



EVIDENCE-INFORMED CROSS-LAYER SECURITY RECOMMENDATIONS FOR WIRELESS NETWORKS AND IOT DEVICES BASED ON PENETRATION TESTING

S. Tleuberdin¹, Zh.Boranbay², K.Malakhov³, T.Aidynov^{4*}

¹Research Institute of Information Security and Cryptology, L.N. Gumilyov Eurasian National University, Astana, Kazakhstan, <https://orcid.org/0000-0001-9170-9936>.

²School of Cybersecurity, Astana IT University, Astana KZ-010000, Kazakhstan, <https://orcid.org/0009-0009-0496-3876>.

³Research Institute of Information Security and Cryptology, L.N. Gumilyov Eurasian National University, Astana 010008, Kazakhstan, <https://orcid.org/0009-0004-5695-2243>

⁴Research Institute of Information Security and Cryptology, L.N. Gumilyov Eurasian National University, Astana 010008, Kazakhstan, <https://orcid.org/0009-0005-3327-9719>.

*Corresponding author: Tolegen Aidynov, e-mail: 951005351309@enu.kz.

Abstract

Smart-city and smart-home systems rely heavily on Wi-Fi, Bluetooth Low Energy, and ZigBee, while many practical risks arise from deployment and implementation weaknesses rather than failures of cryptographic primitives. This paper presents a structured, author-led synthesis of seven previously completed laboratory studies involving adaptive Wi-Fi penetration testing, WPA3-Enterprise attacks, a cross-layer audit of a commercial Wi-Fi smart lock, reverse analysis of a BLE control channel, cross-layer ZigBee assessment, software-defined-radio jamming, and an IoT security laboratory task set. The previously reported experimental findings are normalized by affected layer, attack vector, observable effect, test conditions, and source. On this basis, the study develops a six-level cross-layer taxonomy and 52 stakeholder-oriented recommendations for infrastructure administrators, device and firmware developers, mobile and cloud service developers, monitoring and risk-assessment teams, and testing-process owners. Each recommendation is assigned an evidence level distinguishing direct experimental support, support from external standards or literature, and proposed expert guidance. Priority controls are linked to reproducible acceptance criteria and twelve core metrics, while an extended traceability matrix provides qualitative or quantitative checks for the remaining recommendations. The resulting five-stage procedure is proposed as a testing-remediation-re-verification framework; it has not yet been independently validated as a complete system on an industrial testbed. The contribution is therefore an evidence-informed operational framework rather than a claim that all controls or thresholds are universally validated.



Keywords: penetration testing, Internet of Things, Wi-Fi, WPA3, Bluetooth Low Energy, ZigBee, software-defined radio, cross-layer security audit.

1. Introduction

Wireless connections carry much of the traffic in modern Internet of Things (IoT) systems. The IEEE 802.11 family provides the principal standards base for Wi-Fi connectivity [18], while the worldwide installed base of connected IoT devices continues to grow rapidly [19]. Wi-Fi is joined in homes and offices by two established low-power options: Bluetooth Low Energy (BLE) [5] and ZigBee, which runs on the IEEE 802.15.4 physical and link layers [17], [10]. Many security incidents trace back to installation mistakes or flaws in end-device application logic rather than weaknesses in cryptographic primitives [28].

Wi-Fi security has improved by closing off known attack methods, but each generation has exposed fresh weaknesses. The key reinstallation attack showed that WPA2's four-way handshake could be compromised [38]. Dragonblood found side channels and downgrade paths in WPA3's Simultaneous Authentication of Equals mechanism [39], while research into frame aggregation and fragmentation uncovered implementation flaws across many device types [37]. Reviews of WPA3 [14] and guidance from regulators [27] conclude that migration lowers these risks without eliminating those tied to transitional compatibility modes or optional management frame protection.

BLE devices expose a different set of weaknesses. Instead of network identifiers and IP transport, an attacker may target advertising packets, the central–peripheral relationship, or the Generic Attribute Profile's structure. The Bluetooth Core Specification includes full mechanisms for pairing, bonding, and encryption [5]. Yet educational and low-cost products often leave them disabled, leaving the control characteristic open to anyone. Low-power mesh networks face a different kind of threat. ZigBee's AES-128 encryption, specified in [10], protects message confidentiality and authenticates data, but it cannot shield the radio channel from interference. Affordable software-defined radio (SDR) equipment has also made jamming attacks easy to mount; recent surveys organise the known attack scenarios and countermeasures [30]. Because Wi-Fi, BLE, and ZigBee all operate in the 2.4 GHz band, interference can also cross protocol boundaries.

IoT penetration testing has reached a relatively mature stage. Researchers have proposed automation schemes [9], taxonomies of processes and standards [31], classifications of ethical-hacking tools grounded in research [24], and methods for automatically verifying and replaying attacks against devices [13]. Other work applies reinforcement learning to automate testing [32], [23], [21], [20], while [41] examines how penetration testing can be combined with machine



learning to detect anomalies in smart-city networks. Nevertheless, audit findings often fail to change day-to-day operations. A report may show that an attack is feasible, yet still omit configuration and firmware requirements that are formalized, assigned to an owner, and verifiable. The same weaknesses therefore return: management frames remain unprotected, service requests lack rate limits, control commands lack freshness checks, mass sensor disconnections trigger no response, and mobile application logs retain sensitive data.

This study converts evidence from seven previously completed studies involving members of the present author group into practical guidance for wireless-network and IoT-device developers and administrators. The experiments and their primary results remain attributable to the cited source publications. The new contribution of this manuscript is the structured cross-layer synthesis, the stakeholder-oriented catalogue of security requirements, and the traceability framework connecting observed attack vectors with mitigations, evidence levels, acceptance criteria, metrics, and verification tools.

The study makes four contributions. First, it places adaptive, cross-layer, and radio-physical penetration-testing findings for Wi-Fi, BLE, and ZigBee within one taxonomy. Second, it introduces two-sided traceability, defined here as an explicit link from each observed attack vector to both an implementation requirement and a reproducible verification activity. Third, it defines twelve core quantitative metrics and additional control-specific pass/fail checks. Fourth, it connects one-time penetration-testing findings to continuous monitoring and dynamic risk assessment. The framework is evidence-informed, but its complete industrial effectiveness and the universal validity of its numerical thresholds remain to be established.

Table 1. Relationship between the underlying studies and the new contribution of this manuscript

Source	Previously reported result	Evidence reused here	New synthesis in this paper
[34]	Q-learning-based adaptive Wi-Fi penetration testing	Confirmed Wi-Fi availability vectors and action outcomes	Administrator and testing requirements; baseline and re-verification metrics
[33]	Practical attacks on WPA3-Enterprise	Deauthentication, EAPOL-Start, Evil Twin, and certificate-validation findings	Control-to-test mapping for enterprise wireless authentication
[35]	Cross-layer audit of a commercial smart lock	Radio, network, application, cloud, and credential findings	Cross-layer ownership model and acceptance



			checks for devices and services
[36]	Laboratory analysis of a BLE control channel	GATT access, connection-slot occupation, and replay findings	BLE security requirements and state-changing command tests
[1]	Cross-layer ZigBee and anomaly-detection assessment	Firmware, traffic, and behavior evidence	Monitoring, evidence grading, and risk-assessment requirements
[40]	SDR-based ZigBee jamming study	Availability, interference, and energy-exhaustion findings	Radio-monitoring and resilience acceptance criteria
[2]	IoT penetration-testing laboratory task set	Layered tools, safety procedure, and laboratory thresholds	Five-stage verification procedure and testing-process controls

2. Material and research methods.

The evidence base consists of seven previously completed experimental studies conducted by overlapping members of the present author group in controlled laboratory settings. This section summarizes the source testbeds, tools, procedures, and limitations; Section 3 presents the new cross-layer synthesis and recommendation framework. The present work is a structured author-led evidence map rather than a systematic review of all published wireless and IoT penetration-testing studies.

2.1. Adaptive Wi-Fi testing based on Q-learning

In [34], penetration testing is treated as a Markov decision process and run through tabular Q-learning. Each state records radio conditions alongside signs that show whether the previous action worked: received signal strength, client MAC addresses, BSSID, channel, ESSID, the count of unique clients, how many remain connected afterward, the length of any zero-effect period, whether a handshake was captured, and the EAPOL frame count. Actions combine six to eight attack types-DeauthFlood, BeaconFlood, AuthDOS, WIDSConfusion, RTSCTSFlood, DeauthFloodMDK, and EAPOLStartFlood-with settings for intensity, thread count, and duration. Rewards add small credits for detected EAPOL frames and handshakes, a payment for every client disconnected, and a much larger one when the attack produces a complete denial-of-service event. Written in Python 3, the tool coordinates aircrack-ng, mdk4, tcpdump, and a custom analyser built with Scapy. The testbed used a TP-Link Archer C64 AC1200 router with three different clients: a MacBook M3 Pro, Samsung Galaxy S24+, and Mijia MJJSQ06DY Wi-Fi humidifier. Tests covered both WPA2 and WPA3-Personal.

2.2. Practical attacks on WPA3-Enterprise

In [33], researchers used a controlled setup with a RADIUS server to test several wireless attacks, including deauthentication, authentication denial of service, beacon flooding, EAPOL-Start flooding, man-in-the-middle attacks, and Evil Twin attacks. They worked through reconnaissance, threat analysis, exploitation, post-exploitation, and reporting, using mdk3, mdk4, aircrack-ng, and Wireshark on a Kali Linux host.

2.3. Cross-layer audit of a Wi-Fi smart lock

In [35], researchers audited a commercial Tuya 902V multi-factor smart lock using the PTES methodology. They captured wireless traffic with an Alfa AWUS036AXML adapter and the airodump-ng and aireplay-ng tools, then forced a re-association to obtain the WPA2 four-way EAPOL handshake. An offline dictionary attack followed, using a wordlist containing more than 14 million entries. For the Android application, they used Apktool, Jadx, and MobSF [22]. They also generated network-layer denial-of-service traffic with hping3 and tried cloning contactless credentials through a portable RFID/NFC platform [11].

2.4. Reverse analysis of a BLE control channel

Reference [36] examined the control channel of an educational ESP32-S3 quadcopter running MicroPython. Advertising packets revealed the device, after which an external Linux client using the Bleak library connected to its UART-like GATT service. The client sent eight-byte control frames and subscribed to eighteen-byte telemetry notifications. A hardware sniffer running Sniffle firmware also recorded the normal exchange between the stock controller and the aircraft; the resulting capture was inspected in Wireshark. Tests took place on a stationary bench with the propellers removed, and the aircraft was never flown.

2.5. Cross-layer assessment of a ZigBee ecosystem

In [1], the authors proposed and evaluated a cross-layer assessment scheme for a ZigBee-based smart-home ecosystem. The system draws on firmware-image data, network traffic, and user behaviour. For the network analysis, they used a 100-tree random forest classifier [7] with a 61-dimensional feature vector. Categorical fields were one-hot encoded, and the resulting values were range-normalised. They modelled deployment on a Raspberry Pi 4 single-board computer. To capture ZigBee traffic, the SDR pipeline sampled at 4 MS/s, decimated by 4, and applied a low-pass filter with a 1 MHz cut-off frequency; recordings were saved in PCAP format for later analysis.

2.6. Resilience of a ZigBee network to radio jamming

In [40], researchers used software-defined radio to test how well a ZigBee smart-home network withstands jamming. The setup included an Aqara M1S Gen 2 hub and a P1 motion sensor on channel 25. A CatSniffer intercepted traffic, while a HackRF One transceiver, paired



with a GNU Radio processing pipeline, generated interference. Spectral analysis and waterfall visualisation then characterised the resulting signal.

2.7. Laboratory task set for IoT testing

In [2], researchers built a laboratory complex for testing every layer of a smart-home system, from the physical and link layers through the network and application layers. The setup contains tools for scanning radio signals and diagnosing networks: they count packets, log signal strength for each device, and calculate delivery ratios. The test program also examines ZigBee traffic and frame counters, captures four-way handshakes, and recovers passwords for WEP, WPA/WPA2-PSK, and WPA2-Enterprise. It includes tests for rogue access points and a full audit of smart locks.

The researchers set out a five-stage procedure that others can repeat. First, they prepare the environment; then they identify the active channel and generate controlled interference. Next, they observe how devices respond, including events recorded by the mobile application, before analyzing the spectrum and documenting the findings. Safety rules require a shielded room, the lowest practical starting transmit power with gradual increases, a safety briefing, and compliance with radio-spectrum law.

Table 2. Summary of the experimental base

Study	Object and testbed	Layers and tools	Key confirmed result
[34]	TP-Link Archer C64, three heterogeneous clients, WPA2 and WPA3-Personal	Link layer; aircrack-ng, mdk4, tcpdump, Scapy, Q-learning agent	Sets of effective action combinations differ between WPA2 and WPA3; management frame protection blocks classic scenarios but not resource exhaustion
[33]	WPA3-Enterprise corporate testbed with RADIUS server	Link and authentication layers; mdk3, mdk4, aircrack-ng, Wireshark	Deauthentication and authentication denial of service remain feasible; Evil Twin and MITM succeed when server certificate validation is absent
[35]	Tuya 902V lock, router, Android application, MIFARE Classic 1K card	Radio, network, application and physical layers; Alfa AWUS036AXML, aircrack-ng, hping3, MobSF, Jadx, Apktool	Handshake captured but key not recovered; cloud connectivity disrupted; application defects identified; card sectors and keys fully read

Study	Object and testbed	Layers and tools	Key confirmed result
[36]	ESP32-S3 quadcopter, stock controller, external client, hardware BLE sniffer	Radio, transport (ATT/GATT) and application layers; Sniffle, Wireshark, Bleak	No mandatory pairing or encryption; unauthorised write to the control characteristic; successful frame replay; software reproduction of the stock controller
[1]	ZigBee smart-home ecosystem, gateway on Raspberry Pi 4	Firmware, network traffic, user behaviour; SDR capture to PCAP, Wireshark, random forest classifier	Anomaly detector accuracy 98.64 %, F1 = 0.986, latency about 300 ms on an edge device; absence of SSL pinning in the web interface
[40]	Aqara M1S Gen 2 hub and P1 motion sensor, channel 25	Physical layer; HackRF One, GNU Radio, CatSniffer	A low-power tone jammer causes 100 % packet loss and denial of service without compromising keys
[2]	Smart-home laboratory complex: hub, sensors, sockets, lock, Wi-Fi segment	Physical, link, network and application layers; SDR, KillerBee, aircrack-ng, application analysis tools	Quantitative boundaries of normal and jammed regimes; reproducible five-stage testing procedure

2.8. Construction of the taxonomy and the verification methodology

The evidence base was defined as a purposive corpus of seven author-led studies. A study was included when it: (1) reported a controlled security experiment involving Wi-Fi, BLE, ZigBee, or a related IoT component; (2) described the test object, tools, and procedure; (3) demonstrated an observable security effect; and (4) contained sufficient information to derive a mitigation or verification requirement. For each study, the technology, affected layer, attack vector, observable effect, experimental conditions, limitations, and supporting source were extracted. Overlapping vectors were consolidated when they represented the same underlying weakness. Recommendations were then formulated by identifying the responsible stakeholder and the minimum control intended to prevent, detect, or limit the observed effect. This procedure is a structured evidence mapping of the authors' experimental programme and should not be interpreted as a systematic literature review or independent consensus exercise.

Each recommendation was assigned an evidence level. E1 denotes direct support from one or more of the seven summarized experiments; E2 denotes support from an external standard or



independent publication; and E3 denotes technically motivated guidance that requires further validation. Evidence level indicates the origin of support and does not imply universal effectiveness across devices, protocols, or deployment environments.

The proposed verification process has five stages and is intended to use the same toolset twice: first to establish a baseline and then to assess the system after implementation of selected controls. Stage 1 records the number of associated clients, radio environment, receiver noise floor, packet delivery ratio, device-control interfaces, software versions, and testbed composition. Stage 2 runs documented action episodes. The adaptive tool uses fifteen episodes with an epsilon-greedy policy, while radio-physical trials follow the five-stage safety procedure described in [2]. Stage 3 calculates the applicable metrics. Stage 4 applies the controls specified in the traceability matrix. Stage 5 repeats the trial under equivalent conditions and compares the results. Under this proposed procedure, an implementation is considered compliant only when all pre-defined criteria tied to the tested vector are met. Unless a source study explicitly reports a completed remediation trial, the acceptance test should be interpreted as proposed rather than experimentally validated.

3. The results and discussion

3.1. Cross-layer taxonomy of confirmed vectors

Putting the seven studies into a shared format produces six levels of description, each defined by a consistent group of vectors and observable effects (Table 3). None of the confirmed vectors required breaking a cryptographic primitive. Instead, the weaknesses were practical: an unauthenticated service frame, an exposed radio channel, no resource limit, missing application-level authentication for a command, or a flaw in the supporting software.

Table 3. Taxonomy of confirmed vectors and observable effects

Level	Vector	Observable effect	Confirmed by
Physical (radio channel)	Tone and wideband jamming of the 2.4 GHz band by SDR equipment	Packet loss ratio up to 100 %, device availability 0–20 %, denial of service without key compromise	[40]; [2]
Physical (identification and credentials)	Passive capture of advertising packets, beacons and handshakes; passive determination of channel and network	Disclosure of a persistent device identifier and tracking capability; frequency plan determined within seconds; full readout of card sectors and keys	[34]; [35]; [36]



Level	Vector	Observable effect	Confirmed by
	identifier; contactless credential readout		
Link (802.11, BLE link layer, 802.15.4)	Flooding with deauthentication, beacon and RTS/CTS frames; frame replay; occupation of the single BLE connection slot; association table attacks	Termination of client sessions, degradation and complete loss of service; blocking of the legitimate controller	[33]; [36]; [40]
Authentication and key management	Authentication denial of service, EAPOL-Start flooding, Evil Twin, MITM with a forged certificate, network key compromise, identifier conflicts	Exhaustion of access point and RADIUS server resources; credential interception; unauthorised node association	[33]; [40]
Application (device protocol and companion software)	Unauthorised write to a GATT characteristic; control frame replay; absent SSL pinning; exported components and log analysis; metadata leakage from encrypted traffic	Reproduction of legitimate device control; substitution of management traffic; leakage of sensitive data; construction of a network activity map and identification of the coordinator	[36]; [35]; [1]
Network, cloud and energy	Load generation against the device network interface; battery exhaustion attack; cross-protocol interaction in the 2.4 GHz band	Loss of cloud connectivity and unavailability of notifications; complete discharge of an autonomous sensor in under 16 hours; exploitation of one protocol through another	[35]; [40]
Level	Vector	Observable effect	Confirmed by



Table 3 points to four recurring weaknesses. Features required by the specification can still fall short when compatibility modes are allowed; WPA3's transitional mode and optional BLE bonding illustrate this problem. Across all radio technologies, service requests also lack rate limiting. Even a strong communication channel offers little protection when the application protocol accepts commands without checking their freshness and authenticity. For low-power networks, the most serious issue is separate: encryption and integrity checks do not protect availability, leaving radio-physical attacks effective no matter how strong the algorithms are.

3.2. Recommendations for wireless infrastructure administrators

The recommendations are organized by who must act: infrastructure administrators (A-n), device and firmware developers (D-n), mobile and cloud service developers (S-n), monitoring and risk assessment teams (M-n), and testing-process owners (P-n). Table 4 covers the administrator recommendations.

Table 4. Recommendations for wireless infrastructure administrators

ID	Recommendation	Basis / evidence level
A-1	Migrate the corporate segment to WPA3-Enterprise with mandatory rather than transitional management frame protection; treat the compatibility mode as temporary with a documented decommissioning date	[33]; [14]; E2
A-2	Enforce rate limiting of authentication requests at the access point and at the RADIUS server, since effectiveness of availability attacks under WPA3 is governed by the resources consumed by a formally legitimate EAP procedure	[34]; [33]; E1
A-3	Enforce on client devices a policy of server certificate validation with pinning of the root certification authority and of the expected server name	[33]; E1
A-4	Use high-entropy passphrases where Personal mode remains in use and eliminate WEP entirely; a strong key remains an effective compensating control even when handshake capture cannot be prevented	[35]; [2]; E1
A-5	Segment the network and apply a zero-trust model: place IoT devices and gateways in dedicated virtual networks, micro-segment sensitive services and restrict outbound traffic to an allow list of cloud endpoints	[1]; [28]; E2
A-6	Deploy radio monitoring covering every technology in the band, including BLE and ZigBee contours, since such devices create no network identifier and stay outside the scope of conventional wireless intrusion detection	[36]; [2]; E1



ID	Recommendation	Basis / evidence level
A-7	Introduce continuous spectral monitoring of the 2.4 GHz band by software-defined radio with waterfall visualisation and logging, enabling detection of wideband interference, directed narrowband signals and atypical modulation	[40]; [1]; E1
A-8	Use a combined jamming indicator: simultaneous anomalous rise of received signal level and fall of successfully received packets; baseline values are –50 to –70 dBm at 20–50 packets per minute against –30 to –40 dBm at 0–5 packets per minute under jamming, with thresholds calibrated per installation	[2]; E1
A-9	Treat mass device unavailability as a security event: simultaneous transition of several devices to an unavailable state within 10–30 seconds must raise an alert rather than be logged as a routine fault; recovery times of 5–60 seconds must inform timeout settings	[2]; E1
A-10	Provide redundant notification paths for safety-critical sensors such as leak, smoke and tamper detectors, independent of the primary radio interface, or local on-device alarming	[2]; [40]; E2
A-11	Account for the predictability of the fixed frequency plan: sixteen channels spaced by 5 MHz and a passive scan time of about eight seconds preclude reliance on channel selection as concealment; consider adaptive channel hopping and spatial filtering	[17]; [30]; E2
A-12	Monitor the energy consumption of autonomous devices, treating abnormally rapid discharge as an indicator of an exhaustion attack, since external interaction can fully discharge a CR2450-powered node in under 16 hours	[40]; E1
A-13	Control network joining and key management: limit joining windows, maintain an inventory of legitimate nodes, detect unauthorised nodes with dedicated analysis tools, and consider distributed key management with perfect forward secrecy	[40]; [1]; E1
A-14	Account for client fleet heterogeneity, since devices that do not support the current protocol effectively define the lower bound of segment security	[34]; E1

3.3. Recommendations for device and firmware developers



Table 5. Recommendations for device and firmware developers

ID	Recommendation	Basis / evidence level
D-1	Require pairing, bonding and link encryption before granting access to any characteristic that changes device state	[36]; E1
D-2	Introduce application-level authentication of the control frame: a monotonic counter or nonce combined with a message authentication code over a session key; this is the only measure that eliminates confirmed frame replay irrespective of link encryption	[36]; E1
D-3	Bind the device to a trusted controller and reject unknown central devices at connection establishment rather than at command processing	[36]; E1
D-4	Validate length, value ranges and admissibility of every command against the current state of the device state machine	[36]; E1
D-5	Implement periodic controller presence confirmation and a watchdog with transition to a safe state on timeout or link loss	[36]; E1
D-6	Rate-limit writes to control characteristics and block repeated submission of an identical critical command within a defined window	[36]; E1
D-7	Use resolvable private addresses and minimise identifying fields in advertising packets	[36]; E1
D-8	Design for the possibility of connection slot occupation: provide priority pre-emption for the trusted controller or document the failure scenario and guarantee safe device behaviour within it	[36]; E3
D-9	Provide mandatory support for management frame protection and modern authentication modes in Wi-Fi products, since legacy-only devices force a downgrade across the whole segment	[34]; [35]; E1
D-10	Ensure resilience to load on the network interface: limit concurrent connections, prioritise control traffic and behave predictably when the cloud platform is unreachable	[35]; [16]; E1
D-11	Provide signed firmware updates with anti-rollback protection	[6]; E2

ID	Recommendation	Basis / evidence level
D-12	Replace static contactless credentials with tokens supporting mutual authentication and rolling codes	[35]; [11]; E1
D-13	Log connections, authorisation failures and anomalous control frames with export to an external monitoring system	[36]; E1
D-14	Monitor frame integrity and counter monotonicity: frames with a bad frame check sequence indicate either natural interference or deliberate interference, and a counter discontinuity indicates replay	[1]; [2]; E1
D-15	Minimise metadata leakage: frequency, volume and direction of exchange recoverable from link and network headers allow an activity map to be built and the coordinator to be identified; treat the coordinator as a node of elevated criticality and consider traffic shaping	[2]; [1]; E1
D-16	Provide hardware protection: disable or password-protect UART and JTAG debug interfaces, encrypt and sign the firmware image, use external cryptographic co-processors, apply tamper-evident enclosures and shielding of critical components	[40]; [1]; E2
D-17	Make firmware analysable before deployment: allow extraction of cryptographic indicators such as entropy and crypto-library references and of potentially dangerous API calls, and support protocol stack fuzzing that reveals unknown vulnerabilities without source code access	[1]; [2]; E2
D-18	Design for cryptographic agility in view of the long-term post-quantum threat, providing a migration path to standardised post-quantum schemes within the computational limits of constrained devices	[26]; [1]; E3

3.4. Recommendations for service developers, monitoring and the testing process

Table 6. Recommendations for mobile and cloud services (S), monitoring and risk assessment (M) and the testing process (P)



ID	Recommendation	Basis / evidence level
S-1	Minimise requested permissions and the set of exported components; protect components that must be exported with signature-level permissions	[4]; [22]; E1
S-2	Use current application package signature schemes and abandon v1-only signing, which permits content substitution	[35]; E1
S-3	Eliminate logging of sensitive data and hard-coded keys, tokens and configurations in application code	[35]; [4]; E1
S-4	Apply TLS with certificate pinning on the application-to-cloud and application-to-device channels and verify management interfaces against the OWASP methodology	[1]; [29]; E1
S-5	Rate-limit and harden device and cloud application programming interfaces, providing graceful degradation with explicit user notification	[35]; [16]; E1
S-6	Follow the security-by-design principle: embed protective mechanisms from the design stage, mandate component updates, control the provenance of third-party libraries and remove insecure interfaces and outdated software	[40]; [6]; E2
M-1	Deploy machine-learning traffic anomaly detection on the gateway; a random forest over a 61-dimensional feature vector reached 98.64 % accuracy and F1 = 0.986 with about 300 ms latency and 20 % processor load on an edge device	[1]; [7]; E1
M-2	Complement the network detector with passive radio capture: an SDR pipeline recording to PCAP together with diagnostics of packet counts, per-device signal strength and delivery ratio increases detection completeness	[1]; [2]; E1
M-3	Apply an integral dynamic risk score only after calibration on deployment-specific data. Document the formula, available evidence branches, weights, score version, and alert threshold. The initial weights of 0.4, 0.4, and 0.2 and the threshold of 0.7 are provisional values inherited from the source model and are not universal deployment defaults.	[1]; E3; E3
M-4	Analyse user behaviour only under documented privacy safeguards. Define the minimum necessary event fields, retention period, access controls, and review purpose	[15]; [12]; [1]; E3; E3



ID	Recommendation	Basis / evidence level
	before modelling action sequences. Differential privacy or another privacy-preserving mechanism should be selected and calibrated for the intended analytical task.	
M-5	Evaluate detector robustness using a separately defined validation set, subgroup false-positive analysis, and adversarial or shifted samples. Select the operational threshold according to the documented cost of false positives and missed attacks rather than adopting one universal F1 target.	[1]; [2]; E3; E3
P-1	Plan the audit as cross-layer, since radio-only assessment leads to a false conclusion of security	[35]; [36]; E1
P-2	Include availability scenarios in the test programme, not only confidentiality; for devices controlling physical processes and for alarm sensors, denial of service is the highest-criticality threat	[40]; [28]; E1
P-3	Apply adaptive search for effective action combinations, since a Q-learning agent discovered non-coinciding effective sets for WPA2 and WPA3	[34]; [32]; E1
P-4	Record the testbed configuration and client fleet composition as part of the result, because effectiveness depends strongly on client stack behaviour	[34]; E1
P-5	Re-verify after implementation using the same toolset and metrics as the initial audit	This study; E3
P-6	Test controllers of moving objects exclusively on a stationary bench with actuators disabled, documenting the limitations of the trial	[36]; E1
P-7	Use the reproducible five-stage radio-physical procedure: environment preparation, active channel identification, test interference generation, observation of device reaction with mobile application event logging, and spectral analysis	[2]; [40]; E1
P-8	Observe legal and technical requirements for interference trials: shielded room, minimal initial power with stepwise increase, safety briefing and compliance with radio spectrum legislation	[2]; E2



ID	Recommendation	Basis / evidence level
P-9	Train specialists on real equipment and software-defined radio rather than on simulators alone, covering physical, link, network and application layers	[2]; E1

3.5. Verification matrix and metrics

The practical value of the recommendation catalogue rests on two-sided traceability: every control is linked to an implementation requirement and to a verification activity. Table 7 presents the principal attack-vector-to-control mappings and their reusable acceptance tests. Table 9 supplements it with control-specific checks for recommendations that are not directly represented in Table 7. Together, the tables define a proposed testing-remediation-re-verification loop; they do not imply that every control has already been implemented and validated on an independent industrial testbed.

Table 7. Verification matrix: vector, recommendations and acceptance criterion

Attack vector	Measures	Proposed acceptance criteria	Verification tool
Jamming of the 2.4 GHz band	A-7, A-8, A-9, A-10, A-11, M-2	The jamming event is registered by the monitoring system within the defined time; critical alerts are delivered over the redundant path	HackRF One, GNU Radio, CatSniffer, diagnostics module
Deauthentication and management frame flooding	A-1, A-14, D-9	Share of episodes with complete denial of service equals zero; session recovery time does not exceed the threshold	mdk4, aireplay-ng, airodump-ng, adaptive agent
Authentication denial of service and EAPOL-Start flooding	A-2, D-10	No degradation of service for legitimate clients at the specified interference intensity	mdk4, RADIUS load testing tools
Evil Twin and MITM	A-3, S-4	The client rejects association with an access point presenting an untrusted certificate without user involvement	airgeddon, Wireshark
Handshake capture and offline recovery	A-4, A-1	WEP is disabled; transitional modes are absent unless formally justified; the credential satisfies the approved organizational	aircrack-ng, airodump-ng



Attack vector	Measures	Proposed acceptance criteria	Verification tool
		password policy; and no credential is recovered during the documented offline resistance test. The wordlist result is supporting evidence, not proof of universal password strength.	
Unauthorised GATT access and frame replay	D-1, D-2, D-3, D-6, D-14	A state-changing GATT write is rejected unless an authenticated and encrypted connection with the trusted controller is established; a captured frame is rejected on replay; and the freshness counter remains monotonic.	Bleak, nRF Connect, hardware BLE sniffer, KillerBee
Unauthorised node joining and key compromise	A-13, D-14	Joining outside the permitted window is rejected; an unauthorised node is detected by analysis tools	KillerBee, ZigBee analysis tools
Tracking and metadata leakage	D-7, D-15, A-6	The device identifier changes within the specified interval; the coordinator is not identified from the traffic profile within the observation window	SDR capture to PCAP, Wireshark
Battery exhaustion	A-12, D-10	Discharge rate under external interaction does not exceed the threshold and the anomaly is recorded	Diagnostics module, charge log
Network interface denial of service	D-10, S-5	Local functions and logging are preserved; the user is informed of connectivity loss	hping3, Wireshark
Mobile and web interface defects	S-1, S-2, S-3, S-4, S-6	No excess dangerous permissions, unprotected exported components or secrets in code; certificate pinning is active	MobSF, Jadx, Apktool, OWASP checks
Firmware defects and debug interfaces	D-11, D-16, D-17	Debug interfaces are unavailable; the image is signed; rollback is blocked	Image extraction and



Attack vector	Measures	Proposed acceptance criteria	Verification tool
			analysis, fuzzing
Contactless credential cloning	D-12	A captured credential dump does not grant access on replay	RFID/NFC tooling
Detector evasion	M-1, M-5	Performance meets a threshold selected on an independent validation set, and subgroup false-positive rates remain within documented operational limits under adversarial or shifted samples.	Adversarial sample generation, retraining

Table 8. Verification metrics

ID	Metric	Target value and interpretation
K1	Share of episodes ending in complete denial of service	Zero; a non-zero value indicates reproducibility of full outage in the given configuration
K2	Client session recovery time after interference ceases	Define the deployment-specific threshold before testing and report the mean and 95th percentile. The 5–60 s range is a source-laboratory observation, not a universal limit.
K3	Attack success rate across the predefined action combinations	Zero action sequence reaches the complete denial-of-service condition after mitigation; also report the change from the baseline success rate.
K4	Number of successful unauthorised writes to a control characteristic	Zero when an external central device attempts to connect
K5	Share of accepted replayed frames	Zero when a counter and message authentication code are present
K6	Number of successful unauthenticated or unencrypted state-changing GATT writes	Zero. Sensitive writes are permitted only after authenticated and encrypted establishment with the trusted controller.
K7	Number of unprotected exported components and secrets found in the mobile application	Zero according to static analysis



ID	Metric	Target value and interpretation
K8	Availability of designated safety-critical local functions and security logging under network-interface load	All functions designated as safety critical remain available, and the load event and any degraded service are recorded in the security log.
K9	Packet delivery ratio on the operating channel under interference of a defined power	Report the absolute packet delivery ratio and degradation from baseline. The pass threshold is defined before testing; critical sensors retain delivery through the redundant path.
K10	Time to detect a jamming event by the monitoring system	Does not exceed the locally configured transition time to an unavailable state. The 10–30 s range is provisional and requires site-specific calibration.
K11	F1 score and subgroup false-positive rates of the gateway detector on independent validation and adversarial samples	The acceptable threshold is selected and documented using an independent validation set and operational error costs. The previously reported F1 = 0.986 is a source-study baseline, not a universal target.
K12	Calibrated integral risk score	The score formula, input branches, weights, version, and threshold are documented; weights sum to 1; and calibration is performed on deployment-specific data. The value 0.7 is provisional.

Table 9. Extended traceability for recommendations not directly represented in Table 7

ID	Evidence	Acceptance test	Pass condition	Verification artifact or tool
A-5	E2	Attempt unauthorized communication between the IoT segment, corporate segment, and non-allow-listed external endpoint.	Unauthorized east-west and outbound traffic is blocked while approved services remain available.	Firewall/VLAN configuration; network scan
D-4	E1	Submit malformed, truncated, oversized, and out-of-range commands in	All invalid commands are rejected without	Protocol client; fuzzing



ID	Evidence	Acceptance test	Pass condition	Verification artifact or tool
		each reachable device state.	unauthorized state change or crash.	harness; device log
D-5	E1	Interrupt the controller link and suppress keep-alive messages.	The device enters its documented safe state within the pre-defined timeout.	Controller emulator; event log; physical observation
D-8	E3	Occupy the available BLE connection slot with an unknown central device.	The device remains safe and the trusted controller can regain control according to the documented recovery policy.	BLE clients; connection and recovery test
D-13	E1	Generate failed authorization, replay, and anomalous control-frame events.	Each event is logged with time and device identity and is exportable to the monitoring system.	Device logs; SIEM or log collector
D-18	E3	Review the cryptographic architecture and perform a controlled algorithm or library substitution.	Algorithms and parameters are replaceable without redesign of core application logic, and a migration plan is documented.	Architecture and code review; build test
M-3	E3	Recalculate risk scores from documented test cases using the released formula and configuration.	Results are reproducible; available evidence branches, weights, score version, and calibrated threshold are recorded.	Unit tests; calibration dataset; configuration record
M-4	E3	Audit behavioral data collection, retention,	Only purpose-limited fields are collected; retention	Privacy impact



ID	Evidence	Acceptance test	Pass condition	Verification artifact or tool
		access, and privacy controls.	and access are documented; privacy parameters are reproducible.	assessment; log and access audit
P-1	E1	Review the audit plan against the cross-layer checklist.	Physical, link, network, application, cloud, and management surfaces relevant to the product are covered or explicitly excluded with justification.	Audit plan and checklist
P-2	E1	Review and execute at least one authorized availability scenario for each critical communication path.	Availability impact and recovery are measured, and safety controls remain effective.	Approved test plan; event and recovery logs
P-3	E1	Execute the documented adaptive action set with fixed seeds, reward definition, and stopping rules.	Runs are reproducible and the discovered effective actions are recorded; superiority over baselines is not claimed without comparison.	Adaptive tool; run configuration and logs
P-4	E1	Inspect the final test report for environment metadata.	Hardware models, client types, firmware, software versions, radio settings, and test conditions are recorded.	Report checklist
P-5	E3	Repeat the original test after remediation under equivalent documented conditions.	All pre-defined acceptance criteria are met, or residual failures are reported and	Original toolset; before-and-after report



ID	Evidence	Acceptance test	Pass condition	Verification artifact or tool
			accepted by the risk owner.	
P-6	E1	Review moving-object test preparation before command testing.	The object is stationary, actuators are disabled or physically isolated, and functional-safety limitations are documented.	Safety checklist; test log
P-7	E1	Audit the radio-physical test record for the five required stages.	Preparation, channel identification, controlled interference, reaction observation, and spectral analysis are all documented.	Test record; SDR and application logs
P-8	E2	Verify authorization, isolation, power-control, and spectrum-compliance records before interference testing.	Written authorization is present; the environment is isolated; transmission power is logged and increased stepwise.	Authorization and safety checklist; power log
P-9	E1	Conduct a practical competency assessment using approved laboratory equipment.	The specialist completes the assigned capture, analysis, mitigation, and reporting tasks without violating safety constraints.	Laboratory rubric; assessor record

Note. E1 indicates direct support from the summarized experiments, E2 indicates support from external standards or independent literature, and E3 indicates proposed guidance requiring further validation. A pass condition is operational only after the applicable test environment and threshold have been defined in advance.

The adaptive audit tool calculates K1–K3 and includes them in its report. Protocol-analyser filters and controlled GATT tests provide K4–K6, while the laboratory diagnostics module records K9 and K10. The detection workflow produces K11 and supports K12 when the risk model has been calibrated. The remaining controls use the qualitative and quantitative evidence specified in Table 8. These mappings reduce the need for separate test rigs, but successful implementation still requires configuration-specific thresholds and documented equivalence between baseline and re-test conditions.

3.6. Discussion

The recommendations point to a conclusion also reached independently in [33] and supported by [3] and [8]. For wireless and IoT systems in this class, real security depends mainly on implementation and configuration, not on the protocol's stated strength. Moving to WPA3 changes the attacks that remain practical, but it does not eliminate threats to availability. BLE presents the same problem: its specification includes full pairing mechanisms, yet that protection means little when a product leaves them disabled.

This distinction matters most in low-power networks. The findings in [40] show why: cryptographic protection and availability protection address separate problems, since a low-power tone jammer can disable the network without compromising any key material. For smart-home alarms and safety functions, redundant notification paths (A-10) therefore deserve the same priority as cryptographic safeguards, as does treating widespread unavailability as a security event (A-9).

Design organisations should place some protective functions in the device protocol's application layer. D-2, D-4, and D-5 still work when the channel has been completely compromised, and they do not rely on how the customer's network is configured. Monitoring measures M-1 to M-5 connect the initial audit with day-to-day operations: test results become thresholds for ongoing monitoring.

The study has six principal limitations. First, the source experiments used small testbeds: one router and three clients for adaptive Wi-Fi, one device from each class for the smart-lock and quadcopter studies, and one hub with one sensor on a single channel for the jamming study. These conditions limit statistical strength and external validity. Second, the adaptive agent was not compared with random or fixed-script baselines, so the specific gain from reinforcement learning cannot be quantified. Third, detector performance was evaluated with a general-purpose intrusion-detection dataset [25] rather than a ZigBee-specific traffic corpus, while the firmware and user-behavior branches of the risk model remain conceptual. Fourth, the reported RSSI, packet-rate, recovery-time, risk-score, and alert thresholds are provisional laboratory values requiring deployment-specific calibration. Fifth, the recommendation catalogue is a structured author-led mapping rather than a systematic review, independently coded consensus, or completed industrial validation; E3 recommendations in particular require further evaluation. Sixth, the moving-object trials did not involve physical flight, limiting assessment of functional-safety consequences. The internal technical report used for part of the BLE evidence also limits external reproducibility until its methods and artifacts are made publicly accessible.

4. Conclusion



This study synthesized seven previously completed experimental investigations of wireless-network and IoT-device penetration testing. The source work covered Wi-Fi, Bluetooth Low Energy, and ZigBee, including adaptive auditing, WPA3-Enterprise attacks, cross-layer assessment of a commercial smart lock, BLE control-channel analysis, machine-learning-assisted ZigBee assessment, software-defined-radio jamming, and laboratory security tasks. The primary experimental findings remain attributable to the cited publications; the present contribution is their normalization and integration within a six-level cross-layer taxonomy.

The taxonomy supports 52 recommendations for five stakeholder groups. The recommendations are assigned evidence levels and linked through the principal and extended traceability matrices to proposed acceptance activities, metrics, pass conditions, and verification tools. The work also defines a five-stage testing-remediation-re-verification procedure and twelve core metrics. These elements form an evidence-informed operational framework; they should not be interpreted as proof that every recommendation or numerical threshold has been independently validated across industrial deployments.

The main applied conclusion is that application-layer safeguards—including command authentication, freshness checks, state validation, and safe behavior after link loss—remain important when link-layer controls fail or are misconfigured. Availability also requires controls distinct from cryptography, including spectrum monitoring, combined jamming indicators, backup notification paths for critical functions, and security treatment of widespread device unavailability. Two-sided traceability makes these controls auditable by connecting each requirement to explicit verification evidence.

Future work will validate the catalogue through independent expert review and industrial pilot deployments, report agreement between reviewers who classify vectors and evidence levels, and compare the adaptive agent with random and fixed-script baselines. It will also broaden the device population, produce a dedicated ZigBee traffic dataset, implement the firmware and behavioral branches of the risk model, publish the BLE technical report and supporting artifacts, and calibrate the metrics on deployment-specific data. Extensions to Thread, Matter, and LoRaWAN will be considered only after the present framework has been evaluated on compatible observables.

Thanks: The authors thank the staff of the Scientific Research Institute of Information Security and Cryptology of L.N. Gumilyov Eurasian National University for support in preparing the laboratory testbeds.

Financing. This research was funded by the Committee of Science of the Ministry of Science and Higher Education of the Republic of Kazakhstan (Grant No. BR24992852, ‘Intelligent models and methods of Smart City digital ecosystem for sustainable development and the citizens’ quality of life improvement’)

Conflict of interest: The authors declare that they have no conflicts of interest.

References



1. Aidynov, T., Tleuberdin, S., Nurusheva, A., Satybaldina, D., Altaibek, M., Boranbay, Z. (2025). Cross-layer security assessment in IoT ecosystems: Analyzing vulnerabilities in the Zigbee protocol. In 2025 IEEE European Technology and Engineering Management Summit (E-TEMS), 231–236. <https://doi.org/10.1109/E-TEMS64751.2025.11239104>
2. Aidynov, T., Tleuberdin, S., Boranbay, Z., Tebueva, F., Shaikhanova, A., Satybaldina, D. (2026). Smart home security: Tools, protocols and laboratory tasks for Internet of Things penetration testing. *EKTU Journal of Information and Communication Sciences*, 1(2), 18–33. https://doi.org/10.51885/3134-8025_JICS_2026_2_2
3. Allen, A., Mylonas, A., Vidalis, S., Gritzalis, D. (2024a). Smart homes under siege: Assessing the robustness of physical security against wireless network attacks. *Computers & Security*, 139, 103687.
4. Allen, A., Mylonas, A., Vidalis, S., Gritzalis, D. (2024b). Security evaluation of companion Android applications in IoT: The case of smart security devices. *Sensors*, 24(17), 5465.
5. Bluetooth SIG. (2023). Bluetooth core specification (Version 5.4). Bluetooth Special Interest Group.
6. Bour, G., Bosco, C., Ugarelli, R., Jaatun, M. G. (2023). Water-tight IoT—just add security. *Journal of Cybersecurity and Privacy*, 3(1), 76–94.
7. Breiman, L. (2001). Random forests. *Machine Learning*, 45(1), 5–32.
8. Caballero-Gil, C., Álvarez, R., Hernández-Goya, C., Molina-Gil, J. (2024). Research on smart-locks cybersecurity and vulnerabilities. *Wireless Networks*, 30(6), 5905–5917.
9. Chu, G., Lisitsa, A. (2018). Penetration testing for Internet of Things and its automation. In 2018 IEEE 20th International Conference on High Performance Computing and Communications, 1479–1484.
10. Connectivity Standards Alliance. (2017). Zigbee specification (Revision 22 1.0).
11. Das Dev, A., Shyam, R. (2024). Flipper Zero in action: A comparative review of six methods for enhanced cybersecurity tactics. *International Journal of Research Publication and Reviews*, 5(9), 1317–1321.
12. Dwork, C., Roth, A. (2014). The algorithmic foundations of differential privacy. *Foundations and Trends in Theoretical Computer Science*, 9(3–4), 211–407.
13. Færøy, F., Yamin, M., Shukla, A., Katt, B. (2023). Automatic verification and execution of cyber attack on IoT devices. *Sensors*, 23(2), 733.
14. Halbouni, A., Ong, L.-Y., Leow, M.-C. (2023). Wireless security protocols WPA3: A systematic literature review. *IEEE Access*, 11, 112438–112450.



15. Hamilton, W. L., Ying, R., Leskovec, J. (2017). Inductive representation learning on large graphs. In *Advances in Neural Information Processing Systems* 30, 1025–1035.
16. Huraj, L., Šimon, M., Lietava, J. (2024). Dataset of DDoS attacks on Fibaro Home Center 3 for smart home security. *Data in Brief*, 57, 110991.
17. IEEE. (2020). IEEE standard for low-rate wireless networks (IEEE Std 802.15.4-2020).
18. IEEE. (2021). IEEE standard for information technology—Part 11: Wireless LAN medium access control and physical layer specifications (IEEE Std 802.11-2020).
19. IoT Analytics. (2024). State of IoT 2024: Number of connected IoT devices growing 13% to 18.8 billion globally. <https://iot-analytics.com/number-connected-iot-devices/>
20. Jang, B., Kim, M., Harerimana, G., Kim, J. W. (2019). Q-learning algorithms: A comprehensive classification and applications. *IEEE Access*, 7, 133653–133667.
21. Janisch, J., Pevný, T., Lisý, V. (2023). NASimEmu: Network attack simulator and emulator for training agents generalizing to novel scenarios. In *European Symposium on Research in Computer Security*, 589–608. Springer.
22. Kusreynada, S. U., Barkah, A. S. (2024). Android apps vulnerability detection with static and dynamic analysis approach using MobSF. *Journal of Computer Science and Engineering*, 5(1), 46–63.
23. Li, Q., Zhang, M., Shen, Y., Wang, R., Hu, M., Li, Y., Hao, H. (2023). A hierarchical deep reinforcement learning model with expert prior knowledge for intelligent penetration testing. *Computers & Security*, 132, 103358.
24. Modesti, P., Golightly, L., Holmes, L., Opara, C., Moscini, M. (2024). Bridging the gap: A survey and classification of research-informed ethical hacking tools. *Journal of Cybersecurity and Privacy*, 4(3), 410–448.
25. Moustafa, N., Slay, J. (2015). UNSW-NB15: A comprehensive data set for network intrusion detection systems. In *Military Communications and Information Systems Conference*, 1–6.
26. National Institute of Standards and Technology. (2024). Module-lattice-based key-encapsulation mechanism standard (FIPS 203).
27. National Security Agency. (2019). Cybersecurity technical report: WPA3. <https://media.defense.gov/2019/Jul/16/2002158109/-1/-1/0/CTR-CYBERSECURITY-TECHNICAL-REPORT-WPA3.PDF>
28. Neshenko, N., Bou-Harb, E., Crichigno, J., Kaddoum, G., Ghani, N. (2019). Demystifying IoT security: An exhaustive survey on IoT vulnerabilities and a first empirical look on Internet-scale IoT exploitations. *IEEE Communications Surveys & Tutorials*, 21(3), 2702–2733.



29. OWASP Foundation. (2024). OWASP Mobile Top 10. <https://owasp.org/www-project-mobile-top-10/>
30. Pirayesh, H., Zeng, H. (2022). Jamming attacks and anti-jamming strategies in wireless networks: A comprehensive survey. *IEEE Communications Surveys & Tutorials*, 24(2), 767–809.
31. Sarker, K. U., Yunus, F., Deraman, A. (2023). Penetration taxonomy: A systematic review on the penetration process, framework, standards, tools, and scoring methods. *Sustainability*, 15(13), 10471.
32. Schwartz, J., & Kurniawati, H. (2019). Autonomous penetration testing using reinforcement learning. *arXiv:1905.05965*.
33. Tleuberdin, S., Issainova, A., Adamova, A., Aidynov, T., Samashova, G., Satybaldina, D. (2025). Analysis of vulnerabilities and practical attacks on WPA3-Enterprise in corporate wireless networks. *Procedia Computer Science*, 272, 613–618.
34. Tleuberdin, S., Malakhov, K., Tashatov, N., Satybaldina, D., Yedilkhan, D. (2026). Reinforcement learning-based adaptive penetration testing framework for wireless communication. *International Journal of Advanced Computer Science and Applications*, 17(3), 355–363.
35. Tleuberdin, S., Satybaldina, D., Muratkhan, R., Abisheva, G. (2026). Security audit of Tuya smart lock using penetration testing methodology. *Computers, Materials & Continua*. <https://doi.org/10.32604/cmc.2026.081906>
36. Tleuberdin, S., Satybaldina, D. (2026). Laboratory analysis of the BLE control channel of an ESP32-S3 quadcopter: Discovery, interception, frame replay and software reproduction of control (Technical report). Research Institute of Information Security and Cryptology, L.N. Gumilyov Eurasian National University.
37. Vanhoef, M. (2021). Fragment and forge: Breaking Wi-Fi through frame aggregation and fragmentation. In *30th USENIX Security Symposium*, 161–178.
38. Vanhoef, M., Piessens, F. (2017). Key reinstallation attacks: Forcing nonce reuse in WPA2. In *Proceedings of the 24th ACM SIGSAC Conference on Computer and Communications Security*, 1313–1328.
39. Vanhoef, M., Ronen, E. (2020). Dragonblood: Analyzing the Dragonfly handshake of WPA3 and EAP-pwd. In *2020 IEEE Symposium on Security and Privacy*, 517–533.
40. Zhukabayeva, T., Adamova, A., Boranbay, Z., Benkhelifa, E., Mardenov, Y. (2025). Analyzing the security of ZigBee communication in smart environments via SDR. *Bulletin*



of L.N. Gumilyov ENU. Mathematics, Computer Science, Mechanics Series, 152(3), 6–21.
<https://doi.org/10.32523/bulmathenu.2025/3.1>

41. Zhukabayeva, T., Ahmad, Z., Adamova, A., Karabayev, N., Mardenov, Y., Satybaldina, D. (2025). Penetration testing and machine learning-driven cybersecurity framework for IoT and smart city wireless networks. IEEE Access, 13, 86144–86166.

Information about authors

Saken Tleuberdin – PhD
Research Institute of Information Security and
Cryptography, L.N. Gumilyov Eurasian National
University, Astana, Kazakhstan.
E-mail: sakentleuberdin@gmail.com
ORCID: <https://orcid.org/0000-0001-9170-9936>

Zhandos Boranbay – doctoral student
School of Cybersecurity, Astana IT University,
Astana, Kazakhstan.
E-mail: boranbai_zha_3@enu.kz
ORCID: <https://orcid.org/0009-0009-0496-3876>

Konstantin Malakhov – master's degree in
Cybersecurity
Research Institute of Information Security and
Cryptography, L.N. Gumilyov Eurasian National
University, Astana, Kazakhstan.
E-mail: malakhov04const@gmail.com
ORCID: <https://orcid.org/0009-0004-5695-2243>

Tolegen Aidynov – PhD
Research Institute of Information Security and
Cryptography, L.N. Gumilyov Eurasian National
University, Astana, Kazakhstan.
E-mail: 951005351309@enu.kz
ORCID: <https://orcid.org/0009-0005-3327-9719>