Human Factors in Industrial and Systems Engineering, Vol. 219, 2026, 154–164
https://doi.org/10.54941/ahfe1007786

Zero-Trust Access Control for IoT in Critical Infrastructure Environments

Osama A. Khashan[1], Samar Mouti[2], Nour M. Khafajah[2], Nachaat Mohamed[1], and Waleed Alomoush[3,4]

[1]Homeland Security Department, Rabdan Academy, Abu Dhabi, P.O. Box 114646, UAE
[2]College of Engineering and Computing, Liwa University, Abu Dhabi, P.O. Box 41009, UAE
[3]Plekhanov Russian University of Economics, Knowledge Park, Dubai, UAE
[4]College of Computing and Intelligent Systems, University of Al Dhaid, Sharjah, UAE

ABSTRACT

Static permissions in conventional access control systems for the Internet of Things (IoT) are often persistent even after a device has registered in a deployment. Therefore, a compromised device may retain long-lived privileges through a cloned identity, and this increases the likelihood of unauthorized activity and lateral movement in the context of critical infrastructure environments. This paper presents a user-centred access control model that combines zero-trust principles and short-lived capability tokens. Devices are not trusted by default; each service request explicitly carries verifiable authorization. The policy engine issues tokens that bind device identity, target service, permitted operation, validity window, and contextual constraints. Gateways and services validate tokens for each request and deny requests that are expired or out of scope. As a result, misuse is limited without requiring continuous connectivity to the policy engine. The proposed model is also protocol agnostic, and it transports tokens via application-layer message exchanges across heterogeneous IoT stacks. A simulation-based evaluation using a heterogeneous IoT model assesses credential cloning, unauthorized invocation, and compromised-node scenarios. At high compromise levels, unauthorized request success drops from 74% in the baseline to 6% under the proposed model. The operational cost remains moderate, with a mean end-to-end latency increase of about 20% and total communication overhead between 21.25% and 30.75% across the tested token lifetimes. Overhead is split into token carriage and issuance; issuance cost falls as token lifetime grows. The results show reduced unauthorized requests with bounded per-request verification cost and moderate overhead.

Keywords: Zero trust, Access control, Internet of Things (IoT), Authorization, Critical infrastructure, IoT security

INTRODUCTION

Deployments of the Internet of Things (IoT) into critical infrastructures have to be adapted to limited resources, along with changing operational conditions under adversarial pressure (Stouffer et al., 2023). Actuators are connected intermittently, and battery-powered sensors are required to meet strict control deadlines (Zanasi et al., 2024). Authorization is typically treated statically as an onboarding process in many of these deployments. After a device has been registered, it is provided long-lived permissions for

Received February 4, 2026; Revised April 3, 2026; Accepted April 21, 2026; Available online July 20, 2026

all operations, assuming the availability of continuous communication and benign device behavior over time (Diaz & Mendoza, 2025).

Centralized permission stores can become single points of failure, and revocation may not propagate reliably over intermittent links (Khashan, 2024). Heterogeneous hardware often lacks consistent, enforceable roots of trust. Static access rules allow compromised or cloned devices to continue invoking services, and full re-authorization can induce computational and radio exhaustion that is unsuitable for sleeping, roaming, or duty-cycled nodes (Stouffer et al., 2023).

The expansion of critical infrastructure increases these vulnerabilities across both industrial systems and essential public services that directly affect civil society. Smart-city and safety-relevant IoT deployments, such as emergency response, healthcare, transportation, and municipal infrastructure systems, are subject to high levels of availability and latency constraints, where delays in actuation or unauthorized commands can result in physical disruption or public harm (Khashan et al., 2023). These systems have long equipment lifecycles and contain many legacy devices, thus limiting the ability to frequently update credentials and policies (ISA, 2025). Segmented networks and remote locations limit continuous reachability to centralized policy services; therefore, when an incident occurs, rapid and reliable revocation is required, even under limited network connectivity. Static permissions are dangerous within this context, as a single compromised credential may be valid for extended periods of time, allowing continued malicious use (Zanasi et al., 2024; Khashan & Khafajah, 2018).

Practical adversaries exploit this gap. Replay and relay attacks preserve cryptographic validity while breaking proximity and timing expectations in cyber-physical settings (Albinali & Azzedin, 2025). Credential cloning through supply-chain exposure, debug interfaces, or side-channel extraction enables fabricated legitimacy. When service permissions are bound directly to leaked credentials, authorization becomes brittle and misuse becomes economical (Sasi et al., 2024).

Zero-trust addresses this brittleness by removing implicit trust and requiring explicit verification for each interaction tied to current, request-level evidence (Rose et al., 2020). However, constrained IoT nodes cannot sustain heavyweight policy brokers or deep traffic inspection, and they operate under strict latency, energy, and duty-cycle ceilings (Seitz et al., 2022). The challenge is to realize service-layer zero-trust in a way that keeps verification cost aligned with device capability and critical-infrastructure service objectives (Rose et al., 2020).

A practical path is to replace static permissions with short-lived capability tokens that encode device identity, permitted operation, validity window, and contextual constraints (Li et al., 2022). Devices obtain tokens from a policy engine, and every service invocation must present a token that matches the requested operation (Seitz et al., 2022). Services and gateways validate tokens locally, so authorization depends on per-request proof rather than long-term trust in device state. This shifts access control from one-time admission to continuous, request-by-request authorization while remaining compatible with existing IoT protocols (Fotiou et al., 202; Khashan et al., 2015).

This paper presents a zero-trust access control method for critical-infrastructure IoT using short-lived capability tokens that accompany each service request. The protocol-agnostic design supports heterogeneous deployments and enables intent-based, per-request authorization with limited device overhead. The contributions are threefold: (i) a zero-trust access control model for critical-infrastructure IoT; (ii) a capability-token design and issuance workflow suitable for constrained devices; and (iii) Simulation results confirm improved service protection under cloning and misuse, with limited additional latency and communication overhead. Section 2 surveys related work, Section 3 presents the proposed model, Section 4 reports results and analysis, and Section 5 concludes the paper.

RELATED WORK

Recent work on IoT security increasingly treats authorization as a continuous process rather than a one-time onboarding decision. The study by Wang et al. (2025) review Identity and Access Management (IAM) for future IoT services and argue that conventional IAM patterns, largely shaped by cloud and web assumptions, become brittle when identities and permissions must be managed across heterogeneous, intermittently connected edge-native deployments. Pérez Díaz and Almenares Mendoza (2025) compare authorization models in IoT and emphasize that the post-authentication stage remains the primary bottleneck for many deployments, especially when policies must be fine-grained, context-aware, and enforceable under constrained resources. In parallel, Khashan and Khafajah (2023) address the identity side of the problem by proposing a hybrid centralized and blockchain-based authentication architecture for heterogeneous IoT, showing how distributed trust anchors can harden onboarding and device legitimacy. More recently, Khashan (2025) extends this direction with a fog–blockchain authentication model tailored to smart-city deployments, emphasizing scalable verification under edge constraints, but still framing protection around node authentication rather than request-scoped service authorization.

Within that framing, zero-trust proposals strengthen the requirement that each interaction carry explicit, verifiable authorization. Zhang et al. (2024) push this idea to the network layer through per-packet authorization, which reliance on implicit trust and the opportunity window for misuse when credentials leak or devices are compromised. While packet-level schemes provide an interpretation of "always verify," they also highlight a recurring challenge in constrained settings: verification frequency and signaling overhead must be kept predictable so that security does not destabilize service latency targets.

A practical response has been to encode authorization into compact tokens that can be validated locally at gateways and services. Díaz-Sánchez et al. (2026) propose a zero-trust token authorization approach for scalable distributed firewalls, where cryptographically protected tokens embed signed policy and are verified non-interactively at enforcement points. Their design targets scalability by avoiding the distribution and synchronization of large rule tables, although the model is evaluated in firewall-style contexts rather

than service-specific IoT interactions. At the architectural level, Salim et al. (2026) propose ZT-BlocIoT, combining zero-trust principles with blockchain and optimization mechanisms aimed at scalability and resilience under malicious behavior.

Several token-based designs minimize endpoint burden by treating the blockchain as a trust anchor while keeping IoT devices off-chain. Zhu et al. (2025) introduce G-CapBAC for smart-campus IoT, where capability tokens are managed at the group level and ring signatures support scalable request processing while providing privacy properties. The emphasis on grouping and lightweight verification is aligned with large heterogeneous deployments, though group-based token semantics can be less direct when authorization must reflect request intent at the granularity of individual operations and service contexts. In addition, many recent proposals also integrate cryptographic policy enforcement to tighten authorization scope. Nie et al. (2025) present a zero-trust access control mechanism that combines blockchain with inner-product encryption and smart contracts, aiming for fine-grained control while supporting a self-sovereign style of identity management. These constructions strengthen confidentiality of attribute policies and support micro-segmentation, but they also introduce cryptographic and operational complexity that can be difficult to reconcile with low-latency actuation paths and duty-cycled devices (Khashan, 2025).

A parallel direction is risk-adaptive authorization, where decisions reflect dynamic trust signals rather than static identity alone. Ma and Chiu (2025) propose a risk-based access control engine for zero-trust IoT networks that continuously evaluates trust and risk and adapts mitigation policies accordingly. Cross-domain IoT deployments motivate authorization models that support delegation without reverting to long-lived privileges. Mukta et al. (2025) use a blockchain-backed delegation workflow to propagate capability-style permissions under a zero-trust stance, which is useful when administrative control is distributed. This further motivates request-scoped authorization that remains verifiable at enforcement points even when connectivity and governance boundaries are fragmented.

PROPOSED MODEL

This section describes the user-centred zero-trust authorization model for IoT. The model replaces long-lived permissions with short-lived capability tokens that accompany service request. Authorization is enforced at gateways and services through verification, so constrained devices avoid continuous policy evaluation while operating under request-level verification.

Architecture and Request Flow

The deployment comprises four logical components: the IoT device (D), a constrained sensor or actuator that invokes protected functions; the policy engine (PE), which evaluates access intent and issues short-lived capability tokens; the gateway (GW), an edge enforcement point that validates tokens and forwards only authorized requests; and the protected service (S), which

provides a specific function (e.g., actuation or data access) and verifies tokens directly or relies on GW for pre-filtering. Within the gateway, a token proxy (TP) mediates token requests and returns issued tokens to the device as authorized by PE decisions. Figure 1 summarizes the architecture and request flow among these entities.

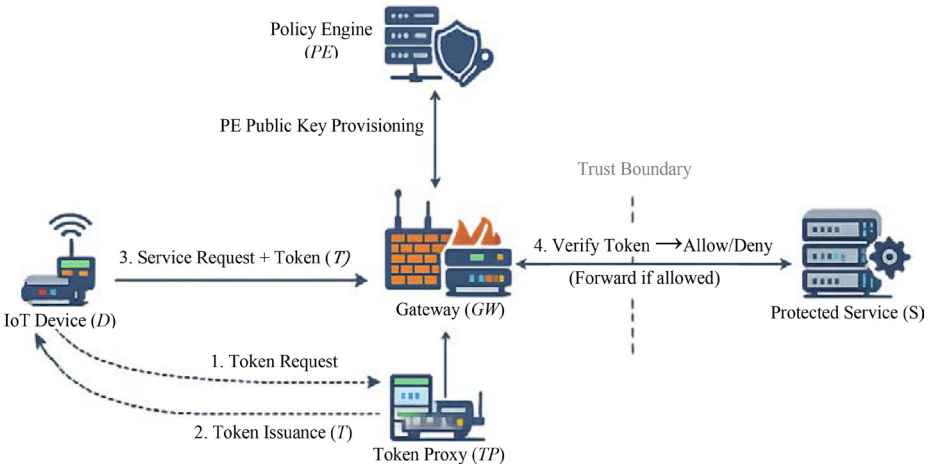

Figure 1: Architecture of proposed zero-trust capability-token authorization model.

To align notation with the proposed model, a token request is denoted as Q, a capability token is denoted as T, and a service request carrying a token is denoted as R. Device and service identities are written as id_D and id_S, and op denotes the requested operation. Token validity uses t_{np} (not-before) and t_{exp} (expiration), and contextual constraints are summarized as ctx. The enforcement point uses t_{now} to assess token validity at verification time.

The model assumes standard cryptographic primitives, namely a digital signature algorithm and a cryptographic hash function. PE holds a private signing key and generates the token signature sig_{PE} over the token fields, while its public verification key is distributed to GW and S using a secure provisioning mechanism so tokens can be validated locally. TP may cache the PE public key and recent token responses to reduce dependence on continuous connectivity during steady-state operation. The device D is not required to store sensitive long-term authorization state beyond its identity material used to request tokens. This choice reflects heterogeneous IoT environments where device vendors and hardware roots of trust are not uniform.

The access flow is request-scoped. When D is intended to invoke an operation on S, it first obtains a short-lived token via TP and PE using Q. The device then issues R carrying T. The enforcement point (either GW or S) makes an allow/deny decision locally based on token verification and contextual checks, using t_{now} to confirm validity. This removes dependence on continuous reachability to PE during normal operation and limits the window of misuse after credential compromise.

Two trust boundaries are central in critical-infrastructure IoT. First, the boundary between the IoT edge and the service network, where *GW* enforces policy before traffic reaches protected services, thereby reducing exposure and supporting segmentation. Second, the boundary between policy evaluation and policy enforcement, in which the *PE* performs heavier policy reasoning when issuing tokens, while *GW*/ *S* perform lightweight, deterministic checks at request time. This separation is intended to keep per-request cost predictable and compatible with latency constraints.

Threat assumptions align with practical misuse patterns. Credential cloning allows an adversary to impersonate a device identity and reuse authorization artifacts. Unauthorized service invocation occurs when a compromised device attempts operations outside its permitted scope. Compromised nodes and lateral movement describe cases where an attacker leverages valid access to pivot across services or zones. The design mitigates these threats by (i) restricting token lifetimes, (ii) binding tokens to service and operation scope, and (iii) encoding contextual constraints that must hold at request time.

Token Issuance and Enforcement

A capability token is a compact authorization object issued by *PE* that binds a device identity to a specific service operation under a short validity window and explicit contextual constraints. The token is defined as:

$$T = \langle id_D,\ id_S, op,\ t_{np}, t_{exp}, ctx,\ sig_{PE} \rangle \tag{1}$$

The token lifetime (L) is calculated as:

$$L = t_{exp} - t_{np} \tag{2}$$

Short lifetimes reduce the impact of stolen or cloned credentials. L is selected to balance security with operational constraints such as duty cycling and intermittent connectivity.

A device requests a token when it intends to access a service. The token request message is:

$$Q = \langle id_D,\ id_S, op,\ t_{np}, ctx_e \rangle \tag{3}$$

where ctx_e is the context evidence supplied by the device or gateway (e.g., current zone, assigned role, or a task identifier). *PE* evaluates Q against stored policies and contextual constraints. If the request satisfies policy, *PE* constructs T, selects (t_{np}, t_{exp}), and sign the token T.

To limit device-side overhead, token encoding is compact and protocol-agnostic. The token can be transported in application-layer messages (e.g., CoAP options, MQTT properties, or HTTP headers), enabling reuse across heterogeneous IoT stacks without modifying lower-layer protocols.

A service invocation is represented as:

$$R = \langle id_D,\ id_S, op,\ payload, T \rangle \tag{4}$$

On receiving R, the enforcement point validates the token. The authorization decision is computed as:

$$Allow(R) = Verify(sig_{PE}) \wedge \left(t_{np} \le t_{now} \le t_{exp} \right) \wedge Match(id_D,\ id_S, op, ctx) \tag{5}$$

where the *Verify* function checks the signature using the *PE* public key, whereas the *Match* function verifies that the token binds the correct device, service, and operation, and that the current context satisfies the constraints in *ctx*. The *Match* function is intentionally lightweight and cost-stable. It avoids complex policy re-evaluation per request and relies on a small set of deterministic validations with predictable cost. This approach preserves request-scoped authorization and reduces the burden on constrained endpoints.

In the presence of suspected compromise, the primary control is token non-renewal. *PE* stops issuing new tokens for the compromised identity, which prevents continued access once existing tokens expire. For urgent response, gateways may also maintain a small deny list of device identifiers or token fingerprints for immediate blocking. This optional mechanism provides rapid containment without requiring large revocation lists or constant synchronization across intermittent links.

The computational burden of verification is placed on GW and S, which are typically more capable than constrained devices. Devices only attach tokens to requests, which reduces device computation and communication overhead. Since tokens are short-lived and operation-scoped, stolen tokens have limited value and cannot be reused beyond their validity window or outside their intended operation scope. This approach directly supports critical-infrastructure environments, where unauthorized commands or delayed actuation can cause operational harm.

ANALYSIS AND RESULTS

This section reports the simulation-based evaluation of the proposed zero-trust capability-token model. The experiments assess two points: reduction of unauthorized service use under compromise scenarios, and the operational cost introduced by request-level enforcement at the gateway and service.

The evaluation used MATLAB R2024b with a discrete-event network model capturing multi-hop forwarding, link loss, and gateway queueing, and an application-layer request/response model implementing token request, token issuance, token carriage, and token verification at the gateway with optional service-side re-verification. The prototype ran on a 64-bit workstation with an Intel Core i7-8655U CPU at 1.90 GHz with 8 MB cache and 16 GB RAM. Each scenario used 30 independent seeded runs and reports mean values.

Three adversarial conditions were modeled: credential cloning, where cloned devices attempted service access using copied identity material; unauthorized invocation, where compromised devices requested operations outside their permitted scope; and compromised-node behavior, where request intensity and target services were varied to emulate pivot attempts. A baseline configuration used long-lived permissions without request-scoped enforcement.

We first evaluate the proposed model under increasing compromise intensity to quantify its ability to block unauthorized service access as device identities are cloned or misused. Figure 2 shows the unauthorized request success rate as the compromised or cloned device fraction increases from 0% to 40%.

Under the baseline, unauthorized requests remain viable after compromise, and the acceptance rate rises from approximately 18% at low compromise levels to 74% at higher levels. Under the proposed model, enforcement at the gateway or service rejects unauthorized requests when the token is expired or out of scope, which keeps the unauthorized success rate between 2% and 6% across the tested range, reaching 6% at 40% compromise. The largest improvement appears when compromised devices attempt operations outside their permitted scope, where scope binding leads to consistent rejection. Any remaining accepted requests align with valid scope and time windows, not with a bypass of verification.

Next, we evaluate the latency impact of request-scoped enforcement as the network size increases. Figure 3 reports end-to-end latency for 10 to 200 devices. The baseline latency ranges from 55 ms to 225 ms across the tested range. With capability-token enforcement, mean latency increases from 72 ms to 250 ms due to token verification at the gateway and the optional second check at the service. Across the tested sizes, the proposed model shows an average latency increase of about 20.4% relative to the baseline. Latency grows smoothly with network size, and the larger rise at 150–200 devices is consistent with gateway queue buildup and channel contention, not with changes in the authorization logic.

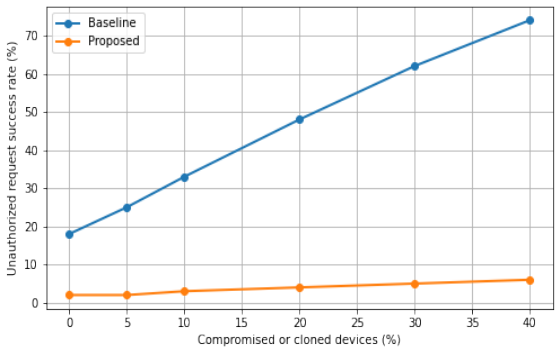

Figure 2: Unauthorized request success rate vs. compromised or cloned device fraction.

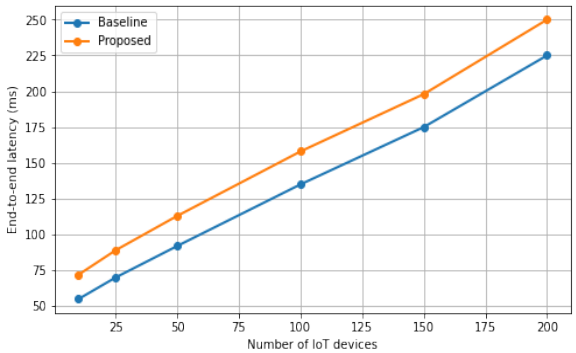

Figure 3: End-to-end latency vs. number of devices for the baseline and proposed model.

Finally, we evaluate communication overhead as a function of token lifetime. Figure 4 separates the overhead into two parts. Token carriage contributes a near-constant per-request overhead because token size is fixed. With a token size of 96 bytes and a baseline request payload of 512 bytes, carriage adds 18.75% overhead per request. Total traffic increases with request volume, but this carriage fraction remains stable under steady request rates. Token issuance adds control-plane overhead that depends on token lifetime. With a 30 s lifetime, issuance overhead reaches about 12%, which yields a total overhead of 30.75%. When token lifetime increases to 300 s, issuance overhead drops to about 2.5%, and total overhead decreases to 21.25%. This decrease occurs because longer-lived tokens require fewer renewals over the same operating period, which reduces token request and issuance exchanges. An intermediate lifetime of 120 s reduces issuance overhead to about 5%, with total overhead of about 23.75%, which illustrates the expected trade-off between refresh frequency and control-plane cost.

As a summary, the results show that unauthorized access is curtailed under the tested misuse cases, while latency remains within a controlled range as the network scales. Communication overhead is driven mainly by token carriage, and renewal traffic decreases as token lifetime increases, which provides a practical tuning parameter for deployment settings.

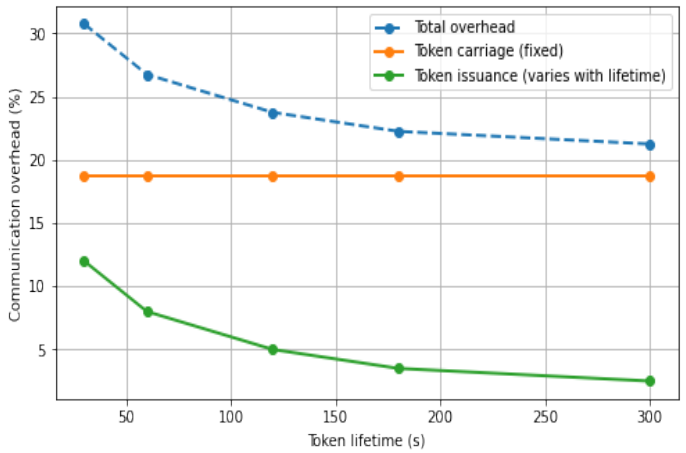

Figure 4: Communication overhead vs. token lifetime, separating token carriage and token issuance overhead.

CONCLUSION

This paper presents a zero-trust, user-centric access control model for heterogeneous IoT deployments in critical infrastructure, where long-lived permissions and weak revocation amplify the impact of device compromise and credential cloning. The design substitutes permanent rights with capability tokens that have a limited lifetime and are issued by a policy engine and enforced by gateways and services per request. Enforcement relies on the

identity of the requesting entity, the service being requested, operation scope, validity window, and contextual constraints, to narrow the exposure window and reduce the privilege surface available for misuse. The gateway enforces authorization while keeping device-side cost low and supporting intermittent connectivity to the policy engine, since most requests are processed at the gateway without requiring continuous reachability to the policy engine. In the simulation experiments, unauthorized request success rate dropped by 68% at high compromise levels, representing an approximately 92% reduction relative to the baseline. The operational impact remained bounded, with mean end-to-end latency increasing by about 20.4% and total communication overhead staying within 21.25%–30.75% under the tested token lifetimes. Longer token lifetimes reduced issuance exchanges and lowered control-plane traffic, which offers a practical parameter to balance refresh cost against exposure duration. Future work will evaluate richer context constraints and varied traffic profiles, and validate the design on a gateway-centric prototype under realistic protocol stacks and intermittent connectivity.

REFERENCES

Albinali, H., & Azzedin, F. (2025). Replay attacks in RPL-based Internet of Things: Comparative and empirical study. Computer Networks, 257, 110996.

Díaz, J. P., & Almenares Mendoza, F. (2025). Authorization models for IoT environments: A survey. Internet of Things, 29, 101430.

Díaz-Sánchez, D., Almenarez-Mendoza, F., Campo-Vázquez, C., & García-Rubio, C. (2026). Zero-trust token authorization with trapdoor hashes for scalable distributed firewalls. Future Generation Computer Systems, 176, 108227.

Fotiou, N., Siris, V. A., Polyzos, G. C., Kortesniemi, Y., & Lagutin, D. (2022). Capabilities-based access control for IoT devices using verifiable credentials. In: 2022 IEEE Security and Privacy Workshops (SPW), pp. 222–228. IEEE, San Francisco, CA.

ISA. (2025). ISA/IEC 62443 series of standards (overview page), accessed 2025. ISA website: http://www.isa.org/standards-and-publications/isa-standards/isa-iec-62443-series-of-standards.

Khashan, O. A. (2024). Blockchain-machine learning fusion for enhanced malicious node detection in wireless sensor networks. Knowledge-Based Systems, 304, 112557.

Khashan, O. A. (2025). Dual-stage machine learning approach for advanced malicious node detection in WSNs. Ad Hoc Networks, 166, 103672.

Khashan, O. A. (2025). Trust based fog-blockchain model for scalable IoT node authentication within smart-city networks. Computer Networks, 264, 111278.

Khashan, O. A., & Khafajah, M. (2023). Efficient hybrid centralized and blockchain-based authentication architecture for heterogeneous IoT systems. Journal of King Saud University – Computer and Information Sciences, 35(2), 726–739.

Khashan, O. A., & Khafajah, N. M. (2018). Secure Stored Images Using Transparent Crypto Filter Driver. International Journal of Network Security, 20(6), 1053–1060.

Khashan, O. A., Khafajah, N. M., Alomoush, W., Alshinwan, M., Alamri, S., Atawneh, S., & Alsmadi, M. K. (2023). Dynamic multimedia encryption using a parallel file system based on multi-core processors. Cryptography, 7(1), 12.

Khashan, O. A., Zin, A. M., & Sundararajan, E. A. (2015). ImgFS: A transparent cryptography for stored images using a filesystem in userspace. Frontiers of Information Technology & Electronic Engineering, 16(1), 28–42.

Li, C., Li, F., Huang, C., Yin, L., Luo, T., & Wang, B. (2022). A traceable capability-based access control for IoT. Computers, Materials & Continua, 72(3), 4967–4982.

Ma, Y.-W., & Chiu, P.-H. (2025). A novel risk-based access control engine in zero trust architecture for IoT network. International Journal of Information Security, 24(3), Article 124.

Mukta, N. N., Gurjar, A., Bhushan, S., Ummadi, V., & Sharma, A. (2026). Design and implementation of the blockchain-enabled permission delegation approach for multi-domain IoT using zero-trust capability-based access control. Blockchain: Research and Applications, 7(1), 100281.

Nie, S., Ren, J., Wu, R., Han, P., Han, Z., & Wan, W. (2025). Zero-trust access control mechanism based on blockchain and inner-product encryption in the Internet of Things in a 6G environment. Sensors, 25(2), 550.

Rose, S., Borchert, O., Mitchell, S., & Connelly, S. (2020). Zero trust architecture. National Institute of Standards and Technology, Gaithersburg, MD (NIST Special Publication 800–207).

Salim, M. M., Kim, M., Singh, S. K., & Park, J. H. (2026). Zero-trust blockchain-enabled framework for scalable and secure IoT networks. Future Generation Computer Systems, 175, 108093.

Sasi, T., Lashkari, A. H., Lu, R., Xiong, P., & Iqbal, S. (2024). A comprehensive survey on IoT attacks: Taxonomy, detection mechanisms and challenges. Journal of Information and Intelligence, 2(6), 455–513.

Seitz, L., Selander, G., Wahlstroem, E., Erdtman, S., & Tschofenig, H. (2022). RFC 9200: Authentication and authorization for constrained environments using the OAuth 2.0 framework (ACE-OAuth). RFC Editor (RFC 9200).

Stouffer, K., Pease, M., Tang, C. Y., Zimmerman, T., Pillitteri, V., Lightman, S., Hahn, A., Saravia, S., Sherule, A., & Thompson, M. (2023). Guide to operational technology (OT) security. National Institute of Standards and Technology, Gaithersburg, MD (NIST Special Publication 800-82 Rev. 3).

Wang, Y., Castillejo, P., Martínez-Ortega, J.-F., & Hernández Díaz, V. (2025). A survey on identity and access management for future IoT services. Computer Networks, 272, 111718.

Zanasi, C., Russo, S., & Colajanni, M. (2024). Flexible zero trust architecture for the cybersecurity of industrial IoT infrastructures. Ad Hoc Networks, 156, 103414.

Zhang, H., Wang, Q., Zhang, X., He, Y., Tang, B., & Li, Q. (2024). Toward zero-trust IoT networks via per-packet authorization. IEEE Communications Magazine, 62(12), 90–96.

Zhu, X., Zou, S., Xu, G., & Xi, J. (2025). Group-capability-based access control with ring signature. Journal of Information Security and Applications, 90, 104014.