Wi-Fi in Constrained Devices

A Literature Review

A Comprehensive Review of Energy Consumption, Security, Performance, and Administration Challenges for IEEE 802.11 in Resource-Constrained Devices

Tom Carpenter, CWNP Director

March 2026, Draft Copy

Contents

Abstract: This literature review examines research on deploying IEEE 802.11 (Wi-Fi) in constrained devices, including IoT sensor nodes, embedded systems, and edge devices, that operate under severe limitations in processing power, memory, energy, and bandwidth. Organized around four focus domains, energy consumption, security, performance, and administration, the review synthesizes findings from over sixty publications (2015–2026), encompassing analytical models, simulations, and empirical testbed evaluations. It begins by establishing a definitional framework based on RFC 7228, which classifies devices into three resource tiers and standardizes terminology for energy supply and power management. Subsequent sections examine how Wi-Fi standards, such as IEEE 802.11ah (HaLow), 802.11ax (Wi-Fi 6), 802.11be (Wi-Fi 7) and the emerging 802.11bn (Wi-Fi 8), have evolved to accommodate constrained devices through mechanisms such as Target Wake Time, Restricted Access Window, Wake-up Radio, and OFDMA scheduling. The security analysis reveals persistent challenges in authentication overhead, management frame vulnerabilities, and the tension between cryptographic rigor and resource scarcity, while the performance section documents gaps between theoretical and measured throughput and the scalability limits of current MAC-layer mechanisms. The administration section evaluates emerging protocols and architectures, including OMA LwM2M, SDN-based management, and automated provisioning, aimed at enabling scalable, energy-efficient device management. The review concludes by identifying cross-cutting themes and future directions for the next generation of constrained Wi-Fi deployments.

Introduction

The widescale use of Internet of Things (IoT) devices has changed the world of wireless networking. Billions of devices, ranging from environmental sensors and industrial monitors to medical wearables and smart home controllers, now require reliable Internet connectivity, and IEEE 802.11 (Wi-Fi) has emerged as a leading candidate technology for connecting these devices. Unlike purpose-built low-power wide-area network (LPWAN) technologies such as LoRa or NB-IoT, Wi-Fi offers the advantages of ubiquitous infrastructure deployment, IP-native communication, high data rates, and mature ecosystem support. However, the application of Wi-Fi to constrained devices introduces a complex set of engineering challenges that the original standard was never designed to address.

Constrained devices, as formally defined by the Internet Engineering Task Force in RFC 7228, operate under tight limits on CPU, memory, and power resources that render many conventional networking assumptions invalid (Bormann, Ersue and Keranen,

2014). A Class 0 and 1 device with fewer than or equal to 10 KiB[1] of RAM cannot execute a full TLS handshake; a coin-cell-powered sensor node cannot sustain the idle listening energy drain of a continuously active Wi-Fi radio. These fundamental mismatches between Wi-Fi's design assumptions and the operational realities of constrained devices have catalyzed a decade of research spanning PHY-layer (Physical Layer) innovations, MAC-layer power management, lightweight security protocols, and scalable device management architectures. Standards bodies have responded with targeted amendments: IEEE 802.11ah (Wi-Fi HaLow) introduced sub-1 GHz (S1G) operation with specialized mechanisms that benefit IoT, IEEE 802.11ax (Wi-Fi 6/HE) brought Target Wake Time and OFDMA scheduling to traditional Wi-Fi, and IEEE 802.11be (Wi-Fi 7/EHT) and IEEE 802.11bn (Wi-Fi 8/UHR)[2] promise further energy efficiency and deterministic latency improvements.

This literature review surveys the academic research addressing the deployment of Wi-Fi and Wi-Fi HaLow in constrained devices, organized around four primary focus domains: energy consumption and battery life, security, performance, and administration and management. For each domain, the review examines the methodologies, findings, and contributions of key publications from 2015 through 2026, identifies recurring themes and research gaps, and synthesizes the state of knowledge with a view to the future as well as the present state. The review draws on a corpus of over fifty papers sourced from the IETF, IEEE, ACM, MDPI, Elsevier, and arXiv, selected for their direct relevance to Wi-Fi in constrained device contexts. By integrating findings across these domains, the review aims to provide a comprehensive reference for students, engineers and administrators working at the intersection of Wi-Fi networking and constrained devices and networks.

Throughout this document, the term Wi-Fi will be used to indicate the traditional Wi-Fi devices that implement the DSSS, HR/DSSS, OFDM, ERP, HT, VHT, HE, and EHT PHYs. At times, the specific amendment, such as IEEE 802.11n for HT, will be referenced. When other IEEE 802.11 PHYs, such as the Sub-1 GHz (S1G) PHY is intended, it will be called out, typically as IEEE 802.11ah or Wi-Fi HaLow.

[1] One kibibyte (KiB) is 1024 bytes and a byte is synonymous with an octet in RFC 7228. Therefore, 10 KiB is 10,240 bytes. KiB is often used to be more precise as kilobyte (KB) in various contexts can mean either 1000 bytes or 1024 bytes. KiB always means 1024 bytes and a byte is always 8 bits in modern systems (though this latter statement has not always been historically true).

[2] The IEEE 802.11 PHYs have names. 802.11-prime is DSSS (Direct Sequence Spread Spectrum). 802.11a is OFDM (Orthogonal Frequency Division Multiplexing). 802.11b is HR/DSSS (High Rate-Direct Sequence Spread Spectrum). 802.11g is ERP (Extended Rate PHY). 802.11n is HT (High Throughput). 802.11ac is VHT (Very High Throughput). 802.11ax is HE (High Efficiency). 802.11be is EHT (Extremely High Throughput). Finally, 802.11bn is UHR (Ultra-High Reliability).

Constrained Devices Defined

Before examining the specific challenges of deploying Wi-Fi in resource-constrained environments, it is essential to establish a precise definition and understanding of what constitutes a constrained device and the taxonomy that governs how such devices are classified. The Internet Engineering Task Force (IETF) published RFC 7228[3], "Terminology for Constrained-Node Networks," in May 2014 to provide the networking community with a shared vocabulary for discussing devices and networks that operate under significant resource limitations (Bormann, Ersue and Keranen, 2014). This section presents the key definitions, device classifications, network constraints, and energy terminology introduced by RFC 7228, which forms the foundational framework for the remainder of this review.

Definition and Characteristics of Constrained Nodes

RFC 7228 defines a constrained node as "a node where some of the characteristics that are otherwise pretty much taken for granted for Internet nodes at the time of writing are not attainable, often due to cost constraints and/or physical constraints on characteristics such as size, weight, and available power and energy" (Bormann, Ersue and Keranen, 2014). This definition is deliberately broad, recognizing that the specific nature and severity of constraints vary widely across device categories. Additionally, as will be seen later, the classes of devices defined in RFC 7228 are not necessarily fixed to hardware specifications, but serve as a scalable guide as to what is constrained in the current context of available hardware[4]. However, the specification identifies several facets that are most commonly constrained: ROM or Flash memory, which limits code complexity and the size of the protocol stack that can be deployed; RAM, which limits the amount of state, buffering, and dynamic data structures the device can maintain; processing power, which limits the computational complexity of protocols and algorithms; available power, which may be drawn from batteries, energy harvesters, or mains supplies; and user interface accessibility, which affects the ability to configure and manage devices (Bormann, Ersue and Keranen, 2014).

The tight limits on these resources lead to hard upper bounds on the state, code space, and processing cycles available to networking software. For many constrained devices, these bounds are not merely performance guidelines but absolute physical constraints.

[3] RFC 7228 is available online at https://datatracker.ietf.org/doc/html/rfc7228.

[4] This time window concern is very important. Remember that RFC 7228 was written in 2014, twelve years before this writing. So, classifying devices based on less than 10 KiB, approximately 10 KiB, and approximately 50 KiB, may not be as relevant today (though those devices would certainly still be considered constrained). The numbers may have risen for these specifications, but they are still constrained in the context of the overall hardware available.

A device with 10 KiB of RAM cannot allocate 12 KiB for a TLS handshake buffer, regardless of software optimization. Similarly, a device powered by a non-replaceable coin-cell battery cannot sustain an always-on radio that draws 100 mA continuously for long time periods, regardless of how efficient the protocol stack is. These hard constraints fundamentally shape the design space for Wi-Fi protocols, security mechanisms, and management architectures for constrained devices.

Classes of Constrained Devices

To facilitate meaningful comparison and design targeting, RFC 7228 introduces a classification system that groups constrained devices into three classes based on their memory resources[5]. As defined in RFC 7228 and Table 1, Class 0 (C0) devices, as defined in the RFC, possess substantially less than 10 KiB of RAM and less than 100 KiB of Flash memory. Class 1 (C1) devices have approximately 10 KiB of RAM and approximately 100 KiB of Flash. Class 2 (C2) devices offer approximately 50 KiB of RAM and approximately 250 KiB of Flash (Bormann, Ersue and Keranen, 2014). These boundaries are approximate and are intended to capture broad categories rather than precise thresholds and should evolve over time based on hardware availability with the hope that hardware will continually become less constrained while also consuming less energy.

The practical implications of these classes for Wi-Fi networking are profound. Class 0 devices are so severely constrained that they typically cannot communicate directly with the Internet in a secure manner. They lack the memory and processing resources to run even lightweight IP stacks and generally require proxies or gateways to mediate their communication. In the context of Wi-Fi, a Class 0 device would be unable to complete a secure IEEE 802.11 connection sequence independently, as the WPA2 or WPA3 four-way handshake alone typically requires several KiBs of state, cryptographic, and other memory-stored values and code, exceeding their capabilities (Bormann, Ersue and Keranen, 2014).

[5] I emphasize again, these memory/RAM values may have changed today, but we can still benefit from classifying today's hardware into the three classes so that we can then choose the right hardware for our needs or implement the right protocol stack for our hardware.

Class Name	Data Size (RAM)	Code Size (Flash Memory)
Class 0 (C0)[6]	<< 10 KiB	<< 100 KiB
Class 1 (C1)	~10 KiB	~100 KiB
Class 2 (C2)	~50 KiB	~250 KiB
Class 3 (C3)	~100 KiB	~500-1,000 KiB
Class 4 (C4)	~300-1,000 KiB	~1,000-2,000 KiB

Table 1: RFC 7228 Constrained Device Classes Summarized (Based on 7228bis-2026[7])

Class 1 devices can use constrained protocol stacks such as CoAP (Constrained Application Protocol) over UDP but generally cannot run a full HTTP stack with TLS in the RAM provided. These devices represent the primary target for lightweight Wi-Fi protocols such as those specified in IEEE 802.11ah, where reduced frame sizes, simplified association procedures, and targeted power-saving mechanisms have been designed to accommodate their limitations. Class 1 devices can participate in Wi-Fi networks if the protocol overhead is carefully managed, but they remain unable to support the full range of networking capabilities expected of conventional Wi-Fi clients, such as laptop and desktop computers (Bormann, Ersue and Keranen, 2014).

Class 2 devices have sufficient resources to support most standard protocol stacks, including TCP/IP and potentially lightweight TLS implementations. However, they still benefit significantly from constrained-aware protocols that reduce overhead, minimize energy consumption, and streamline management operations[8]. In the Wi-Fi and Wi-Fi HaLow domains, Class 2 devices can operate as standard 802.11 stations but achieve

[6] If you haven't worked in the IoT space, you may be quick to assume that such constrained devices are no longer with us, however, that is not the case. To extend battery life, less memory is optimal (as the devices drain less energy during sleep with less memory). Additionally, memory costs have risen at the time of writing due to the overspending on a technology that is valuable, but not as valuable as hyped in this author's opinion (namely generative AI). Therefore, less memory equals a significantly lower price for hardware. These, among some other factors continue to drive the use of the highly constrained devices, particularly in utility and other large-scale deployments.

[7] The 2026 update to RFC 7228 (7228bis) adds Class 4 to the microcontroller group and adds a new group (J) that includes Classes 10, 15, 16, 17, and 19, which are general purpose devices and are "not really constrained devices" per the RFC. In most cases, these J-group devices do not impose constraints, other than excessive energy consumption.

[8] Additionally, remember that even for Class 2 devices, the total RAM available is the total RAM available. The device obviously requires more than a network stack. It must have application code to process local metrics from sensors or commands to actuators. This code must reside somewhere as well. Careful programming, loading code just-in-time, can help, but eventually you still hit the ceiling in constrained devices.

substantially better battery life and operational efficiency when connected to access points that support 802.11ah, 802.11ax TWT, or other IoT-oriented mechanisms. The class system thus provides a practical framework for matching protocol features to device capabilities (Bormann, Ersue and Keranen, 2014).

Before leaving this definition of constrained device classes, it is useful to consider the design decisions that can be made to address some of these limitations[9]. Hardware architecture dictates how these memory constraints are managed. While single-chip designs strictly limit available RAM by forcing the Wi-Fi stack and application logic to share resources, dual-chip designs overcome this by offloading network operations to a dedicated Wi-Fi coprocessor. However, this separation introduces an energy tradeoff, as powering discrete microprocessors increases consumption and reduces battery life. Additionally, the more RAM in the device, the more power required during sleep states. By intentionally limiting the RAM you gain a significant benefit in battery life. As one can see, tradeoffs are important and must be considered.

Constrained Networks

RFC 7228 also defines constrained networks as distinct from constrained nodes. A constrained network exhibits characteristics such as low bitrate and throughput, high packet loss rates, high variability of packet loss, asymmetric link capacities, penalties for transmitting larger packets, intermittent reachability, and lack of advanced IP services (Bormann, Ersue and Keranen, 2014). These network-level constraints interact with device-level constraints to create compounding challenges. A constrained device operating on a constrained network must not only minimize its own resource usage but must also cope with unreliable, low-bandwidth communication channels that exacerbate the costs of protocol overhead, retransmissions, and management operations[10].

Constrained networks arise particularly in sub-1 GHz IEEE 802.11ah deployments, where channel bandwidth is limited to 1, 2, 4, 8, or 16 MHz and data rates may be as low as 150 Kbps. The end nodes are Wi-Fi HaLow nodes, the intermediate nodes (gateways/access points) are Wi-Fi HaLow nodes, and therefore the network is constrained. At these rates, protocol overhead, including MAC headers, acknowledgments, and security encapsulation, consumes a disproportionate fraction of available bandwidth, directly impacting the energy efficiency and application layer

[9] In constrained device networks, design must go down to the end node hardware level to effectively implement a solution that balances performance, energy consumption, and security.

[10] The RFC indicates a difference between constrained networks and challenged networks. While constrained networks may have nodes with full capabilities and the constraints are derived more from the end nodes than the network itself, challenged networks are always constrained by the network itself. This explanation is a simplification, but it is sufficient for our purposes. For more on challenged networks, see RFC 4838 and Kevin Fall's 2003 paper titled "A Delay-Tolerant Network Architecture for Challenged Internets."

throughput experienced by constrained devices. Additionally, the extended ranges achievable with 802.11ah (up to 1 km) introduce propagation-related packet loss that further constrains effective throughput, operational reliability, and increases retransmission energy costs (Bormann, Ersue and Keranen, 2014).

Power and Energy Terminology

One of the most significant contributions of RFC 7228 is its formalization of power and energy terminology for constrained devices[11]. The specification defines four energy supply categories: E0 (event energy-limited) devices, such as those powered by energy harvesting, which receive energy on an unpredictable or event-driven basis; E1 (period energy-limited) devices, such as those with rechargeable batteries, which receive periodic energy replenishment; E2 (lifetime energy-limited) devices, such as those with non-replaceable primary batteries, whose total energy budget is fixed at deployment; and E9 (no energy limitations) devices, which are mains-powered and face no energy constraints (Bormann, Ersue and Keranen, 2014).

Complementing the energy supply categories, RFC 7228 defines three power management strategies: P0 (normally-off) devices that are usually in a state where they are not connected to the network and must reattach when communication is required; P1 (low-power) devices that appear to be connected to the network but may experience significantly higher latency due to sleep cycles; and P9 (always-on) devices that maintain continuous connectivity with minimal latency (Bormann, Ersue and Keranen, 2014).

With the energy category and power management strategies, we can create combinations to meet our needs. For example, an E0 device coupled with a P0 strategy could communicate when energy is available through energy harvesting and simply remain idle otherwise. It could gather data, communicate the data and then return to a low/no power state. However, an E2 device could be coupled with the same P0 strategy to extend the non-replaceable battery to years instead of months on the same battery power.

[11] Terminology is the bane of technology professionals. Having a standardized set of terms for a specific domain is essential for developing and advancing that domain expertise. CWNP recommends use of the IETF terminology when communicating, documenting, or referencing constrained devices.

Energy Class Name	Energy Limitation	Power Source Example
E0	Event energy-limited	Event-based harvesting
E1	Periodic energy-limited	Battery that is periodically recharged or replaced
E2	Lifetime energy-limited	Non-replaceable primary battery
E9	No limitation to available energy	Mains-powered

Table 2: RFC 7228 Energy Classes Summarized

Power Management Class	Strategy	Communication Ability
P0	Normally-Off	Reattach when required
P1	Low-Power	Appears connected, perhaps with high latency
P9	Always-On	Always connected

Table 3: RFC 7228 Power Management Strategies

For Wi-Fi-connected constrained devices, the radio interface often consumes the largest portion of total energy. Duty cycling, periodic switching between active and sleep states, is the primary strategy for reducing radio energy consumption. The interaction between a device's energy supply category and its power management strategy determines the feasible operating envelope for Wi-Fi communication. An E2/P1 device (non-replaceable battery, low-power mode) might achieve multi-year operation on IEEE 802.11ah (Wi-Fi HaLow) with TWT-scheduled wake-ups, while the same device using standard 802.11n (Wi-Fi) with continuous active mode would deplete its battery within days (or even hours, depending on the battery capacity). Understanding these categories is essential for evaluating the energy-related findings discussed in

subsequent sections of this review and for selecting and implementing appropriate devices (Bormann, Ersue and Keranen, 2014).

An additional factor that impacts energy consumption is the combination of the channel access medium and the traffic scenario. For example, (Santi, Tian, Khorov, and Famaey, 2019) found that a low traffic network generally performs better related to end device energy consumption using CSMA/CA (Carrier Sense Multiple Access with Collision Avoidance), while a high traffic network generally performs better using RAW (Restricted Access Window) and having more RAW slots rather than fewer. At the same time, using more RAW slots was shown to increase the latency. This shows the complexity of designing for constrained devices and the importance of engineering expertise in the process. This will be discussed in significant depth in the section titled "Energy Consumption and Battery Life."

Implications for Wi-Fi Protocol Design

The RFC 7228 framework reveals a fundamental tension in applying Wi-Fi to constrained devices. Wi-Fi was originally designed with comparatively powerful devices in mind, such as laptops, desktops, and access points, that operate with abundant memory, processing power, and energy because they are either mains-powered or Power over Ethernet (PoE)-powered. Adapting this technology to Class 0, Class 1, and even Class 2 devices requires rethinking assumptions at every layer of the protocol stack: PHY-layer modulation and coding must be efficient at low data rates; MAC-layer access mechanisms must minimize idle listening and contention overhead; security protocols must achieve adequate protection without exceeding device memory and processing budgets; and management protocols must support large-scale device fleets without imposing unsustainable per-device overhead. The sections that follow examine how the research community has addressed each of these challenges, using the RFC 7228 classification system as a common reference point.

Energy Consumption and Battery Life

Energy consumption is widely recognized as the single most critical challenge in deploying Wi-Fi in constrained devices. The IEEE 802.11 protocol family was originally designed for devices with essentially unlimited power supplies, and its default operating modes, continuous channel monitoring, active contention-based access (CSMA/CA), and complex signal processing, consume orders of magnitude more energy than competing low-power wireless technologies such as Bluetooth Low Energy (BLE) or IEEE 802.15.4-based protocols. For battery-powered IoT nodes that must

operate for months or years without maintenance, achieving acceptable energy efficiency requires fundamental changes to how Wi-Fi radios are controlled, how channel access is scheduled, and how devices transition between active and sleep states. This section reviews the substantial body of research addressing energy consumption and battery life in Wi-Fi- and Wi-Fi HaLow-connected constrained devices, organized around the key technical themes that have emerged in the literature.

Wi-Fi HaLow Energy Models

Foundational Analytical Models

The IEEE 802.11ah (Wi-Fi HaLow) standard, operating in the sub-1 GHz band, was specifically designed to support power-constrained devices and has consequently been the focus of extensive energy modeling research. Bel, Adame, and Bellalta, developed one of the earliest analytical energy models for 802.11ah WLANs, identifying and characterizing all major energy consumption processes in a self-powered wireless sensor node operating in 802.11ah power-saving mode (Bel, Adame, and Bellalta, 2015). Their model provides a foundational energy budget decomposition, quantifying energy consumption across four distinct states: transmission, reception, idle listening, and doze. A key finding of this early work was that idle listening dominates energy consumption in low-traffic IoT scenarios, a result that has been repeatedly confirmed by subsequent research and that underscores the critical importance of the TWT and sleep mechanisms introduced in 802.11ah. The authors demonstrated that battery lifetime prediction based on their model enables system designers to plan battery replacement cycles or size energy storage components prior to deployment, and they validated that the IEEE 802.11ah Task Group's proposed power-saving parameters are near-optimal for typical IoT scenarios.

Building on this foundation, (Santi, Tian, Khorov, and Famaey, 2019) developed a more detailed analytical model specifically targeting the two key MAC mechanisms for energy reduction in 802.11ah: the Restricted Access Window (RAW) and Target Wake Time (TWT). RAW uses scheduled transmission time slots so that end nodes know when they can transmit with reduced contention rather than requiring operation with continued mass contention. The AP can segment end nodes into groups that contend for the medium during shared RAWs[12]. TWT is more node-specific in that a scheduled power management plan can be negotiated between the AP and each station, reducing

[12] RAW is not available in Wi-Fi chipsets and is unique to Wi-Fi HaLow. In standard Wi-Fi today, OFDM is used to bring efficiency to the channel rather than time-segmented nodes. So the segmentation is frequency-based rather than time-based. In Wi-Fi HaLow chipsets, RAW is widely supported, for example, in Morse Micro MM6108 and MM6104, Newracom NRC7292 and NRC7394, and Palma Ceia PCS2100 and PCS2500.

the time that the radio is required to be powered on for data reception. Both modes can be used together.

The (Santi, Tian, Khorov, and Famaey, 2019) model computes average energy consumption during a RAW slot and was validated against an extended 802.11ah simulator incorporating an energy life-cycle model, achieving a maximum deviation from simulation of only 10%. The study showed that RAW reduces energy consumption by limiting channel access contention, while TWT further extends sleep periods to drastically lower idle listening energy. The combination of RAW and TWT in 802.11ah was shown to provide superior energy efficiency compared to legacy 802.11 power-saving mechanisms for constrained devices, and TWT was found to be particularly beneficial for low-duty-cycle IoT nodes that transmit infrequently but must maintain connectivity. This model has become a reference tool for pre-deployment lifetime estimation in the 802.11ah research community.

Empirical Validation and Hardware Measurements

While analytical models provide essential theoretical foundations, the gap between theoretical predictions and real-world device behavior has been a persistent concern in the energy modeling literature. Maudet, Andrieux, Chevillon, and Diouris made a contribution by presenting direct power consumption measurements on a commercially available 802.11ah device, providing the first in-depth, empirical foundation for IoT energy analysis of Wi-Fi HaLow (Maudet, Andrieux, Chevillon, and Diouris, 2024). Their work integrated the current drawn by the power amplifier during transmission and qualified each power-saving mode of the standard in practice. A critical finding was that the power amplifier current during transmission is a frequently underestimated contributor to overall energy consumption. The study confirmed that 802.11ah power-saving mechanisms function as specified and that sleep mode current is low enough to make coin-cell-powered operation realistic for low-duty-cycle applications, but it also revealed that energy per payload byte increases significantly with overhead from headers, acknowledgments, and retransmissions, suggesting that payload aggregation and reliable channels are key to efficiency (Maudet, Andrieux, Chevillon, and Diouris, 2024).

The same research group subsequently published a refined power consumption model that accounts for factors neglected in prior literature: the number of nodes in the network, retransmission attempts, and exchanges related to higher communication layers such as TCP/IP stack overhead (Maudet, Andrieux, Chevillon, and Diouris, 2025). This refined model demonstrated that network density has a significant and previously undercharacterized impact on per-device energy consumption, as more competing stations raise energy due to increased contention, collisions, and retransmissions. Higher-layer protocol overhead, including TCP keep-alives, DHCP exchanges, and TLS

handshakes, was shown to represent a substantial fraction of total energy in constrained nodes. The work identified a maximum viable network size beyond which energy consumption becomes unsustainable for battery-powered sensors, providing a critical design tool for network operators who must bound the number of devices per access point while maintaining target battery lifetimes.

The most recent empirical contribution comes from Xu, Kane, Liu, McKague, and Li, who developed a forecast model for predicting the energy consumption of Wi-Fi HaLow devices with TWT enabled, validated through direct experimental measurements on commercially available hardware (Xu, Kane, Liu, McKague, and Li, 2025). Unlike prior analytical models, this work closes the gap between theoretical energy estimates and real-world device behavior by capturing all operational phases: wake-up, TWT negotiation, transmission, and sleep. The experimental model revealed that actual hardware consumes meaningfully more energy than simplified analytical models predict, due to protocol overhead in TWT negotiation phases. The study also found that optimal TWT interval configuration is highly dependent on application traffic pattern, and that mismatched configurations can negate energy benefits, providing actionable recommendations for managing energy costs in real-world Wi-Fi HaLow IoT deployments.

An important takeaway from this research is that the management of RAW and TWT is key to energy consumption in Wi-Fi HaLow, and therefore battery lifetime for battery-powered constrained devices, and custom software may be required to accomplish the goals of a given project. For example, using the right batteries and managing the radio power state and other device power states, a field node could operate for several years on non-replaceable batteries. For devices using energy harvesting and rechargeable batteries, even longer battery lifespan could potentially be achieved. It requires careful planning of energy consumption and device configuration and implementation[13].

Target Wake Time Mechanisms for Wi-Fi

TWT in IEEE 802.11ax (Wi-Fi 6)

Target Wake Time (TWT) was originally introduced in IEEE 802.11ah and subsequently adopted and extended in IEEE 802.11ax (Wi-Fi 6), making it the most widely available power-saving mechanism for IoT devices in modern Wi-Fi networks. Sianipar et al. presented a comparative ns-3[14] simulation study of three TWT scheduling models

[13] Interestingly, for wireless engineers designing and deploying constrained node IoT solutions, mastering battery technology becomes one of the most important skills. Which batteries last longer physically? Which batteries can sustain performance at different temperature levels? Which batteries can handle more recharge operations? And many other questions must be addressed.

[14] Ns-3 (network simulator version 3) is a simulator often used in research to evaluate the performance of protocols. It is freely available at nsnam.org.

contrasted against Continuously Active Mode (CAM) in Wi-Fi 6 networks, evaluating the tradeoff between energy efficiency and quality of service for both real-time and non-real-time IoT uplink traffic (Sianipar, Arif, Syahrial, Yunida, Walidainy, and Munadi, 2025). Their results demonstrate dramatic energy improvements: CAM (always-on) achieves the lowest delay (0.14–0.6 ms) but the highest energy consumption (9.2–9.8 nJoules/bit), while TWT scheduling can reduce consumption to 0.38–0.53 nJoules/bit, an approximately twenty-fold improvement. The study found that one scheduling model can achieve a low energy consumption level and is recommended for non-real-time, delay-tolerant IoT use cases, while another scheduling model provides the best energy-QoS balance for mixed workloads with both real-time and background traffic. The research shows the importance of tuning the TWT scheduling model for the need and not simply accepting default settings.

The energy benefits of TWT are further contextualized by Farhad and Pyun's comprehensive survey of IEEE 802.11ah MAC-layer features for massive IoT, which documents that TWT enables long, protocol-scheduled sleep intervals that make 802.11ah suitable for IoT devices targeting multi-year battery lifetimes (Farhad and Pyun, 2022). While focused on Wi-Fi HaLow and not traditional Wi-Fi, their survey identifies TWT as one of several complementary mechanisms, alongside RAW and TIM segmentation, that collectively reduce contention, beaconing overhead, and idle listening energy. Most relevant to Wi-Fi deployments, the survey identifies dynamic, adaptive TWT parameter tuning under heterogeneous traffic loads as a key open challenge, noting that the optimal TWT configuration depends on a complex interaction between traffic patterns, network density, and device capabilities.

Though this research (Farhad and Pyun, 2022) was focused on 802.11ah, the TWT procedures and algorithms impact implementations in 802.11ax and beyond as well. Engineers should consider the kind of traffic generated on the network, the density of network nodes and traffic, and the end node capabilities when configuring TWT parameters. Many TWT parameters are tunable only when the software is customized in the end nodes and they are used when negotiating TWT configuration with the APs. These settings include the *Wake Interval Mantissa and Exponent* used in the interval formula (Interval = *Mantissa* x $2^{Exponent}$) and the *nominal minimum wake duration*, which is the time in microseconds that the client must stay awake for data exchanges during a session. Such customizations may require custom coding of the Wi-Fi drivers or firmware on the end nodes. In some cases, APs may expose these settings through a CLI configuration interface, but most do not and instead depend on the client to offer the parameters in a TWT request.

Restricted TWT in IEEE 802.11be (Wi-Fi 7)

The latest evolution of TWT is the Restricted Target Wake Time (R-TWT) feature introduced in IEEE 802.11be (Wi-Fi 7). R-TWT allows for improved priority for device traffic that is low-latency demanding. Mozaffari Ahrar et al. evaluated R-TWT using ns-3 simulation across diverse service types and network sizes, assessing worst-case latency, energy efficiency, collision rate, and throughput (Mozaffari Ahrar et al., 2025). Their findings confirm that R-TWT achieves bounded worst-case latency for IoT and real-time traffic while outperforming legacy Power Saving Mode and DCF in energy efficiency, making Wi-Fi 7 the first Wi-Fi generation potentially suitable for deterministic industrial IoT applications. Performance scales stably as network size increases, confirming R-TWT's viability for dense IoT deployments. This represents a significant advance: previous Wi-Fi power-saving mechanisms forced a trade-off between energy efficiency and latency determinism, whereas R-TWT achieves both simultaneously.

It is important to note that R-TWT is not required by the Wi-Fi Alliance to gain the Wi-Fi Certified 7 status. Additionally, at the time of writing, common client devices and APs do not seem to support R-TWT as they indicate no support for it. If this feature becomes available, it may add significant enhancements to mission critical communications across Wi-Fi networks[15].

Idle Listening Optimization

The Idle Listening Problem

Multiple independent research efforts have converged on the finding that idle listening, which is the energy spent monitoring the channel when no data is being sent or received, is the dominant source of energy drain for low-traffic Wi-Fi IoT devices. Park and Adnan provided definitive evidence of this phenomenon, demonstrating that idle listening surpasses active transmission energy in low-to-moderate traffic IoT scenarios (Park and Adnan, 2016). Their work proposed two complementary mechanisms: a downclocking scheme that allows a station to enter a semi-sleep state while maintaining channel access capability during overheard frames, and a frame aggregation scheme that extends semi-sleep duration by reducing transmission frequency. The combined scheme achieves substantial energy improvements applicable to any 802.11-based constrained device without requiring changes to the access point or infrastructure. In

[15] Given that the focus of Wi-Fi 8 (802.11bn) is on high reliability, it is possible that we will see R-TWT or some evolved or replacement solution as a major focus in the next iteration and that manufacturers will focus more on the implementation of the same.

other words, it can be implemented on the end node in drivers or firmware to accommodate reduced energy consumption.

Kim, Lee, Ahn, Park, Suh, and Park extended this line of research with MILD (Minimizing Idle Listening via Down-Clocking), a scheme that adaptively reduces the radio's clock speed during idle listening intervals to cut processing power without sacrificing channel access responsiveness (Kim, Lee, Ahn, et al., 2025). The authors confirmed that idle listening remains a significant but addressable energy drain even in devices using Adaptive-PSM (A-PSM), and that standard PSM (Power Saving Mechanisms) and A-PSM mechanisms are insufficient for battery-constrained IoT. MILD is notable for its backward compatibility with both legacy 802.11n/ac and modern 802.11ax standards, making it applicable across the full range of Wi-Fi generations deployed in IoT settings, while achieving a 55% improvement in energy efficiency. The approach is particularly effective for IoT nodes with bursty, infrequent traffic patterns where idle periods dominate the device duty cycle.

Cross-Layer Energy Optimization

Another research project took a different approach by experimentally characterizing the IEEE 802.11 power-saving mechanism on a commercial Wi-Fi IoT module (Texas Instruments CC3235SF) under the sparse uplink traffic that dominates IoT scenarios (Venkateswaran et al., 2021). Their key finding was that even with PSM enabled, battery life is only approximately 30% of what it would be on a truly idle connection, exposing a fundamental inefficiency caused by timing misalignment between TCP transmissions and beacon receptions. The authors built a platform-agnostic power consumption model and proposed five power optimization strategies culminating in a standard-compliant cross-layer algorithm that extends battery life by 24–31% without requiring protocol modifications. Their work demonstrates that network round-trip time and the relative timing between uplink transmissions and beacon intervals are the dominant factors governing energy efficiency, not merely the PSM on/off state. This finding is significant because it reveals energy optimization opportunities that exist entirely within the existing 802.11 standard framework, requiring only software changes on the device side.

Additional Factors in Wi-Fi Energy Efficiency

Lee proposed an interference-aware opportunistic dynamic energy saving mechanism that adaptively optimizes operating clock frequencies for signal processing when the mobile station transmits on partial sub-channels, exploiting channel availability windows to reduce processing energy (Lee, 2017). The work identified that WLAN's inherently complex signal processing (OFDM, MIMO) consumes significantly more energy than simpler wireless standards, creating a per-transmission energy

disadvantage for IoT applications. The interference-aware component avoids activating the mechanism when channel interference is high, which would degrade performance without energy benefit. This approach achieves substantial energy efficiency improvement for sparse, low-duty-cycle IoT traffic patterns typical of sensor nodes. While the implementation of this research requires chipset changes, an important factor that it reveals does not: the use of narrower channels. Devices may consume less energy, in most cases, transmitting the same data in an IoT context on a 20 MHz channel than on a wider channel. In large part, this is due to the nature of small payloads in IoT communications, However, it is important to note that this is not always the case. For example, (Liu and Choi, 2023) found that with Samsung S10, regardless of bandwidth, it consumed a constant 1455 mW of power during transmission. In other devices, they found reduction in energy consumption as the bandwidth narrowed.

A recent research paper provides complementary empirical data by measuring current consumption in IEEE 802.11ax (Wi-Fi 6) and 802.11ac devices across all MAC states under varying traffic loads (Lemercier and Orgerie, 2025). Their study reveals that Wi-Fi 6 shows significant improvements in sleep and idle state energy efficiency compared to 802.11n, but higher current draw in active transmit and receive states due to more complex signal processing. The energy consumption patterns differ substantially between 802.11ac and 802.11ax, potentially making prior-generation measurement data unreliable for modeling modern devices. These measurements provide calibration-quality data for simulators such as ns-3 and directly benefit battery-constrained IoT devices that spend most of their time in doze or sleep modes between infrequent transmissions.

Wake-Up Radio Approaches

Wake-up Radio (WuR) represents a fundamentally different approach to the idle listening problem. Rather than optimizing when the primary Wi-Fi radio sleeps and wakes, WuR adds a secondary ultra-low-power radio interface that remains always listening and wakes the primary Wi-Fi radio only when needed, enabling the main interface to remain in deep sleep for extended periods. One paper surveyed use cases for 802.11-based WuR across smart homes, industrial monitoring, healthcare, and asset tracking, and described the IEEE 802.11ba amendment that standardizes this mechanism (Garcia-Villegas, López-Aguilera, Demirkol, and Aspas, 2020). A WuR secondary interface can reduce primary Wi-Fi radio energy consumption by up to two orders of magnitude in low-traffic IoT deployments by completely eliminating idle listening. The authors found that WuR is most beneficial for event-driven IoT applications where long dormant periods are interrupted by infrequent burst data transmissions, and that the latency trade-off between WuR-enabled deep sleep and on-demand wake-up is manageable for the many IoT use cases.

Sánchez Vital's doctoral dissertation provides the most comprehensive analysis of multi-radio architectures for energy-efficient Wi-Fi IoT, combining a primary Wi-Fi interface with secondary WuR or LPWAN interfaces (Sánchez Vital, 2024). The research demonstrated that a multi-radio architecture with WuR as secondary interface reduces total IoT device energy consumption by up to nearly two orders of magnitude compared to conventional single-radio Wi-Fi, while maintaining latency comparable to always-on operation. The dissertation also analyzed AP Power Save mechanisms under discussion in IEEE 802.11bn (Wi-Fi 8), finding that AP power consumption can be reduced by one-third on average using the new mechanisms. This work highlights unresolved trade-offs between energy efficiency, data rate, and latency that remain as open research challenges in Wi-Fi IoT standardization.

In a later paper (Sanchez-Vital, Roger, Gomez, Carles, Garcia-Villegas, and Eduard, 2024), the same primary author addresses WuR in the context of wireless mesh networks. For clarification, WuR is further defined as "the main or primary radio, which is a high-bandwidth, but power-hungry communication interface, and a secondary radio, referred to as WuR. This secondary radio's only duty is to 'wake up' or activate the primary radio on demand (upon detection of a wake-up signal) so that high-throughput, low-latency data may be exchanged. It offers the capacity to wake up a receiver node instantly to exchange urgent data, while allowing that node to sleep for extended periods of time." This paper found that Wi-Fi with WuR can come much closer to competing with Wi-Fi HaLow, BLE, IEEE 802.15.4, and LoRaWAN related to energy efficiency, when considering the battery lifetime capabilities, while still offering burst traffic at much higher data rates. In the end, the research shows that WuR improves energy efficiency over single-radio idle power save operations by up to 50 times, which is a significant advantage for constrained devices. This finding is also illustrated in Sanchez Vital, et al., 2023, "Energy-Efficient Wireless Mesh Networks with IEEE 802.11ba: A New Architecture."

At the time of writing, WuR is in its infancy and there is little-to-no support for it in the market. The major hindrance to adoption seems to be the implementation of the secondary wake up radio requirement. This functionality requires a complete overhaul of traditional Wi-Fi device design. For small devices, it also becomes a space design factor. In the end, we will have to wait and see how commonly it is implemented and the types of devices that implement it. Consider that there is little value in implementing it in constrained nodes when APs do not implement support for it[16].

[16] WuR requires a different modulation of the WuR Wake Up frames than that used in traditional Wi-Fi. Some methods have been developed, though inconsistent and very complex, to emulate these signals with current 11n and 11ac hardware, ultimately hardware changes are likely required to enable switching to OOK (on-off keying) modulation instead of OFDM modulation schemes. This is another constraint on its deployment.

Additionally, a presentation was given in May of 2024 to the 802.11bn (Wi-Fi 8) group on the integration of WuR into Wi-Fi 8. If the presentation's recommended features are implemented, such as support for 160 MHz and 320 MHz channels and aggregation of UHR and WuT PPDUs for improved bandwidth utilization, it may add some interest to the implementation of the required extra WuR radio. Additionally, in March of 2024, a presentation was given to the group on the benefits of WuR for power metering, showing continued interest in WuR's capabilities. Ultimately, the implementation of the WuR functionality will come down to market demand for it and with the energy enhancements provided by TWT and other PMS solutions, it will be interesting to see if WuR is fully adopted, at least for constrained devices and laptops and some other use cases.

AP-Side Power Saving

A notable recent development is the extension of power-saving mechanisms from the station side to the access point side. Sanchez-Vital, Belogaev, Gomez, Famaey, and Garcia-Villegas provided the first comprehensive overview and analysis of the AP Power Save framework being introduced in IEEE 802.11bn (Wi-Fi 8), which addresses AP-side energy consumption in the Wi-Fi standard (Sanchez-Vital, et al., 2024). Prior Wi-Fi standards focused exclusively on client and station power saving, though they may coordinate with the AP. The new mechanisms, Scheduled Power Save, Semi-Dynamic Power Save, and Cross-Link Power Save (as they are referenced in the research literature), supplemented by Wake-up Radio integration, can reduce AP power consumption by an average of up to 28%. Battery-powered APs, increasingly relevant for IoT gateway scenarios in remote or infrastructure-limited deployments, stand to benefit most, as always-on AP behavior is the dominant energy waste in low-traffic periods. However, backward compatibility with legacy devices remains the primary open challenge, as legacy stations cannot anticipate AP sleep periods. This will be an area to watch in the future.

Clarity Moment: In temporary Wi-Fi deployments, energy efficiency is not only important for the clients but also for the AP. In many cases, the deployment needs to be as quick and simple as possible. Additionally, mains power is not always available and, even when generators are used, more significant power draw devices may need the mains power more. In any case, deploying APs that run on batteries is more common than you might first think. This can be achieved with custom-built APs, APs power from batteries that provide Power over Ethernet (PoE), or off-the-shelf APs that support battery power. Today, most of the off-the-shelf battery-capable APs/routers are very limited. For example, Spectrum launched Invincible WiFi™ in early 2026, which supports Wi-Fi 7 and has included battery backup that can run the router for up to 8 hours without mains power. Other solutions, such as the RCN Technologies Pop-Up

Network Kits (PNKs), incorporate mid-range to enterprise-grade Wi-Fi routers into a Pelican case with a large battery included to power the router. These kits can run from 8 to 16 hours depending on configuration. If one can improve the energy efficiency of the APs significantly, this could be extended by 30-60% without other hardware changes. Additionally, you could add extra batteries to the kit to extend the operational life[17].

Energy Harvesting Integration and MAC Layer Issues

For constrained devices that operate in E0 (event energy-limited) environments, energy harvesting integration is a critical enabler. A recent paper investigated the integration of solar energy harvesting into a dense Wi-Fi network serving IoT medical devices, introducing a MAC-layer algorithm for energy consumption mitigation aligned with the IEEE 802.11be amendment (Famitafreshi, Melià-Seguí, and Afaqui, 2022). Their key finding was that common and recommended general Wi-Fi MAC-layer behavior is incompatible with energy-harvesting operation in dense deployments, and MAC operational modifications are required to prevent overloading the limited harvested energy budget. Solar harvesting can sustain continuous Wi-Fi operation for e-health IoT devices in indoor environments if properly coupled with adaptive MAC scheduling, but the study demonstrates that MAC-layer energy management is as important as the choice of energy harvesting technology for achieving long-lived, battery-free IoT operation.

Interestingly, the study showed that one can reduce the size of the solar panels from 47 cm^2 to 7 cm^2 and still power the constrained nodes for medical device operation using the introduced algorithms (Famitafreshi, Melià-Seguí, and Afaqui, 2022). The goal is something called Energy Neutrality or Energy Neutral Operation (ENO), which effectively means that the solar panels are harvesting all the energy required to operate for 24 hours within that same 24 hours. This does not mean that no battery or energy buffer is used. Wi-Fi is bursty, not only in traffic, but in energy consumption. A single transmission will require more energy than the solar panel provides at that moment. However, energy is buffered into a battery, and the battery is used to power the actual radio. To realize the impact of going from 47 square centimeters to 7 square centimeters, consider that the former is about the size of a credit card and the latter is about the size of a postage stamp. That is a significant achievement and bodes well for field medical pop-ups as well as agricultural use and many others[18].

[17] The FAA restricts batteries to 100 Wh for air travel. The 98 Wh batteries in the RCN PNKs are strategically chosen to comply with this.

[18] Few studies evaluate multiple energy harvesting techniques combined, but this is also an option for improved energy harvesting and may well support even more traffic-intensive Wi-Fi communications in the future. For example, a vehicle can have a piezoelectric energy harvester and solar energy harvesters working in unison.

Implementing these algorithms, and others like them, requires driver/firmware modification. In this case, the changes must be made in clients and APs, which greatly limits its use case with off-the-shelf hardware. However, in the real world, it is not uncommon to implement custom solutions for unique needs and with constrained devices this is more common than one might assume. For example, Espressif announced in late 2023 that they had shipped more than one billion ESP8266 modules since launch in 2014 with nearly 200 million shipped in 2022 alone. These modules are heavily customized in most deployments.

Several factors must be considered in real-world implementations:

- The availability of the energy source: light, vibration, heat, RF, etc. Sufficient energy must be provided throughout every day (not just the average day) to power the device, assuming ENO.
- The conversion and movement to energy storage. Several options are available for this, and the components needed are rectifiers (for conversion from AC to DC), a DC-DC converter for voltage regulation and a supercapacitor[19] and/or battery for energy storage.
- The powering of the Wi-Fi chipset from energy storage.
- The additional heat factor within the overall design, if applicable[20].

Additionally, when implementing customized drivers/firmware, one is not required to start from scratch. Software Development Kits (SDKs) and Application Programming Interface (API) implementations may be possible. For example, Expressif ESP-IDF, Silicon Labs GSDK, and Infineon/Cypress SDK can be used to configure Wi-Fi behavior, power management, provisioning, and over-the-air operations. With these solutions, the underlying driver is provided, and the integration layer is where customization occurs. With Linux-based embedded systems, directly customizing the drivers is an option. On the AP side, it is common to customize firmware like OpenWRT, DD-WRT, or a proprietary fork from a System on a Chip (SoC) vendor.

Additional Enhanced Power Save Mode Mechanisms

A final potential modification to power management methods in 802.11 is presented in "An Enhancement for IEEE 802.11 STA Power Saving and Access Point Memory Management Mechanism" (Bhargava and Raghava, 2022). Beyond TWT and WuR, this study proposed enhancements to the basic IEEE 802.11 Power Save Mode to improve its suitability for constrained devices. The proposal is an adaptive Listen Interval scheme

[19] Supercapacitors are less frequently used than rechargeable batteries because they are less capable in intermittent energy source scenarios, such as when the sunlight is unavailable for significant periods.
[20] This is mostly only a concern in high temperature deployments above 100 degrees F where the ambient heat can degrade the life of the battery.

that adjusts dynamically based on the station's battery status, combined with improved AP buffer memory management to maximize packet retention for sleeping devices. The change requires code modification, but not chipset modification, in both the clients and the AP. Their simulation results showed that the proposed scheme outperforms standard 802.11 PSM in both power consumption at the station and memory utilization at the AP, while handling edge cases such as missed beacon frames and delivery of unicast, multicast, and broadcast data to power-saving stations. The mechanism is backward-compatible with existing 802.11 infrastructure, enabling deployment on commercial Wi-Fi networks without hardware changes because a vendor specific value is used for the Listen Interval information and will be ignored if it is not included. Effectively, the modification adds the ability for the client to request a change to the Listen Interval when its available battery life is reduced, thereby extending the life of the device.

Security Attacks against Battery-Powered Devices

An important but underexplored intersection exists between energy management and security. Kim, Park, Lee, and Lee identified a critical vulnerability in next-generation Wi-Fi power-saving mechanisms: trigger frame-based uplink transmissions introduced in 802.11ax can be exploited by attackers to force IoT sensor nodes out of sleep mode, causing rapid battery depletion (Kim, Park, Lee, and Lee, 2024). Their Secure Triggering Frame-Based Dynamic Power Saving Mechanism (STF-DPSM) authenticates trigger frames before the device responds, preventing battery draining/depletion attacks while maintaining energy efficiency. The mechanism improves energy efficiency by 55.69% over conventional methods and reduces delay by 44.7% compared to full per-frame encryption. This work demonstrates that security and energy efficiency must be co-designed in next-generation Wi-Fi IoT systems, as power-saving mechanisms without authentication create exploitable energy vulnerabilities.

Stated clearly, this research addresses a Denial of Service (DoS) attack that can be launched against battery-powered or energy-harvesting/battery-powered combo devices where the battery is drained faster than the system design can accommodate. Devices using the newer Wi-Fi 7 multi-link operation (MLO) are even more susceptible to such attacks (due to the use of multiple radios that can be exploited). The novel position of this paper is that it suggests only implementing encryption for trigger frames when an anomaly is detected so that performance is only impacted when a system is under attack (Kim, Park, Lee, and Lee, 2024). Given that the recommendation requires encrypting frames that are not encrypted in the standard specification, this change would also require driver/firmware modifications for implementation.

Synthesis and Themes

One frequent theme discovered throughout this section has been the requirement of customized drivers/firmware to implement the most effective solutions for constrained devices. In cases where only the client must be modified, the process is very straightforward. In the cases where the AP must also be modified, it removes the potential for using typical enterprise hardware in the deployment and requires either custom-built APs or the use of consumer-grade devices that can run custom firmware. However, in many use cases, the development of a custom AP solution is justified and is more common than many engineers that work only in typical corporate deployments realize.

The transition of Wi-Fi from high-throughput mobile connectivity to a viable IoT protocol hinges on solving the idle listening problem. Research consistently identifies that for constrained devices, the energy spent merely monitoring the channel for potential traffic far outweighs the energy used for active transmission. While IEEE 802.11ah (Wi-Fi HaLow) pioneered solutions like Restricted Access Window (RAW) and Target Wake Time (TWT) to address this, empirical evidence shows a persistent "reality gap" between theoretical models and actual hardware. Factors such as power amplifier draw, protocol overhead (TCP/IP and TLS handshakes), and network density can significantly shorten predicted battery lifespans, necessitating more robust, measurement-backed planning tools. Even still, Wi-Fi HaLow is the primary choice for constrained IoT devices for many engineers, if an 802.11 protocol is to be used.

The advancement of TWT in Wi-Fi 6 and the introduction of Restricted TWT (R-TWT) in Wi-Fi 7 represent a shift toward deterministic power saving. By allowing devices to negotiate specific "wake" windows, TWT can reduce energy consumption per bit by up to twenty-fold. However, the literature warns that default settings are rarely sufficient; engineers must tune parameters like the Wake Interval Mantissa and Exponent to match specific application traffic patterns. Wi-Fi 7's R-TWT adds a layer of latency-sensitive priority, though its practical utility remains limited by current hardware support and certification requirements.

Emerging architectures like Wake-up Radio (WuR) and AP-side power saving offer paths to even more extreme efficiency. WuR (802.11ba) utilizes a secondary, ultra-low-power radio to "trigger" the primary Wi-Fi radio only when data is pending, potentially extending battery life by 50 to 100 times in event-driven scenarios. Simultaneously, upcoming Wi-Fi 8 (802.11bn) standards are beginning to address the energy drain of the Access Point itself—a critical consideration for temporary, battery-powered deployments or remote gateways where "always-on" behavior is unsustainable.

The literature highlights the necessity of cross-layer and security-aware optimization. Energy efficiency is not just a PHY/MAC layer problem; it is affected by TCP/IP timing alignment and even the size of energy-harvesting components. New power-saving mechanisms introduce vulnerabilities, such as trigger-frame-based battery depletion attacks, where attackers force nodes to stay awake. Achieving true Energy Neutral Operation (ENO) requires a thoughtful design that integrates efficient hardware, adaptive software drivers, and authenticated management frames to protect the device's limited energy stores.

Key Takeaways for IoT Engineers and Administrators

- **Prioritize Idle Listening Mitigation:** Since idle listening is the primary energy drain in low-traffic scenarios, focus on maximizing "doze" time rather than just optimizing transmission power.
- **Tune TWT Parameters Manually:** Do not rely on default TWT settings, if possible. Adjust the Wake Interval and minimum wake duration within the firmware to align with your specific sensor reporting frequency.
- **Account for Protocol "Tax":** Factor in the energy cost of higher-layer protocols (DHCP, TCP keep-alives, TLS handshakes). These often represent a larger percentage of the energy budget than the raw data payload.
- **Evaluate Hardware-Specific Power Curves:** Actual hardware (e.g., ESP32 vs. 8-bit MCU + SPI Wi-Fi) consumes more energy than analytical models suggest due to protocol negotiation overhead. Use empirical measurement for battery sizing.
- **Leverage Narrower Channels for IoT:** In many chipsets, transmitting over a 20 MHz channel is more energy-efficient for small IoT payloads than using wider 40/80 MHz channels, which require more complex signal processing.
- **Design for Energy Neutrality (EN):** When using energy harvesting, optimize MAC scheduling to reduce the required solar panel size. Small postage-stamp-sized panels can power Wi-Fi nodes if the software prevents energy-budget "burst" violations.
- **Secure Your Power-Saving Triggers:** Enable Protected Management Frames (PMF) or implement authenticated trigger-frame logic to prevent attackers from remotely waking your devices and "bleeding" the battery dry.
- **Consider Multi-Radio WuR for Extreme Longevity:** If your application requires "instant" wake-up but has long dormant periods, look toward 802.11ba (WuR) solutions as they become available, as they outperform standard PSM by orders of magnitude in lab tests.

Security

Security in Wi-Fi-connected constrained devices presents a unique set of challenges that arise from the fundamental tension between the computational requirements of modern cryptographic protocols and the resource limitations of IoT devices. Standard Wi-Fi security mechanisms, such as WPA2, WPA3, TLS, and DTLS, were designed for devices with ample processing power, memory, and energy. When applied to Class 0, Class 1, and even Class 2 constrained devices, these mechanisms impose overhead that can degrade performance, accelerate battery depletion, and in some cases exceed device capabilities entirely. This section reviews academic literature on security challenges and solutions for Wi-Fi in constrained devices and what we can learn from them, organized around the key themes that have emerged from the research.

Authentication and Key Management Challenges

Overhead of Standard 802.11 Authentication

The standard IEEE 802.11 authentication and key management framework, which relies on IEEE 802.1X, EAP-TLS, and the WPA2/WPA3 four-way handshake, imposes substantial computational, memory, and network overhead on participating stations. Kim, Min, and Han directly quantified this overhead for constrained devices, demonstrating that standard 802.11 combined with 802.1X EAP-TLS requires approximately 111 megacycles of computation, 14,960 bytes of memory, and 472.87 milliseconds of network time for a single authentication exchange on an ATmega128-class microcontroller (Kim, Min, and Han, 2017). These figures are prohibitive for Class 0 and Class 1 devices, where processing cycles, RAM, and energy are all scarce. The authors proposed a new authentication and key management mechanism that delegates computationally heavy operations to a more powerful Station-side Authentication Server agent, reducing the computation cost by approximately 98.5%, network cost by 79%, and memory requirements by 66% while maintaining security guarantees. The mechanism requires no pre-configured security information between the access network and IoT service domains, enabling zero-touch provisioning.

While the method suggested in (Kim, Min, and Han, 2017) would require that custom drivers/firmware be loaded on the clients and APs[21], it presents a viable solution to a real problem: implementing secure communications for extremely constrained devices. Consider that the selection of hardware is a balance of technical requirements and unit

[21] In this case, the solution would require a new Information Element (IE) called the IoT Domain Element (IDE) and a custom AKM Suite beyond that defined in the standard. Additionally, on the AP side a custom 802.1X modified solution would be required.

economics. While one could simply acquire more powerful end nodes and implement the right energy solution for the more powerful devices, these devices also typically cost more. If the solution includes 20-30 devices, it may not be a concern. What about deployments with 1,000 or even 10,000 nodes. Now, an increase of $5 per node makes a difference and this is just to meet security demands. It may be the case that it is worth investing in the custom development of a solution that costs roughly the same as the extra hardware or less to allow for the use of hardware that is too constrained for standard Wi-Fi operations, but would otherwise meet the needs of the solution and, therefore, allow for more efficient energy management over time. At the same time, it is important to consider the technical debt[22] incurred by implementing a custom solution that must be managed in the future, which may outweigh the cost of the more capable hardware. Each scenario must be carefully evaluated.

In the introduction to a 2019 special issue of Sensors, Li, et al. provide a broader perspective on the authentication challenge in their review of the papers dedicated to security in constrained devices (Li, Song, and Iqbal, 2019). Their statements characterize the root causes of security deficiency: devices are not designed with effective security features, standard cryptographic solutions are too computationally expensive, and IoT networks are too heterogeneous for uniform security solutions in many deployments.

The solution to these problems is not simple. It requires careful planning and consideration of the balance required among security, performance, and energy efficiency. For this reason, many solutions use end-to-end encryption (rather than Layer 2 encryption) so that data can be encrypted on the device first, then transmitted on an open network. Lower layers are still exposed to various vulnerabilities, but the use of lightweight, secure encryption and decryption within the originating and destination applications ensures that the data is protected in transmission. It is also for this reason that some engineers choose to go with a "less intense" protocol than Wi-Fi for their implementation. They can achieve the security required on less capable devices, even at lower layers, without demanding excessive memory or processing power. While the intent of this paper is not to fully extrapolate the engineering and design process, it is essential that one understand its importance in wireless IoT deployments.

WPA3 Vulnerabilities

The introduction of WPA3, intended to address well-known WPA2 vulnerabilities, has not fully resolved the security challenges for constrained devices. Saini, Halder, and Baswade documented that WPA3 itself contains exploitable vulnerabilities, including

[22] Technical debt occurs when you implement a solution that is cheaper, easier, or faster now, but you pay for it in the future. In this case, you pay for it by exiting the typical Wi-Fi ecosystem with a customized Wi-Fi deployment that will require future software maintenance and possible reengineering should the original hardware no longer be available.

SAE (Dragonfly) timing and cache-based side-channel attacks (Saini, Halder, and Baswade, 2022). Their work introduced a two-stage intrusion detection system where the access point performs a lightweight first-stage check at regular intervals, and a full machine-learning classifier runs on the backend WLAN controller only when an anomaly is detected (second-stage). This architecture explicitly manages the computational burden on resource-limited hardware, reducing the load at the constrained AP by orders of magnitude compared to running full ML inference on the AP itself. The two-stage approach achieved over 99% accuracy on WPA3 attack classification.

Tarish provides complementary evidence of authentication weakness by documenting that the majority of commercially deployed Wi-Fi IoT devices lack meaningful authentication security, relying only on weak shared passphrases susceptible to brute-force and dictionary attacks (Tarish, 2022). The study proposed integrating the Secure Remote Password Protocol (SRPP) into Wi-Fi network protocols for IoT applications (a higher layer solution), providing zero-knowledge password proof that prevents passive eavesdropping and man-in-the-middle attacks without additional hardware cost. While SRPP addresses credential-level vulnerabilities, it represents only one layer in what must be a multi-layered defense for constrained Wi-Fi devices. However, the author of the paper was vague on the implementation. Is it a Layer 2 solution that required driver/firmware modification in the clients and APs? Is it only an Application Layer solution that is implemented for higher layer authentication and encryption? Certainly either (or both) could be implemented but implementing it at Layer 2 would require software modification in all devices. While this is a common theme across energy efficiency, security, and performance, it would be helpful if the author had more clearly defined the details of the intended system. What we can learn from this is that security can be addressed anywhere in the protocol stack and, when possible, security at multiple layers is best[23].

Additional WPA3 vulnerabilities include well-known potential attacks against the DragonFly handshake and other brute force methods. But, in most cases these are addressed through patches. However, in custom IoT deployments the addressing of these issues becomes the responsibility of the developer and care should be used to ensure proper and secure implementation.

[23] Interestingly, some research even focuses on security at the Physical Layer. For example, Channel-Based Secret Key Generation (CB-SKG) uses channel state information to establish keys, RF fingerprinting for PHY Layer authentication uses the "imperfections" in the oscillators, mixers, and power amplifiers to uniquely identify a device, Artificial Noise (AN) injection uses MIMO to send real data in the direction of the legitimate target while simultaneously injecting artificial noise in the null space, and Physical Layer Integrity (PLI) uses Time-of-Arrival (ToA) or Angle-of-Arrival (AoA) from known location sources (like fixed location sensors) to validate that the source could be located where it's supposed to be. These are just a few examples of security at the Physical Layer, though most of them are currently only "in the lab."

Higher Layer Protocol Overhead and Delegation

TLS and DTLS Performance on Constrained Devices

Transport Layer Security (TLS) and its datagram variant DTLS are the IETF's recommended security protocols for IoT communication, but their performance on constrained devices has been a persistent concern. Restuccia, Tschofenig, and Baccelli provided the first systematic experimental comparison of DTLS 1.2, TLS 1.2, DTLS 1.3, and TLS 1.3 running directly on real low-power IoT microcontrollers, measuring bytes-over-the-air, memory footprint, and energy consumption across multiple implementations and configurations (Restuccia, Tschofenig, and Baccelli, 2020). Their findings revealed that TLS/DTLS 1.3 do not always increase overhead versus their 1.2 predecessors and in some configurations decrease it, making 1.3 a viable candidate for constrained wireless deployments. However, implementation quality varies considerably, and the choice of implementation has as much impact on overhead as the protocol version itself. For Wi-Fi-connected IoT devices, TLS 1.3 represents a practical and deployable security solution, but current implementations still have significant optimization potential.

The paper specifically noted that Flash memory footprint of TLS 1.3 was approximately 20 percent greater, but the RAM footprint was the same on average with one implementation using 25 percent more RAM and another using 30 percent less (Restuccia, Tschofenig, and Baccelli, 2020). Interestingly, TLS with TLS-ECC (Elliptic Curve Cryptography) instead of TLS-PSK (Pre-Shared Key) required a smaller memory footprint in 1.3 than that required in 1.2. TLS 1.3 enhances security while providing faster "start of data." This result is due to the use of shorter handshakes in initial setup.

Like TLS, DTLS uses more Flash memory in version 1.3 than version 1.2 and, when it comes to RAM, it frequently uses slightly less in version 1.3 than version 1.2, though it is minimal, being measured at less than 30 bytes of variance in most cases (Restuccia, Tschofenig, and Baccelli, 2020).

Of greatest interest, however, are the results showing that the implementation has a significant impact on memory consumption and energy efficiency. The researchers evaluated both Mbed TLS[24] and WolfSSL[25] libraries and found that, depending on the implemented authentication and key establishment method, each library has its strengths (though WolfSSL wins in most cases). The largest impact on energy efficiency is between password-based and certificate-based authentication and key establishment, with certificate-based methods being many times more costly than password-based methods (Restuccia, Tschofenig, and Baccelli, 2020). For this reason, one must use

[24] Mbed-TLS can be found at: https://github.com/Mbed-TLS/mbedtls
[25] WolfSSL can be found at: https://github.com/wolfSSL/wolfssl

caution in selecting both a library for TLS/DTLS support and the authentication and key establishment method so that the optimum balance of security and efficiency are achieved.

In "Evaluating CoAP End to End Security for Constrained Wireless Sensor Networks," Fournaris, Giannoulis, and Koulamas (2019) evaluated DTLS-based security for CoAP using Contiki OS 3, specifically comparing Pre-Shared Key (PSK) and Elliptic Curve Cryptography (ECC) cipher suites. A critical finding was that the transition from unsecured CoAP to any DTLS mode represents the most significant ROM overhead (~76%), whereas moving from PSK to 256-bit ECC adds a comparatively smaller marginal cost, suggesting ECC offers a better security-to-memory tradeoff for capable nodes. However, the study highlighted a severe performance penalty when using Radio Duty Cycling (RDC) with ECC; while RDC reduces handshake energy consumption, it pushes latency over 110 seconds due to timer expirations and retransmissions. Consequently, the authors recommend PSK as the most balanced suite for general wireless sensor network use, noting that ECC is only viable when sessions are maintained long-term to amortize the extreme energy cost of the initial handshake.

Another paper proposed dynamic adaptation of TLS parameters based on current resource availability and security requirements for wireless critical infrastructure (Bodenhausen, Grote, Rademacher, and Henze, 2024). Given that the use of TLS can result in increased overhead when used for end-to-end security in constrained devices (Rademacher, et al., 2022), their adaptive approach allows constrained wireless devices to use tighter security when resources permit and lighter-weight configurations during resource-scarce conditions, rather than being locked into a single fixed configuration. The work identified significant untapped configurability within TLS that current constrained device implementations do not leverage, representing a practical optimization path that requires no protocol changes.

This solution harkens back to the earlier discussion of changed communication parameters for low battery states with the goal of extending battery life. Here we see it applied to security in that, ideally, the strongest security should always be used but, realistically, one must identify the most important factor, and it is not required that this decision be made universally. Instead, the decision can be made contextually, lightening the processing load when energy sources are getting low, but still using "enough" security for the overall requirements. This concept is further bolstered by much research, including Suárez-Albela, et al., 2018, which addressed the differences in performance between ECC and RSA used with TLS (Suárez-Albela, Fernández-Caramés, Fraga-Lamas, and Castedo, 2018). However, in this research, interestingly, the more secure ECC performed better than RSA on constrained devices. This result is in part because ECC can use smaller key sizes and achieve greater key strength than RSA.

For example, an ECC key of 256-bits in length is stronger than an RSA key of 2048-bits in length[26].

Security Offloading to Access Points and Proxies

A recurring architectural pattern in the literature is the delegation of security computations from constrained devices to more capable network elements. Nofal et al. proposed a framework to offload TLS security computation from constrained Wi-Fi stations to access points, empirically measuring the energy and processing overhead that TLS imposes on IoT stations (Nofal, Tran, Dezfouli, and Liu, 2021). Their framework models the AP association problem as a multi-objective optimization problem and demonstrated that offloading achieves approximately 15x energy savings on constrained stations (0.173 J versus 2.729 J per handshake) and reduces time from 21 seconds to 1.3 seconds. The framework is notable in that it does not require changes to the hardware in the IoT devices, making it potentially deployable in existing constrained device ecosystems. However, the devices must be capable of implementing the software-based solution. A centralized management gateway handles monitoring, key distribution, and AP handover, providing a unified administrative control plane. This solution requires customized software on the AP and clients to establish the pre-shared key higher layer connection between them and software on the AP to proxy to and from a TLS session with remote nodes using standard TLS exchanges. Additionally, it is possible that the AP now becomes the bottleneck as it will be responsible for handling all TLS sessions for all associated end devices.

ESSE (Efficient Secure Session Establishment) splits DTLS into a separate handshake phase and an encryption/transmission phase, delegating the handshake burden to a more capable proxy node while preserving end-to-end secure communication semantics (Kang, Park, Kwon, and Jung, 2015). This separation is significant because it demonstrates that end-to-end security guarantees can be maintained despite delegation, which is a non-trivial result that addresses concerns about proxy-based approaches weakening the security model. The research found that the DTLS handshake is more than nine times higher than that of the record layer (encryption) processing. By delegating this handshake to a proxy, the constrained node avoids this significant energy drain. While the paper focused on 802.15.4-type networks, Wi-Fi and other IP-linked sensor networks can benefit from ESSE when standard security protocols are too costly to run directly on constrained nodes.

[26] Key strength can be confusing. It is a measure of the resistance to being broken by a brute-force attack or specific cryptanalytic method. For RSA keys, to achieve 128-bit strength, the key size must be 3072 bits long. That's 3072 bits that must be stored in memory and is quite impactful on a C0 or C1 class constrained device. However, ECC keys meeting 128-bit strength are only 256 bits long (as to the ECC Curve Size). Therefore, stronger security is more efficient from a memory consumption standpoint in this case.

The most comprehensive proxy architecture in the research literature is the "IoT Proxy," a modular edge security component that externalizes security functions from resource-limited IoT devices by routing their network traffic through a secure gateway equipped with Virtual Network Security Functions (Canavese, Mannella, Regano, and Basile, 2024). The VNSFs include a Virtual Private Network (VPN) terminator and an Intrusion Prevention System (IPS) using Machine Learning (ML)-based "oblivious authentication" to identify and classify connected devices without decrypting payloads. By offloading all significant security computation to the proxy, even the most resource-constrained Wi-Fi IoT devices can benefit from enterprise-grade protection without hardware upgrades. The architecture is modular and deployable in real-world settings, providing a scalable path to securing large fleets of legacy or constrained Wi-Fi IoT devices. Docker containers can be deployed to implement the solution.

The convergence of multiple research threads points toward proxy and edge computing architectures as the most practical approach to securing constrained Wi-Fi IoT devices at scale. Canavese et al.'s IoT Proxy, Nofal et al.'s AP-based TLS offloading, Kim et al.'s SAS-based authentication delegation, and Kang et al.'s ESSE handshake delegation all implement variants of the same pattern: move expensive security functions off the constrained device and onto a more powerful network. This pattern is emerging as a practical consensus architecture for securing constrained Wi-Fi IoT at scale, with the specific implementation varying based on the security function being offloaded and the deployment context.

An edge security approach has several advantages for constrained Wi-Fi devices. First, it does not require changes to the constrained devices' hardware, making it deployable with legacy and already-fielded IoT hardware. Second, security policy and algorithm upgrades can be applied at the proxy or gateway level without requiring firmware updates on thousands of constrained endpoints in some cases. Third, the approach naturally aligns with network architecture trends, as edge computing nodes are increasingly deployed for latency and bandwidth optimization and can serve double duty as security enforcement points[27].

Management Frame Vulnerabilities

Deauthentication and Disassociation Attacks

A fundamental architectural vulnerability in IEEE 802.11 is that management frames, including deauthentication and disassociation frames, are not authenticated by default unless WPA3 is enabled or a vendor chooses to enable it by default, which is rare. This

[27] One person's definition of edge is another person's definition of infrastructure. In this context, we mean the location on the network where IoT devices connect: the router, gateway, coordinator, sink, or whatever it's called according to the protocol in use.

allows any attacker to forge these frames and disrupt network connectivity. Gebresilassie, Rafferty, Chen, Cui, and Abu-Tair developed a Transfer Learning and CNN-based intrusion detection system to detect deauthentication and disassociation denial-of-service attacks in IoT Wi-Fi environments (Gebresilassie, Rafferty, Chen, Cui, and Abu-Tair, 2023). Their system achieved 99.36% detection accuracy with a very low false negative rate of 0.002, using a new dataset (Wi-Fi Association_Disassociation Dataset) collected from real IoT traffic. The authors emphasized that constrained IoT Wi-Fi devices are disproportionately impacted by management frame attacks because they have no ability to verify frame authenticity at the device level, and because forced reassociation consumes energy that battery-powered devices cannot afford. The solution presented is an end-to-end solution including detection, real-time data collection, parsing, and visualization in Elastic Stack (ELK (Elasticsearch, Logstash, and Kibana)) and is sufficiently detailed for duplication. The data set generated for this project is available to anyone for their analysis at: https://github.com/samsonkg/Wi-Fi-Association_Disassociation-Dataset.

KRACK, FragAttacks, and Multi-Channel Man-in-the-Middle

Another paper provided a comprehensive review of Multi-Channel Man-in-the-Middle (MC-MitM) attacks against WPA-protected Wi-Fi networks, tracing attacks from 2014 through FragAttacks in 2021 (Thankappan, Rifà-Pous, and Garrigues, 2022). Their analysis documented that IoT devices are among the most vulnerable to these attacks because they often run outdated firmware and cannot receive or apply security patches quickly. This context is still relevant today as even the newest IoT devices are often using firmware/drivers that are 5 or more years old. KRACK demonstrated fundamental weaknesses in the 802.11 four-way handshake, while FragAttacks revealed vulnerabilities in frame aggregation mechanisms, both affecting all WPA-protected devices including constrained ones. Protected Management Frames (PMF), the standard countermeasure, are not consistently supported by low-cost IoT hardware, creating a systemic gap in the security lifecycle: constrained devices are provisioned with security assumptions that degrade over time but cannot be easily updated.

Clarity Moment: Remember that many constrained IoT devices are deployed with unique firmware or drivers to address the energy efficiency, performance, and administration issues presented in this paper. This custom solution must also address security concerns, but patching now becomes the issue. Unlike vendor-supplied hardware and software, where most of the issues presented above were repaired in patches within days or weeks, the internal organization must patch the software when new vulnerabilities are discovered, and this frequently takes weeks or months due to the delay in awareness and the workload that already exists. This reality often results in much longer time gaps between the discovery of an issue and the repair of the same

and is another factor that must be carefully considered when choosing to implement a custom solution.

TWT Attacks

Liu and Choi address three TWT attacks categorized as battery depletion attack, packet loss attack, and throughput throttle attack (Liu and Choi, 2023). These attacks are defined as follows:

- **Battery depletion attack:** Given that a TWT-capable device must establish a TWT session with the AP, the attacker can tear down the session with a TWT action frame sent to the client spoofing the AP's identity. The result is that the client will have to stay awake and consume more energy. Repeating this process can quickly deplete the battery on the device.
- **Packet loss attack:** Given that the client wakes up at the negotiated time slot within the TWT session, an attacker can spoof the client and tear down the TWT session with the AP. The result is that the AP will send frames to the client thinking it is awake and the frames (and packets they contain) will be lost.
- **Throughput throttle attack:** Given that TWT specified the duration in which a client can communicate with the AP, an attacker can spoof the client and set up a TWT session as the target client with a very low duty cycle. The result is that the client experiences very low throughput.

As can be seen, all of these attacks exploit the ability to spoof either the client or the AP. Therefore, all three attacks can be thwarted by enabling protected management frames (802.11w, Management Frame Protection), but this solution comes with its own problems. Many clients experience difficulties when enabling 802.11w and this can be exacerbated in constrained devices. It also adds extra processing requirements on the node. However, it is an option for constrained devices that can implement it. The other option, recommended in the paper (Liu and Choi, 2023), is to implement anomaly detection on the clients and APs, but this will also add extra processing requirements.

As an alternative to 802.11w and anomaly detection, one could revert back to custom drivers/firmware for the clients and APs. In this case, the TWT configuration could be set statically at the time of deployment and linked to the MAC addresses. Then, the client and AP will simply ignore any reconfiguration or tear down requests post-connection. This would require custom logic in the implementation. Alternatively, negotiations could be allowed, but custom authentication logic could be coded into the deployment. In either case, the three TWT attacks referenced by Liu and Choi could be avoided, but the requirement is, again, the introduction of technical debt to the solution and one must weigh the cost of that. Switching to another non-Wi-Fi protocol may be the better option for a specific project, such as Wi-Fi HaLow, which would likely

implement the full stack on a chip[28]. Also, due to its use of lower frequency bands, lower transmit power helps to extend battery life on Wi-Fi HaLow SoC implementations. Additionally, these Wi-Fi HaLow chips can typically sleep "deeper" than Wi-Fi chips because they can sleep for much longer periods.

Clarity Moment: When implementing Wi-Fi in constrained devices, the implementations often use SoC solutions for the Wi-Fi (like ESP32, for example) and then the SoC handles all the connection, security, etc., in its memory. Therefore, even if the host device has extremely constrained memory, it may be able to use traditional Wi-Fi without problems. However, in many cases, such as industrial, medical, and long-life infrastructure, an 8-bit or 16-bit microcontroller may be used, coupled with an SPI/SDIO Wi-Fi transceiver that requires the host MCU to run the TCP/IP stack and manage the security handshake, etc. These devices are built intentionally to last a very long time on battery, and they simply cannot handle all the Wi-Fi security operations within their constraints.

Intrusion Detection Systems

In addition to the earlier cited work on intrusion detection/prevention, several additional research projects are revealing. Örs, Aydin, Bogatarkan, and Levi presented a machine-learning-based Wi-Fi intrusion detection system specifically designed for the resource-constrained and heterogeneous nature of IoT networks (Örs, Aydin, Bogatarkan, and Levi, 2021).Their approach uses a single multi-class classifier operating on encrypted data collected from the wireless data link layer, covering both benign traffic and six categories of IoT-specific attacks with 96.85% accuracy. The system is described as scalable because no device-specific training is required, a critical advantage when managing diverse fleets of constrained devices. Working at the encrypted data link layer enables privacy-preserving intrusion detection without requiring packet decryption, and the single-model approach substantially reduces deployment complexity compared to systems requiring per-device classifiers.

In (Dalal, Akhtar, Gupta, Karamchandani, Kasbekar, and Parekh, 2022), nine different attacks were tested, and it was discovered that eight of them were exposed by the AP under test and were undetected by an IDS. From here, the authors developed a signature-based IDS that can detect all eight of the attacks. The research output, instructions and scripts for the attacks and the code for the IDS, are available at: https://github.com/neildalal/WPA3-Attacks-IDS. The IDS code is based on PCAP

[28] I just want to insert a reminder here that security is not the only concern. Tests should be performed for a specific project to ensure energy efficiency, performance, and security all meet your requirements at a minimum. Optimally, efficient administration and management will also be included.

analysis but could be easily modified to incorporate live captures for real-time detection using a dedicated monitoring device such as a Raspberry Pi.

The preceding paper also points out an important fact: even when PMF is enabled, deauthentication attacks may work after the initial Open System Authentication request and before the 4-way handshake, effectively resulting in a DoS attack (Dalal, Akhtar, Gupta, Karamchandani, Kasbekar, and Parekh, 2022). These kinds of DoS attacks can't be resolved without either modifying the AP firmware (adding logic to ignore a deauth request between the initial authentication and the 4-way handshake within a specified time window constraint) or implementing an IDS/IPS solution so that action can be taken by the system or by an administrator.

An additional Wi-Fi intrusion detection project is available on GitHub as the Raspberry Pi Pico W: Wi-Fi Intrusion Detection System (WIDS)[29]. This project detects Evil Twin attacks, RSSI anomalies, and channel flooding. It provides a real-time dashboard and runs on the Raspberry Pi Pico W and requires basic configuration to function. It is an excellent learning tool for understanding wireless IDS through source code investigation. As a very simple algorithmic solution, it provides a foundation Python script that can be modified to test, learn, or even implement as a basic IDS system.

The intrusion detection literature consistently demonstrates that passive wireless monitoring combined with machine learning is the most practical security detection layer for constrained Wi-Fi IoT networks. The constrained devices themselves lack the resources to perform intrusion detection locally, making network-level or infrastructure-level detection essential. The two-stage architecture proposed by Saini et al., the single multi-class classifier approach of Örs et al., and the CNN-based system of Gebresilassie et al. all converge on architectures that place detection intelligence at the network infrastructure level or network edge rather than on the constrained devices.

Synthesis and Themes

A key issue in securing constrained Wi-Fi devices lies in the heavy computational and memory requirements of standard protocols like 802.1X and EAP-TLS, which can consume prohibitive amounts of cycles and RAM on Class 0 and Class 1 devices and, in some cases, even Class 3 devices. Research indicates that standard authentication can take nearly half a second and significant memory on lower-end microcontrollers, making "zero-touch" provisioning and delegation mechanisms essential. Engineers must weigh the unit economics of more expensive, capable hardware against the long-term technical debt of maintaining custom, lightweight security drivers and firmware.

[29] https://github.com/flatmarstheory/Wi-Fi-Intrusion-Detection-System/blob/main/README.md

When implementing higher-layer security, the choice of library and protocol version is as impactful as the hardware itself. TLS 1.3 and DTLS 1.3 have proven to be viable for constrained environments, often offering faster handshakes and comparable RAM footprints to their predecessors. ECC provides a much more efficient security-to-memory ratio than RSA; for instance, a 256-bit ECC key offers greater strength than a 2048-bit RSA key while being significantly easier for a constrained CPU to process and consuming less RAM.

A recurring architectural solution for resource scarcity is security offloading. By delegating the intensive handshake process, such as the DTLS handshake or TLS session establishment, to a more capable edge gateway or AP, devices can achieve up to 15x energy savings. This "proxy" pattern allows legacy or extremely constrained devices to benefit from enterprise-grade protection, such as VPN termination and ML-based monitoring, without requiring hardware upgrades or complex local processing.

Engineers must account for the inherent vulnerabilities in Wi-Fi management frames. Even with WPA3, devices remain susceptible to deauthentication attacks and TWT exploits that can lead to rapid battery depletion or permanent denial of service. Because constrained devices lack the resources for local threat detection, the consensus in the literature points toward network-level IDS. Using machine learning at the infrastructure or edge level allows for high-accuracy threat classification without placing any additional burden on the IoT endpoints themselves.

Key Takeaways for IoT Engineers and Administrators

- **Evaluate the "Security vs. Unit Cost" Trade-off:** Custom security logic saves hardware costs but incurs "technical debt" in the form of specialized firmware maintenance and patching.
- **Prioritize ECC over RSA:** Use Elliptic Curve Cryptography for a better security-to-memory/energy ratio, as it achieves higher bit-strength with smaller key sizes.
- **Optimize TLS Implementation:** Select high-quality libraries (e.g., WolfSSL) and favor TLS 1.3 for its shorter handshakes and reduced "start of data" latency.
- **Adopt a Proxy/Edge Architecture:** Delegate heavy cryptographic handshakes to a gateway or AP to extend the battery life of Class 0 and Class 1 devices.
- **Enforce Management Frame Protection (PMF):** Always enable 802.11w where possible to mitigate deauthentication and TWT-based battery depletion attacks.
- **Implement Network-Level IDS:** Do not rely on local device logs for security; use passive wireless monitoring and ML at the infrastructure level to detect anomalies.
- **Consider Multi-Layered Security:** Since Layer 2 remains vulnerable to DoS, implement lightweight end-to-end encryption at the application layer to protect data integrity.

- **Plan for Patching Cycles:** In custom deployments, the responsibility for security patches shifts to the developer; ensure a scalable mechanism exists to update firmware across the fleet.

Performance

Performance evaluation of Wi-Fi in constrained devices encompasses throughput, latency, packet delivery reliability, and scalability under the dual challenge of resource limitations and large device populations. Unlike conventional Wi-Fi deployments where performance is primarily a matter of user experience, constrained device performance directly impacts operational viability: a sensor node that cannot reliably deliver its data within the application's latency window fails its mission, and a network that cannot scale to the required number of devices is fundamentally unsuitable for the deployment. This section reviews the performance literature, examining both the capabilities of IEEE 802.11 standards designed for IoT and the real-world performance that constrained devices actually achieve.

IEEE 802.11ah Throughput and Range

Theoretical Characterization

IEEE 802.11ah (Wi-Fi HaLow) was designed as the first Wi-Fi standard specifically targeting IoT connectivity, operating in the sub-1 GHz band with data rates from 150 Kbps to 78 Mbps on a single spatial stream and ranges up to 1 km. Additionally, the standard supports more than 8,000 associated stations per AP and stations can be grouped (based on the association ID (AID)) by similar characteristics like location or battery level. The paper "Throughput and Range Characterization of IEEE 802.11ah" systematically characterized the throughput and range capabilities of 802.11ah, comparing it against IEEE 802.11n and 802.11ac (Baños-Gonzalez, Afaqui, Lopez-Aguilera, and Garcia-Villegas, 2016). Their analysis demonstrated that 802.11ah achieves improved performance over 802.11n and 802.11ac in terms of power received at long range across varying packet error rates, which is a key advantage for power-constrained devices far from the AP. Frame aggregation was shown to significantly improve effective throughput for constrained IoT nodes; its absence leads to notable efficiency losses due to protocol overhead. The sub-1 GHz band provides better obstacle penetration and range than 2.4 GHz and 5 GHz Wi-Fi, directly benefiting embedded sensor nodes in industrial or building environments.

Tian, Santi, Seferagić, Lan, and Famaey provided the most comprehensive recent survey of IEEE 802.11ah research, documenting that the standard supports up to 8,192 stations per access point and is up to six times more energy efficient than IEEE 802.15.4 while offering higher throughput and lower latency jitter than BLE (Tian et al., 2021). Their survey identified that RAW and TWT mechanisms are critical to handling contention among constrained IoT devices, and that improper configuration causes severe throughput and delay degradation. In addition, 802.11ah introduced TIM (Traffic Indication Map) segmentation, which allows the TIM to be broken into smaller groups to which only a subset of stations belong. This configuration allows stations to sleep longer before waking to monitor for downlink traffic indicated in Delivery TIMs (DTIMs) and then only wake again for their TIM group period if it is indicated in the DTIM.

The paper further explains the benefit of 802.11ah compared to Zigbee/BLE on the one hand and LoRa/NB-IoT on the other. 802.11ah offers moderate range (longer than Zigbee/BLE and shorter than LoRa/NB-IoT), in the 1-kilometer range, while also offering speeds of multiple megabits per second. This capability allows it to fill a gap not addressed by the other lower data rate protocols (Tian et al., 2021).

Practical Measurements

Maudet, Andrieux, Chevillon, and Diouris conducted indoor and outdoor measurement campaigns to evaluate 802.11ah performance in the EU-868 MHz frequency band using commercial Newracom NRC-7292 hardware (Maudet, Andrieux, Chevillon, and Diouris, "Practical", 2023). Wi-Fi HaLow achieved nearly 6 Mbps throughput for a 2 MHz bandwidth channel and ranges up to 1 km at 23 dBm transmit power, validating the standard’s claimed specifications. PHY-layer configuration (bandwidth, guard interval, modulation and coding scheme) significantly affects measured throughput and latency, providing network designers with levers to tune performance for specific IoT device constraints. It is important to note that, when using a module like the NRC-7292 with a host like a Raspberry Pi or other host device, the UART interface will not be able to achieve the highest supported 802.11ah data rates in many cases. On the other hand, SPI should be more than sufficient for such use cases.

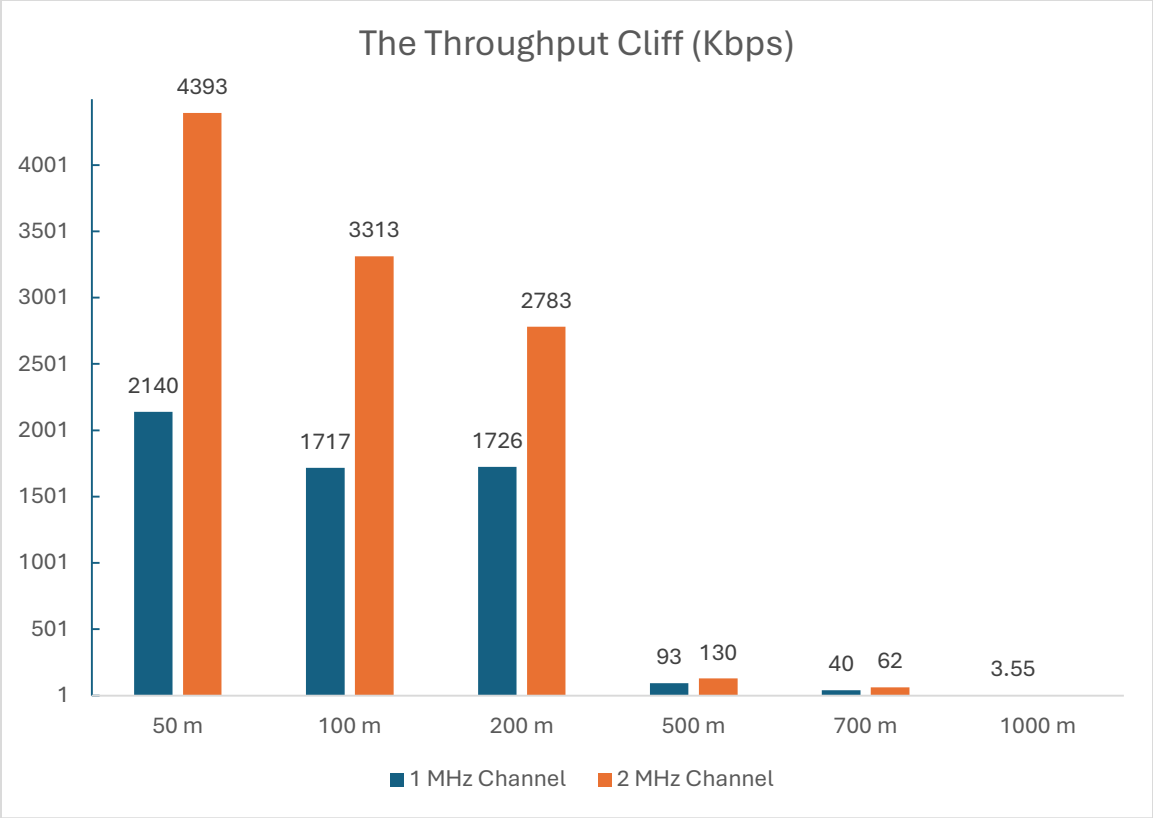
This paper also found that round trip time (RTT) was significantly better for 4 MHz channels than 1 MHz channels at 40 ms compared to 150 ms (Maudet, Andrieux, Chevillon, and Diouris, "Practical", 2023). They also indicated that the use of the short guard interval instead of the long guard interval improved performance by roughly 11 percent. Finally, the research showed that MCS (Modulation and Coding Schemes) of 0 to 2 provided very stable connections with almost 0 percent packet loss and the higher MCS indexes, such as MCS 7, experienced significant packet loss fluctuations over time.

This suggests that critical IoT alerting systems should use the lower MCS rates to provide improved reliability.

A very important paper provided the first authentic experimental field evaluation comparing Wi-Fi HaLow against IEEE 802.11n and LoRa across distances up to 1,000 meters in a smart grid context (Kane, Liu, McKague, and Walker, 2023). Their results revealed a significant gap between theoretical and actual performance: at 50 meters using MCS7 with 4 MHz bandwidth, actual throughput was only 35–71% of the theoretical maximum (4,680 Kbps versus 13,500 Kbps theoretical). Latency increased sharply with distance, from 8–10 ms at 50–200 meters to 60–116 ms at 500–700 meters, making time-sensitive control applications infeasible at longer ranges. Packet loss was negligible below 200 meters but became substantial at 500 meters and beyond. The study concluded that Wi-Fi HaLow is suitable for smart metering, firmware updates, home energy management, and EV charging IoT use cases, but not for distribution automation requiring sub-4-second latency at long range.

Probably the most important factor in the paper was the apparent "cliff" of throughput between 200 meters and 500 meters (Kane, Liu, McKague, and Walker, 2023). A partial reproduction of a graph in the paper is presented below, showing this extreme falloff. Interestingly, the "cliff" occurred between 100 and 200 meters at another location and in both locations, there was no benefit of a 4 MHz channel over a 1 or 2 MHz channel beyond 500 yards. Additionally, the researchers found that line of sight was required to achieve better throughput than LoRa at 1 kilometer. Some of these issues may be addressed with better antenna selection, but this is beyond the scope of the research paper.

Another paper focused on the extreme edge, evaluating Wi-Fi HaLow performance on very low-power IoT nodes using off-the-shelf hardware with free-space path loss emulation (Scharer, Polonelli, and Magno, 2023). Energy per bit (EPB) as low as 27 nJ/bit was achieved at close range, and 45 nJ/bit at a free-space equivalent distance of 158 meters, which is highly competitive for battery-constrained sensor nodes. Applied to a wind turbine blade monitoring wireless sensor network, Wi-Fi HaLow projected a two-fold reduction in EPB and a ten-fold increase in effective bit rate compared to BLE. Additionally, the study found that a maximum UDP payload throughput of 17.9 Mbps could be achieved with only 1.5% packet loss on an 8 MHz wide channel. This study provides the strongest evidence to date that Wi-Fi HaLow can uniquely balance range, data rate, and power efficiency for demanding embedded IoT applications. The performance results in this study were better than those in previous studies and, given that they did not use the same hardware, this improvement could be based on improved hardware, or it could be based on the conditions during the test.



Clarity Moment: Research only goes so far. When you are engineering a solution, particularly at range, it is essential to perform your own tests in the real environment where you will deploy the system. The fact that we have mixed results in the research using different hardware, in different contexts, likely with different protocol stacks, and more, and we see different results should inform you that it is necessary to test your equipment with your software in your environment[30].

802.11ah MAC Layer Efficiency and the RAW Mechanism

Corrected RAW Performance Models

The RAW mechanism is 802.11ah's primary tool for managing channel access contention among large numbers of IoT devices. Khorov, Krotov, Lyakhov, Yusupov, Condoluci, Dohler, and Akyildiz made a critical contribution by identifying and correcting systematic errors in prior RAW studies (Khorov, et al., 2019). Their new mathematical model yields significantly more accurate performance predictions,

[30] If you wish to read about the process of designing, implementing, and testing an SoC for 802.11ah, the IEEE Communications Magazine article *Wi-Fi HaLow for Long-Range and Low-Power Internet of Things: System on Chip Development and Performance Evaluation* is a good place to start. It is a brief read and is available online at: https://www.researchgate.net/publication/353592733_WiFi_HaLow_for_Long-Range_and_Low-Power_Internet_of_Things_System_on_Chip_Development_and_Performance_Evaluation.

demonstrating that existing models oversimplified RAW behavior and overestimated real system throughput, sometimes by several times, particularly for short RAW slot durations. The corrected model accounts for three IoT traffic patterns (single-packet, random-size batch, and full-buffer) and enables accurate optimization of RAW slot count and duration to balance throughput, packet loss, and energy for thousands of constrained devices sharing a single AP.

One of the most interesting insights in this paper (Khorov, et al., 2019) was the clarification of RAW slot times related to backoff counters in the CSMA/CA process. Many early researchers assumed that the backoff counter would transfer from one RAW slot to the next in which a STA could participate. However, this is not how Wi-Fi HaLow works. Instead, it implements two "universes" of EDCAF states. The first is the EDCAF state related to unrestricted access or the time in which any STA is allowed to transmit if it wins access to the medium regardless of RAW slots. In other words, it is the time in which a RAW slot is not active. When leaving this unrestricted access time and entering a RAW slot, STAs will save their EDCAF state, including the backoff counter, into the "first backoff function state." Then when the unrestricted access time resumes, STAs can reload that first backoff function state and resume the countdown. However, there is also a "second backoff function state" and this is the one used within RAW slots. This EDCAF state is reset at the beginning of each RAW slot, so a new random number is selected and the countdown begins again. The result of this operation is that more collisions could potentially happen at the start of a slot and STAs may have to skip several slots before winning contention due to the reset each time. Both realities result in reduced throughput for STAs on the network and drive the benefit seen with longer RAW slot time windows in dense STA networks; however, shorter RAW slot times are better in networks with fewer STAs. Once again, one can see that proper configuration of parameters is key.

NOTE: The preceding paragraph contains some of the most important information to performance optimization in 802.11ah networks. However, not all off-the-shelf systems provide configuration parameters and custom firmware/drivers will be needed in those cases. Some, such as Morse Micro's Linux/hostapd-based implementation, expose the RAW configuration. In the hostapd case, the hostapd_s1g.cfg file can be configured so that the duration, number of slots, and start time can all be configured in the beacon announcements of the AP.

An additional factor that must be considered related to RAW slot durations is that a STA cannot start transmission of a frame if the slot time does not have enough time left to complete it. Therefore, the frame sizes traversing a network should be considered when configuring the slot durations. Additionally, research has shown that the best throughput typically comes with increased packet loss ration (PLR). In fact, the highest

throughput is often achieved when PLR exceeds 10 percent. Therefore, throughput is increased but latency may also be increased for some time-critical applications. To provide the QoS for those time-critical applications, overall throughput may have to be sacrificed. Also, the number of STAs in a RAW slot group impacts the throughput. Generally, when you have saturated traffic, you want fewer STAs in each group and when you have unsaturated traffic, you want more in each group. Properly configured hardware will see overall throughput increase to around 120-140 STAs in a group with unsaturated traffic. Saturated traffic should typically have less than 10 STAs (or even less than 5) in a group.

Sljivo, et al. investigated how RAW grouping and TIM segmentation influence scalability, throughput, latency, and energy efficiency under bidirectional TCP/IP traffic (Sljivo, Kerkhove, Tian, Famaey, Munteanu, Moerman, Hoebeke, and De Poorter, 2018). They established quantitative limits: up to 20 IP cameras can be reliably connected via IEEE 802.11ah with a maximum average data rate of 160 kbps, while up to 6,960 stations transmitting every 60 seconds can be connected over 1 km with no lost packets under optimal RAW/TIM configuration. TCP behavior was found to be severely impacted by the long delays introduced by the 802.11ah link layer, limiting end-to-end throughput for constrained nodes. The study underscores the importance of RAW and TIM parameter tuning as suboptimal configuration leads to significant throughput degradation and increased latency. They also show the importance of balancing energy efficiency and latency. With proper configuration, STAs can sleep 97 percent of the time or more; however, latency increases and in fact is often doubled with the most extreme configurations designed to conserve energy. The last note regarding Sljivo, et al. is that the cause of latency, specifically increased RTT, in 802.11ah is often that there is insufficient time in the RAW slot in which the frame is transmitted from the STA to the AP to also provide the TCP ACK (when TCP is in use). The AP has to wait until the next slot time to transmit the ACK to the STA. Implementers can design a spoofed ACK mechanism into the AP so that it can send an immediate TCP ACK to the STA, resolving this issue in most cases. However, the risk of acknowledging a TCP segment that actually fails delivery on the "other side" of the AP must be considered when implementing such a model.

Riza, et al. used a Markov Chain analytical model to systematically evaluate 802.11ah performance under varying mobility, node counts, and collision probability (Riza, Gunawan, and Arifin, 2023). Increasing the number of nodes leads to decreased throughput and increased delay and energy consumption, revealing clear scalability limits. Mobility degrades throughput and raises energy consumption, while higher collision probability causes throughput to drop and delay to rise in a direct inverse relationship. RAW slot duration must be carefully tuned to the actual contention level. At first glance, Riza, et al. seems to conflict with Khorov, et al, but the apparent

contradiction is on the focus of throughput analysis. Riza, et al. focus on individual STA throughput and Khorov, et al. focus on system throughput (overall throughput of the network consisting of an AP and the associated STAs). So, for the individual STA, throughput goes down as the RAW slot duration goes up. But for the system, throughput goes up as the RAW slot duration goes up. The designer must determine which is more important. How many APs do you need? How many STAs should be on each AP? How many RAW slot times and what duration should they use? These are all factors that impact the performance of the individual STAs and the network as a whole.

Scalability Testing in Real-World Testbeds

Chounos, et al. presented one of the first real-world IEEE 802.11ah testbed evaluations specifically designed to uncover unexpected performance behaviors in office deployments (Chounos, Kyriakou, and Korakis, 2025). Their results confirmed that intense network contention among constrained IoT devices causes significant throughput degradation that is not fully captured by simulation. Adjacent Channel Interference (ACI) in closely deployed wireless links drastically degrades performance and must be considered in dense IoT deployments. Energy consumption under contention conditions increases substantially, raising concerns for battery-operated sensor nodes. These real-world results expose performance behaviors that make optimal configuration planning essential before large-scale IoT deployment and these authors recommend a "3-Channel Rule," which suggests that you should use every third channel and not the "next" channel due to ACI carrier sense backoff. Some of the key findings of this research are shown in Table 4.

Metric	Simulation Predictions	Test Bed Reality
Max Throughput (1 MHz)	~0.6 Mbps	~0.45 Mbps
Energy per bit	Stable	Increases with contention
ACI Impact	Minimal	Up to 40% degradation

Table 4: Chounos, et al. findings in an implemented test bed for 802.11ah

802.11ax OFDMA for IoT

Multi-User Access and Dense Deployment Performance

IEEE 802.11ax (Wi-Fi 6) introduced Orthogonal Frequency-Division Multiple Access (OFDMA) for uplink transmissions, a fundamental departure from the pure contention-

based CSMA/CA mechanism that has governed 802.11 MAC-layer access since the standard's inception. While contention-based access is still there, it also provided for triggered delivery in the uplink from multiple STAs concurrently. Qu, et al. provided a comprehensive survey and performance evaluation of 802.11ax, demonstrating that the standard achieves four times the average per-user throughput of legacy 802.11 in high-density deployments (Qu et al., 2018). OFDMA enables simultaneous multi-user uplink transmissions, significantly reducing access delay for numerous low-data-rate IoT devices compared to CSMA/CA contention alone. Power efficiency features including TWT and Trigger-Based OFDMA allow constrained, battery-operated IoT devices to reduce idle listening and extend operational lifetime, though it has not been implemented in pure IoT deployments as much up to this point. Dense IoT deployments also benefit from Spatial Reuse and BSS Coloring, which reduce inter-device interference without requiring constrained nodes to perform complex interference management.

Avdotin et al. specifically studied OFDMA resource allocation for real-time applications with strict latency and reliability requirements (Avdotin, Bankov, Khorov, and Lyakhov, 2019). Their algorithm decreases delays for real-time IoT traffic by orders of magnitude compared to standard CSMA/CA, enabling sub-millisecond latency targets. Reliability of up to 99.999% is achievable for constrained IoT devices under the proposed algorithm, meeting industrial IoT requirements. Non-real-time traffic throughput is reduced only insignificantly, confirming that real-time QoS guarantees do not come at the cost of bulk data performance. Of course, there is a limit on the number of STAs that can be supported in a cell and once this limit is reached performance falls sharply. In fact, this limit is the *number of total RUs available* in the channel *minus the number of RUs granted for random access times two*:

$$(RUmax - RUrnd) \bullet 2$$

For example, if you have 18 RUs and 2 are assigned random access, the calculation is 16 times 2 or 32 total devices before you begin to fall below the 99.999% rate. In other words, the reliability at the highest level is only there for small networks and not for real-world networks that are often much larger. However, to give fair due to the researchers, they have revealed when the reliability begins to fall off and they have also provided the calculations we can use to see how many STAs should be able to perform with a desired level of reliability. Then again, if you are implementing OFDMA in 802.11ax and require 99.999% reliability, you may be required to implement more APs to achieve the goal. This work identifies OFDMA uplink scheduling as the key enabler for deterministic low-latency data transmission from resource-constrained stations.

Clarity Moment: Most IoT communications are very tiny when compared to traditional Wi-Fi data payloads. You can see this in the design of other wireless IoT protocols, for example, those based on IEEE 802.15.4, in that their payload size limits are in the KiB range and not the MiB range. Of course, those protocols can use higher layer fragmentation to get larger payloads through the network, but the reality is that sensor data is typically measured in kilobytes and even bytes rather than hundreds of kilobytes or megabytes.

Cross-Technology Comparisons

Understanding Wi-Fi HaLow's and Wi-Fi's performance relative to competing IoT wireless technologies is essential for deployment decision-making. Many of these comparisons have already been shown in this paper. Additionally, Verhoeven et al. evaluated Wi-Fi HaLow, NB-IoT, and LoRa across four smart city application scenarios including underground and above-ground deployments (Verhoeven, Kempinski, and Meratnia, 2022). Wi-Fi HaLow provides significantly higher throughput than LoRa and NB-IoT but with shorter effective range under difficult propagation conditions, a key trade-off for constrained device deployment planning. No single technology was found to be universally optimal: Wi-Fi HaLow is better suited to applications requiring higher data rates and IP-based communication, while LoRa excels in deep coverage or low-data-rate scenarios. Underground and obstructed environments significantly degrade Wi-Fi HaLow performance relative to open-area limits, underscoring that constrained device requirements must be matched to the specific wireless technology.

Additionally, LoRaWAN has constraints on duty cycle that vary by regulatory domain (Carpenter, et al., 2024). For example, in the United States, a STA cannot transmit for more than 400 ms. In Europe, a STA cannot exceed 1% duty cycle. Data rates are measured in bits per second (bps) or kbps instead of Mbps, like Wi-Fi. NB-IoT has its constraints as well. First, it is a subscriber-based network solution requiring subscription fee for access to the carrier. But it provides a very reliabile performance level as it operates in licensed bands. It supports downlinks up to 26 or 127 kbps and uplinks up to 62 or 159 kbps depending on the release version (13 vs. 14). NB-IoT is declining in the United States with LTE-M becoming more popular, but it is more available in China and Europe.

The field comparison by Kane et al. provides complementary evidence specific to smart grid applications, confirming Wi-Fi HaLow's superiority to some other IoT protocols in throughput and IP compatibility while documenting its limitations at extended range compared to protocols like NB-IoT, LTE-M, and LoRaWAN (Kane, Liu, McKague, and Walker, 2023). The performance of Wi-Fi HaLow begins to drop at 200 meters, and after 100 meters, narrower channels perform better than wider channels. Additionally,

latency spikes significantly at 500 meters (56-75 ms compared to 8-10 ms at shorter ranges).

Together, these cross-technology studies establish that Wi-Fi HaLow occupies a distinct niche: higher throughput and native IP support compared to LPWAN alternatives, but with range and coverage limitations that must be factored into deployment planning for constrained device networks.

Protocol Overhead Impacts

Protocol overhead is a recurring theme in constrained device performance research. Karmakar et al. surveyed the complex interactions between PHY/MAC layer enhancements in IEEE 802.11n and 802.11ac and their impact on transport and application layer protocols (Karmakar, Chakraborty, and Chattopadhyay, 2017). While these enhancements significantly raise physical data rates, they can have negative impacts on reliable end-to-end protocols, particularly TCP, reducing actual end-to-end throughput for constrained devices. Frame aggregation improves efficiency but requires buffering resources that constrained embedded devices may not have. Block acknowledgement reduces per-frame ACK overhead but adds complexity. TCP throughput collapse occurs when large packet bursts are sent to a constrained device that cannot handle the volume causing packet loss and adjustment of TCP window sizes. Depending on the TCP/IP implementation, these issues may occur again and again over time.

The “performance anomaly” problem is what occurs because of multi-rate STAs, which is inherent to Wi-Fi operation, where the data rate of client STAs varies by their received signal strength indicator (RSSI) or signal-to-noise ratio (SNR). This performance anomaly problem was defined in detail all the way back in 2006 in Razafindralambo, et al. as “a bad time sharing between stations transmitting at high bit rate (fast stations) and stations transmitting at slow bit rate (slow stations) … results in an unfair throughput, with slow stations throttling fast stations’ traffic” (Razafindralambo, Guérin-Lassous, Iannone, and FDIDA, 2006). The survey identifies open challenges in coordinating PHY/MAC and upper-layer protocols, many of which disproportionately impact resource-limited embedded devices.

Paul et al. addressed contention-related performance deterioration by proposing an adaptive contention window scheme for the 802.11 MAC layer (Paul, Karthikeyan, Bhavadharini, and Karthik, 2018). Network performance in IoT deteriorates significantly due to distributed contention among resource-constrained devices, and MAC-layer contention directly impacts transport-layer congestion, which creates a "snowball effect" where back-off delays at the hardware level are misinterpreted by protocols like TCP as network congestion, leading to unnecessary reductions in

transmission rate. The study found that standard 802.11 mechanisms suffer a sharp decline in Packet Delivery Ratio (PDR) as the number of nodes increases. For example, the proposed Adaptive Contention Window (ACW) maintains a PDR of roughly 95% at 50 nodes, whereas standard mechanisms drop significantly lower under the same density. Their adaptive scheme dynamically adjusts to actual device density, improving throughput and packet delivery ratio while confirming that standard 802.11 MAC parameters designed for laptops and smartphones are poorly suited to large-scale IoT deployments.

Razzaq and Rao proposed an IoT-QoS algorithm that simultaneously addresses security constraints and the limited hardware resources of Wi-Fi IoT devices (Razzaq and Rao, 2024). Their algorithm monitors energy levels, communication quality, and queuing delay at APs, using a streamlined identity management system to reduce authentication and authorization overhead based on a Random Forest machine learning algorithm. The proposed IoT-QoS algorithm achieves a PDR of 98.2%, significantly outperforming traditional methods that struggle with higher packet loss in constrained environments. The research demonstrates that throughput is increased to 4.2 Mbps, compared to only 2.1 Mbps in conventional Wi-Fi IoT models, effectively doubling the data handling capability of the constrained devices. Queuing delay was reduced to 12 ms, which is critical for the "time-sensitive applications" mentioned by the authors. Designers should note that the "streamlined" nature of the system comes from a centralized IDMS that handles authentication for all IoT devices at the Access Point (AP) level. This offloads the heavy cryptographic processing from the individual constrained devices. Performance evaluation in NS-3 demonstrated improvements across all measured metrics compared to conventional Wi-Fi IoT models, confirming that security overhead and identity management are major contributors to performance degradation, not just hardware limitations.

Shared-Channel and Performance

Da Silva, et al. investigated the paradigm of cooperative Wi-Fi sensing and communication on shared channels, evaluating the impact on latency, jitter, and packet delivery ratio as IoT devices simultaneously transmit sensed data (Da Silva, Cotrim, and Margi, 2025). The Wi-Fi sensing addressed here[31] is that of obtaining information about objects or people in the environment as they act as radio signal reflectors, diffractors, and/or scatterers as defined in Meneghello, et al., 2022. At lower sampling rates, acceptable performance was achievable (packet delivery ratio exceeding 80%, latency below 50 ms), but performance degraded significantly as sensor count and sampling frequency increased. Performance begins to fall off at a sampling rate of 200

[31] Wi-Fi sensing is in its infancy and is mostly relegated to the research literature at this point. But as it evolves and becomes desirable to organizations, engineers must be prepared to address it.

Hz. This study demonstrates that shared-channel Wi-Fi sensing is viable for constrained IoT at moderate data rates but requires careful network planning at higher density.

Synthesis and Themes

The performance of constrained Wi-Fi devices is defined by a balance between range, throughput, and power consumption. While IEEE 802.11ah (Wi-Fi HaLow) serves as a cornerstone for long-range 802.11 IoT, research highlights a significant disparity between theoretical maximums and field reality. Practical measurements reveal a "performance cliff," where throughput drops sharply beyond 200 meters, and latency spikes to levels that may disqualify time-sensitive industrial automation. While sub-1 GHz frequencies offer superior penetration, achieving competitive performance at the kilometer mark often necessitates strict line-of-sight and specialized antenna configurations.

At the MAC layer, the efficiency of resource-constrained networks hinges on the RAW and TWT mechanisms. Because 802.11ah resets backoff counters at the start of each RAW slot, short slot durations in dense environments can lead to increased collisions and "starvation" of individual nodes. Conversely, while 802.11ax (Wi-Fi 6) introduces OFDMA to provide deterministic access and sub-millisecond latency, these benefits are mathematically capped by the number of available Resource Units. As device density exceeds these physical limits, the reliability of the network degrades, shifting the burden of performance back to architectural decisions such as AP density and frequency reuse patterns.

Protocol overhead and other capacity challenges, specifically the "performance anomaly" where slow-rate legacy devices throttle high-speed nodes, remain a persistent bottleneck. The mismatch between standard TCP/IP behaviors and the long-sleep cycles of IoT nodes can lead to throughput collapse and unnecessary retransmissions. Current literature suggests that the most successful deployments move away from "one-size-fits-all" configurations, instead utilizing adaptive contention windows and offloaded identity management to shield constrained CPUs from the exhaustive processing requirements of standard network stacks.

Key Takeaways for IoT Engineers and Administrators

- **Account for the "Performance Cliff":** Do not rely on theoretical sub-1 GHz range; anticipate a significant drop in throughput and a sharp rise in latency (e.g., from 10ms to >60ms) once devices exceed 200 meters.
- **Optimize RAW Slot Duration:** In high-density 802.11ah networks, use longer RAW slot durations to mitigate the throughput loss caused by backoff counter resets at slot boundaries.

- **Match Interface to Data Rate:** Ensure host-to-module interfaces (e.g., SPI vs. UART) do not become the bottleneck; SPI is generally required to realize the full multi-megabit potential of Wi-Fi HaLow.
- **Prioritize Lower MCS for Reliability:** For critical alerting systems, lock devices to lower Modulation and Coding Schemes (MCS 0–2) to maintain link stability and near-zero packet loss at the expense of peak throughput.
- **Implement the "3-Channel Rule":** In dense deployments, utilize every third channel to minimize Adjacent Channel Interference (ACI), which can degrade real-world performance by up to 40% compared to simulations.
- **Scale OFDMA Carefully:** When using 802.11ax for real-time tasks, limit the number of active stations per AP to the number of available Resource Units (minus random access RUs) to maintain 99.999% reliability.
- **Mitigate TCP Throughput Collapse:** Use spoofed ACK mechanisms or lightweight application-layer protocols to prevent the long latencies of IoT link layers from triggering TCP congestion control loops.
- **Environment-Specific Testing is Mandatory:** Because hardware implementation and local RF conditions vary wildly, empirical field testing in the target environment is the only reliable way to validate a performance budget.

Administration and Management

The administration and management of 802.11-connected constrained devices includes device provisioning, configuration, monitoring, firmware updates, security patch distribution, and network-level resource management across potentially large device fleets. These operational challenges are compounded by the defining characteristics of constrained devices: limited processing and memory resources restrict the complexity of management agents that can run on-device; limited energy budgets constrain the frequency and duration of management operations; and the sheer scale of IoT deployments, potentially thousands of devices per AP, demands automated, scalable management architectures that traditional WLAN management frameworks were never designed to support. This section reviews the literature on management challenges and solutions for Wi-Fi in constrained devices, organized around the key architectural and protocol themes.

IEEE 802.11ah Resource Management

MAC-Layer Resource Allocation

Farhad and Pyun's comprehensive survey provides the most thorough assessment of Wi-Fi HaLow's (IEEE 802.11ah's) built-in resource management capabilities for massive IoT deployments (Farhad and Pyun, 2022). The standard supports more than 8,100 stations per AP through a hierarchical Association ID (AID) structure that enables scalable IoT fleet management within a single basic service set. RAW allows the AP to assign time slots to groups of constrained stations, directly managing channel access and reducing collision overhead, which is a fundamental administrative control mechanism that operates at the MAC layer. TWT enables the AP to schedule wake and sleep times for individual IoT stations, serving both energy management and resource allocation functions. TIM segmentation reduces the overhead of beacon-based signaling for large-scale deployments where only a fraction of devices are active in any interval. Centralized Authentication Control reduces delay for large device fleets to approximately 5 milliseconds (compared to more than 20 ms), critical for managing large-scale deployments where devices may frequently enter and leave sleep states.

The survey identifies dynamic RAW configurations and machine-learning-based massive access algorithms as key future research directions for autonomous, scalable management of Wi-Fi HaLow IoT networks (Farhad and Pyun, 2022). The challenge of adapting RAW, TWT, and TIM parameters in real time to heterogeneous and time-varying IoT traffic remains unsolved in the standard, requiring management intelligence that goes beyond static configuration and, potentially, custom firmware/drivers if vendors do not begin to provide these configurable parameters. The interplay between these mechanisms, where optimizing one parameter (e.g., TWT interval) affects the others (e.g., RAW slot utilization), creates a complex multi-dimensional optimization problem that current administrative tools cannot address automatically. For example, while RAW settings can reduce energy consumption by keeping non-active STAs in sleep mode for more than 98 percent of the time, there is a trade-off where using more RAW slots increases energy efficiency but dramatically increases latency.

Device Association Management

Device association, the process by which an IoT station connects to an access point, is a critical management function that directly affects deployment efficiency and operational reliability. Nofal et al. proposed a framework that models AP association as a multi-objective optimization problem, solved using genetic algorithms to balance security requirements, throughput capacity, and energy efficiency (Nofal, Tran,

Dezfouli, and Liu, 2021). Their framework supports 35% more stations than competing approaches and defines association constraints based on signal thresholds, security level requirements, and per-AP computation capacity. They found that offloading the TLS handshake saves approximately 15x the energy compared to a conventional approach where the IoT device performs the handshake itself. APs sharing the same SSID form a management-transparent network where IoT devices associate based on security capacity and throughput requirements rather than just received signal strength, providing a principled basis for Wi-Fi fleet management decisions. The recommended management architecture allows administrators to trigger a STA handover when an AP reaches its capacity or a STA moves, ensuring the network remains balanced.

Authentication Offloading

The computational cost of IEEE 802.11 authentication creates a significant management challenge for constrained devices. As discussed in the security section, Kim, Min, and Han demonstrated that standard 802.11 combined with 802.1X EAP-TLS requires resources far exceeding what Class 0 and Class 1 devices can provide (Kim, Min, and Han, 2017). From an administration perspective, their delegation-based approach has important implications: the AP authenticates the Station-side Authentication Server once, and the SAS then guarantees authenticity for all registered IoT stations, greatly improving scalability for thousands of simultaneous device enrollments. The design allows security algorithm upgrades by replacing the SAS without affecting existing constrained devices, a key operational advantage that decouples security management from individual device capabilities.

This delegation pattern aligns with a broader trend in IoT management architecture: moving management intelligence and computational burden from constrained endpoints to network infrastructure elements. The management implications extend beyond authentication to include firmware distribution, policy enforcement, and monitoring or any function that requires more resources than the constrained device can provide or sustain.

IoT Management Protocols

LwM2M, SNMP, and CoAP-Based Management

The choice of management protocol is foundational to the administration of constrained Wi-Fi devices. Silva, Rodrigues, et al. provided a comprehensive evaluation of IoT management platforms and protocols, covering both network-level protocols (SNMP, NETCONF) and device-level protocols (CoAP Management Interface, OMA LwM2M) (Silva, Rodrigues, Al-Muhtadi, Rabêlo, and Furtado, 2019). Their survey identified SNMP (adapted as LNMP (LoWPAN Network Management Protocol) for 6LoWPAN,

but for 802.15.4 networks) as the dominant network management protocol valued for its simplicity and low memory footprint, though it lacks configuration resources for dynamic IoT management. OMA LwM2M over CoAP/UDP was highlighted as the most capable device management protocol for constrained IoT, supporting firmware-over-the-air updates, configuration control, and secure communications with minimal overhead. No single platform was found to address all IoT management requirements, and the survey identified energy-saving management, security standardization, and real-time context-awareness as the most pressing open challenges. Additionally, a common issue is the lack of a unified data model; even if two devices use LwM2M, they may not be interoperable if they define their objects differently. Another note is that MQTT can be an optimal communication mechanism for administrative operations that are not time-critical because it uses a publish/subscribe model so that STAs are not required to constantly poll the network.

Sinche et al. provided a complementary survey confirming that traditional SNMP is inadequate for constrained IoT devices due to its overhead, while newer protocols like CoAP-based COMI and LwM2M are purpose-built for constrained device management (Sinche, Raposo, Armando, Rodrigues, Boavida, Pereira, and Silva, 2020). The use of ASN.1 encoding used by SNMP is more CPU and memory intensive than the binary-efficient CBOR used by CoAP/COMI. OMA LwM2M was identified as the leading standard for remote IoT device administration, providing standardized objects for firmware updates, configuration, monitoring, and access control. RESTCONF and YANG data models show promise but require adaptation for resource-constrained nodes. One argument for the use of 802.11 instead of 802.15.4 is that standard RESTCONF/YANG payloads often exceed the 127 bytes of payload offered by 802.15.4, triggering fragmentation, and this is not a concern in 802.11 communications. Also, traditional network management is often "pull-based" where the manager asks and the device answers; however, CoAP allows an observe option that uses a "push-based" or publish/subscribe-like model where the device only sends updates when a state changes or a threshold is met. This shift reduces "heartbeat" or polling traffic, which preserves bandwidth in wireless channels and prevents network congestion in high-density deployments. The survey identified scalability to millions of constrained devices, security of management channels, energy-aware management operations, and support for heterogeneous wireless access as remaining challenges.

Clarity Moment: The IoT management protocols mentioned here are just a small representation of the possibilities. Many systems can be managed through REST APIs that rely only on HTTPS for communications. Others use remote CLI scripting across SSH. The key point is to have some method for mass remote management whenever possible because one is often managing many hundreds or even thousands of IoT nodes. Configuring or reconfiguring them one-by-one is simply not scalable.

SDN-Based Management

Software-Defined IoT Device Management

Software-Defined Networking offers a fundamentally different approach to IoT management by separating the control plane from the data plane, enabling centralized, programmable management of distributed constrained device networks. Mavromatis et al. presented the Software-Defined IoT Management (SDIM) framework, an SDN-enabled architecture for dynamic provisioning and fault detection across multi-domain wireless sensor networks at the network edge (Mavromatis, Colman-Meixner, Silva, Vasilakos, Nejabati, and Simeonidou, 2020). SDIM edge deployments lowered average device provisioning time by up to 46% compared to LwM2M and 60.3% compared to NETCONF Light, while reducing average operational fault detection time by approximately 33% versus LwM2M. CPU consumption was reduced and energy savings of up to 20% were achieved during device provisioning, extending device lifetime. The framework's superior provisioning speed is rooted in its use of Link Layer Discovery Protocol (LLDP) and MAC/IP address filtering rather than traditional IP-layer control messaging. This allows the SDN controller to maintain a WSN topology inventory and add flow rules more efficiently. The ability to dynamically provision IoT devices enables machine-to-machine communication without manual intervention, achieving near-autonomous network administration.

Li, Su, et al. proposed a Knowledge-Driven SDN architecture for IoT that adds a knowledge plane using IoT-generated data to drive intelligent network management (Li, Su, Ding, Lindgren, Liu, Prehofer, Riekki, Rahmani, Tarkoma, and Hui, 2020). SDN separation of control and data planes provides IoT networks with programmability, standardized APIs, global network view, and reduced management overhead. Energy-aware management, including duty cycles, sleep scheduling, and data aggregation, can be orchestrated via the SDN controller, extending operational lifetime of constrained devices. SDN-based mobility management using distributed hashing enables seamless device handover in Wi-Fi networks without disrupting IoT data flows. The architecture introduces IoT-proxies that translate SDN control commands into device-specific management primitives, bridging the gap between network administration and constrained device capabilities. It distinguishes between SDN Controllers and IoT Controllers. While the SDN controller optimizes general network resources and flow tables, the IoT controllers are service-specific. This allows admins to manage network performance (bandwidth, routing) separately from application-level device management (task scheduling, service logic).

OTA Firmware Updates

Secure Update Protocols for Constrained Devices

Over-the-air (OTA) firmware updates are a critical administration capability for maintaining the security, functionality, and compliance of deployed IoT device fleets. Schnor, et al. proposed MUP (MYNO Update Protocol), a secure OTA firmware update protocol for MQTT-connected constrained IoT devices (Schnor, Sahlmann, Nowak, and Clemens, 2021). MUP uses a two-phase approach: a signed manifest containing firmware hash and metadata is delivered first, allowing the device to authenticate the update before committing to the energy-intensive firmware download. The protocol achieves successful firmware updates on 32 KB RAM constrained devices without requiring TLS, using only ECDSA for authentication and AES for data security. Optimal slice size of 600 bytes with reply-to-address optimization delivers 87.8 KB firmware in 81.54 seconds, practical for wireless IoT devices with intermittent connectivity. Security properties include resistance to replay attacks, DoS and resource exhaustion, eavesdropping, and firmware injection, all without requiring per-device pre-shared keys.

Mahfoudhi, et al. presented a proof-of-concept OTA firmware update system for constrained NB-IoT devices, providing detailed power and latency measurements that are relevant to Wi-Fi-connected constrained devices as well (Mahfoudhi, Sultania, and Famaey, 2022). The system achieves zero packet loss at RSSI −85 to −80 dBm in field conditions, with transmission latency scaling linearly from 178 seconds for 2 KB to 4,338 seconds for 1 MB. Power consumption for a 120 KB update is approximately 38 J, and the system implements version control, integrity verification, segment retransmission, and backward compatibility. A key design feature is bootloader/application separation using the standard A/B partition approach, ensuring the device remains operable if an update fails, which is a critical safety requirement for remotely managed field deployments. Power measurements confirm that a 1 MB update consumes only 11% of a 100 mAh battery, establishing practical viability for constrained, battery-powered devices, assuming updates occur infrequently as 11% battery drain is significant for long-term deployed constrained devices. The management logic can be applied to 802.11 networks, though the power and latency measurements are specific to NB-IoT.

Automated Provisioning

Zero-Touch Device Onboarding

The initial provisioning and configuration of constrained IoT devices is one of the most labor-intensive aspects of IoT fleet management. Silva, Mitschang, et al. proposed an automated six-step methodology supported by a software toolchain for IoT device

provisioning (Silva, Mitschang, Képes, Wieland, Hirmer, and Breitenbücher, 2016). The approach distinguishes three device classes: Class 1 (directly connected), Class 2 (configurable/constrained), and Class 3 (not configurable). Automated provisioning reduces manual configuration effort to near-zero via SSH-based deployment for the gateways and Class 1 devices and constrained device configuration through the gateways. A Device Ontology using OWL 1.1 stores binding information for sensors and actuators, enabling automated discovery and configuration of wireless IoT devices.

Sousa, et al. addressed the identity management and authentication dimension of provisioning with YubiAuthIoT, which combines One-Time Password for user/administrator authentication with ECDSA public-key cryptography for device identity provisioning on the FIWARE IoT platform (Sousa, Magalhães, Resende, Martins, and Antunes, 2021). The full provisioning workflow completes in an average of 1,137.8 milliseconds (with nearly 40 percent of the provisioning latency attributed to the device-side cryptographic operations), and the OTP server handles 83.86 provisioning requests per second. Device pools enable decentralized fleet administration with hierarchical authentication. The system eliminates human configuration errors, identified as the dominant failure mode in IoT security, through semi-autonomous provisioning and secures the provisioning trigger through a physical hardware token held by the administrator, preventing unauthorized or remote-only device onboarding. While focused on FIWARE platforms, this research suggests the need for secure enrollment, fleet segmentation, lifecycle security, and minimizing the credential gap (devices managed via unsecure protocols or hardcoded admin credentials).

Self-Adaptive Device Management

Moualla et al. (2022) addressed the scalability failures of static IoT Device Management (DM) platforms by proposing a self-adaptive architecture for LwM2M servers (Moualla, Douet, Bolle, and Rutten, 2022). Legacy DM platforms designed for home devices fail to scale to IoT requirements because mass campaign operations such as fleet-wide firmware updates overwhelm static server configurations. The constraint-solving approach in this research provides formal guarantees on scaling decisions, avoiding both under-provisioning (devices unreachable for updates) and over-provisioning (wasted resources). Unlike static configurations, this autonomic approach scales the DM layer horizontally in response to "bursty" operations, such as fleet-wide firmware updates, while utilizing specific constraints to prevent scaling oscillations. Validated within Orange's industrial environment, the architecture demonstrates that by monitoring CPU, memory, and active session counts, a DM platform can maintain formal performance guarantees and high availability without the manual intervention typically required by legacy deployments.

From an administrative perspective, Alfonso, et al. highlight that IoT systems are inherently unstable due to the "dynamic nature" of their environments, where unexpected events like signal fluctuations and device aging can compromise system integrity (Alfonso, Garcés, Castro, Cabot, 2021). For engineers, the core management task is ensuring QoS, encompassing communication, computation, and "things" (devices), by implementing self-adaptive systems. These systems allow administrators to move beyond static configurations by enabling the network to modify its behavior at runtime in response to environmental shifts. This is particularly critical in high-stakes domains like healthcare or industrial monitoring, where a management failure resulting in late alerts could have catastrophic consequences.

Engineers must design management frameworks capable of detecting and responding to six primary categories of dynamic events: client mobility, variable data transfer rates, sensor-detected emergencies, software failures/aging, network connectivity fluctuations, and cyber-attacks (Alfonso, Garcés, Castro, Cabot, 2021). Effective administration relies on constant monitoring of specific QoS metrics, primarily CPU and memory consumption, to identify when edge or fog nodes are becoming overloaded. The study notes that while resource consumption is the most frequently monitored feature, comprehensive management should also include factors like power consumption, especially for battery-powered devices, and communication latency to ensure the system meets real-time requirements.

To maintain system health, the authors identify four key adaptation strategies that administrators can employ: data flow reconfiguration, auto-scaling of services, semi-automatic software deployment/upgrades, and task offloading (Alfonso, Garcés, Castro, Cabot, 2021). Data flow reconfiguration, the most common strategy, allows managers to reroute traffic to nodes with the best available resources or lowest latency. The use of containerization technology (e.g., Docker) is presented as a vital management tool, offering faster deployment and better efficiency on heterogeneous edge hardware compared to traditional virtual machines. For future-proofing IoT implementations, administrators are encouraged to adopt "orchestration servers" that automate resource provisioning and software allocation based on predefined rules, thereby reducing the manual overhead of managing thousands of distributed devices.

Attarha and Förster identified a complementary gap: current IoT management systems cannot re-configure low-power edge devices in a timely manner, creating a fundamental disconnect between operational requirements and management capabilities (Attarha and Förster, 2024). Their micro-service modularization technique allows individual functional components of a constrained IoT device to be independently managed without recompiling the underlying firmware or affecting other services, reducing management overhead by targeting only affected components.

The approach explicitly balances manageability, performance, and design constraints, acknowledging that management operations themselves consume scarce resources on constrained devices.

Network Slicing for IoT

Fami, et al. proposed network slicing in Enterprise Wi-Fi networks through dynamic AP association to support differentiated IoT service provisioning (Fami, Hammami, Pham, and Nguyen, 2022). Their approach enables differentiated management of IoT devices with different QoS requirements on shared Wi-Fi infrastructure without hardware virtualization. A Reinforcement Learning-based online algorithm achieves near-optimal throughput while meeting slicing service requirements without full system state knowledge, enabling autonomous Wi-Fi management at scale. The state knowledge required is the SNR matrix of all consumers, the minimum bandwidth requirements, and the continuous network-wide observations of states. The stable matching heuristic provides near-real-time association decisions suitable for resource-limited Wi-Fi controllers. Dynamic AP association serves as the key lever for service isolation in Wi-Fi IoT networks, replacing per-device QoS configuration and providing a practical framework for managing co-existing IoT services on a common Wi-Fi network.

Synthesis and Themes

The administration of constrained Wi-Fi and Wi-Fi HaLow device fleets represents a shift from manual, per-device configuration to automated, architectural orchestration. Because Class 0 and Class 1 devices lack the local resources to host traditional management agents, the literature emphasizes a "delegation" pattern. By offloading resource-intensive tasks, such as the 802.1X authentication handshake or complex identity management, to a Station-side Authentication Server (SAS) or an edge gateway, administrators can scale deployments to thousands of nodes while extending battery life by up to 15x. This architectural decoupling also allows for seamless security and protocol upgrades at the infrastructure level without requiring physical or firmware interventions on the constrained endpoints themselves.

Effective management at scale further requires a transition from "pull-based" protocols like traditional SNMP to "push-based" or "observe" models found in OMA LwM2M and CoAP. These modern protocols utilize binary-efficient CBOR encoding and asynchronous notifications, significantly reducing the "heartbeat" traffic that otherwise congests wireless channels in high-density environments. The emergence of Software-Defined Networking (SDN) and Network Slicing provides a programmable framework for managing heterogeneous IoT traffic. By separating the control plane, administrators can dynamically provision devices, isolate service types (e.g., healthcare monitoring vs.

HVAC control), and achieve near-autonomous fault detection, reducing provisioning times by nearly half compared to legacy methods.

The lifecycle management of these devices hinges on robust Over-the-Air (OTA) update mechanisms and self-adaptive system designs. Modern update protocols mitigate the risks of "bricking" remote hardware by employing A/B partition schemes and manifest-based pre-verification, ensuring that a 1 MB update does not deplete a device's entire energy budget. As environments shift due to signal fluctuations or node mobility, self-adaptive architectures that monitor CPU, memory, and latency in real-time allow the network to horizontally scale or reconfigure data flows autonomously. This "knowledge-driven" approach transforms the role of the administrator from a manual configurator to an orchestrator of intelligent, self-adaptive systems.

Key Takeaways for IoT Engineers and Administrators

- **Prioritize OMA LwM2M over SNMP:** Utilize LwM2M over CoAP/UDP for device management to benefit from smaller binary footprints (CBOR) and "push-based" reporting that preserves bandwidth and battery.
- **Offload Cryptographic Management:** Centralize authentication and TLS handshakes at the Access Point or an edge gateway to shield constrained CPUs from prohibitive cycle consumption.
- **Implement A/B Partitioning for OTA:** Always use a dual-partition bootloader approach for firmware updates to ensure a "fail-safe" state, preventing permanent denial of service during remote maintenance.
- **Leverage SDN for Rapid Provisioning: Adopt** Software-Defined Networking to reduce device onboarding time and automate the application of flow rules across large, multi-domain fleets.
- **Apply the "Observe" Pattern:** Configure devices to send updates only upon state changes or threshold breaches rather than constant polling to minimize network-wide contention.
- **Utilize Network Slicing for QoS:** Use dynamic AP association and reinforcement learning-based slicing to isolate critical IoT traffic from bulk data without requiring hardware virtualization.
- **Adopt Manifest-Based Updates:** Deliver a small, signed manifest first to allow the device to verify update integrity and metadata before committing to the energy-heavy firmware download.
- **Monitor "Fog" Node Health:** Shift monitoring focus from individual endpoints to edge/fog nodes, using CPU and memory saturation as primary triggers for auto-scaling or task offloading.

- **Design for Self-Adaptation:** Build management frameworks capable of autonomous data flow reconfiguration to respond to environmental shifts like signal fading or node mobility without manual intervention.

Conclusion

The integration of Wi-Fi and Wi-Fi HaLow into resource-constrained environments represents a significant evolution in IoT, shifting the 802.11 protocol from a high-throughput consumer utility to a potential industrial and agricultural backbone. This literature review has explored the challenges of adapting a traditionally "power-hungry" technology to devices with constraints in memory, processing, and energy. By synthesizing current research, you can see that successful deployment hinges not on a one-size-fits-all approach, but on the strategic application of the RFC 7228 framework to balance the competing demands of energy efficiency, security, performance, and management.

The taxonomy provided by RFC 7228 remains the essential lens through which constrained nodes are viewed. While hardware capabilities have advanced since the standard's inception, the classification of Class 0 through Class 2 devices continues to dictate the feasibility of standard networking stacks. Research confirms that for Class 0 and Class 1 nodes, the "standard" assumptions of the Internet, such as the ability to maintain large TCP buffers or perform complex TLS handshakes, are physically unattainable. Understanding these hardware tiers, alongside the energy supply categories (E0–E9), allows engineers to move beyond generic IoT design and into specialized optimizations that account for the physical limits on code space and power draw.

Energy consumption remains the primary hurdle for Wi-Fi in the IoT space, with idle listening identified as the most significant drain on battery life. The literature shows the transformative role of IEEE 802.11ah (Wi-Fi HaLow) and its mechanisms, such as RAW and TWT, which allow devices to achieve multi-year lifespans by maximizing sleep intervals. However, empirical studies reveal a "reality gap" where protocol overhead and transmission power often exceed theoretical models. Achieving true ENO through harvesting requires a cross-layer approach, where MAC-layer scheduling is tuned to the specific bursty nature of Wi-Fi traffic to prevent depleting limited energy buffers.

Security in constrained environments is a study in computational trade-offs, where the overhead of WPA3 and EAP-TLS can become a vector for exhaustion attacks. The research highlights two primary paths forward: the adoption of more efficient

cryptographic primitives, such as Elliptic Curve Cryptography (ECC) instead of RSA, and the use of architectural delegation. By offloading intensive handshakes to a Station-side Authentication Server (SAS) or edge proxy, even the most limited Class 0 devices can benefit from enterprise-grade protection. The emergence of battery-depletion attacks targeting power-saving triggers necessitates that security and energy management be co-designed rather than treated as independent variables.

The performance of Wi-Fi IoT is characterized by a "performance cliff" at extended ranges, particularly for sub-1 GHz deployments. While Wi-Fi HaLow offers a unique niche among IoT-specific protocols of megabit-per-second throughput at distances up to one-kilometer, real-world factors like ACI and line-of-sight requirements often degrade these theoretical maximums by up to 40%. The literature suggests that in high-density or long-range scenarios, reliability must be prioritized over peak speed, often requiring engineers to lock devices into lower MCS rates. However, the use of OFDMA in Wi-Fi 6 provides a path toward deterministic latency, provided the network size remains within the physical limits of available Resource Units.

Managing massive fleets of constrained devices requires a fundamental shift from manual configuration to automated, push-based orchestration. Traditional "pull" protocols like SNMP are increasingly replaced by binary-efficient models such as OMA LwM2M over CoAP, which minimize the "heartbeat" traffic that can congest wireless channels. SDN and network slicing have emerged as tools for isolating critical IoT traffic and reducing provisioning times. The literature emphasizes that lifecycle management, particularly secure OTA updates, must utilize fail-safe mechanisms like A/B partitioning to ensure that a management operation does not inadvertently result in a permanent denial of service for remote hardware.

The successful deployment of Wi-Fi for constrained devices requires engineering expertise and real-world field testing. The research consistently demonstrates that while standard Wi-Fi provides the necessary IP compatibility and throughput, it is the specialized amendments, 802.11ah, 802.11ax, and the forthcoming 802.11bn, that provide the tools to overcome resource limitations. As the ecosystem matures, the focus must remain on bridging the gap between theoretical protocol design and the messy reality of RF environments, ensuring that constrained nodes remain secure, performant, and administratively viable with sufficient energy throughout their operational lifetimes.

Bibliography

Alfonso, I., Garcés, K., Castro, H., Cabot, J. “Self-Adaptive Architectures in IoT Systems: A Systematic Literature Review.” Journal of Internet Services and Applications, 2021. DOI: 10.1186/s13174-021-00145-8.

Attarha, Shadi, and Förster, Anna. "Empowering IoT Applications with Flexible, Energy-Efficient Remote Management of Low-Power Edge Devices." Proceedings of the 2023 International Conference on Embedded Wireless Systems and Networks (EWSN), 2024. DOI: 10.48550/arXiv.2405.01578.

Avdotin, Evgeny, Bankov, Dmitry, Khorov, Evgeny, and Lyakhov, Andrey. "OFDMA Resource Allocation for Real-Time Applications in IEEE 802.11ax Networks." 2019 IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM) / BlackSeaCom, IEEE, 2019. DOI: 10.1109/BlackSeaCom.2019.8812774.

Baños-Gonzalez, Victor, Afaqui, M. Shahwaiz, Lopez-Aguilera, Elena, and Garcia-Villegas, Eduard. "Throughput and Range Characterization of IEEE 802.11ah." arXiv preprint, 2016. DOI: 10.48550/arXiv.1604.08625.

Bel, Albert, Adame, Toni, and Bellalta, Boris. "An Energy Consumption Model for IEEE 802.11ah WLANs." arXiv preprint, 2015. DOI: 10.48550/arXiv.1512.03576.

Bhargava, Vishal, and Raghava, N. "An Enhancement for IEEE 802.11 STA Power Saving and Access Point Memory Management Mechanism." Electronics, vol. 11, no. 23, 2022, article 3914. DOI: 10.3390/electronics11233914.

Bodenhausen, Jörn, Grote, Laurenz, Rademacher, Michael, and Henze, Martin. "Adaptive Optimization of TLS Overhead for Wireless Communication in Critical Infrastructure." 2024 8th Cyber Security in Networking Conference (CSNet), 2024. DOI: 10.48550/arXiv.2411.01971.

Bormann, Carsten, Ersue, Mehmet, and Keränen, Ari. "Terminology for Constrained-Node Networks." RFC 7228, Internet Engineering Task Force, May 2014, https://www.rfc-editor.org/rfc/rfc7228.

Bormann, Carsten, Ersue, Mehmet, Keränen, Ari, Gomez, Carles "Terminology for Constrained-Node Networks – draft-ietf-iotops-722bis-05" RFC 7228bis, Internet Engineering Task Force, March 2026, https://datatracker.ietf.org/doc/draft-ietf-iotops-7228bis/.

Canavese, Daniele, Mannella, Luca, Regano, Leonardo, and Basile, Cataldo. "Security at the Edge for Resource-Limited IoT Devices." Sensors, vol. 24, no. 2, 2024, article 590. DOI: 10.3390/s24020590.

Carpenter, T., Foster, L., Morgan, P., Ramoul, D., Lessard, M., and Davis, J. “CWICP: Certified Wireless IoT Connectivity Professional Official Study and Reference Guide.” Certitrek Publishing, 2024.

Chounos, Kostas, Kyriakou, Katerina, and Korakis, Thanasis. "Scalability and Performance Evaluation of IEEE 802.11ah IoT Deployments: A Testbed Approach." 2025 IEEE International Conference on Distributed Computing in Sensor Systems (DCOSS-IoT), IEEE, 2025. DOI: 10.1109/DCOSS-IoT65416.2025.00131.

Da Silva, Hugo F., Cotrim, Jorge, and Margi, Cíntia. "Performance Evaluation of Shared-Channel Wi-Fi Sensing and Communication in IoT Networks." 2025 IEEE Latin-Caribbean Conference on IoT (LCIoT), IEEE, 2025. DOI: 10.1109/LCIoT64881.2025.11118540.

Dalal, N., Akhtar, N., Gupta, A., Karamchandani, N., Kasbekar, G. S., and Parekh, J., "A Wireless Intrusion Detection System for 802.11 WPA3 Networks," 2022 14th International Conference on COMmunication Systems & NETworkS (COMSNETS), Bangalore, India, 2022. DOI: 10.1109/COMSNETS53615.2022.9668542.

Fami, Foroutan, Hammami, Nessrine, Pham, Chuan, and Nguyen, Khoa T. "Slicing Wi-Fi Networks for Differentiated IoT Service Provisioning." IEEE Wireless Communications and Networking Conference (WCNC), IEEE, 2022. DOI: 10.1109/WCNC51071.2022.9771816.

Famitafreshi, Golshan, Melià-Seguí, Joan, and Afaqui, Muhammad Shahwaiz. "Enabling Energy Harvesting-Based Wi-Fi System for an e-Health Application: A MAC Layer Perspective." Sensors, vol. 22, no. 10, 2022, article 3831. DOI: 10.3390/s22103831.

Farhad, Asim, and Pyun, Jae-Young. "Resource Management for Massive Internet of Things in IEEE 802.11ah WLAN: Potentials, Current Solutions, and Open Challenges." Sensors, vol. 22, no. 23, 2022, article 9509. DOI: 10.3390/s22239509.

Fournaris, Apostolos P., Giannoulis, Spyridon, and Koulamas, Christos. "Evaluating CoAP End to End Security for Constrained Wireless Sensor Networks." 2019 10th IFIP International Conference on New Technologies, Mobility and Security (NTMS), IEEE, 2019. DOI: 10.1109/NTMS.2019.8763857.

Garcia-Villegas, Eduard, López-Aguilera, Elena, Demirkol, Ilker, and Aspas, Juan. "IEEE 802.11-Enabled Wake-Up Radio: Use Cases and Applications." Sensors, vol. 20, no. 1, 2020, article 66. DOI: 10.3390/s20010066.

Gebresilassie, Samson Kahsay, Rafferty, Joseph, Chen, Liming, Cui, Zhan, and Abu-Tair, Mamun. "Transfer and CNN-Based De-Authentication (Disassociation) DoS Attack Detection in IoT Wi-Fi Networks." Electronics, vol. 12, no. 17, 2023, article 3731. DOI: 10.3390/electronics12173731.

Kane, Luke, Liu, Vicky, McKague, Matthew, and Walker, Geoffrey R. "An Experimental Field Comparison of Wi-Fi HaLow and LoRa for the Smart Grid." Sensors, vol. 23, no. 17, 2023, article 7409. DOI: 10.3390/s23177409.

Kang, Namhi, Park, Jiye, Kwon, Hyeokjin, and Jung, Souhwan. "ESSE: Efficient Secure Session Establishment for Internet-Integrated Wireless Sensor Networks." International Journal of Distributed Sensor Networks, vol. 11, no. 5, 2015. DOI: 10.1155/2015/393754.

Karmakar, Raja, Chakraborty, Sandip, and Chattopadhyay, Samiran. "Impact of IEEE 802.11n/ac PHY/MAC High Throughput Enhancements over Transport/Application Layer Protocols — A Survey." arXiv preprint, 2017. DOI: 10.48550/arXiv.1702.03257.

Khorov, Evgeny, Krotov, Alexander, Lyakhov, Andrey, Yusupov, Ruslan, Condoluci, Maurizio, Dohler, Mischa, and Akyildiz, Ian. "Enabling the Internet of Things With Wi-Fi HaLow — Performance Evaluation of the Restricted Access Window." IEEE Access, vol. 7, 2019, pp. 127402–127415. DOI: 10.1109/ACCESS.2019.2939760.

Kim, Ki-Wook, Min, Sung-Gi, and Han, Youn-Hee. "An Authentication and Key Management Mechanism for Resource Constrained Devices in IEEE 802.11-based IoT Access Networks." Sensors, vol. 17, no. 10, 2017, article 2170. DOI: 10.3390/s17102170.

Kim, Dongdeok, Lee, Harim, Ahn, Hyeongtae, Park, Jae-Hyeon, Suh, Young-Joo, and Park, Young Deok. "MILD: Minimizing Idle Listening Energy Consumption via Down-Clocking for Energy-Efficient Wi-Fi Communications." Sensors, vol. 25, no. 4, 2025, article 1155. DOI: 10.3390/s25041155.

Kim, So-Yeon, Park, So-Hyun, Lee, Jung-Hoon, and Lee, Il-Gu. "Secure Triggering Frame-Based Dynamic Power Saving Mechanism against Battery Draining Attack in Wi-Fi-Enabled Sensor Networks." Sensors, vol. 24, no. 16, 2024, article 5131. DOI: 10.3390/s24165131.

Lee, Il-Gu. "Interference-Aware Opportunistic Dynamic Energy Saving Mechanism for Wi-Fi Enabled IoTs." Future Internet, vol. 9, no. 3, 2017, article 38. DOI: 10.3390/fi9030038.

Lemercier, François, and Orgerie, Anne-Cécile. "Towards an Energy-Efficient Wi-Fi: An Experimental Study on Recent Standards Power Consumption." IEEE WCNC 2025, IEEE, 2025. DOI: 10.1109/WCNC61545.2025.10978325.

Li, Shancang, Song, Houbing, and Iqbal, Muddesar. "Privacy and Security for Resource-Constrained IoT Devices and Networks: Research Challenges and Opportunities." Sensors, vol. 19, no. 8, 2019, article 1935. DOI: 10.3390/s19081935.

Li, Yuhong, Su, Xiang, Ding, Aaron Yi, Lindgren, Anders, Liu, Xiaoli, Prehofer, Christian, Riekki, Jukka, Rahmani, Rahim, Tarkoma, Sasu, and Hui, Pan. "Enhancing the Internet of Things with Knowledge-Driven Software-Defined Networking Technology: Future Perspectives." Sensors, vol. 20, no. 12, 2020, article 3459. DOI: 10.3390/s20123459.

Liu, Ruofeng and Choi, Nakjung. "A First Look at Wi-Fi 6 in Action: Throughput, Latency, Energy Efficiency, and Security." Proc. ACM Meas. Anal. Comput. Syst. 7, 1, Article 25, 2023. DOI: 10.1145/3579451.

Mahfoudhi, Farouk, Sultania, Ashish Kumar, and Famaey, Jeroen. "Over-the-Air Firmware Updates for Constrained NB-IoT Devices." Sensors, vol. 22, no. 19, 2022, article 7572. DOI: 10.3390/s22197572.

Maudet, Sébastien, Andrieux, Gilles, Chevillon, Romain, and Diouris, Jean-François. "Evaluation and Analysis of the Wi-Fi HaLow Energy Consumption." IEEE Internet of Things Journal, vol. 11, no. 18, 2024, pp. 29775–29789. DOI: 10.1109/JIOT.2024.3401862.

Maudet, Sébastien, Andrieux, Gilles, Chevillon, Romain, and Diouris, Jean-François. "Practical Evaluation of Wi-Fi HaLow Performance." Internet of Things, vol. 24, 2023, article 100957. DOI: 10.1016/j.iot.2023.100957.

Maudet, Sébastien, Andrieux, Gilles, Chevillon, Romain, and Diouris, Jean-François. "Refined Energy Consumption Model of an STA in a Wi-Fi HaLow Network." IEEE Transactions on Communications, vol. 73, 2025. DOI: 10.1109/TCOMM.2025.3535868.

Mavromatis, Alex, Colman-Meixner, Carlos, Silva, Antonio P., Vasilakos, Xenofon, Nejabati, Reza, and Simeonidou, Dimitra. "A Software-Defined IoT Device Management Framework for Edge and Cloud Computing." IEEE Internet of Things Journal, vol. 7, no. 3, 2020, pp. 1718–1735. DOI: 10.1109/JIOT.2019.2949629.

Meneghello, Francesca, Cheng Chen, Carlos de M. Cordeiro and Francesco Restuccia. "Toward Integrated Sensing and Communications in IEEE 802.11bf Wi-Fi Networks." IEEE Communications Magazine, 2022. DOI:10.1109/MCOM.001.2200806.

Moualla, Ghada, Douet, Marc, Bolle, Sébastien, and Rutten, Éric. "Self-adaptive Device Management for the IoT Using Constraint Solving." Proceedings of the 17th Conference on Computer Science and Intelligence Systems (FedCSIS 2022), 2022. DOI: 10.15439/2022F80.

Mozaffari Ahrar, Erfan, Wilhelmi, Francesc, Galati-Giordano, Lorenzo, Imputato, Pasquale, Menth, Michael, and Avallone, Stefano. "R-TWT in Wi-Fi 7 and Beyond: Enabling Bounded Latency, Energy Efficiency, and Reliability." IEEE ETFA 2025, IEEE, 2025. DOI: 10.1109/ETFA65518.2025.11205686.

Nofal, Ramzi A., Tran, Nam, Dezfouli, Behnam, and Liu, Yuhong. "A Framework for Managing Device Association and Offloading the Transport Layer's Security Overhead of WiFi Devices to Access Points." Sensors, vol. 21, no. 19, 2021, article 6433. DOI: 10.3390/s21196433.

Örs, Faik Kerem, Aydin, Mustafa, Bogatarkan, Aysu, and Levi, Albert. "Scalable Wi-Fi Intrusion Detection for IoT Systems." 12th IFIP International Conference on New Technologies, Mobility and Security (NTMS 2021), IEEE, 2021. DOI: 10.1109/NTMS49979.2021.9432662.

Park, Eun-Chan, and Adnan, Muhammad. "Improving Energy Efficiency in Idle Listening of IEEE 802.11 WLANs." Mobile Information Systems, vol. 2016, 2016, article 6520631. DOI: 10.1155/2016/6520631.

Paul, Anand, Karthikeyan, N., Bhavadharini, R. M., and Karthik, S. "Wireless Networking Performance in IoT Using Adaptive Contention Window." Wireless Communications and Mobile Computing, vol. 2018, 2018, article 7248040. DOI: 10.1155/2018/7248040.

Qu, Qiao, Li, Bo, Yang, Mao, Yan, Zhongjiang, Yang, Annan, Yu, Jian, Gan, Ming, Li, Yunbo, Yang, Xun, Aboul-Magd, Osama, Au, Edward, Deng, Der-Jiunn, and Chen, Kwang-Cheng. "Survey and Performance Evaluation of the Upcoming Next Generation WLAN Standard - IEEE 802.11ax." arXiv preprint, 2018. DOI: 10.48550/arXiv.1806.05908.

Rademacher, Michael, Linka, Henrik, Konrad, Jannis, Horstmann, Thorsten and Jonas, Karl, "Bounds for the Scalability of TLS over LoRaWAN," Mobile Communication - Technologies and Applications; 26th ITG-Symposium, Osnabrueck, Germany, 2022. URL: https://www.researchgate.net/publication/360779154_Bounds_for_the_Scalability_of_TLS_over_LoRaWAN.

Razafindralambo, T., Guérin -Lassous, I., Iannone, L., and Fdida, S. "Dynamic packet aggregation to solve performance anomaly in 802.11 wireless networks." Proceedings of the 9th ACM international symposium on Modeling analysis and simulation of wireless and mobile systems (MSWiM '06), 2006. DOI: 10.1145/1164717.1164761.

Razzaq, Ali Ahmed, and Rao, Kunjam Nageswara. "Improving the Performance of IoT Devices That Use Wi-Fi." International Journal of Reconfigurable and Embedded Systems, vol. 13, no. 3, 2024, pp. 748–757. DOI: 10.11591/ijres.v13.i3.pp748-757.

Reaz, Khan, and Wunder, Gerhard. "ComPass: Proximity Aware Common Passphrase Agreement Protocol for Wi-Fi Devices Using Physical Layer Security." IMIS 2021, 2021. DOI: 10.48550/arXiv.2103.06763.

Restuccia, Gabriele, Tschofenig, Hannes, and Baccelli, Emmanuel. "Low-Power IoT Communication Security: On the Performance of DTLS and TLS 1.3." IFIP/IEEE PEMWN 2020, 2020. DOI: 10.48550/arXiv.2011.12035.

Riza, Tengku Ahmad, Gunawan, Dadang, and Arifin, Ahmad Syarif. "The Evaluation of IEEE 802.11ah Performance Based on the Effect of Mobility, Node's Number, and Traffic Using the Markov Chain Model." Journal of Communications, vol. 18, no. 5, 2023, pp. 310–317. DOI: 10.12720/jcm.18.5.310-317.

Saini, Rahul, Halder, Debajyoti, and Baswade, Anand M. "RIDS: Real-time Intrusion Detection System for WPA3 Enabled Enterprise Networks." arXiv preprint, 2022. DOI: 10.48550/arXiv.2207.02489.

Sánchez Vital, Roger. "Contributions to Energy-Efficient Multi-Radio Architectures for Novel IoT Scenarios". 2024. Universitat Politècnica de Catalunya, PhD dissertation. DOI: 10.5821/dissertation-2117-442724.

Sanchez-Vital, Roger, Belogaev, Andrey, Gomez, Carles, Famaey, Jeroen, and Garcia-Villegas, Eduard. "A Primer on AP Power Save in Wi-Fi 8: Overview, Analysis, and Open Challenges." IEEE Communications Magazine, vol. 63, 2024. DOI: 10.1109/MCOM.004.2400486.

Sanchez-Vital, Roger, Gomez, Carles, and Garcia-Villegas, Eduard. "Exploring the Boundaries of Energy-Efficient Wireless Mesh Networks with IEEE 802.11ba." Internet of Things, vol. 28, 2024. DOI: 10.1016/j.iot.2024.101366.

Sanchez-Vital, Gomez, Carles, and Garcia-Villegas, Eduard. "Energy-Efficient Wireless Mesh Networks with IEEE 802.11ba: A New Architecture." Proceedings – IEEE Symposium on Computers and Communications, 2023. DOI: 10.1109/ISCC58397.2023.10218013.

Santi, Serena, Tian, Le, Khorov, Evgeny M., and Famaey, Jeroen. "Accurate Energy Modeling and Characterization of IEEE 802.11ah RAW and TWT." Sensors, vol. 19, no. 11, 2019, article 2614. DOI: 10.3390/s19112614.

Scharer, Nicolas, Polonelli, Tommaso, and Magno, Michele. "Pushing Wi-Fi HaLow to the Extreme Edge: A Performance Study on a Low-Power IoT Node." 2025 IEEE International Workshop on Advances in Sensors and Interfaces (IWASI), IEEE, 2025. DOI: 10.1109/IWASI66786.2025.11121933.

Schnor, Bettina, Sahlmann, Kristina, Nowak, Michael, and Clemens, Vera. "MUP: Simplifying Secure Over-The-Air Update with MQTT for Constrained IoT Devices." Sensors, vol. 21, no. 1, 2021, article 10. DOI: 10.3390/s21010010.

Sianipar, Virgo Jaya, Arif, Teuku Yuliar, Syahrial, Syahrial, Yunida, Yusra, Walidainy, Hubbul, and Munadi, Ramzi. "Impact of Target Wake Time (TWT) on Energy Efficiency in Wi-Fi 6 Networks for Real-Time and Non-Real-Time Applications." Proceedings of COSITE 2025, IEEE, 2025. DOI: 10.1109/COSITE68330.2025.11414306.

Silva, Ana Cristina Franco Da, Mitschang, Bernhard, Képes, Kálmán, Wieland, Matthias, Hirmer, Pascal, and Breitenbücher, Uwe. "Automating the Provisioning and Configuration of Devices in the Internet of Things." Complex Systems Informatics and Modeling Quarterly, no. 9, 2016, pp. 28–43. DOI: 10.7250/csimq.2016-9.02.

Silva, Jonathan de C., Rodrigues, Joel J. P. C., Al-Muhtadi, Jalal, Rabêlo, Ricardo A. L., and Furtado, Vasco. "Management Platforms and Protocols for Internet of Things: A Survey." Sensors, vol. 19, no. 3, 2019, article 676. DOI: 10.3390/s19030676.

Sinche, Sérgio, Raposo, Duarte, Armando, Ngombo, Rodrigues, Augusto, Boavida, Fernando, Pereira, Vasco, and Silva, Joel. "A Survey of IoT Management Protocols and Frameworks." IEEE Communications Surveys & Tutorials, vol. 22, no. 2, 2020, pp. 1168–1190. DOI: 10.1109/COMST.2019.2943087.

Sljivo, Arno, Kerkhove, David, Tian, Le, Famaey, Jeroen, Munteanu, Adrian, Moerman, Ingrid, Hoebeke, Jeroen, and De Poorter, Eli. "Performance Evaluation of IEEE 802.11ah Networks With High-Throughput Bidirectional Traffic." Sensors, vol. 18, no. 2, 2018, article 325. DOI: 10.3390/s18020325.

Sousa, Patrícia R., Magalhães, Luís, Resende, João S., Martins, Rolando, and Antunes, Luís. "Provisioning, Authentication and Secure Communications for IoT Devices on FIWARE." Sensors, vol. 21, no. 17, 2021, article 5898. DOI: 10.3390/s21175898.

Suárez-Albela, Manuel, Fernández-Caramés, Tiago M., Fraga-Lamas, Paula, and Castedo, Luis, "A Practical Performance Comparison of ECC and RSA for Resource-Constrained IoT Devices," 2018 Global Internet of Things Summit (GIoTS), Bilbao, Spain, 2018. DOI: 10.1109/GIOTS.2018.8534575.

Tarish, Hiba A. "Enhanced IoT Wi-Fi Protocol Standard's Security Using Secure Remote Password." Periodicals of Engineering and Natural Sciences, vol. 10, no. 1, 2022. DOI: 10.21533/pen.v10i1.2728.

Thankappan, Manesh, Rifà-Pous, Helena, and Garrigues Olivella, Carles. "Multi-Channel Man-in-the-Middle Attacks Against Protected Wi-Fi Networks: A State of the Art Review." Computer Networks, vol. 209, 2022, article 108918. DOI: 10.48550/arXiv.2203.00579.

Tian, Le, Santi, Serena, Seferagić, Amina, Lan, Julong, and Famaey, Jeroen. "Wi-Fi HaLow for the Internet of Things: An up-to-date Survey on IEEE 802.11ah Research." Journal of Network and Computer Applications, vol. 182, 2021, article 103036. DOI: 10.1016/J.JNCA.2021.103036.

Venkateswaran, Shyam Krishnan, Tai, Ching-Lun, Ben-Yehezkel, Yoav, Alpert, Yaron, and Sivakumar, Raghupathy. "Extending Battery Life for Wi-Fi-Based IoT Devices: Modeling, Strategies, and Algorithm." ACM CoNEXT 2021, ACM, 2021. DOI: 10.1145/3479241.3486699.

Verhoeven, Richard, Kempinski, Stash, and Meratnia, Nirvana. "Performance Evaluation of Wi-Fi HaLow, NB-IoT and LoRa for Smart City Applications." ACM International Conference on Information Technology for Social Good (GoodIT '22), ACM, 2022. DOI: 10.1145/3551663.3558596.

Xu, Zhiqi, Kane, Luke, Liu, Vicky, McKague, Matthew, and Li, Yuefeng. "Energy Consumption Modeling for Wi-Fi HaLow Networks." IEEE Open Journal of the Communications Society, 2025. DOI: 10.1109/OJCOMS.2025.3578864.

Zhang, Junqing, Duong, Trung Q., Woods, Roger, and Marshall, Alan. "Securing Wireless Communications of the Internet of Things from the Physical Layer: An Overview." Entropy, vol. 19, no. 8, 2017, article 420. DOI: 10.3390/e19080420.