ISSN E: 3006-1466
ISSN P: 3006-1458
CONTEMPORARY JOURNAL OF SOCIAL SCIENCE REVIEW
CJSSR
CONTEMPORARY JOURNAL OF SOCIAL SCIENCE REVIEW
Vol.02 No.04 (2024)

Difficulties in IoT Networking Using Transmission Control Protocol/Internet Protocol (TCP/IP) Architecture

Syed Irtaqa Naqi Naqvi
Department of Information Technology, Superior University, Lahore, Pakistan
Muhammad Haseeb Iqbal
Department of Information Technology, Superior University, Lahore, Pakistan
Sidra Yousaf
Department of Software Engineering, Superior University, Lahore, Pakistan
Dr. Syed Asad Ali Naqvi
Associate Professor, Department of Information Technology, Superior University, Lahore, Pakistan
Saleem Zubair Ahmed
Professor, Department of Software Engineering, Superior University, Lahore, Pakistan

Abstract
Lately, the "Internet of Things" (IoT) has acquired claim since it offers the capacity to interface an enormous number of gadgets with restricted assets. Most of the present Web of Things gadgets depend on TCP/IP conventions, especially IPv6. The discoveries up until this point, nonetheless, highlight the requirement for a more prominent variation of the TCP/IP convention stack to the Web of Things climate. The IETF has buckled down lately to alter the convention stack for IoT arrangement circumstances. Both new and stretched out conventions have been added to the TCP/IP convention stack because of these drives. Be that as it may, there are in every case new difficulties arising. In this paper, we look at the numerous arrangements recommended by the IETF and analyze the mechanical hardships of broadening TCP/IP to the Web of Things climate. We accept that a data driven network design would be a more down to earth way to deal with help IoT applications than standard IP-based frameworks, which are inadequate or lacking.
Keywords*: Internet of Things (IoT), Control Protocol/Internet Protocol (TCP/IP), Network Architecture*

INTRODUCTION

The organization of various PC types to give a broad extent of noticing and control applications is insinuated as the "Internet of Things" (IoT). Present day IoT structures use the open rules of the TCP/IP show suite, which was encouraged many years earlier for the wired overall Web, as the systems administration answer for handle the range of gadgets and applications from a few suppliers. In any case, as we will cover underneath, there are a few tremendous contrasts between IoT organizations and regular wired PC organizations. Settling these disparities will significantly affect network design and give significant obstructions to the reception of TCP/IP advances in the Web of Things setting. The objective of this examination is to completely look at the issues that the IoT biological system is confronting and to frame an arrangement for settling them later on.

IoT networks regularly utilize low-energy Layer-2 innovations like IEEE 802.15.4, Bluetooth LE, and low-power Wi-Fi as a result of force imperatives. These innovations generally have lower transmission rates and a lot more modest MTU than standard Ethernet associations. Subsequently, one of the underlying difficulties in planning IoT network conventions is changing bundle size for restricted associations. IoT hubs may not generally be turned on to save energy, rather than wired networks. Remote lattice arrangements are likewise expected for correspondence when an IoT framework is sent in regions without wired network foundation, like rainforests, undersea, or battle zones. The TCP/IP convention configuration presently faces extra troubles thus: First, the first IP addressing configuration doesn't permit network organizations to utilize the multi-connect subnet model. A versatile directing framework is the third thought. Furthermore, broadcast and multicast are expensive on a battery-fueled network since a solitary transmission requires a few multi-jump sending and may awaken a few lethargic hubs.

Most IoT applications use various sensors and actuators to play out an arrangement of normal checking and control abilities. Fast and adaptable assistance for name arrangement and exposure, security protection for data grouping and incitation works out, and a resource arranged correspondence interface like Illustrative State Move (REST) are crucial for their arrangement thoughts. Tragically, existing responses for these issues — an extensive parcel of which are comprehensively used in contemporary Web progressions — don't meet the necessities of Web of

ISSN E: 3006-1466
ISSN P: 3006-1458
CONTEMPORARY JOURNAL OF SOCIAL SCIENCE REVIEW
CJSSR
CONTEMPORARY JOURNAL OF SOCIAL SCIENCE REVIEW
Vol.02 No.04 (2024)

Things conditions. For example, in various IoT course of action circumstances, standard DNS-based name organizations are pointless due to the shortfall of structure support for committed servers. Application-layer content stores and delegates are frequently unbeneficial in strong association settings with unpredictable organization. Additionally, to the extent that show errands and resource usage, IoT contraptions are genuinely upset by channel-based security shows like TLS and DTLS, which are supposed to safeguard REST coordinated efforts.

Each of the aforementioned challenges is covered in detail in the remainder of this essay. Our goal is to understand the architectural justification for the difficulties that arise while implementing TCP/IP in an Internet of Things setting. Additionally, we review current approaches to issues that have been standardized or are currently being explored at the IETF and discuss why they often fall short of addressing the stated concerns. This research aims to offer guidance and insights for the development of the next IoT network topologies.

NETWORK LAYER PROBLEMS

IP, especially IPv6, was created for the modern Internet environment, where wired servers PCs, and laptops interact. This section will examine how IP has been modified to adapt IP and its partner protocols to the IoT environment, as well as the characteristics of hosts and networks that IP currently presumes do not exist in the IoT world.

MTU

In IoT networks, MTUs for limited low-energy associations are habitually somewhat low. For instance, IEEE 802.15.4-2006 determines a greatest actual layer outline size of 127 bytes. This stands as a glaring difference to the present IP organizations, which much of the time expect a base MTU of 1500 bytes or higher. The IPv6 convention has two plan choices that lead to issues for short MTU lines. Some time before the idea of the Web of Things arose, during the 1990s, this convention was created for the traditional Web. Most importantly, in light of the fact that IPv6 utilizes a 40-byte fixed-length header with discretionary expansion headers, short parcels have a high convention cost. Also, restricted associations can't meet the base MTU size of 1280 bytes expected by the IPv6 convention for all IPv6-able organizations.

6LoWPAN consolidates a transformation layer between the connection layer and the organization layer to help IPv6 in 802.15.4 organizations. This layer utilizes two strategies to take care of the recently portrayed issues: connect layer fracture and header pressure. Header pressure eliminates superfluous fields (such stream name and traffic class) and copy data (like the connection point identifier in the IPv6 address, which can be found from the L2 Macintosh address). The pressure strategy for UDP and expansion headers, which are regularly used in the Web of Things to account for application payloads, is additionally determined. Since association layer discontinuity darkens the genuine MTU size of 802.15.4, the organization layer appears to work over a standard-consistent association that can uphold a 1280-byte MTU. Nonetheless, it is far-fetched that numerous IoT applications will send bundles bigger than as far as possible.

Multiple-link network

Two forms of Layer-2 networks are distinguished by the current IPv4 and IPv6 subnet model: point-to-point connections, where only two nodes share the connection, and multi-access links, where several nodes use the same access medium. Both assume that communication between nodes within the same subnet may occur over a single hop. The IoT mesh network, on the other hand, is composed of several Layer-2 links that are connected without the need for any Layer-3 equipment (like IP routers). In effect, this creates a multi-link subnet model that the original IP addressing design did not anticipate (Hinden, R., & Deering, S. 2006).

The IETF community chose to adopt a 1:1 mapping between Layer-2 links and IP subnets in place of the multi-link subnet paradigm, as explained in RFC 4903, "Multi-Link Subnet Issues". Many current protocols already rely on the "one-hop" reachability concept, which is the main source of issues. First, TTL/HopLimit handling is affected when forwarding across several connections inside the subnet. In IP organizations, it is standard practice to limit correspondence to a particular subnet by setting the TTL/Bounce bind to 1 or 255 and guaranteeing that the worth remaining parts steady upon gathering. The multi-interface subnet engineering will impede any convention that utilizes this strategy since hubs that communicate IP traffic across various associations will unavoidably decrease the TTL/Jump Cutoff esteem. The subsequent issue is that interface perused multicast can't deal with multi-connect subnets assuming there are deficient multicast steering abilities. Along these lines, a few

ISSN E: 3006-1466
ISSN P: 3006-1458
CONTEMPORARY JOURNAL OF SOCIAL SCIENCE REVIEW
CJSSR
CONTEMPORARY JOURNAL OF SOCIAL SCIENCE REVIEW
Vol.02 No.04 (2024)

more established innovations that depend on connect perused multicast, like ARP, DHCP, Neighbor Revelation, and other steering conventions, will never again work on multi-interface subnets.

The aforementioned difficulties are essentially caused by a misalignment between the new IoT mesh networks and the conventional IP subnet notion. To get over these technical obstacles, one must either split the mesh network into several subnets with distinct prefixes or employ Layer-2 technologies to transparently glue a large number of connections into a single network (similar to bridging multiple Ethernet segments). A certain amount of intra-subnet routing capacity is required for the first approach; this will be covered in Section 2.4. Since prefix allocation needs to be communicated throughout the mesh network (for example, through prefix delegation) and link formation in a mesh may change over time in a dynamic environment, the second alternative complicates network architecture.

Multicasting performance

IP multicast is utilized broadly in numerous IP-based conventions to achieve one of two errands: telling everybody in a gathering or suggesting a conversation starter without knowing who to inquire. In any case, sending multicast bundles is a huge trouble for little IoT network organizations. To start with, since most remote Macintosh conventions refuse connect layer ACK for multicast, lost bundles can't be recuperated at the connection layer. Second, the source should communicate at the most reduced normal connection speed across all collectors in light of the fact that the concurrence of numerous Macintosh conventions (like different Wi-Fi variants) and connection layer rate transformation might result in multicast beneficiaries encountering changing information transmission speeds. Third, IoT hubs may every so often change to resting mode to preserve energy, which could bring about them missing specific multicast bundles. At last, when hubs are associated through a lattice organization, a multicast parcel should be shipped by means of various bounces over an immense number of pathways. This could awaken a ton of lethargic hubs and strain the generally obliged network assets.

To tackle the issues with multicast support, more seasoned conventions should be changed to restrict the utilization of IP multicast prior to being utilized in restricted IoT conditions. At the point when IoT hubs need to pass admonitions on to an enormous number of collectors, they can briefly store bundles at a known area and trust that the beneficiaries will get them through unicast on-request (in light of their dozing designs) rather than multicasting them. Rather than flooding the organization with multicast messages, clients can focus on data gathering by sending inquiries to choose hubs that are designed to answer questions when they wish to communicate them to a gathering. These innovative methods circumvent the difficulties of multicast support while still allowing for sleeping nodes by substituting on-demand unicast pulling for multicast.

IPv6 Neighbor Revelation enhancement for 6LoWPAN is one case of convention transformation (Stenberg, M., and Barth, S. (2016)). Multicast is utilized in the first IPv6 ND (Selander et al. 2017) to identify copy addresses, learn default passage switches, and make an interpretation of adjoining IP locations to Macintosh addresses. While adjusting ND functionalities to 6LoWPAN, the streamlined convention empowers the obliged hubs to revive Switch Notice data on request utilizing Switch Sales messages, rather than utilizing the switch to communicate Switch Ads consistently, which will either awaken or miss the hubs that are resting. The switches can likewise answer demands for address goal and copy address identification for the end has by keeping a vault of host addresses. This permits questioning hubs to send unicast messages with their inquiries to the default switches.

MPL, an alternate methodology proposed by the IETF roll WG, fundamentally changes the sending semantics of multicast across restricted networks (Hui, J., and Kelsey, R. 2016). Without requiring a multicast steering convention to keep up with geography data, MPL synchronizes among MPL forwarders (i.e., MPL hubs) utilizing controlled flooding to circulate multicast parcels over the entire multicast space. Each multicast bundle is recognized by both a succession number and the parcel generator ID to distinguish duplication. Moreover, ongoing parcels are stored by the MPL forwarders in a sliding-window design (otherwise called a FIFO cushion) for use in ensuing retransmissions. The ongoing ZigBee IP standard presently incorporates this new multicast sending instrument.

Mesh routing of networks

Common IoT network topologies can be classified as either star or peer-to-peer (Hui, J., & Thubert, P. 2011). The center point hub, (for example, a Bluetooth ace hub) goes about as the default passage for the fringe hubs in a star organization, making steering setup direct. In any case, the sign inclusion of a solitary center point hub restricts the sending size of the star geography, which makes it unacceptable for wide-region applications. Because the

ISSN E: 3006-1466
ISSN P: 3006-1458
CONTEMPORARY JOURNAL OF SOCIAL SCIENCE REVIEW
CJSSR
CONTEMPORARY JOURNAL OF SOCIAL SCIENCE REVIEW
Vol.02 No.04 (2024)

nodes in the mesh design relay packets to each other, there is more coverage. Effective packet forwarding inside the mesh requires a routing system since flooding the whole network is unaffordable. Network or link layers can be used to build mesh network routing. Using Layer-2 forwarders, the link-layer technique—referred to as mesh-under in the IETF jargon (Kushalnagar et al. 2013)—connects several connections into a single "one-IP-hop" subnet. Route-over, a network-layer technique, relies on IP routers to forward messages across several hops. The current solutions in each of these two categories are described in the remainder of this paragraph.

To help mesh networks constructed using IEEE 802.15.4 connections with link-layer routing, the IEEE made the 802.15.5 norm (Jacobson, V. et al. 2009). The fundamental strategy for relegating L2 addresses is to initially build a crossing tree over the lattice organization. The foundation of the tree gives its youngsters constant connection layer address blocks, which the kids thus give sub-blocks to their posterity. This tending to methodology guarantees that hubs with a similar precursor will have interface layer tends to that fall inside a similar reach. Every hub makes a 2-jump neighbor data set subsequent to relegating a location, which incorporates the neighbors' bounce distance, tree level, and address block range. From that point onward, the hubs start illuminating their nearby neighbors about the nearby connection state. While sending bundles to an objective that is multiple bounces away, the shipper hub utilizes a basic heuristic to choose the following jump that is relatively close from the traversing tree root (and subsequently more used to the organization design). The detriment of this approach is that assuming extra hubs join the organization powerfully, it very well may be important to rehash the location distribution methodology to represent topological changes.

RPL (IPv6 Steering Convention for Low-Power and Lossy Organizations) is presently the business standard answer for the lattice network directing issue, which is tended to by the IETF utilizing a course over approach. Like IEEE 802.15.5, RPL utilizes a spreading over tree called an Objective Situated DAG (DODAG) to address a gathering of hubs, with all coordinated ways finishing at the root. At the point when two hubs impart in a DODAG, their parcels initially go up to the root hub or a common progenitor prior to advancing downlink to the objective. Dissimilar to IEEE 802.15.5, which allots geography subordinate L2 addresses, RPL makes no assumptions as for IP address appropriation. This really stops coordinating segment assortment past the use of typical prefixes. Staying aware of such a controlling table is hard for centers near the root, which in the most over the top horrible situation need to keep on overcoming sections for every device in the subnet. Besides, RPL offers a "Non-Putting away" choice in which the root hub is the sole one keeping up with the directing table. The root hub should contain the entire source course data in the bundle header while sending messages through Down Connection ways. The "Non-Putting away" choice lessens memory use on non-root hubs, however it causes downstream bundle header sizes to develop, which is dangerous for little MTU organizations.

It ought to be noticed that the essential directing issue in IoT network networks is monitoring steering data for each host in a multilink design. In traditional IP organizations, this isn't an issue since switches or self-learning extensions might assist with sending and steering engineering. Notwithstanding, in constrained IoT settings, per-have courses are either stayed aware of by every cross section center using guiding shows, which uses a lot of memory, or talked with the IP bundle as source courses during sending, which dismisses the association layer's low MTU need. Because of IP's host-arranged correspondence semantics, coordinating will continue to be an immense difficulty in IP-based IoT network structures.

DIFFICULTIES AT THE TRANSPORT LAYER

TCP, the main transport layer protocol on the Internet, provides reliable delivery and congestion control at the transport layer of the TCP/IP architecture. For many years, TCP has been developed to effectively transport massive volumes of data across an extended point-to-point connection with low latency requirements. Every byte in the stream must be sent in sequence, as it reflects the communication between the sender and the recipient as a byte stream.

In any case, an assortment of correspondence designs that TCP is unprepared to deal with are normally experienced by IoT applications. To begin with, gadgets may oftentimes go into rest mode inferable from energy impediments, which makes it unthinkable for IoT applications to keep a supported association. Second, the expense of interfacing is exorbitant since a huge part of IoT network basically includes a small measure of information. Third, a few applications might require low inertness and can't endure the postpone brought about by the TCP handshake (e.g., gadget incitation). TCP's all together conveyance and retransmission procedure might cause head-of-line blockage

ISSN E: 3006-1466
ISSN P: 3006-1458
CONTEMPORARY JOURNAL OF SOCIAL SCIENCE REVIEW
CJSSR
CONTEMPORARY JOURNAL OF SOCIAL SCIENCE REVIEW
Vol.02 No.04 (2024)

and extreme deferral when utilized with lossy remote organizations. Furthermore, connect layer robotized rehash demand (ARQ), a component of most of remote Macintosh conventions, may weaken TCP execution on the off chance that the L2 retransmission delay is more noteworthy than the TCP RTO (Fairhurst, G., and Wood, L. 2002). Albeit some modern IoT guidelines, as ZigBee IP, actually require TCP backing, increasingly more IoT conventions, as BACnet/IP and CoAP (Tang, S. et al. 2020), have decided to involve UDP as the vehicle layer convention and integrate transport functionalities into the application layer, in this way transforming the vehicle layer into a multiplexing module. The meaning of use level outlining was featured by these advances (D. D. et al. 1990). Application-level outlining is a strategy that empowers an organization to distinguish explicit application information units (ADUs). This empowers more adaptable vehicle support, including the utilization of elective retransmission methodologies for various types of ADUs, more effective information conveyance through in-network storing, and that's only the tip of the iceberg. Tragically, application-level outlining isn't adequately upheld by the ongoing TCP/IP design, which keeps applications from implanting application semantics in network-level bundles.

Difficulties behind the application layer

Most Internet of Things applications use a resource arranged request response correspondence technique. For example, while control applications use actuators to request exercises from authentic articles, checking applications search for data from sensors. These applications seem to be available day Web organizations, which use the REST (Valid State Move) plan to confer at the application layer (Taking care of, R. T. 2000). Because of the Web's enormous omnipresence, the Snare of Things pack has been managing incorporating the REST plan into IoT applications. For example, the IETF focus working get-together made the "Constrained Application Show" (CoAP) standard (Stenberg, M., Barth, S., and Pfister, P. 016), a UDP-based data transmission show made arrangements for a constrained setting, to enable REST-style correspondence for Web of Things applications. The need of executing REST at the application layer shows how critical capacities like resource divulgence, putting away, and security are not maintained at the lower layers of the TCP/IP designing. We will see how existing IoT applications fill in those openings and the obstructions of their responses in this part.

Resource Discovery

A resource discovery mechanism, which enables programs to request or start operations on resources, is frequently required by resource-oriented communication architectures. A technique for resource discovery in common IP networks is DNS-based Service Discovery (DNS-SD) (Cheshire, S., & Krochmal, M. 2013). The potential of this technology to support IoT applications is, however, severely limited.

DNS-SD's essential objective is to work with administration disclosure when the help frequently alludes to a functioning application, (for example, a printing administration working on a printer). Then again, assets with regards to the Web of Things are more broad; they could contain IoT gadgets, sensor information, and different things notwithstanding administrations. Consequently, a more complete way to deal with perceiving different assets is expected for IoT asset ID. For instance, as opposed to utilizing DNS records to distinguish assets, CoAP utilizes a URI-based naming instrument (like HTTP). Subsequently, the IETF center working gathering made Center RD (Winter, T. et al. 2012), a CoAP-based asset disclosure technique that utilizes less restricted asset registry (RD) servers to hold metadata about assets put away on different gadgets.

Second, traditional service discovery frequently uses multicast when local contexts do not provide access to dedicated services like DNS and CoRE-RD. For example, DNS-SD sends messages for name resolution and service discovery on the local network using Multicast DNS (mDNS) (Cheshire, S., & Krochmal, M. 2013). However, link-local multicast does not work in Internet of Things environments.Synchronizing resource Meta-information across the network in a peer-to-peer fashion is one way to use multicast. The Home Networking Control Protocol (HNCP) (Zhang, Y., et al. 2015), for instance, is being developed by the IETF home net WG. It would use a synchronization technique dictated by the Distributed Node Consensus Protocol (DNCP) (Uma Maheswari, P., et al. 2019) to distribute home network settings.

It ought to be noticed that the failure of the organization and transport layers of TCP/IP to distinguish the assets showed by application-layer identifiers is the reason for such arrangements. For instance, the IPv6 Neighbor Revelation convention can find settings at the organization layer and lower, yet DNS-SD SRV records generally use IP locations and port numbers to distinguish administrations. Resource discovery should be a basic component

ISSN E: 3006-1466
ISSN P: 3006-1458
CONTEMPORARY JOURNAL OF SOCIAL SCIENCE REVIEW
CJSSR
CONTEMPORARY JOURNAL OF SOCIAL SCIENCE REVIEW
Vol.02 No.04 (2024)

of an efficient IoT network design since it is so prevalent in IoT applications that it eliminates the need for apps to develop their own unique solutions.

The process of caching

The client, or resource requester, and the server, or resource holder, must both be available simultaneously according to the TCP/IP communication architecture. Restricted devices, however, could often switch to sleep mode in IoT setups to conserve energy. Furthermore, it is sometimes difficult to build strong links between communication partners due to the dynamic and/or interrupted network environment. Because of this, caching and proxies are commonly used in IoT systems to guarantee efficient data dissemination. In addition to temporarily storing the return data until the inquiring nodes awaken, For the resting hubs, the picked intermediary hub can get to assets. Comparative solicitations from different hubs that utilization a similar intermediary can likewise be served by the stored information, which speeds up reaction times and monitors network transmission capacity. The asset beginning server may likewise appoint intermediary hubs (otherwise called turn around intermediary) to handle demands for its benefit to diminish client traffic and go down when important.

Despite the fact that CoAP and HTTP are helpful for application-level reserving, they have huge downsides with regards to the Web of Things. For the clients to exploit the substance reserving highlights, they should initially pick a forward or switch intermediary hub. The pre-designed reserve areas won't help each client hub. The asset revelation approach permits clients to find neighborhood intermediaries as needs be. Notwithstanding, the general framework turns out to be more mind boggling because of this methodology. Second, the recently picked intermediary point can quit working under powerful organization conditions with irregular availability. Exactly when the association geology changes, clients need to either re-plan or re-find delegates or quit using stores and mediators endlessly out. Third, when stores and delegates discourage the beginning to end affiliations expected by current security standards, protecting application data ends up being significantly really testing.

For caching to be viable and versatile with regards to the Web of Things, the organization design should have astute stores all through the organization and permit projects to utilize them without bringing about arrangement or correspondence costs. The organization layer should incorporate storing into the sending system and know about the application layer assets for each organization parcel to investigate the reserves as it traversed the organization. To make in-network reserves protected and trustworthy, an essential change in the security engineering is likewise required.

Privacy

IoT applications need a lot of security precautions since they directly interact with the real world. Channel-based security, such as TLS (Rescorla, E. 2018) and its datagram variant DTLS (Shelby et al. 2014), is the most widely used security architecture for IP-based applications. It creates a secure channel of communication between the client and the resource server. However, for some reasons, secured-channel solutions are not effective in IoT situations.

- The cost of setting up a protected channel is the primary test with channel-based security. Prior to communicating the underlying application information, TLS and DTLS require at least two security handshake rounds to validate a channel and arrange security boundaries.
- The second challenge is that until a channel is closed, its state must remain the same on both ends. Memory use may be severely strained in a heavily meshed network when a device has to connect to several peers simultaneously. It is important to note that this challenge creates a challenging trade-off when paired with the first. The attempt to solve one problem (for instance, reducing memory use by creating on-demand short-lived channels) could make the other problem worse.
- At last, when application information has left the channel, channel-based security doesn't ensure demand reaction security. This is especially difficult when application information is stored by center boxes (like reserves and intermediaries). Both asset proprietors and asset requestors need to have confidence in the center boxes' capacity to apply access control arrangements accurately and give exact, unaltered information.

The aforementioned limitations highlight the necessity of a unique security strategy for Internet of Things applications. An alternative paradigm called object-based security (Shelby et al. 2014) has been proposed by the IETF, which protects the application information unit rather than the channel that sends the information. For

ISSN E: 3006-1466
ISSN P: 3006-1458
CONTEMPORARY JOURNAL OF SOCIAL SCIENCE REVIEW
CJSSR
CONTEMPORARY JOURNAL OF SOCIAL SCIENCE REVIEW
Vol.02 No.04 (2024)

anyone who gets the information to confirm its legitimacy, paying little mind to the way things were obtained, every information thing ought to have the important validation data (like advanced marks). The creator of the information can scramble it with the goal that main the assigned beneficiaries can decipher it in situations when information mystery is an issue. Outside of the IoT arena, similar ideas for object-based security have surfaced, such as the continuing attempts to safeguard JSON objects at the IETF Jose WG (Barnes, R. 2014).

RECONSIDERING THE ARCHITECTURE

"All problems in computer science can be solved by another level of indirection," according to the well-known indirection idea. It does not, however, address the problem of excessive indirection layers, which is a true reflection of the current architecture of IoT networks.

An IP-based IoT stack's tiered design is shown in Figure 1. IoT applications frequently employ CoAP or HTTP as their communications protocol to support the REST interface. Applications usually need to connect to common services (such as object security support and the CoAP Resource Directory) above the message layer. To secure the communication channel, TLS and DTLS are placed just above the transport layer. Other infrastructure services, including ICMP, DHCP, Neighbor Discovery (ND), DNS, and RPL, are also necessary to support IP network connections.

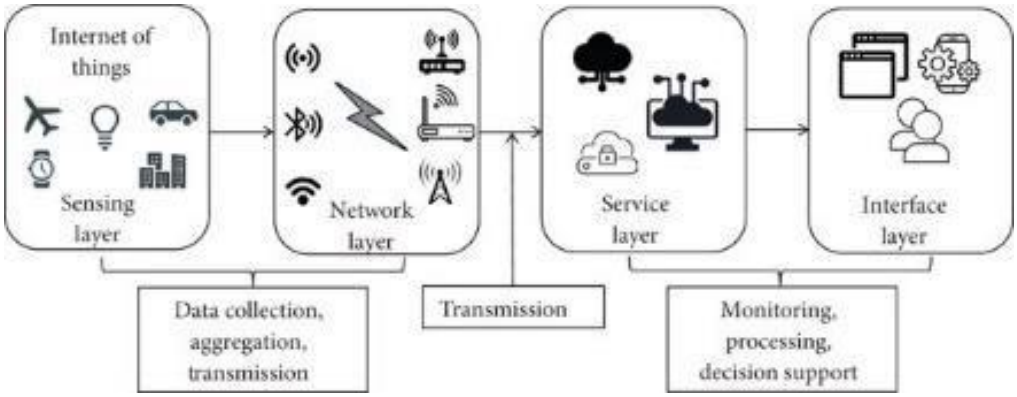

Figure 1: Typical architecture for IoT systems

Reconsidering the association stack and focusing in on its imperative parts from the standpoint of the application yields a very surprising picture, as shown in Figure 2. As a choice rather than "everything over IP," IoT applications have embraced "everything over REST." Any data transport, including 6LoWPAN and UDP, can be used at the lower some portion of an IoT stack. All of the assistance parts that unexpected spike popular for a single impression of the application data unit (ADU) given by IoT applications are completed by a Relieving educating show at the middle regarding the stack. A gigantic befuddle between the targets of IoT applications and the structure reality of TCP/IP is shown by the difference between this new viewpoint and the layered point of view on the old stack..

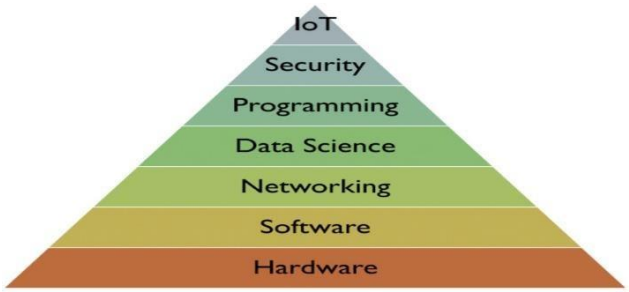

Figure 2: An IoT stack from the application's standpoint

Many sub-modules that implement crucial functions make up the REST layer:

ISSN E: 3006-1466
ISSN P: 3006-1458
CONTEMPORARY JOURNAL OF SOCIAL SCIENCE REVIEW
CJSSR
CONTEMPORARY JOURNAL OF SOCIAL SCIENCE REVIEW
Vol.02 No.04 (2024)

- A communication technique that uses URIs to send application-layer data to network locations.
- Effective data dissemination is ensured by a caching system.
- The confidentiality and integrity of individual ADUs are protected by an object security mechanism.
- A module for controlling congestion that may apply multiple techniques depending on the network conditions.
- Resource discovery and naming setup to assist with application operations.
- A system for sequencing large amounts of data that are too big to fit within one ADU
- A reliability mechanism that permits packets to be retransmitted and rearranged per the needs of the application.

These attributes are currently upheld by the application layer conventions (counting the REST interface). In any case, assuming such attributes had been remembered for the principal organization, they might have had more achievement. For example, organization and connection level info could assist blockage with controlling pursue better choices. Reserving might be more viable assuming stores were scattered over the organization as opposed to depending on specific reserving intermediaries. The organization layer should give URI-based sending, REST connection points, and item security to utilize in-network storing, which makes reserved content effortlessly found, open, and validated. Eventually, this convention stack upgrade created a more direct and viable engineering that firmly reflected the Data Driven Organization (ICN) thought.

ICN architectures like NDN manage lower-layer network problems in addition to enabling the functionalities needed by IoT applications. It returns packet flow control to the applications and employs the same ADU across layers. The reduced stack does not impose arbitrary minimum MTU requirements; instead, it lowers the number of packet headers. Because ubiquitous caching makes it possible for data to be efficiently reused by several users, it is inherently multicast-friendly. In addition to enabling scalable routing and forwarding via application layer names, its data-oriented communication removes the requirement to address and route to a large number of sensor nodes. IoT devices with constrained resources and erratic connection are more suited for data-centric security as they lower channel-based security solutions' overhead. The architectural simplicity leads to a smaller application software code size, a smaller device energy and memory footprint, and improved network resource utilization as compared to the current IP-based IoT stack. The IRTF has already taken notice of the potential for IoT through ICN, and as interest in IoT technology grows, we expect it to become a significant research topic.

CONCLUSION

The TCP/IP protocol stack was created in the early 1980s to use cable connections to link mainframe computers. The fundamental idea behind the architectural design remained the same, despite changes in the protocol stack following the publication of the IP standard. The IP architecture cannot easily accommodate IoT networks, a novel application type, without significant protocol stack changes.

In this study, we looked at the network and transport layer barriers to TCP/IP implementation in IoT networks. We also spoke about how protocols at the application layer, such as CoAP, offer solutions for necessary characteristics that are not possible at lower levels. By contrasting the current IoT stack with the architecture required from the application's point of view, the discrepancy was brought to light. In contrast to the existing application layer solutions, we proposed an architectural change that places REST-related elements in the core network layer, producing a more effective architecture. Using the ICN architecture, this new IoT stack would more successfully and organically include the required capabilities into the network.

REFERENCES

Tang, S., Shelden, D. R., Eastman, C. M., Pishdad-Bozorgi, P., & Gao, X. (2020). BIM assisted Building Automation System information exchange using BACnet and IFC. *Automation in Construction*, *110*, 103049.

Shang, W., Yu, Y., Droms, R., & Zhang, L. (2016). Challenges in IoT networking via TCP/IP architecture. *NDN Project*, *2*.

Barnes, R. (2014). *Use cases and requirements for JSON object signing and encryption (JOSE)* (No. rfc7165).

Cheshire, S., & Krochmal, M. (2013). RFC 6763: DNS-based service discovery.

ISSN E: 3006-1466
ISSN P: 3006-1458
CONTEMPORARY JOURNAL OF SOCIAL SCIENCE REVIEW
CJSSR
CONTEMPORARY JOURNAL OF SOCIAL SCIENCE REVIEW
Vol.02 No.04 (2024)

Cheshire, S., & Krochmal, M. (2013). Rfc 6762: Multicast dns.

D. D. Clark and D. L. Tennenhouse. Architectural Considerations for a New Generation of Protocols. SIGCOMM Comput. Commun. Rev., 20(4):200–208, Aug. 1990.

Deering, S., & Hinden, R. (2017). *Internet protocol, version 6 (IPv6) specification* (No. rfc8200).

Rescorla, E. (2018). *The transport layer security (TLS) protocol version 1.3* (No. rfc8446).

Fairhurst, G., & Wood, L. (2002). *Advice to link designers on link Automatic Repeat reQuest (ARQ)* (No. rfc3366).

Fielding, R. T. (2000). *Architectural styles and the design of network-based software architectures*. University of California, Irvine.

Hinden, R., & Deering, S. (2006). *IP version 6 addressing architecture* (No. rfc4291).

Hui, J., & Kelsey, R. (2016). *Multicast protocol for low-power and lossy networks (MPL)* (No. rfc7731).

Hui, J., & Thubert, P. (2011). *Compression format for IPv6 datagrams over IEEE 802.15. 4-based networks* (No. rfc6282).

Jacobson, V., Smetters, D. K., Thornton, J. D., Plass, M. F., Briggs, N. H., & Braynard, R. L. (2009, December). Networking named content. In *Proceedings of the 5th international conference on Emerging networking experiments and technologies* (pp. 1-12).

Kim, E., Kaspar, D., Gomez, C., & Bormann, C. (2012). *Problem statement and requirements for IPv6 over low-power wireless personal area network (6LoWPAN) routing* (No. rfc6606).

Kushalnagar, N., Montenegro, G., & Schumacher, C. (2007). IPv6 over low-power wireless personal area networks (6LoWPANs): overview, assumptions, problem statement, and goals.

Rescorla, E., & Modadugu, N. (2012). *Datagram transport layer security version 1.2* (No. rfc6347).

Selander, G., Mattsson, J., Palombini, F., & Seitz, L. (2017). Object security of coap (oscoap). *Internet Engineering Task Force (IETF) Internet-Draft work in progress*.

Shelby, Z., Chakrabarti, S., Nordmark, E., & Bormann, C. (2012). *Neighbor discovery optimization for IPv6 over low-power wireless personal area networks (6LoWPANs)* (No. rfc6775).

Shelby, Z., Hartke, K., & Bormann, C. (2014). *The constrained application protocol (CoAP)* (No. rfc7252).

Shelby, Z., Krco, S., & Bormann, C. (2014). CoRE Resource Directory; draft-ietf-core-resource-directory-02.

Stenberg, M., & Barth, S. (2016). *Distributed Node Consensus Protocol* (No. rfc7787).

Stenberg, M., Barth, S., & Pfister, P. (2016). *Home networking control protocol* (No. rfc7788).

Winter, T., Thubert, P., Brandt, A., Hui, J., Kelsey, R., Levis, P., ... & Alexander, R. (2012). *RPL: IPv6 routing protocol for low-power and lossy networks* (No. rfc6550).

Uma Maheswari, P., Manickam, P., Sathesh Kumar, K., Maseleno, A., & Shankar, K. (2019). Bat optimization algorithm with fuzzy based PIT sharing (BF-PIT) algorithm for Named Data Networking (NDN). *Journal of Intelligent & Fuzzy Systems*, *37*(1), 293-300.

Zhang, Y., Raychadhuri, D., Grieco, L. A., Baccelli, E., Burke, J., Ravindran, R., & Wang, G. (2015). Icn based architecture for iot-requirements and challenges. *ICNRG Draft draft-zhang-icn-iot-architecture-00*.