A comprehensive analysis of 6LoWPAN for IoT

[1]Anireekshith Narayana, [2]Tarun Sharma, [3]Ramesh Chandra Veeturi

***Abstract*: The explosion of innovative, low-priced sensors and devices popularly known as the Internet of Things(IoT) have great business potential. 6LoWPAN is an important IP (Internet Protocol) layer based technology used to connect devices in IoT. 6LowPAN is a IP based networking technology that carries IPV6 packets efficiently using small link layer frames as prescribed by IEEE 802.15.4. However there are confrontations such as packet size of IPV6 is much more immense than the most immense IEEE 802.15.4 frame size, challenges while compressing IPV6 header in Low Power Wireless Personal Area Networks (6LowPAN). Excess use of multicast causes 6LoWPAN inefficient and sometimes unrealistic in a "low power and lossy network".**

This paper is a survey on different approaches used for Header Compression such as LowPAN_HC1, LowPAN_HC2 and LowPAN_IPHC. These approaches are covered in "RFC 4919 (IPV6 Over Low Power Wireless Personal Area Networks (6LowPAN): Overview, Assumptions, Problem Statement and Goals)", "RFC 4944 (Transmission of IPV6 packets over IEEE 802.15.4 networks)", "RFC 6282 (Compression format for IPV6 datagrams over IEEE 802.15.4 based networks)", "RFC 6568 (Design and application spaces for IPV6 Over Low Power Wireless Personal Area Networks)"

***Index Terms*: Internet of Things (IoT), 6LowPAN, Header Compression (HC)**

I. INTRODUCTION

A device in the Internet of Things(IoT) possess unique identifier and is capable of sending/Receiveing sensor data via gateway over a network. Internet of Things comprises of distributed devices with sensors which are used to monitor the environmental and physical conditions. A typical Internet of thing scenario consists of several end-nodes, routers and a co-ordinator. End nodes are nothing but sensors, sometimes end nodes also acts as routers. Routers job is to route the data from end-nodes to co-ordinator. Co-ordinator sends the sensor data and acts as a gateway to the Internet. This paper, fixated on compression aspects of IPV6 header and methodologies used for Next headers compression. The major issues in Header compression are it needs optimization to make the payload light weight to transmit within a small link layer frames. RFC 4944, designates transmission of IPV6 packets over IEEE 802.15.4 networks has been discussed in this paper. Apart from this various RFCs that are related to Header Compression and Next Header compression are discussed in this paper. The rest of this paper is organized as following. Section II contains details about 6LowPAN and IPV6 with basics of IPV6 header structure, Neighbor discovery and their existing challenges. Section III explains related work which includes alternate options, details and techniques available for Header compression and Section IV concludes with survey outcomes by identifying the enhancements, future research, scope and directions..

II. INTRODUCTION TO 6LOWPAN AND IPV6

In a typical IoT protocol stack, network layer sends IP datagram from source network to the destination network. IP datagrams are forwarded by Routers and delivered them to their final destination. If we consider internet as an immensely colossal physical network, 32-bit IP address which is unique, is assigned to each host that is used in all communication with that host. A typical IPV4 environment uses 32-bit address that allows total of 232 addresses or 4.3 billion addresses. In a IoT world, every second a new IoT asset is getting added to the network, and this makes difficult to assign IPV4 address to each and every device. In addition, IPV4 requires configuring the IP address manually or stateful configuration through Dynamic Host Configuration Protocol[DHCP]. IPV4 has a complex header structure, which integrates all the options in the base header. IPV6 overcomes from these limitations: Extended Address Space ; The address format is elongated from 32 bits to 128 bits. Every node on the universe gets IP address with IPV6[1]. Stateless Autoconfiguration; When a IoT device comes up, it might ask for a network prefix from the IPV6 router on link with the avail of this network prefix, a device can autoconfigure its own IP address utilizing either MAC address or private desultory number. Simpler Header Structure; In comparison with header structure of IPV4, IPV6 has a simpler header format with a fixed size of 40 bytes. Source address has been assigned 16 bytes, Destination address has been assigned 16 bytes and only 8 bytes are assigned for general header information, which makes IPV6 header less weight.

Improved support for options and extensions : IPV4 Header is bulky which includes all the options in the base header where as in IPV6 options are optional and added only if they are needed to so called Extension Header. Fig 1 depicts the Header Structure of IPV6.

There are certain challenges in implementing IPV6 over tiny IoT devices. Large number of packets are transmitted over wire as part of the Neighbor Discovery. IPV6 Header contains fields which are not necessarily required for lossy and low-power network. Hence Header compression is required in 6LoWPAN to accommodate lightweight payload.

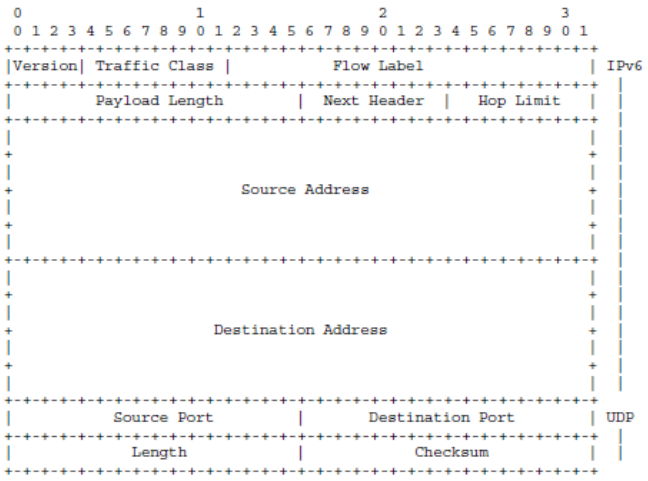

Fig.1 IPV6 Header structure

The 6LoWPAN Architecture :

LoWPAN's are broadly classified in to 3 categories : Simple LoWPAN, Extended LoWPAN and Adhoc LoWPAN[2] as shown in Fig 2. A Simple LoWPAN network contains LoWPAN nodes which are connected to the internet using a LoWPAN Edge Router. An Extended LoWPAN network contains LoWPAN nodes which are connected to the internet using multiple Edge Routers. These Edge routers are linked using a backbone link. Adhoc LoWPAN as the name specifies operates without any infrastructure. Edge Routers play an important role in handling compression and Neighbor Discovery for the LoWPAN[2]. A node in a LoWPAN network can be Host or Router. Edge router and router sends the IPV6 prefix to all the nodes in LoWPAN network[2]. LoWPAN network might contain the same LoWPAN nodes the same time. LoWPAN nodes can move across the LoWPAN network, between Edge Routers and even between LoWPANs[2].

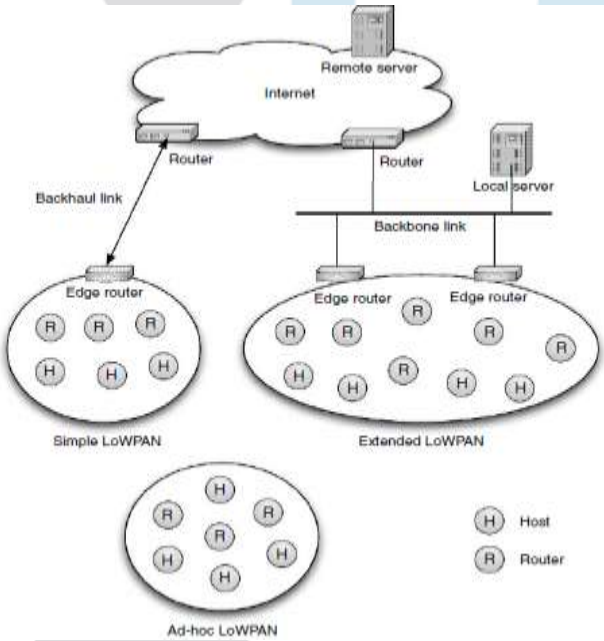

Fig. 2 The 6LoWPAN Architecture[2]

The Protocol Stack :

Fig 3 shows the comparison of TCP/IP network layer and 6LoWPAN layer with IPV6. In 6LoWPAN protocol stack, an adaptation layer has been introduced in network layer as per RFC 4944. TCP is not used in Transport layer as it is not suitable in low power and lossy network. Edge Router is responsible for adaptation between these two protocol formats[2]. As part of the application protocols, its recommended to use IoT specific application protocols such as AMQP, MQTT , XMPP instead of HTTP which is more suitable for web based application on a traditional IP network[3].

Header Compression according to RFC :

MTU of 1280 bytes is required for IPV6(Max Transmission Unit), which is fulfilled by 6LoWPAN Adaption layer. RFC 4944 defines mechanisms for the Header compression in IEEE 802.15.4. IEEE 802.15.4 allows IEEE 64-bit extended address or 16-bit address with 6LoWPAN. IEEE 802.15.4 has max frame size of 102 octets, which is derived by removing 25 octet (frame overhead) from maximum packet size of 127 octets.

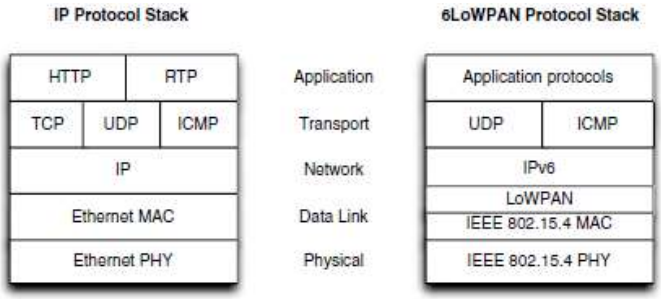

Fig. 3 IP and 6LowPAN Protocol stack

Security aspects of Link Layer will take up 21 octets, leaving packet size of 81 octets [4]. Next layer protocols are left with only 41 octets as 40 octets are required for IPV6 Header. Header of the Upper layer protocol (eg., UDP) uses 8 octets, hence left with only 33 octets for actual application data.

A 802.15.4 device can have 2 types of address a) IEEE EUI-64 address b) 16-bit short address. 16 zero bits are appended to the 16-bit PAN id to get first 32-bits of interface identifier. Thus formed 32-bit address is concatenated with short address (16-bit) to give rise to resultant address of 48-bit. Next interface identifier is formed as per RFC 2464[4].

The IID for an interface is based on the EUI-64. EUI-64 is formed by adding FFFE hexadecimal between the fourth and fifth octet of the EUI. The interface identifier is then formed by complementing the lowest order bits of the first octet of the EUI-64.

RFC 4944 explains simple method of Header Compression in LoWPAN_HC1. LoWPAN_HC1 assumes that IPV6 Header fields and 6LoWPAN fields are common. Version is IPV6, Link-local addresses are used in Source and destination and can be obtained from lower layers. Lower layer provides the payload length or derived from the fragment Header's "datagram-size" field. In LoWPAN_HC1 the TC and the FL are set to zero(Traffic Class & Flow Label). The Next header is either TCP, UDP or ICMP. Hop-limit is carried in-line. As a result of LoWPAN_HC1, IPV6 packet can be compressed to 2 octets. In LoWPAN_HC2 compression of source port, destination port and length is allowed. Checksum field is carried in-line as per LoWPAN_HC2.

LoWPAN_HC1 and LoWPAN_HC2 have certain drawbacks. i.e., these mechanisms are best suitable for unicast communication with Link-local address. LoWPAN_HC1 is not suitable for practical implementation because while routing with LoWPAN_HC1, addresses needs to be carried in-line. RFC 6282 define LoWPAN_IPHC which is a context based for effective compression of IPV6 addresses, which includes unique Local, Global and multicast address[6]. RFC 6282 also define a mechanism called LoWPAN_NHC for the compression of arbitrary next Header[6].

Header Compression with LowPAN_IPHC :

LoWPAN_IPHC considers the below encoding format for 6LoWPAN communication: version field is set to IPV6, In LoWPAN_IPHC the TC and the FL are set to zero(Traffic Class & Flow Label), Lower layer provides the payload length, Source set the Hop limit value, 6LoWPAN interfaces addresses are formed with IID directly derived from either the 16-bit short IEEE 802.15.4 addresses or the 64-bit extended address[6]. LoWPAN_IPHC when used efficiently can result in Header length of 2 octets, with Local-local communication. While routing this can results in Header compression of 7 octets.

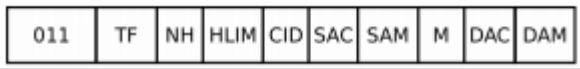

Fig. 4 LoWPAN_IPHC format

Bits 011 in the encoding format indicate LoWPAN_IPHC is used for compression. 2-bit Traffic class and flow label indicate compression further 8-bit TF is split into 2-bit ECN and 6-bit DSCP. If NH is set, then LoWPAN_NHC is used for compressing Next Header, if NH not set, then Next Header is carried in-line. 2 bit HLIM(Hop Limit) is set to different values by the source i.e., 01 indicates Hop Limit as 1, 10 indicates Hop Limit as 64, 11 indicates Hop Limit as 255, 00 indicates Hop Limit carried in-line. If 1 bit CID is set, it indicates 8-bit context Identifier Extension follows the DAM field. If CID is set to 0, it indicates context Identifier Extension is not used. If 1 bit SAC is set to 1, it indicates stateful context based compression, if SAC is set to 0, it indicates stateless compression of source address. 2-bit SAM field indicates how many bits of source address are carried in-line when SAC is enabled or disabled. If 1-bit M is set to 1, it indicates multicast address is used as the dest(destination) address, if M is to 0, it indicates dest(destination) address is not a multicast address. If 1-bit DAC is set to 1, it indicates stateful context based compression, if DAC is set to 0, it indicates stateless compression of destination address. 2-bit DAM field indicates how many bits of destination address are carried in-line when DAC is enabled or disabled.

III. RELATED WORK

This section briefs about the related work done in 6LoWPAN Header compression. Sreejesh V.K. et al; [8] implemented the header compression scheme proposed in [7] in which throughput is increased by 8% and transmission energy is reduced by 5%. He used Contiki to implement the idea. This paper details about the LoWPAN-IPHC unused bit combinations and multicast addresses compression and default UDP port numbers.

Sameer.K.Alsudany et al;[9] proposed a mechanism which improvises the proposed methodology [10] in which the most frequently used IP address prefixes are mapped to the context identifiers. As per [10], context system contains 2 tables, a Context Table (CT) and Context Table Buffer(CTB). Context Table used to map frequently used IP address prefix and CTB stores other prefixes. As per [9], a system model which is improvised version of [10], contains both CT and CTB. Proposed mode will work on the packets coming inside the 6LOwPAN network and packets leaving the 6LoWPAN network. Algorithm compares the source address prefix and destination address prefix with Context Table and take actions for the incoming packets similarly based on the SAC and DAC fields, decompressor takes actions for the outgoing packets.

Laizhong Cui et al; [10] proposed a design which contains Dynamic 6LoWPAN Context Table Maintaining Algorithm. As per the algorithm first row of the Context Table(CT) contains the default IPV6 address prefix and remaining rows are occupied by the frequent IPV6 address prefixes. In [10], two scenarios are described which includes incoming and outgoing traffic at the 6LBR/Edge Router. When a packet arrives at the 6LBR, proposed algorithm extracts the corresponding source address and destination address, suffix portion of the IPV6 address is directly fed to 6LoWPAN Packet Header Generator. Later comparison of Prefix with CT entries happens. If there is a match, CID is given to Packet Header Generator. If there is no matching entry, then prefix is directly given to Packet Header Generator and prefix is entered in CTB and corresponding weight is increased. For outgoing traffic, Header Decompressor checks the packet, if SAC field and DAC field of the LoWPAN_IPH are zero, then packet uses stateless compression. In this case, IPV6 address can be directly given to packet generator and CTB. On the Other hand, if either Source Address or Destination address's SAC field and DAC field contains One, then stateful and Context-based Compression has been used. Now CID can be retrieved from Context Identifier Exension field of LoWPAN_IPHC header, then IPV6 prefix is retrieved from Context Table(CT). Finally IPV6 address is generated by combining prefix and suffix.Finally, complete content and organizational editing before formatting.

IV. CONCLUSION

In this paper we mainly focused on Header compression approaches such as LoWPAN_HC1, LoWPAN_HC2 and LoWPAN_IPHC in 6LoWPAN network. Header compression mechanisms discussed in various RFCs are analyzed to send the packet over network which is having limited processing power and memory, short range communication. Still lot of research has to be conducted in the way of sharing contexts, maintaining the context table. Addressing the issues of security in 6LoWPAN is still open and an important area.

REFERENCES

[1] IPv6 Essentials, 2nd Edition, Silvia Hagen.

[2] 6LoWPAN : The wireless Embedded Internet, Wiley Series in Communications Networking and Distributed Systems, Zach Shelby and Carsten Bormann

[3] Rahul Shrimali et al; "Header Compression and Neighbor Discovery in 6LoWPAN based IoT - a survey" IEEE WiSPNET 2016 conference

[4] IETF Transmission of IPv6 packet over IEEE 802.15.4 networks" RFC 4944, September 2007

[5] IETF Transmission of IPv6 packet over Ethernet networks" RFC 2464, December 1998

[6] IETF Compression Format for IPV6 Datagrams over IEEE 802.15.4 based networks, RFC 6282, September 2011

[7] W. Huiqin, and D. Yongqiang, "An Improved Header Compression Scheme for 6LoWPAN Networks", Ninth International Conference on Grid and Cloud Computing, Nov. 2010.

[8] Sreejesh V.K and G. Santhosh Kumar "Implementation and Evaluation of an Improved Header Compression for 6LoWPAN"

[9] Sameer K. Alsudany et al; A Dynamic Context-Based Header Compression Algorithm for 6LoWPAN Networks.