Comparison of networking protocols for edge devices in IoT systems

S1956488

The Internet of Things (IoT) refers to the process of connecting physical objects to the internet. IoT refers to any system of physical devices or hardware that receive and transfer data over networks without any human intervention. IoT devices are used all across the world in various industries to improve processes, and many of these industries rely on analyzing the data in real time or at least almost real time. The challenge is that the data being processed and analysed is so massive that for it to be analysed and processed in real time, it would have to be done close to where the data is generated. In a regular system, this would be impossible but due to edge devices and edge computing it is now possible. Edge computing brings data analysis, computation and storage closer to where the data is being gathered. This would enhance the IoT applications performance as it would no longer suffer latency issues cause by the data having to travel long distances. Furthermore it saves cost for the company as the processing, analysis and computation is done locally, in house, reducing the amount of data that has to be sent to a cloud-based location. IoT edge computing depends on devices to receive, process and output the IoT data which would involve a system of connectivity dependent on devices and sensors. Since the data being processed is exceptionally large, keeping the computation near the edge would prevent latency and operational issue. This is where edge devices come in. An edge device is typically internet-enabled and comprised of sensors. A common edge device is used to transmit data between the local network and the cloud. It is used to translate between the protocols used by the local devices into the protocols used by the cloud where the data will be further processed. Intelligent edge devices however, have built in processors, sensors and on-board analytics or artificial intelligence capabilities. This allows the device to process certain amounts of data directly on it, rather than uploading it to the cloud, and making decisions in milliseconds, hence improving the latency. Examples of edge devices in IoT systems are in autonomous vehicles. In the autonomous platooning of truck convoys, edge devices would make it possible to remove the need for drivers except in the first one as the trucks would be able to communicate with each other in ultra-low latency (see figure 1).Another example is for in-hospital patient monitoring. As of current, monitoring devices like glucose monitors are either not connected or are storing large amounts of the data in 3rd party clouds which could potentially be a security concern for the healthcare provider. An edge device on the hospital site could process data privately, hence maintaining data privacy. Since these edge devices captures, processes and communicates the data, it would have to communicate through some protocols just like other IoT devices. There are various networking protocols that edge devices can use, all with their own advantages and disadvantages.

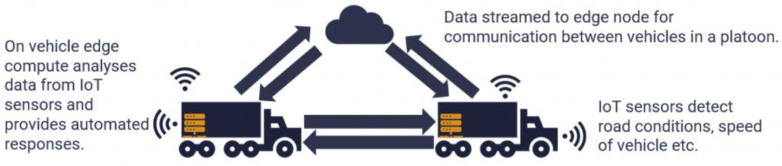
On vehicle edge compute analyses data from IoT sensors and provides automated responses.
Data streamed to edge node for communication between vehicles in a platoon.
IoT sensors detect road conditions, speed of vehicle etc.

Figure 1

Network protocols is an established set of rules that determine how data is transmitted between devices in the same network. It allows devices connected to the network to communicate with each other , irrespective of internal and structural differences. These protocols can be classified into 3 major categories; communication, management and security. Communication protocols are vital to the functionality of a network. These protocols describe the format and rules that whereby data is transferred over the internet. They also handle authentication, error detection, syntax, semantics and synchronization that both analog and digital communications must abide by to function. Examples of communication protocols are HTTP, TCP and UDP. Network management protocols define the policies and procedures used to monitor, manage and maintain the computer network. It also ensures a stable connection and optimal performance by communicating these needs across the network. Examples of management protocols would SNMP and ICMP. Lastly, security protocols are used to ensure that the data being transferred stays safe and secure. It also defines how the network secures the data from any attempts to review or extract by illegitimate means. This ensures the privacy as no unauthorized user will be able to access the data. Examples of security protocols are SSL, SFTP and HTTPS.

In general most edge devices should be able to run common network protocols such as CoAP (Constrained Application Protocol) ,MQTT(Message Queue Telemetry Transport) and AMQP (Advanced Message Queuing Protocol) . However, as shown in some scientific articles, the protocols chosen would need to be tailored to the requirements of the edge devices being used. Hence certain networking protocols would have an advantage over the other. To compare these protocols, there are various challenges that could be used to determine the suitability of a protocol over the other. These challenges are, security, quality of service, scalable, mobility, energy efficiency , throughput and reliability. Since the main advantages and usage of edge devices, are the low-latency that allows it to analyse and process data in real-time, and the security (where since the data is processed locally and not sent to a cloud, data privacy is ensured), most research papers found talk mainly about communication and security protocols. To compare the protocols, we will be looking at the security and pros and cons of different protocols that are used in edge devices in IoT systems.

Firstly we will be looking at the networking security protocols and we will be comparing the security quality of these protocols. When communicating data in edge devices, data-transmission should be secure so that “man-in-the-middle” and similar attack can be avoided. Some of the criterias being looked at is how vulnerable is the data so malicious users when it is being transferred by the edge device, trust issues of the protocol and how much we can trust that the data being transferred by the edge device is unaltered. The protocols being analysed and compared are shown below with a brief summary in table 1.

Protocols	Brief Description
SSL/ TLS	A protocol used for protecting sensitive data and securing internet connections. SSl allows both server to server communication and client to server communication. All data transferred through SSL is encrypted thus stopping any unauthorized person from accessing it. TLS is the successor of SSL. They both have similar goals but TLS has some changes such as the removal of support of legacy encryption algorithm and Zero Round Trip Time to enhance its performance.
IPSec	IPSec is a security suite that secure communication at the IP layer. It uses Internet Key Exchange (IKE) protocol to undergo negotiations like deciding on an encryption algorithm. IKE operates in 2 phases. In the first phase, the secure connections is established. In the second phase, the summetryc key used for encrypting messages is finalized. It is commonly used in virtual private networks to secure the communication between edge devices and edge gateways.
SSH	Secure Shell is used for secure login to a remote server. It is a low-level transport protocol that provides secure encryption, authentication, and message integrity protection. It is a common tool used in edge devices for the maintenance of said edge device. Message Authentication Code is paired with each packet sent to ensure data integrity
TCP/IP	TCP ensures that sent data packets sent in sequence without any errors. This is achieved by acknowledging that certain data was received.

SMTP	A protocol developed to ensure efficient and reliable transfer or electronic mail over multiple networks if necessary.

Table 1

As mentioned above, there are multiple ways to analyse and compare the security of network protocols for edge devices used in IoT systems. Some of these ways that were used to analyse the above protocols are; setting up a simulation testbed and using it to test security hypothesis , propose a new attack on the protocol in question and comparing the different implementations of the protocols to determine the better protocols.

Generally, TLS has great security qualities as it is free of security holes. However, when used in real-life, sometimes developers take shortcuts to save time and resources. This leaves TLS vulnerable to many different attacks , spanning over different versions and implementations of TLS. These attacks are listed below in table 2 and the attacks' assumptions to be feasible in table 3.

Attack	Description
Compression Attack	The attack happens only when TLS is used with TLS compression. Compression algorithms work in predictable ways. This attack exploits that feature and utilizes this fact to obtains the user's session key.
ChangeCipherSpec Message Drop	This Attack works only on SSL 2.0 which has since become outdated. The ChangeCipherSpec message can be dropped in this version which causes 2 parties to be in an insecure state.
Version Rollback	The version of TLS is downgraded to the lowest possible by the attacker to reduce the security between the 2 devices communicating
Padded Oracle	Padding is the act of adding data before or after a payload before encrypting. These attacks potentially allow the attacker to decrypt padded encrypted data
Cipher Suite Rollback	The attacker changes the messages during the handshake protocol. The attacker replaces the ciphers with weaker ciphers that are easy to attack. If any of the ciphers are accepted by the server then the communication and data transfer is compromised
CBC Attack	The attacks happens on connections using cipher block chaining mode of encryption. This is primarily a vulnerability in TLS1.0
SSL Stripping	The attacker tries to remove the usage of TLS completely by modifying the header data in another protocol that requests TLS.

Table 2

Type of attack	Running JS code in Victims Browser	Vulnerable sessions	Security Measures
CBC Attack	Yes	SSl 3.0/TLS1.0	Newer TLS versions
CBC Attack	Yes	TLS1.0-1.2	Add random time delays and Use RC4
Compression Attack	Yes	TLS1.0-1.1,TLS 1.2 with SPDY ext	Disable HTTP deflate Compression
Compression Attack	Yes	TLS 1.0-1.2	Disable HTTP compression

Padding Oracle Attack	Unnecessary	TLS 1.0-1.2	Disable SSL 2.0 support
Padding Oracle Attack	Unnecessary	TLS 1.0-1.2	Disable RSA encryption suites

Table 3

Another significant threat 2 TLS is TLS interception. It is a common practice in organisations to monitor their employees web traffic and is accomplished using the middlebox technology(acts as a man in the middle to communicate information between the user and the server) which is a device that uses a TLS interception proxy. The fact that TLS interception is common and goes unknown to the user shows how trustworthy TLS protocol is.

As for SSH, it has 2 main critical functions which are authentication and secure communication. The most widely used method of authentication used by SSH is the passwords and public key authentication. As for securing the communication between 2 devices, SSH uses a cipher suite that is negotiated on to allow the devices to communicate through symmetrical encryption. One common way to attack a server or device using SSH protocols is to the brute force or dictionary attack where the attacker attempts to guess the configured password. As brute-force attack is implementation agnostic, to stage this attack, researchers set up a "honeypot", which is meant to look like the real system to the attacker. Corporations use production honeypots to lower the risk of a successful attack by having a simulated system that appears real and would waste the hackers time. their methods can then be analysed to strengthen the system.

A dictionary attack on the other hand happens when the attacker attempts to guess the correct password to gain access to the SSH server and intercept data. They then use an SSH implementation to recover plaintext from ciphertext. The attack happens by sending the client a ciphertext block and depending on the response, sometimes the first 14 bits of the plaintext can be recovered. Further knowledge of what the plaintext contains, the attack success further improves.

IPsec is a suite of network protocols meant to provide secure communication between two devices.IPSec uses 2 different formats to secure IP packets which are Encapsulating Security Payload (ESP) and Authentication Header(AH). IPSec protocol suite lies in the Internet Key Exchange (IKE) protocol. This protocol handles negotiating security associations, which are a set of encryption algorithms and keys. IPSec provides integrity protection in the form of Message Authentication Codes (MAC) which can be done prior to or after encryption. As stated in the "Analysis of Network Protocols for Secure Communication" article, Researchers have considered the use case of IPSec to build a VPN and demonstrated practical attacks against every MAC-thenencrypt configuration of IPSec. These attacks are :

1) Chosen plaintext where the attacker can recover an arbitrary IPsec protectected plaintext
2) Fragmentation attack where the attacker creates packets that allow for replies to be received while bypassing authentication checks

As for TCP/IP protocols, there are various attacks that these protocols are prone to. These are:

1) Packet Sniffing. This enables a hacking device to obtain any incoming or outgoing data between a client and a server
2) Spoofing. The attacker simulates this attack to gain access to a computer or server whereby the attacker only needs the IP address of the trusted port.

We will now look at Network communication protocols used in edge devices and their comparisons. We will look at the common communication protocols that are reviewed most often by researchers and discussing their advantages and disadvantages over each other. The following table (table 4)gives a brief description of each protocol and their pros and cons .

Protocols	Description	Advantages	Disadvantages
HTTP (Hyper text transfer protocol)	A layer 7 protocol that is used for transferring a hypertext between 2 or	- Memory usage and CPU usage are low	- Lacks encryption capabilities

	more systems. It works on a client server model.	because of lesser concurrent connections. - Errors can be reported without closing connections.	- higher power consumption to establish communication and transfer data
TCP	Provides a reliable stream delivery by using sequenced acknowledgement. It is a connection-oriented protocol and is used for communicating over a network	- ensures the data reaches the destination on time, without duplication. - Automatically breaks data into packets before transmission	- Cannot be used for broadcast and multicast connections
UDP	It provides basic but unreliable message services. It is used when we want faster transmission but it adds no flow control, reliability or error-recovery functions	- Can be used for broadcast and multicast connections - Faster than TCP	- Does not ensure the same data sending reliability as TCP
ARP	Helps map logical addresses to physical addresses acknowledged in a local network.	- Arp cache contains MAC addresses so they do not need to be memoriesed	- Susceptible to spoofing attacks - Susceptible to ARP denial-of-services by hackers
IP	A protocol through which data is sent from one host to another over the internet	- encrypts data being transferred - routing data becomes more scalable and economical	- labor intensive,complex and provne to errors
DHCP	It is used for the method of automating the process of configuring devices on IP networks. It also allows devices to use services such as NTP,DNS or other protocols based on TCP or UDP	- reuses ip address, conserving the total number of IP addresses used - centralized management of IP Adresses	- tracking internet activity becomes tedious - computer with DHCP cannot be used as servers, as their IPs change over time.
FTP	The protocol enables file sharing between hosts locally and runs on top of TCP	- enables large file sharing	- lacks security - lacks encryption capabilities

		- able to resume file sharing if interrupted	
SMTP (simple mail transfer protocol)	A push protocol used to transfer electronic mail reliably and efficiently.	- Ease of installation - Connects to any system without restrictions	- Certain firewalls can block the ports used with SMTP

Table 4

As there are many different networking protocols used by edge devices in IoT systems, each one has certain specifications and benefits. However, at the moment it is almost impossible to determine which is the best protocol to be used. As we had seen above,research for security protocols are extensive and yet we do not have a definite best protocol to be used. With each protocol it has to be dissected and extensively tested to prevent any forms of vulnerability. The same can be said for the network communication protocols whereby there is not a single protocol without any flaws. Although some of the protocols mentioned above may look promising such as TCP and TLS, and may help serve the main function of the edge device which is to connect networks, reduce latency and promote security and data privacy, we have to remember that a particular implementation of a protocol in an edge device in a system may not work well in another system or another device and such adoption would be harmful to the overall security and reliability of the edge device.Although there isn't any research papers that are able to conclude that a certain protocol is better than the other in all scenarios, researchers are using current findings and testing data to further improve the current protocols . For example TSL was made to replace SSL eventhough they function the same way. The only difference is that TSL is more secure. Researchers are also using the current findings to create new protocols with better performance and security. For example, researchers, Y. Keshtkarjahromi, Y. Xing and H. Seferoglu, have developed a new protocol that is meant for edge computing called Coded Cooperative Computation Protocol (C3P) by taking into account the heterogeneous resources of edge devices. As of current C3P has an efficiency in terms of resource utilization of more than 99%, its task completion delay is very close to optimal coded cooperative computation solutions and it improves task completion delay significantly. With extensive research going into current networking protocols and into creating new protocols, it would be safe to say that in the future it would be easier to compare the protocols as they would have improved greatly hence it would be easier to choose a protocol that would allow the edge device to function optimally in an IoT system.

Bibliography

1) S. Al-Sarawi, M. Anbar, K. Alieyan and M. Alzubaidi, "Internet of Things (IoT) communication protocols: Review," 2017 8th International Conference on Information Technology (ICIT), 2017, pp. 685-690, doi: 10.1109/ICITECH.2017.8079928.

The authors compare the different wireless communication technologies and protocols used in IoT devices to determine which is the best. The technologies and protocols are compared based on different criteria which include frequency bands, network, topology, power consumption, range, security and others.

2) D. Caballero, F. Gonzalez and S. A. Islam, "Analysis of Network Protocols for Secure Communication," 2021 9th International Symposium on Digital Forensics and Security (ISDFS), 2021, pp. 1-6, doi: 10.1109/ISDFS52919.2021.9486356.

The authors conduct an analysis of the security properties of network protocols used in IoT devices, mainly network security protocols. The authors give a brief description of each protocol and then states the various attacks that each protocol is susceptible to and how they take place.

3) Y. Keshtkarjahromi, Y. Xing and H. Seferoglu, "Dynamic Heterogeneity-Aware Coded Cooperative Computation at the Edge," 2018 IEEE 26th International Conference on Network Protocols (ICNP), 2018, pp. 23-33, doi: 10.1109/ICNP.2018.00013.

The authors present their findings about a networking protocol used in edge computing that takes into account the heterogeneous resources of edge devices. According to the researches the protocol improves the reliability and efficiency of these edge devices by a significant amount.

4) “Network Protocols”, ”Manage Engine OpManager ”. Available at: https://www.manageengine.com/network-monitoring/network-protocols.html (Accessed : 01 November 2022)

The article talks about the different networking protocols used at different layers of the network architecture and their advantages and disadvantages.

5) Anna Triantafyllou, Panagiotis Sarigiannidis, Thomas D. Lagkas, "Network Protocols, Schemes, and Mechanisms for Internet of Things (IoT): Features, Open Challenges, and Trends", Wireless Communications and Mobile Computing, vol. 2018, Article ID 5349894, 24 pages, 2018. https://doi.org/10.1155/2018/5349894

The authors, researchers from the University of Western Macedonia and the University of Sheffield International Faculty, present their findings on the various network protocols and technologies used in IoT systems and the various challenges in using each protocol.

6) Sharma, Yogesh & Gokan Khan, Michel & Al-Dulaimy, Auday & Khoshkholghi, Mohammad Ali & Taheri, Javid. (2020). Networking models and protocols for/on edge computing. 10.1049/PBPC033E.

The authors write about the various networking models used in edge computing and the different protocols used in each layer of the edge computing network model.

7) Priya Pedamkar, “Introduction to Edge Computing Architecture”, “Educba”. Available at: https://www.educba.com/edge-computing-architecture/. (Accessed : 01 November 2022).

The article is about the edge computing architecture. It explains how edge computing works, the protocols used and the devices involved.

8) Kmbh,” Types of Network Protocols and Their Uses”, “GeeksforGeeks”,24th November 2021. Available at : https://www.geeksforgeeks.org/types-of-network-protocols-and-their-uses/. (Accessed : 01 November 2022).

The article shows the different network protocols available, how they are split into 3 different group, the uses for each group and it give a brief description of each protocol.

9) Cabe Atwell, “Fundamentals: What is an edge device?”, “Fierce Electronics”, 26th April 2021. Available at : https://www.fierceelectronics.com/electronics/fundamentals-what-edge-device. (Accessed : 01 November 2022).

The article explains what an edge device is, its uses and its challenges.

10) “What is IOT Edge Computing?”, “Red Hat”, 29th July 2022. Available at : https://www.redhat.com/en/topics/edge-computing/iot-edge-computing-need-to-work-together, (Accessed : 01 November 2022).

The article explains the difference between an IOT device and an edge device, and how Iot and edge computing are related.

11) Bill Bither, “ WHAT IS AN EDGE DEVICE AND WHY IS IT ESSENTIAL FOR IOT?”, “machinemetrics”, 07th January 2021, Available at : https://www.machinemetrics.com/blog/edge-devices. (Accessed : 02 November 2022).

The article talks about edge devices and it is integrated in IoT systems. It also talks about how edge devices are able to improve IoT systems.

12) “what is an edge device?”, “Chooch”, 2nd April 2021, Available at : https://chooch.ai/computer-vision/what-is-an-edge-device/. (Accessed : 02 November 2022)

The article talks about edge devices and how they work in IoT Systems. It also talks about the advantages of edge devices and how Artificial intelligence can work on these edge devices.

13) Nikolai Siersted, “10 Edge computing use case examples”, “STL Partners”, Available at: https://stlpartners.com/articles/edge-computing/10-edge-computing-. (Accessed : 02 November 2022).

The article gives 10 examples of edge computing and how it is used in IOT systems.