern network infrastructure. Data center interconnects, telecommunications backhaul links, and Network Function Virtualization (NFV) platforms routinely operate at 100 Gbps and increasingly at 200–400 Gbps, requiring processing of over 148 million packets per second (Mpps) at minimum-sized 64-byte frames on a single 100 Gigabit Ethernet (GbE) interface [1]. At these rates, the time budget for processing a single packet is approximately 6.7 nanoseconds, which corresponds to roughly 20 Central Processing Unit (CPU) clock cycles on a 3 GHz processor, leaving virtually no margin for the overhead introduced by traditional kernel-based networking stacks [2].

These performance constraints are equally critical for operators of Critical Information and Communication Infrastructure (CICI) in the Republic of Kazakhstan. According to Article 1, paragraph 24 of the Law of the Republic of Kazakhstan No. 418-V "On Informatization" dated 24 November 2015 (as amended through January 2026), CICI objects are defined as information and communication infrastructure objects whose disruption may lead to social or technogenic emergencies, or significant negative consequences for national defense, security, the economy, or the life-sustaining activities of the population [3]. CICI facilities span energy supply, transportation, water and gas distribution, financial systems, and public healthcare networks. Article 51 of the same Law mandates that CICI objects undergo mandatory information security testing, and Chapter 7-1 establishes requirements for incident response and information security event monitoring. These regulatory obligations translate into concrete technical requirements: high-throughput data transmission, inline cryptographic protection, and real-time intrusion detection — all of which must be delivered without compromising the operational continuity of critical services.

The legal framework distinguishes between two related but distinct categories of regulated objects, and this distinction has direct architectural implications. Article 1 of Law No. 418-V defines informatization objects (Article 1, paragraph 4) as the broader category encompassing electronic information resources, software, internet resources, and information and communication infrastructure. Critical Information and Communication Infrastructure objects (CICI, Article 1, paragraph 24) constitute a more narrowly scoped subset to which significantly stricter security requirements apply: mandatory testing under Article 51, alignment with state information system standards under Chapter 6, integration with the National Coordination Center for In-

Volume 1 (2026), Issue 2
pp. 31–54

TSARKA SCIENCE

eISSN: 3134-6057
pISSN: 3134-6049

formation Security event monitoring system, and incident response obligations under Chapter 7-1 [3]. The architectural recommendations developed in this article are calibrated to the CICI subset specifically; their applicability to general informatization objects is qualitatively similar but the throughput, latency, and verification thresholds are typically less stringent. This distinction matters when interpreting the regulatory mapping in Section 3.7: the three integrated findings apply unconditionally to CICI deployments, while informatization objects of lower criticality may admit relaxed compliance approaches that fall outside the scope of the present analysis.

Recent empirical evaluations confirm that the conventional Linux networking stack cannot meet these requirements. Du and Nikolaev (2025) demonstrated on AMD EPYC 9005 servers equipped with 100 Gbps Intel E810 Network Interface Cards (NICs) that a single Linux process achieves less than 25 Gbps throughput, while DPDK-based applications reach near line rate using a single CPU core — a fourfold efficiency gap with direct cost and performance implications for high-speed deployments [4]. Even with kernel-side optimizations such as New API (NAPI) polling, Generic Receive Offload (GRO), and multi-queue support, the kernel networking stack typically saturates at 10–40 Gbps depending on packet size and processing complexity [5].

The Data Plane Development Kit (DPDK), originally developed by Intel and now maintained as a Linux Foundation project, addresses these limitations by providing a complete user-space framework for high-performance packet processing. DPDK bypasses the kernel networking stack entirely, allowing applications to interact directly with NIC hardware through Poll Mode Drivers (PMDs). By eliminating interrupt overhead, reducing memory copies, and leveraging CPU optimizations such as Single Instruction Multiple Data (SIMD) extensions, cache-line alignment, and Non-Uniform Memory Access (NUMA) -aware allocation, DPDK enables single-core packet forwarding rates that approach the theoretical maximum throughput of 100 GbE interfaces [6].

However, high-speed packet processing alone is insufficient for CICI deployments. The integration of cryptographic protection (Internet Protocol Security — IPsec, Media Access Control Security — MACsec) and Deep Packet Inspection (DPI) for intrusion detection introduces additional computational load that must be absorbed without sacrificing throughput. The challenge lies in maintaining wire-speed performance while applying these security functions, a problem that DPDK addresses through its rte_security and cryptodev subsystems and through integration with high-performance pattern matching libraries such as Hyperscan [7]. Field-Programmable Gate Array (FPGA) -based SmartNICs offer a complementary acceleration path, with recent designs (e.g., FpgaNIC, SuperNIC, and reconfigurable pipelines on Xilinx Alveo U200) achieving 100 Gbps line-rate processing for various network functions [8, 9].

1.1 Research Problem and Knowledge Gap

TSARKA SCIENCE
Volume 1 (2026), Issue 2
pp. 31–54
eISSN: 3134-6057
pISSN: 3134-6049

Despite extensive vendor documentation and individual technology evaluations, the existing literature lacks a systematized analytical framework that simultaneously addresses three dimensions critical for CICI applications: (i) per-core throughput at 100 Gbps and beyond, (ii) cryptographic protection overhead, and (iii) regulatory compliance requirements specific to the Republic of Kazakhstan. Most published studies treat these dimensions in isolation: performance benchmarks rarely incorporate cryptographic load, security-focused evaluations typically operate at sub-100 Gbps rates, and Kazakhstani regulatory compliance has not been mapped to concrete technical architectures in peer-reviewed literature.

1.2 Aim and Objectives

The aim of this study is to develop a systematized analytical framework for evaluating DPDK-based architectures for the acceleration and protection of network traffic at 100 Gbps and beyond, with explicit consideration of CICI protection requirements in the Republic of Kazakhstan.

To achieve this aim, the following objectives are formulated:

1. To analyze the architectural components and acceleration mechanisms of DPDK that enable wire-speed packet processing at 100 Gbps, with quantitative comparison against alternative kernel-bypass and kernel-resident technologies.

2. To define a set of evaluation criteria (per-core throughput, latency, cryptographic overhead, scalability) and apply them to compare DPDK packet processing models (run-to-completion, pipeline, event-driven, graph-based).

3. To assess the performance impact of integrating cryptographic protection (IPsec, MACsec) and Deep Packet Inspection within the DPDK data plane, distinguishing between inline and look-aside acceleration modes.

4. To evaluate the prospects of FPGA-based acceleration in combination with DPDK for the design of switch-encryptor devices applicable to CICI inter-facility links.

5. To map the resulting architectural recommendations to the regulatory requirements established by the Law of the Republic of Kazakhstan “On Informatization”.

1.3 Scientific Novelty

The scientific novelty of this work consists in the following:

1. Systematization of DPDK acceleration mechanisms in the context of CICI protection requirements — for the first time in peer-reviewed literature, the technical capabilities of DPDK at 100 Gbps are mapped to the regulatory framework of the Republic of Kazakhstan (Law No. 418-V), establishing a direct correspondence between architectural choices and compliance obligations.

2. A comparative methodology with quantitative evaluation criteria — the work proposes a four-dimensional evaluation framework (throughput, latency, cryptographic overhead, scalability) and applies it to DPDK processing models using

empirical data drawn from peer-reviewed publications (2021–2026), vendor performance reports, and reference benchmarks, rather than relying on qualitative comparison alone.

3. A conceptual architecture of an FPGA-based switch-encryptor integrated with DPDK — based on synthesis of recent FPGA SmartNIC research (FpgaNIC, USENIX ATC 2022; SuperNIC, HPCA 2024; reconfigurable pipelines on Xilinx Alveo U200), the work proposes a hybrid architecture suitable for wire-speed protection of inter-facility CICI links, with explicit identification of the verification challenges that must be addressed for production deployment.

1.4 Article Structure

The remainder of this article is organized as follows. Section 2 (Materials and Methods) describes the methodology, evaluation criteria, and data sources used in the analysis. Section 3 (Results and Discussion) presents the analytical results across seven subsections covering DPDK architecture, comparative performance of kernel-bypass technologies, processing models, cryptographic protection, DPI capabilities, FPGA integration prospects, and integrated discussion. Section 4 (Conclusion) summarizes findings and identifies directions for further research.

2. Materials and Methods

2.1 Research Design

This study employs a systematic comparative analysis methodology combining (i) structured literature review of peer-reviewed publications and authoritative technical documentation from 2021–2026, (ii) quantitative comparison of architectural alternatives based on published performance data, and (iii) mapping of technical capabilities to the regulatory requirements of the Republic of Kazakhstan. The research design follows the principles of comparative architectural evaluation established in systems engineering literature [10] and is adapted to the specific domain of high-speed packet processing.

The analysis does not include original hardware experiments. Instead, it relies on a curated dataset of published performance measurements, ensuring reproducibility and traceability of every quantitative claim. This approach is consistent with review-style research papers where the contribution lies in synthesis, classification, and identification of architectural trade-offs rather than in the generation of new measurement data.

2.2 Source Selection Criteria

The literature corpus was assembled through a structured search across four categories of sources, with explicit inclusion and exclusion criteria.

Inclusion criteria:

1. Peer-reviewed publications in IEEE, ACM, USENIX, MDPI, and Springer venues,

TSARKA SCIENCE
Volume 1 (2026), Issue 2
pp. 31–54
eISSN: 3134-6057
pISSN: 3134-6049

published between 2021 and 2026, addressing DPDK architecture, performance evaluation, kernel-bypass packet processing, FPGA-based SmartNICs, or cryptographic acceleration at line rates of 25 Gbps or higher.

2. Official technical documentation from the DPDK Project (releases 21.11 LTS through 25.03), Intel, NVIDIA, AMD/Xilinx, and Chelsio Communications, with publication or revision dates within the analysis window.

3. Standards and Request for Comments (RFC) documents issued by the Institute of Electrical and Electronics Engineers (IEEE) and the Internet Engineering Task Force (IETF) related to 100/200/400 GbE, IPsec, MACsec, and Transport Layer Security (TLS).

4. Regulatory acts of the Republic of Kazakhstan governing information security and CICI protection, accessed through the official "Adilet" legal information system.

Exclusion criteria:

1. Non-peer-reviewed blog posts, vendor marketing materials lacking quantitative substantiation, and presentations without accompanying published proceedings.

2. Publications older than 2021, except for foundational works that establish baseline definitions — such works are cited only when their content has not been superseded by more recent peer-reviewed studies.

3. Studies operating exclusively at line rates below 25 Gbps, where the architectural constraints differ substantially from those at 100 Gbps and above.

The final corpus consists of 28 sources, of which 19 are peer-reviewed publications from 2021–2026, 6 are official technical documents from 2024–2025 vendor releases, 2 are Kazakhstani regulatory acts, and 1 is a foundational reference cited for definitional purposes.

2.3 Evaluation Criteria and Metrics

To enable consistent comparison across architectural alternatives, four quantitative evaluation criteria were defined. The selection of criteria reflects both the technical priorities of high-speed packet processing and the operational requirements of CICI deployments.

Volume 1 (2026), Issue 2
pp. 31–54

TSARKA SCIENCE

eISSN: 3134-6057
pISSN: 3134-6049

Table 1. Evaluation criteria and metrics applied in the comparative analysis

Criterion	Metric	Unit	Threshold for 100 GbE Compliance
1	2	3	4
C1. Per-core throughput	Sustained packet processing rate per CPU core under reference workload	Mpps	≥ 30 Mpps for L2 forwarding; ≥ 15 Mpps for L3 forwarding with 64-byte packets
C2. Per-packet latency	Mean and 99th percentile end-to-end latency from ingress to egress	ns / µs	≤ 10 µs mean; ≤ 50 µs at p99 for non-cryptographic forwarding
C3. Cryptographic overhead	Throughput degradation when AES-256-GCM encryption is enabled	%	≤ 30% for inline mode; ≤ 60% for look-aside mode
C4. Scalability	Aggregate throughput as a function of allocated CPU cores	Mpps per added core	Near-linear scaling up to NUMA boundary; sub-linear acceptable beyond

The thresholds in Table 1 are derived from two sources: (i) the theoretical maximum of 148.8 Mpps for 64-byte packets at 100 GbE line rate, scaled by typical efficiency factors observed in published benchmarks [11], and (ii) latency budgets reported as acceptable for industrial control system traffic in CICI environments [12].

2.4 Comparison Workflow

The comparative analysis follows a four-step workflow applied uniformly across all architectural alternatives:

1. Architectural decomposition — each alternative (DPDK processing model, cryptographic acceleration mode, FPGA integration approach) is decomposed into its constituent components: data path, control path, memory subsystem, and external interfaces.

2. Quantitative characterization — published measurements are mapped to the four evaluation criteria. Where direct measurements are unavailable, derivations from cited data are explicitly indicated.

3. Trade-off identification — for each alternative, the analysis identifies the dominant trade-offs between throughput, latency, flexibility, and operational complexity.

4. Regulatory mapping — each alternative is evaluated against the requirements of Articles 51 (mandatory information security testing) and Chapter 7-1 (incident re-

TSARKA SCIENCE
Volume 1 (2026), Issue 2
pp. 31–54
eISSN: 3134-6057
pISSN: 3134-6049

sponse and event monitoring) of the Law No. 418-V of the Republic of Kazakhstan, identifying where the architectural choice supports or constrains compliance.

2.5 Reference Hardware Configurations

To ensure that quantitative comparisons remain meaningful, performance data are normalized to a small set of reference hardware configurations frequently encountered in the literature. Three reference configurations are used (Table 2).

Table 2. Reference hardware configurations used for performance normalization

Configuration	CPU	NIC	DPDK Version	Source
1	2	3	4	5
Ref-1: Intel 100G	Intel Xeon Scalable (Ice Lake / Sapphire Rapids), 3.0+ GHz	Intel E810-CQDA2 (100 GbE)	23.11 LTS	[13]
Ref-2: NVIDIA 100G	AMD EPYC 9005 series or Intel Xeon Platinum	NVIDIA ConnectX-6 Dx / ConnectX-7 (100 GbE)	23.11 LTS	[4, 14]
Ref-3: FPGA SmartNIC	x86 host (varies)	Xilinx Alveo U200 / U280 (100 GbE)	Custom PMD	[9, 15]

When performance figures from publications using different hardware are compared, normalization is performed by extracting per-core throughput rather than aggregate throughput, since per-core efficiency is largely independent of NIC vendor at 100 GbE rates [16].

2.6 Limitations of the Methodology

The methodology has three explicit limitations that should be considered when interpreting the results:

1. Reliance on published measurements introduces potential bias toward configurations and workloads that vendors and researchers choose to publish; corner-case scenarios may be underrepresented.

2. Absence of original experiments means that interaction effects between cryptographic load, DPI processing, and high-throughput forwarding under specific CICI traffic profiles cannot be directly observed.

3. Regulatory mapping is performed against the current text of Law No. 418-V (as amended through January 2026); subsequent legislative amendments may alter the

TSARKA SCIENCE

eISSN: 3134-6057
pISSN: 3134-6049

compliance landscape.

These limitations define directions for further research, particularly the construction of a dedicated CICI traffic profile dataset and execution of original benchmarks on Kazakhstani-deployed equipment, which are beyond the scope of the present article.

3. Results and Discussion

3.1 DPDK Architectural Decomposition

The Data Plane Development Kit is organized as a layered framework in which each layer addresses a specific source of overhead in traditional kernel-based networking. The decomposition presented in Table 3 identifies the architectural mechanism, the corresponding overhead it eliminates, and the resulting performance contribution at 100 GbE rates.

Table 3. DPDK architectural mechanisms and their contribution to 100 Gbps performance

Layer	Mechanism	Eliminated Overhead	Quantitative Impact
1	2	3	4
EAL	Hugepage allocation (2 MB / 1 GB)	TLB misses on 4 KB pages	TLB miss reduction up to 41%; reported in TurboMem (2026) [17]
EAL	NUMA-aware memory binding	Cross-socket memory access (~70 ns penalty)	Eliminates ~40–70 ns latency per cross-NUMA reference [18]
EAL	CPU core isolation (lcore binding)	Context switching, scheduler interference	Removes ~1–5 µs scheduler-induced jitter per packet [19]
PMD	Polling instead of interrupts	Interrupt service routine overhead (~1–3 µs per IRQ)	Enables deterministic per-packet latency in nanosecond range [6]
PMD	Burst-oriented Rx/Tx (32–64 packets)	Per-packet function call overhead	Amortizes call overhead; ~3.5× single-core gain with AVX-512 vectorization [13]
Core libs	rte_mbuf with cache-line aligned metadata	Cache line bouncing across cores	Two-cache-line metadata (128 B) keeps hot data resident [20]

TSARKA SCIENCE
Volume 1 (2026), Issue 2
pp. 31–54
eISSN: 3134-6057
pISSN: 3134-6049

1	2	3	4
Core libs	rte_mempool with per-core local cache	Lock contention on shared allocators	Lock-free allocation; sustained 100+ Mpps in mempool stress tests [17]
Core libs	rte_ring (lockless CAS FIFO)	Mutex acquisition for inter-core queues	Sub-100 ns enqueue/dequeue latency [6]

The Environment Abstraction Layer (EAL) provides the foundation by managing hugepages, NUMA topology, and CPU affinity. As demonstrated by Yang (2026) in the TurboMem study, hugepage utilization combined with lock-free memory pool design and transparent huge page auto-merging reduces Translation Lookaside Buffer (TLB) misses by up to 41% and increases mempool throughput by up to 28% relative to standard 4 KB pages [17]. The TurboMem design is particularly relevant for sustained 100 Gbps operation because it addresses three concurrent sources of memory subsystem overhead: lock contention on shared mempool rings, cache-coherence ping-pong between cores, and TLB pressure from thousands of small pages. For a 100 Gbps application processing 148 Mpps, a single TLB miss costing approximately 30–50 ns would consume 50–75% of the 6.7 ns per-packet budget, making efficient hugepage management a structural prerequisite rather than a discretionary optimization. The TurboMem approach extends standard DPDK hugepage usage by automating the merging of transparent huge pages, removing the manual configuration burden that has historically been a deployment friction point for DPDK-based CICI installations.

The Poll Mode Driver replaces interrupt-driven I/O with a continuous polling loop. While this dedicates CPU cores to packet reception (cores running at 100% utilization during polling), it eliminates the 1–3 µs cost of interrupt service routines and the associated cache disturbance. Modern PMDs such as mlx5 (NVIDIA ConnectX), ice (Intel E810), and cxgbe (Chelsio T6) implement burst-oriented Application Programming Interfaces (APIs) — rte_eth_rx_burst() and rte_eth_tx_burst() — that process up to 32 or 64 packets per call, amortizing function-call overhead across the batch.

Beyond the standard DPDK optimizations, recent compiler-level techniques such as PacketMill (Farshin et al., 2021) demonstrate that further per-core efficiency gains are achievable through metadata management optimization and source-to-source code transformation [2]. PacketMill modifies the way packet metadata flow through the processing pipeline by integrating custom buffer layouts directly with the driver — an approach termed X-Change — and applies compiler-driven specialization to eliminate redundant operations introduced by generic packet processing frameworks. The reported result is per-core throughput sufficient to sustain 100 Gbps on

Volume 1 (2026), Issue 2
pp. 31–54

TSARKA SCIENCE

eISSN: 3134-6057
pISSN: 3134-6049

commodity Intel Xeon hardware at small packet sizes, addressing the regime where new packet data arrive at the NIC roughly an order of magnitude faster than main memory access latency. For CICI deployments operating on hardware that may not always include the latest-generation NICs with full inline offload, such software-level techniques represent a meaningful path to maintaining performance margins under combined cryptographic and inspection workloads.

3.2 Comparative Performance of Kernel-Bypass Technologies (Criterion C1)

Table 4 presents per-core throughput data for representative kernel-bypass and kernel-resident technologies, normalized to the reference hardware configurations defined in Section 2.5. All measurements correspond to Layer-3 forwarding workloads with 64-byte packets unless otherwise indicated.

Table 4. Per-core throughput comparison across packet processing technologies (64-byte packets, L3 forwarding)

Technology	Per-core Throughput	Hardware Reference	Source
Linux kernel stack	< 25 Gbps (≈ 4–6 Mpps effective)	Ref-2 (AMD EPYC 9005 + Intel E810)	Du & Nikolaev (2025) [4]
AF_XDP	~10–20 Mpps	Comparable to Ref-2	Shah & Naik (2023) [5]
XDP/eBPF	up to 24 Mpps	Reported on Intel Xeon	[5]
DPDK (mlx5 PMD)	≥ 30 Mpps; near 100 Gbps line rate	Ref-2	NVIDIA Performance Report DPDK 23.11 [14]
DPDK (ice PMD, AVX-512)	~40–45 Mpps	Ref-1	Intel Ethernet 800 Series Guide [13]
DPDK (cxgbe PMD)	up to 75 Mpps (Tx/Rx aggregate)	Chelsio T6 100 GbE	Chelsio benchmark report [21]
FPGA SmartNIC (Alveo U200)	100 Gbps wire-speed	Ref-3	Song et al. (2024) [9]; Sage et al. (2024) [15]

Two observations from Table 4 warrant emphasis. First, the gap between the Linux kernel stack and DPDK on identical hardware is approximately fourfold per core, as quantified by Du and Nikolaev (2025) [4]; this gap scales with the number of cores

TSARKA SCIENCE
Volume 1 (2026), Issue 2
pp. 31–54
eISSN: 3134-6057
pISSN: 3134-6049

and translates directly into deployment cost differences for CICI operators. Second, the spread among DPDK PMDs themselves is significant — the cxgbe PMD on Chelsio T6 reaches up to 75 Mpps for small packets [21], outperforming general-purpose Intel and NVIDIA PMDs. This suggests that PMD selection is not a secondary implementation detail but a primary architectural decision that should be made with reference to the specific traffic profile of the target deployment.

A complementary research direction worth noting in the broader academic context is the Joyride architecture proposed by Du and Nikolaev (2025), which approaches the same throughput problem from the opposite end: rather than bypassing the kernel entirely, Joyride seeks to integrate kernel-bypass principles into a microkernel-style redesign of the Linux network stack while preserving compatibility with existing socket-based applications [4]. The Joyride evaluation that produced the fourfold gap measurement cited above is itself part of an argument that DPDK and traditional Linux networking represent two extremes of a design space rather than fundamentally incompatible approaches. For CICI deployments, the practical implication is that DPDK-based architectures should be viewed not as the only conceivable solution but as the currently most performant option within an evolving research landscape. Future work in this space — particularly the maturation of microkernel-style network stacks and the continued development of AF_XDP — may eventually narrow the gap, though no current alternative meets the 100 Gbps thresholds defined in Section 2.3 for typical CICI workloads.

The 30 Mpps threshold defined in Section 2.3 (Criterion C1) is achievable only with DPDK or FPGA-based solutions on the reference configurations. Linux, AF_XDP, and XDP/eBPF fall below the threshold for L3 forwarding at 100 GbE, confirming that DPDK is not merely a performance optimization but a necessary architectural choice for compliance with the throughput requirements of CICI inter-facility links.

3.3 Comparative Analysis of DPDK Processing Models (Criteria C1, C2, C4)

DPDK supports four distinct packet processing models, each with characteristic trade-offs between per-core throughput, end-to-end latency, and scalability. Table 5 summarizes the comparative assessment using the criteria defined in Section 2.3.

Table 5. Comparative assessment of DPDK packet processing models

Model	Throughput (C1)	Latency (C2)	Scalability (C4)	Optimal Use
1	2	3	4	5
Run-to-Completion (RTC)	Highest per-core (no inter-core transfer)	Lowest (~1–3 μs)	Linear with cores + NICs	Simple L2/L3 forwarding

Volume 1 (2026), Issue 2
pp. 31–54

TSARKA SCIENCE

eISSN: 3134-6057
pISSN: 3134-6049

1	2	3	4	5
Pipeline	Medium (ring overhead 50–100 ns/stage)	Medium (3–10 μs)	Per-stage parallelism	Multi-stage feature-rich processing
Event-Driven (Eventdev)	Variable (scheduler dependent)	Variable; HW-assisted < SW	Dynamic load balancing	High per-packet variance
Graph Framework	Medium-high (optimized batching)	Medium (3–8 μs)	Flexible node parallelism	Complex feature pipelines

The Run-to-Completion (RTC) model achieves the highest per-core efficiency by eliminating inter-core packet transfer entirely. Each core executes the full processing pipeline — reception, classification, modification, transmission — without enqueuing packets to other cores. This eliminates the 50–100 ns cost of ring-buffer enqueue/dequeue operations and avoids the cache-line bouncing that occurs when packets traverse core boundaries [6]. RTC is the model of choice for CICI applications dominated by uniform L2/L3 forwarding, such as inter-facility VPN gateways.

The Pipeline model distributes processing stages across dedicated cores, with rte_ring lockless First-In-First-Out (FIFO) queues passing packets between stages. The model becomes advantageous when stages have substantially different computational costs, allowing bottleneck stages to be parallelized independently. The price is per-packet latency increase, which empirical measurements place at 50–100 ns per ring-buffer hop [22]. For pipelines with 3–4 stages, accumulated latency reaches 200–400 ns, still well below the 10 μs threshold of Criterion C2.

The Eventdev framework provides hardware- or software-based event scheduling with atomic, ordered, and parallel scheduling types. Hardware-backed event devices (available on certain System-on-Chip platforms and SmartNICs) perform scheduling without consuming host CPU cycles, but their availability is limited and their integration introduces vendor-specific dependencies that complicate CICI deployments where vendor diversification is often a regulatory expectation.

The Graph framework, introduced in DPDK 20.05 and matured through the 23.11 LTS and 25.03 releases, represents the most modern approach. It models packet processing as a Directed Acyclic Graph (DAG) of nodes connected by edges, with the framework handling automatic batching, cache-aware node dispatch, and runtime reconfiguration [23]. The mixi-PGW reference implementation (Yang, 2025) demonstrates a Packet Gateway pipeline built on the Graph framework that sustains high

TSARKA SCIENCE
Volume 1 (2026), Issue 2
pp. 31–54
eISSN: 3134-6057
pISSN: 3134-6049

per-core throughput while supporting 0.5–1.0 million users per Distributor core on a single-socket Intel Xeon system [18].

A complementary development is the integration of the Programming Protocol-independent Packet Processor (P4) language with DPDK through the p4c-dpdk compiler. P4-DPDK enables developers to describe packet processing pipelines in P4 — a high-level domain-specific language designed for protocol-independent packet processing — and automatically generate DPDK-specific C code targeting the host CPU [29]. This approach lowers the development barrier for complex monitoring and filtering functions: features that previously required deep expertise in DPDK internals can be expressed in declarative P4 syntax, with the compiler handling the translation to optimized DPDK API calls. For CICI operators, P4-DPDK is particularly relevant because it enables the same P4 description to target both software (host CPU + DPDK) and hardware (FPGA SmartNIC) execution environments, providing a unified development workflow across the tiered architecture proposed in Section 3.5.

3.4 Cryptographic Protection: Inline versus Look-Aside (Criterion C3)

The integration of cryptographic operations into the DPDK data plane is supported by two complementary subsystems: cryptodev (generic crypto accelerator API) and rte_security (protocol-aware security processing for IPsec and MACsec). The implementation choice between inline and look-aside processing has substantial performance implications, quantified in Table 6.

Table 6. Performance impact of AES-256-GCM cryptographic protection on 100 GbE throughput

Acceleration Mode	Description	Throughput Degradation	Per-core Cost	Latency Impact
No encryption (baseline)	L3 forwarding, no crypto	0%	Reference	Reference
Software AES-NI (look-aside)	CPU-only, AES-NI instructions	~60–70%	3–5 cores per 100 Gbps	+1–3 µs
QAT look-aside acceleration	Intel QuickAssist Technology offload	~25–40%	1–2 cores for control	+5–15 µs
NIC inline crypto offload	NVIDIA ConnectX-6 Dx, Intel E810	< 10%	< 0.5 core (control plane only)	+0.5–1 µs

The data in Table 6 are synthesized from NVIDIA technical briefs on inline IPsec

Volume 1 (2026), Issue 2
pp. 31–54

TSARKA SCIENCE

eISSN: 3134-6057
pISSN: 3134-6049

offload [25], Intel QAT performance documentation [26], and peer-reviewed evaluations of AES-GCM throughput on modern Intel and AMD processors [27]. The threshold defined in Criterion C3 (≤ 30% degradation for inline; ≤ 60% for look-aside) is satisfied by NIC inline offload and partially by QAT look-aside, while pure software AES-NI fails the threshold.

For CICI deployments subject to mandatory inline cryptographic protection of inter-facility traffic, the architectural implication is unambiguous: software-only encryption cannot meet 100 GbE throughput targets without dedicating 30–50% of available CPU cores exclusively to cryptographic operations. NIC inline offload is the recommended baseline, with QAT look-aside reserved for non-standard algorithms or protocols that the NIC cannot handle natively.

A practical hybrid architecture distributes the cryptographic load: the NIC handles standard IPsec or MACsec for the majority of traffic, while DPDK software paths process exception flows requiring custom protocols. This approach maintains compliance with C3 while preserving the algorithmic flexibility required for evolving threat landscapes.

3.5 Deep Packet Inspection on DPDK

DPI extends beyond header-based filtering by analyzing packet payloads to identify applications, detect malicious patterns, and enforce compliance policies. For CICI operators, DPI directly supports the requirements of Chapter 7-1 of Law No. 418-V regarding information security event monitoring and incident response [3]. At 100 GbE, software DPI faces fundamental computational constraints that vary substantially with the chosen pattern matching engine.

Table 7. Throughput characteristics of DPI engines integrated with DPDK

Engine	Architecture	Per-core Throughput	Notes
1	2	3	4
Suricata-DPDK 7.0+	Stateful IDS/IPS with detection engine	1–10 Gbps with full ruleset; > 60% drop reported at 100 Gbps with small packets	DPDK capture mode native since v7.0 [28]
Hyperscan (regex)	Hybrid NFA/DFA, SIMD-accelerated	10–20 Gbps per core with ~10,000 patterns	Streaming mode preserves cross-packet state [7]

TSARKA SCIENCE

Volume 1 (2026), Issue 2
pp. 31–54

eISSN: 3134-6057
pISSN: 3134-6049

1	2	3	4
nDPI (protocol classification)	*Application protocol identification*	*10–30 Gbps per core (header-based)*	*Lightweight, not signature-based*
FPGA-accelerated DPI (NFA)	*Approximate Nondeterministic FAs in hardware logic*	*Wire-speed beyond 100 Gbps*	*Češka et al. (2019/2024) [30]*

Empirical evaluation by Almaraz et al. (GLOBECOM 2024) provides a clear illustration of the limits of software DPI: at 100 Gbps with small packets and four CPU cores, Suricata-DPDK exhibits packet drop rates exceeding 60% even with relatively large 1500-byte packets [11]. A custom Programming Protocol-independent Packet Processor (P4) -based DPDK pipeline operating on the same hardware achieved substantially lower drop rates, confirming that monolithic Intrusion Detection System (IDS) / Intrusion Prevention System (IPS) engines cannot, in their current form, deliver wire-speed inspection at 100 Gbps without architectural decomposition.

The practical implication for CICI deployments is a tiered DPI architecture: hardware pre-classification on the NIC (using rte_flow rules) directs only flagged or unclassified flows to software inspection, reducing the volume requiring deep analysis by an order of magnitude. The remaining flows are inspected by Hyperscan-based engines on dedicated DPDK cores, with stateful analysis (such as Suricata's flow tracking) reserved for a further-reduced subset.

The growing prevalence of encrypted traffic introduces an additional constraint on conventional DPI. With over 90% of contemporary Internet traffic protected by TLS 1.3 or QUIC, payload-based signature matching is largely ineffective on the underlying content [31]. Encrypted Traffic Analysis (ETA) addresses this limitation by extracting features that remain visible despite encryption: TLS handshake parameters, certificate metadata, packet-size and inter-arrival-time distributions, and flow-level statistical patterns. JA4+ fingerprints, introduced by Althouse et al. in 2023 to address the deficiencies of the earlier JA3 method under TLS extension randomization in modern browsers, have been demonstrated to support malware detection in encrypted traffic without requiring decryption [32]. Empirical evaluation by Holík et al. (CNSM 2024) shows that the JA4 family of fingerprints — including JA4 (client-side), JA4S (server-side), and JA4X (X.509 certificate-based) — achieves higher accuracy than JA3 while remaining stable against extension reordering, making them suitable for line-rate computation within DPDK-based capture pipelines [32]. For CICI environments operating under Chapter 7-1 of Law No. 418-V, which mandates information security event monitoring, JA4+ fingerprinting offers a regulatory-compliant

approach to threat detection that does not require breaking end-to-end encryption — a property that aligns with both technical feasibility at 100 Gbps and the privacy considerations inherent in monitoring critical infrastructure traffic. For deployments where decryption is operationally required, TLS-inspecting proxies remain an option, with the inline crypto offload paths described in Section 3.4 mitigating the performance penalty of decrypt-inspect-reencrypt cycles.

3.6 FPGA-Based Acceleration in Combination with DPDK

The recent body of FPGA SmartNIC research (2022–2026) demonstrates that wire-speed 100 Gbps processing with reconfigurable hardware logic is feasible on commercially available platforms. Four reference designs are particularly relevant for CICI applications.

Table 8. Recent FPGA SmartNIC designs relevant to CICI applications

Design	Year	Platform	Reported Throughput	Key Capability
1	2	3	4	5
FpgaNIC (USENIX ATC)	2022	Custom FPGA + GPU integration	100 Gbps	GPU-virtual-address DMA; on-path data-path accelerator [8]
SuperNIC (HPCA)	2024	FPGA prototype	100 Gbps with 196 ns scheduling overhead	Multi-tenant NT-DAG offload; up to 40% latency reduction [33]
Reconfigurable Pipeline	2024	Xilinx Alveo U200	100 Gbps, low latency	Fully reconfigurable match-action units; runtime rule installation [9]
P4 + HLS on Alveo	2024	AMD FPGA SmartNIC	~95 Gbps	First combined P4 and High-Level Synthesis on AMD FPGA [15]

These designs converge on a common architectural pattern: the FPGA implements the high-throughput data path (forwarding, classification, optionally encryption), while a host CPU running DPDK provides control-plane management, exception handling, and rule installation through a custom PMD. The conceptual architecture of an FPGA-based switch-encryptor for CICI inter-facility links — proposed as a synthesis

TSARKA SCIENCE
Volume 1 (2026), Issue 2
pp. 31–54
eISSN: 3134-6057
pISSN: 3134-6049

of these approaches — would consist of three components.

A particularly important architectural property demonstrated by the reconfigurable pipeline design on Xilinx Alveo U200 (Song et al., 2024) is the ability to modify match-key fields and match-table sizes at runtime without recompiling the Hardware Description Language (HDL) code or reloading the FPGA bitstream [9]. Processing rules and action instructions can be installed dynamically through a configuration module, enabling protocol changes to be propagated within milliseconds rather than the hours typically required for full FPGA synthesis and reprogramming. For CICI deployments, this capability transforms FPGA acceleration from a one-time engineering investment into an operationally adaptive solution: a switch-encryptor that initially supports IPsec ESP can be reconfigured to handle MACsec or a new encapsulation format without taking the device offline for re-flashing, addressing one of the principal operational concerns historically associated with hardware-based security devices.

1. Hardware data path (FPGA): 100 GbE Media Access Control (MAC) and Physical Layer (PHY) blocks; programmable parser; forwarding database with Content-Addressable Memory (CAM) / Ternary CAM (TCAM) lookup; inline AES-256-GCM encryption engine implementing IPsec Encapsulating Security Payload (ESP) or MACsec encapsulation.

2. Control plane (host CPU + DPDK): custom FPGA PMD interfacing via Peripheral Component Interconnect Express (PCIe) Generation 4; key management (Internet Key Exchange version 2 — IKEv2 — for IPsec, MACsec Key Agreement — MKA — for MACsec); telemetry and configuration through rte_metrics and rte_telemetry.

3. Exception path: Direct Memory Access (DMA) -based forwarding of unhandled or suspicious packets to the host for software inspection by Suricata or Hyperscan.

The challenges for production deployment include the development and verification cost of custom FPGA designs, the requirement for formal verification of hardware encryption to meet certification requirements analogous to FIPS 140-3, secure key storage within the FPGA fabric (typically realized through dedicated key storage with anti-tamper protection), and the development of a stable PMD interface that bridges FPGA-specific descriptor formats with the standard DPDK ethdev API.

3.7 Integrated Discussion and Mapping to Regulatory Requirements

The comparative analysis presented in Sections 3.1–3.6 yields three integrated findings that bear directly on the regulatory framework of the Republic of Kazakhstan.

Finding 1: DPDK is the necessary baseline for CICI inter-facility links at 100 Gbps. The empirical fourfold gap between Linux kernel networking and DPDK on identical hardware [4] establishes that kernel-bypass is not optional for compliance with throughput requirements at modern line rates. Under the mandatory information security testing regime of Article 51 of Law No. 418-V, demonstrating sustained throughput

TSARKA SCIENCE

eISSN: 3134-6057
pISSN: 3134-6049

under cryptographic load is a testable criterion that kernel-resident solutions cannot meet at 100 Gbps.

Finding 2: Inline cryptographic offload is the only practical path to compliant 100 Gbps protection. The performance data in Table 6 demonstrate that software-only AES-256-GCM imposes throughput penalties exceeding the 60% threshold defined in Criterion C3, while NIC inline offload remains below 10%. For CICI operators, the architectural recommendation is to specify NIC inline crypto capability as a baseline procurement requirement.

Finding 3: Tiered DPI architectures are required for incident response compliance. Chapter 7-1 of Law No. 418-V mandates information security event monitoring; the data in Table 7 indicate that monolithic software DPI (Suricata-DPDK as configured for typical IDS deployments) cannot sustain 100 Gbps. A tiered architecture combining hardware pre-classification, Hyperscan-based pattern matching on dedicated cores, and FPGA-accelerated DPI for highest-volume segments offers a feasible compliance path.

These findings collectively support the central thesis of this work: that DPDK, augmented by inline cryptographic offload and FPGA-based acceleration, constitutes a viable architectural foundation for 100 Gbps CICI protection in the Republic of Kazakhstan, with each architectural choice traceable to a specific regulatory obligation.

The methodology has limitations consistent with those declared in Section 2.6. Notably, the absence of measurements obtained on equipment deployed in Kazakhstani CICI facilities means that the architectural recommendations are derived from internationally published benchmarks rather than from the specific traffic profiles of domestic critical infrastructure. Construction of such a domestic measurement dataset is identified as a priority direction for further research.

4. Conclusion

This article has presented a systematized analytical framework for evaluating Data Plane Development Kit (DPDK) -based architectures for the acceleration and protection of network traffic at 100 Gbps and beyond, with explicit consideration of Critical Information and Communication Infrastructure (CICI) protection requirements established by the Law of the Republic of Kazakhstan No. 418-V “On Informatization”.

In addressing the five objectives formulated in Section 1.2, the work has produced the following results.

Regarding Objective 1 (architectural decomposition of DPDK), the analysis identified eight distinct mechanisms across the Environment Abstraction Layer, Poll Mode Drivers, and core libraries, each addressing a specific source of overhead in traditional kernel-based networking. Quantitative data from peer-reviewed publications (2021–2026) demonstrate that these mechanisms collectively close a fourfold per-core efficiency gap between Linux kernel networking and DPDK on identical hard-

TSARKA SCIENCE
Volume 1 (2026), Issue 2
pp. 31–54
eISSN: 3134-6057
pISSN: 3134-6049

ware at 100 GbE rates.

Regarding Objective 2 (comparison of processing models), the four DPDK models — Run-to-Completion, Pipeline, Event-Driven, and Graph framework — were evaluated against the four criteria defined in Section 2.3. The Run-to-Completion model emerges as optimal for uniform forwarding workloads characteristic of CICI inter-facility links, while the Graph framework provides the strongest foundation for complex multi-feature pipelines.

Regarding Objective 3 (cryptographic protection assessment), quantitative comparison of inline and look-aside acceleration modes established that only Network Interface Card (NIC) inline cryptographic offload meets the throughput-degradation threshold of Criterion C3 at 100 Gbps, with degradation below 10%. Software-only AES-256-GCM imposes penalties exceeding 60% and is therefore unsuitable for CICI deployments requiring mandatory inline encryption of inter-facility traffic.

Regarding Objective 4 (FPGA integration prospects), synthesis of recent FPGA SmartNIC research — including FpgaNIC (USENIX ATC 2022), SuperNIC (HPCA 2024), and reconfigurable pipelines on Xilinx Alveo U200 (MDPI 2024) — supports the feasibility of an FPGA-based switch-encryptor architecture for CICI applications. The proposed conceptual architecture combines wire-speed hardware data path with DPDK-based control plane, identifying formal verification of cryptographic logic and secure key management as the principal challenges for production deployment.

Regarding Objective 5 (regulatory mapping), the three integrated findings of Section 3.7 establish direct correspondence between architectural choices and the requirements of Chapter 7-1 and Article 51 of Law No. 418-V: DPDK as the necessary baseline for throughput compliance, inline cryptographic offload as the practical path to encrypted-traffic compliance, and tiered Deep Packet Inspection architectures as the feasible approach to incident response monitoring.

4.1 Scientific Contribution

The scientific novelty declared in Section 1.3 is substantiated through three concrete contributions of this work.

First, the article provides — for the first time in peer-reviewed literature — an explicit mapping between DPDK architectural mechanisms and the regulatory framework of the Republic of Kazakhstan, transforming architectural choices from technical preferences into traceable compliance decisions.

Second, the four-criterion evaluation methodology (per-core throughput, latency, cryptographic overhead, scalability) with quantitative thresholds derived from line-rate budgets enables consistent comparison across heterogeneous published benchmarks, replacing qualitative comparison with measurable criteria.

Third, the conceptual FPGA-based switch-encryptor architecture synthesizes

2022–2026 SmartNIC research into a deployment-oriented design with explicit identification of verification and key-management challenges, providing a research direction grounded in current FPGA capabilities rather than in speculative future hardware.

4.2 Limitations and Future Work

The principal limitations of this study, as declared in Section 2.6, are the reliance on published rather than original measurements and the absence of CICI-specific traffic profile datasets from Kazakhstani facilities. These limitations define the priority directions for further research.

1. Construction of a CICI traffic profile dataset based on observations from operating Kazakhstani critical infrastructure, with appropriate anonymization and regulatory compliance, to enable validation of the architectural recommendations against domestic operational conditions.

2. Original benchmarking of DPDK 25.03 with inline cryptographic offload on hardware procurable in the Republic of Kazakhstan, reporting per-core throughput, latency distributions, and encryption overhead under realistic CICI workloads.

3. Prototype implementation of the proposed FPGA-based switch-encryptor on a commercially available Xilinx Alveo U200 or U280 platform, with formal verification of the AES-256-GCM hardware engine against a published reference implementation, addressing the verification challenge identified in Section 3.6.

4. Extension of the methodology to 200 Gbps and 400 Gbps line rates, where the per-packet time budget falls to 3.4 ns and 1.7 ns respectively, requiring re-examination of the trade-offs between SIMD-accelerated software paths and hardware-based processing.

The technical and regulatory landscape of high-speed packet processing in the Republic of Kazakhstan is positioned for substantial development in the coming years, driven by CICI digitalization initiatives, the implementation of post-quantum cryptographic transitions, and the convergence of 5G-Advanced and 6G research programs. The framework proposed in this article is intended as a foundation for evaluating the architectural decisions that this development will require, and as a reference point for subsequent research grounded in domestic measurement and prototyping.

Thanks. The authors thank the reviewers for their constructive feedback and helpful suggestions, which have substantially enhanced the methodological rigor and quantitative grounding of this paper.

Financing. This work was conducted without external funding.

Conflict of interest. The authors declare that there is no conflict of interest.

TSARKA SCIENCE

Volume 1 (2026), Issue 2
pp. 31–54

eISSN: 3134-6057
pISSN: 3134-6049

References

1. IEEE 802.3 Standard. (2022). IEEE Standard for Ethernet, Section 4: Physical Layer and Management Parameters for Operation Above 10 Gb/s. IEEE Std 802.3-2022.

2. Farshin, A., Barbette, T., Roozbeh, A., Maguire, G. Q., & Kostić, D. (2021). PacketMill: Toward Per-Core 100-Gbps Networking. In Proceedings of the 26th ACM International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS '21), 1–17. URL: https://doi.org/10.1145/3445814.3446724

3. Law of the Republic of Kazakhstan No. 418-V dated 24 November 2015 "On Informatization" (with amendments and additions as of 18 January 2026). Adilet Legal Information System. URL: https://adilet.zan.kz/eng/docs/Z1500000418

4. Du, Y., & Nikolaev, R. (2025). Joyride: Rethinking Linux's Network Stack Design for Better Performance, Security, and Reliability. arXiv preprint arXiv:2509.25015. URL: https://arxiv.org/abs/2509.25015

5. Shah, R., & Naik, P. (2023). Kernel-Bypass Techniques for High-Speed Network Packet Processing: A Survey. ACM Computing Surveys, 55(3), 1–38. URL: https://doi.org/10.1145/3551644

6. DPDK Project. (2025). Data Plane Development Kit: Programmer's Guide, Release 25.03. Linux Foundation. URL: https://doc.dpdk.org/guides-25.03/prog_guide/

7. Wang, X., Hong, Y., Chang, H., Park, K., Langdale, G., Hu, J., & Zhu, H. (2019). Hyperscan: A Fast Multi-Pattern Regex Matcher for Modern CPUs. In Proceedings of the 16th USENIX Symposium on Networked Systems Design and Implementation (NSDI '19), 631–648.

8. Wang, Z., He, H., Luo, Z., Zhang, J., Yu, M., Chen, J., & Alonso, G. (2022). FpgaNIC: An FPGA-Based Versatile 100Gb SmartNIC for GPUs. In Proceedings of the 2022 USENIX Annual Technical Conference (ATC '22), 967–986.

9. Song, X., Lu, R., & Guo, Z. (2024). High-Performance Reconfigurable Pipeline Implementation for FPGA-Based SmartNIC. Micromachines, 15(4), 449. URL: https://doi.org/10.3390/mi15040449

10. Bass, L., Clements, P., & Kazman, R. (2021). Software Architecture in Practice (4th ed.). Addison-Wesley Professional.

11. Almaraz, J., Bou-Harb, E., & Crichigno, J. (2024). Scalable Heavy Hitter Detection: A DPDK-Based Approach for High-Speed Network Monitoring. In Proceedings of the IEEE Global Communications Conference (GLOBECOM 2024). URL: https://research.cec.sc.edu/files/documents/globecom_2024_2_1_0.pdf

12. International Electrotechnical Commission. (2022). IEC 62443-3-3: Industrial Communication Networks — Network and System Security — Part 3-3: System Security Requirements and Security Levels. Edition 1.0.

13. Intel Corporation. (2024). Intel Ethernet 800 Series Network Adapter — Perfor-

mance Evolution on DPDK. Intel Network Builders Technology Guide. URL: https://builders.intel.com/docs/networkbuilders/

14. NVIDIA Corporation. (2024). NVIDIA NICs Performance Report with DPDK 23.11. URL: https://fast.dpdk.org/doc/perf/DPDK_23_11_NVIDIA_NIC_performance_report.pdf

15. Sage, T., Khan, M. R., Patel, P., Leeser, M., & Skadron, K. (2024). Extracting TCP/IP Headers at High Speed for the Anonymized Network Traffic Graph Challenge. In Proceedings of the IEEE High Performance Extreme Computing Conference (HPEC 2024). URL: https://arxiv.org/abs/2409.07374

16. Pirelli, S., & Candea, G. (2022). A Simpler and Faster NIC Driver Model for Network Functions. In Proceedings of the 16th USENIX Symposium on Operating Systems Design and Implementation (OSDI '22), 225–241.

17. Yang, J. (2026). TurboMem: High-Performance Lock-Free Memory Pool with Transparent Huge Page Auto-Merging for DPDK. arXiv preprint arXiv:2603.18690. URL: https://arxiv.org/abs/2603.18690

18. Yang, J. (2026). Implementation and Performance Optimization of a DPDK-Based Packet Gateway: The mixi-PGW Architecture. Preprints.org, 2026010019. URL: https://www.preprints.org/manuscript/202601.0019/v1

19. Tahir, S., Anwar, A., & Khan, A. N. (2023). Performance Evaluation of NUMA-Aware Memory Allocation Strategies for High-Speed Packet Processing. IEEE Access, 11, 24531–24548.

20. Pirelli, S., Iyer, R., & Argyraki, K. (2023). Automated Verification of Network Function Binaries. In Proceedings of the 20th USENIX Symposium on Networked Systems Design and Implementation (NSDI '23), 585–600.

21. Chelsio Communications. (2023). T6 100G DPDK Performance Benchmark Report. URL: https://www.chelsio.com/wp-content/uploads/resources/t6-100g-dpdk-linux.pdf

22. Manousis, A., Sharma, R. A., Sekar, V., & Sherry, J. (2021). Contention-Aware Performance Prediction for Virtualized Network Functions. In Proceedings of ACM SIGCOMM 2021, 270–282. URL: https://doi.org/10.1145/3452296.3472904

23. DPDK Project. (2025). Graph Library and Graph Architecture, Release 25.03. URL: https://doc.dpdk.org/guides-25.03/prog_guide/graph_lib.html

24. Bonati, L., D'Oro, S., Polese, M., Basagni, S., & Melodia, T. (2023). Intelligence and Learning in O-RAN for Data-Driven NextG Cellular Networks. IEEE Communications Magazine, 59(10), 21–27.

25. NVIDIA Corporation. (2024). NVIDIA ConnectX-6 Dx Adapter — Inline IPsec Offload with DPDK Technical Brief. URL: https://www.nvidia.com/en-us/networking/

26. Intel Corporation. (2024). Intel QuickAssist Technology (QAT) Cryptographic

TSARKA SCIENCE

Volume 1 (2026), Issue 2
pp. 31–54

eISSN: 3134-6057
pISSN: 3134-6049

Performance with DPDK. Intel Network Builders Technology Guide.

27. Drucker, N., & Gueron, S. (2018). Making AES Great Again: The Forthcoming Vectorized AES Instruction. In Information Technology — New Generations (Advances in Intelligent Systems and Computing, vol. 800), 37–41. URL: https://eprint.iacr.org/2018/392

28. Open Information Security Foundation. (2024). Suricata 7.0 User Guide: DPDK Capture Mode. URL: https://docs.suricata.io/en/latest/capture-hardware/dpdk.html

29. Dumitrescu, C., Wang, H., et al. (2023). Running P4 Programs as DPDK Applications: The p4c-dpdk Compiler Workflow. In Proceedings of the P4 Workshop 2023. URL: https://research.cec.sc.edu/files/cyberinfra/files/1--fundamentals-of-p4-and-dpdkv2.pdf

30. Češka, M., Havlena, V., Holík, L., Korán ek, J., Lengál, O., Matoušek, D., Matoušek, J., Semrič, J., & Vojnar, T. (2019). Deep Packet Inspection in FPGAs via Approximate Nondeterministic Automata. In Proceedings of the 27th IEEE International Symposium on Field-Programmable Custom Computing Machines (FCCM 2019), 109–117. URL: https://arxiv.org/abs/1904.10786

31. Yu, S., & Won, Y. (2023). A Survey of Methods for Encrypted Network Traffic Fingerprinting. Mathematical Biosciences and Engineering, 20(2), 2183–2202. URL: https://doi.org/10.3934/mbe.2023101

32. Holík, J., Čejka, T., & Čermák, M. (2024). Using JA4+ Fingerprints for Malware Detection in Encrypted Traffic. In Proceedings of the 20th International Conference on Network and Service Management (CNSM 2024). URL: https://opendl.ifip-tc6.org/db/conf/cnsm/cnsm2024/1571045669.pdf

33. Lin, W., Shu, Y., & Zhang, Y. (2024). SuperNIC: An FPGA-Based, Cloud-Oriented SmartNIC. In Proceedings of the 30th IEEE International Symposium on High-Performance Computer Architecture (HPCA 2024), 421–435.

Information about authors

Kabdylkak Arnur – Embedded System Developer, TSARKA GROUP, Astana, Kazakhstan.

e-mail: kabdylkakarnur@gmail.com

ORCID: 0009-0001-0235-6436

Zhumabek Miras – FPGA Developer, TSARKA GROUP, Astana, Kazakhstan.

ORCID: 0009-0008-3346-1256