ZigBee IP Network Performance, Part I

Unicast Messaging under SEP2.0 HAN Messaging Profile

Document Part Number 53-301017 Rev 0.1

April 2013

ZigBee IP Network Performance, Part I
Unicast Messaging under
SEP2.0 HAN Messaging Profile

ZigBee IP Network Performance, Part I
Unicast Messaging under
SEP2.0 HAN Messaging Profile

Exegin Technologies Limited
Printed in Canada

Revision History

Rev. Number	Date	Notes
0.1	13/04	Preliminary

ZigBee IP Network Performance, Part I
Unicast Messaging under
SEP2.0 HAN Messaging Profile

Table of Contents

List of Figures

ZigBee IP Network Performance, Part I
Unicast Messaging under
SEP2.0 HAN Messaging Profile

1 Introduction

This paper presents the results of a set of tests, ongoing, that conducted in conjunction with Cisco System and Grid2Home in order to try to understand the performance characteristics of a ZigBee IP network running on a set of 802.15.4 (2006) radios, in the context of a particular network size and topology, and loading.

2 Background

Utilities and other agencies have become increasing interested in providing communications systems to the electric meter and to the devices in the home. A candidate wireless networking standard to achieve this is the IEEE 802.15.4 used in conjunction with version 6 of the internet protocol (IP). The synthesis of the two along with a number of other protocol components is being manifest as ZigBee IP, an IP base mesh networking protocol that provides TCP and UDP socket based facility for standard internet applications such as HTTP (web), telnet , ICMP, etc. that runs on 802.15.4 complaint radios.

The utility community is interested, in particular, in the use of HTTP and HTTPs in order that their applications can communicate using RESTful web services in accordance to CIM IEC WG 13 and 14, IEC 61970-301 FDIS, Third Edition

ZigBee IP introduces a number of new technologies, to the area of 802.15.4 based radios, in order to provide Internet protocol (IP) and in particular IPv6 to the domain of small wireless devices. These include 6LoWPAN (HC and ND) which provide, respectively IP header compression and neighbour discovery and ROLL-RPL which provides the mesh routing facility for the network. The following figure depicts the current state of the ZigBee IP network stack in the context of a Smart Grid application:

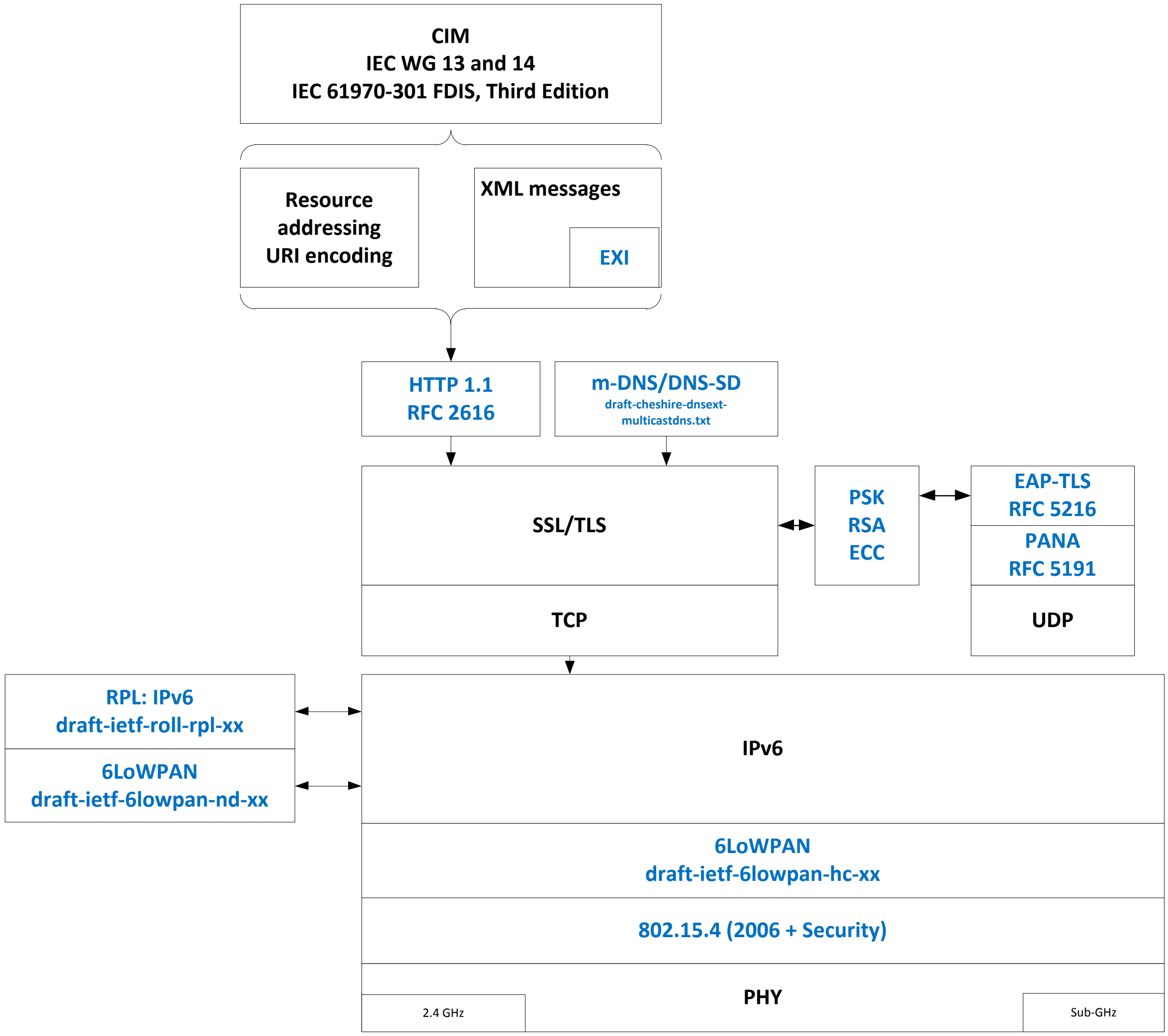

Figure 1: ZigBee IP Network Stack

where additional network support components which comprise the key components of ZigBee IP that support mesh routing (ROLL-RPL), 802.15.4 adaptation (6LoWPAN) network authentication (PANA, EAP, TLS) along with application support with HTTP, SSL/TLS, m-DNS/DNS-SD noted in blue, have been included in the picture.

From the outset many questions surrounded the efficacy of the implementation including but not limited to network through put and in particular network throughput in light of perceived network loading. This paper details the authors' attempts to answer that question.

3 Messaging profile

In measuring network performance it is critical to understand the context of the measurements and the required ambient conditions. Without having prior knowledge of what the network loading would be for the group used a much discussed messaging pattern which comprised:

The following message profile was executed:

- Price. All 30 clients query server for Price on 3 minute interval, synchronized to top of hour, flat randomization of +- 1 minute. Base server return payload set at 900.
- DRLC (Demand Response Load Control). All 30 clients query server for DR event list on 3 minute interval, synchronized to top of hour, flat randomization of +-1 minute. Base server return payload set at 2400 bytes.
- Instantaneous Demand. 3 nodes, one per tier, query for instantaneous demand on 2 second interval. No randomization applied. Base server return payload fixed at 300 bytes.

Within the messaging profile two particular cases were considered; one where the payload was transmitted as plain xml text or POX (Plain Old XML) and the other being the case where the XML data would be compressed using the EXI (Efficient XML Interchange) data format as specified by the W3C.

By using grammar based EXI compression on the XML payloads it was found that at least 80% reduction of payload size could be realized and the tests conducted in this paper used that figure to simply reduce the payload sizes of each of the messages without encoding the payload data using EXI. Under these conditions the respective payloads under simulated compressed conditions became:

- Price. Base server return payload set at 900 bytes (POX) or 180 bytes (EXI = 20% of POX).
- DRLC Base server return payload set at 2400 bytes (POX) or 480 bytes (EXI).
- Instantaneous Demand Base server return payload fixed at 300 bytes (POX) or 60 bytes (EXI).

Pictorially, with some simplifications being made for presentation purposes, the messaging profile, for POX data payloads, can be displayed as

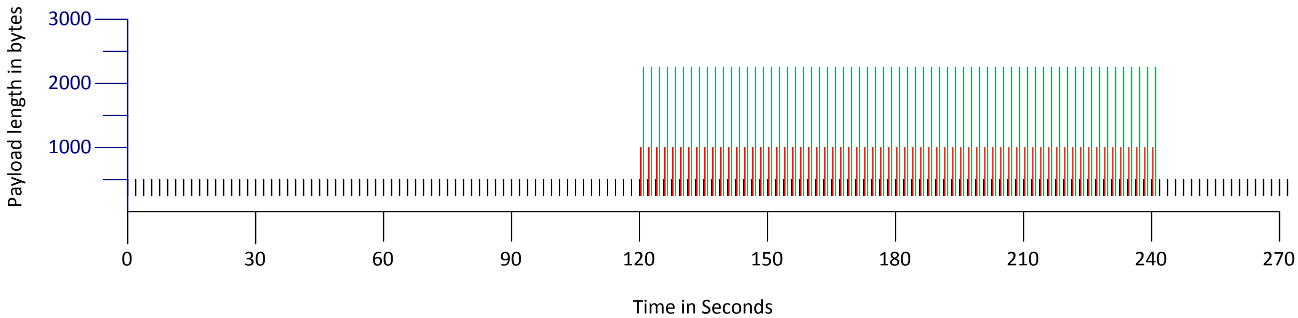

Figure 2: Messaging Profile POX

Where:

Instantaneous Demand 3 nodes, one on each tier
DRLC all 30 nodes
Price all 30 nodes

4 Test setup

The test harness comprised 31 ZigBee IP nodes. One node was configured as the DAG (Dynamic Acyclical Graph) root; this node also server as the HTTP server for the message requests. The nodes were connected in a tiered hierarchy using coaxial cable. Resistive splitters and attenuators were introduced to ensure that sufficient separation between the tiers of nodes to guarantee some of the test messages would travel over 3 hops to reach their destination, dictated by the messaging profile.

The resistive splitters introduced 20 dB attenuation and further 30 dB attenuators were added between tiers to provide a separation of 70 dB between tiers in order to ensure that the RF signal across two tiers was sufficiently weak that the routing algorithms would dynamically create the required number of hops in the network.

The following figure details the test configuration:

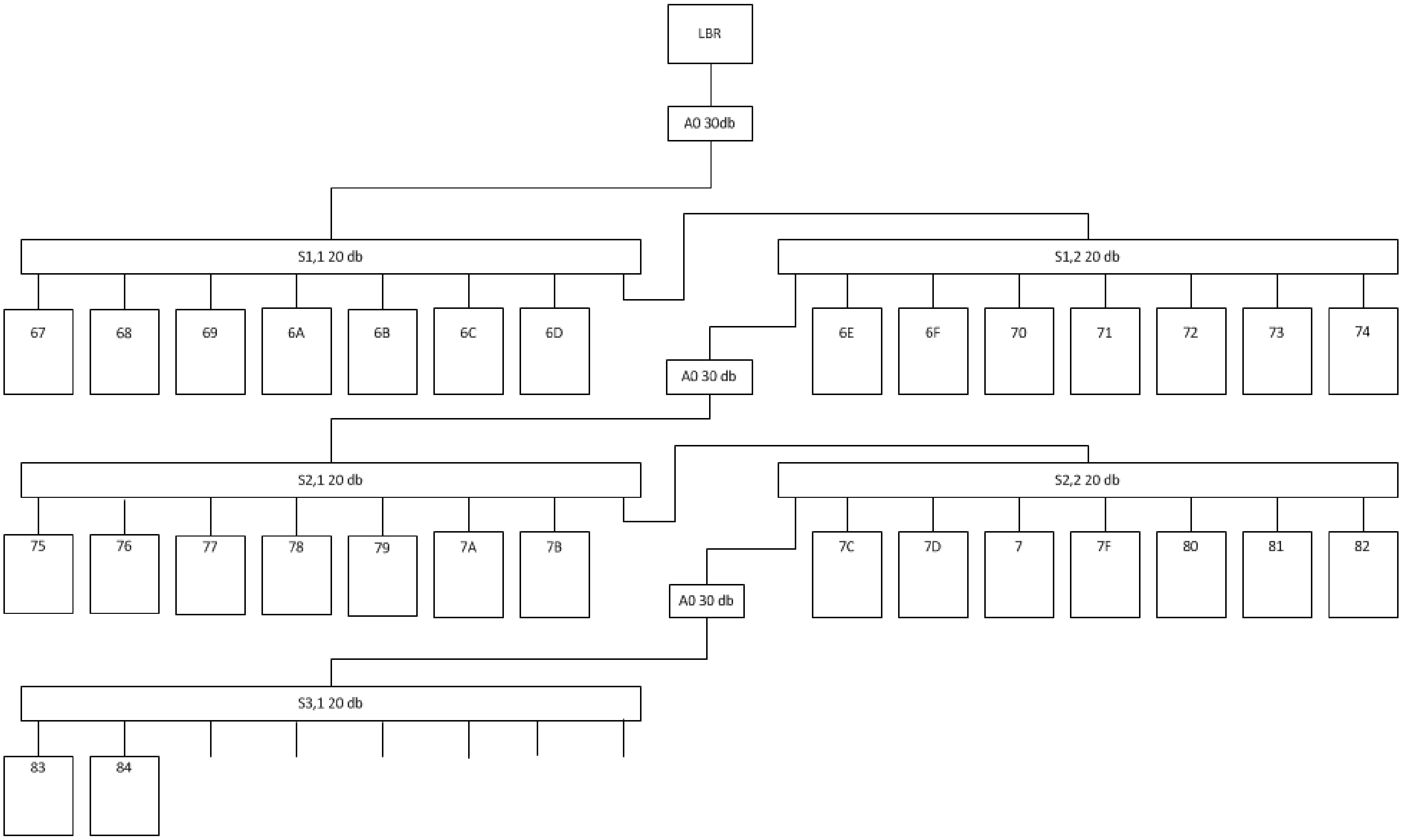

Figure 3: Test Configuration

The node designated LBR (Local Border Router) functioned as the DAG root and also as the systems HTTP server, in effect mimicking an electric meter. The components labeled Sn,m are the 8-way resistive splitters and those components labeled An are the inline attenuators. The remaining components bring the hexadecimal number labels 67 through 84 are the ZigBee IP client nodes: the numbers reflecting the last two digits of the devices MAC addresses.

Unused taps on splitters were terminated with 50 Ohm resistors.

On network start-up all of the nodes were allowed to come out of reset at approximately the same time. Over time the nodes dynamically settled into a stable network topology using the RPL routing algorithms (objective function 0). The final topology was left entire to the algorithms without any external coercion order than the use of RF attenuators between the nodes. A typical network topology is depicted below:

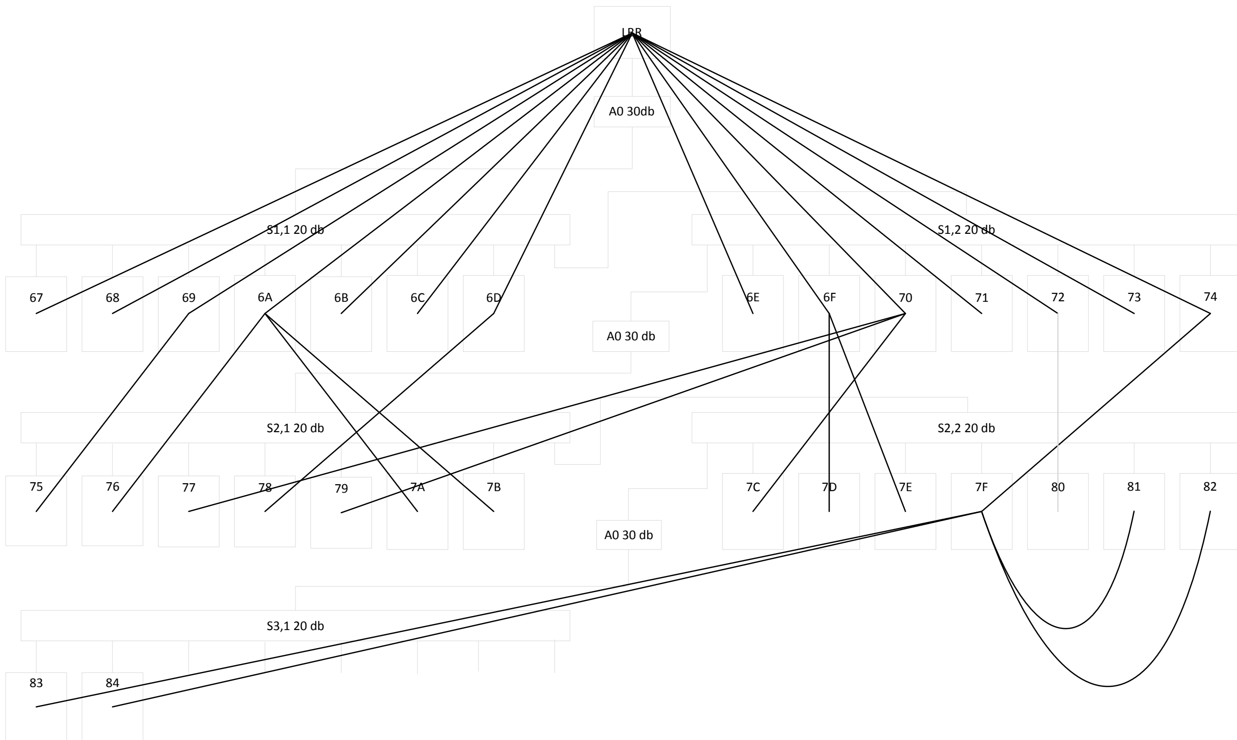

Figure 4: Route Structure POX test

After a period of a few minutes the network topology was, for the most part, stable, with only the occasional instance where a node changed parent within the same tier, but never to a parent from a next higher tier.

The results discussed in this section are based on the routing structure shown in Figure 4.

5 Tests

Two independent sets of tests were conducted, one without compression of the xml payloads for the Instantaneous Demand, DRLC and Price messages and one with the payload size reduced to 20% of the original payload, simulating compression. The payloads themselves did not specifically contain the message structures for the message profile messages but stings of random ACSII data of the appropriate length. The individual transactions comprised of the client node (end node) making an HTTP GET request of the server (the LBR in this case) and the server responding with the appropriate data. A typical transaction is shown below:

```
GET /echo/300 HTTP/1.1
Keep-Alive: 60
Content-Length: 0
```

```
HTTP/1.1 200 OK
Server: Exegin emHTTPD/1.0
Content-Length: 300
Content-Type: text/plain
0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789:;<=>?@ABCDEF
GHIJKLMNOPQRSTUVWXYZ0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXY
Z0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789:;<=>?@ABCDE
FGHIJKLMNOPQRSTUVWXYZ0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWX
YZ0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXY
```

with the request in blue type and the response in green type.

All of the tests were conducted without transport layer security, i.e. using HTTP and not HTTPs.

Once it was determined, by examining the routing tables of some of the ZIP nodes, that the routing structure was stable the client nodes' applications were started through an out of band control. Depending on the type of node the node either emitted HTTP requests of the server that included Instantaneous Demand messages along with DRLC and PRICE messages or just DLRC and PRICE messages. The timing of the nodes was such that the nodes would make their DRLC and PRICE request in a random distribution around the 3 minute mark with a variation of +/- one minute (Figure 2: Messaging Profile depicts is an abstraction of this randomness). The nodes and their application were left to execute this behaviour in a continuous cycle for a period on the order of 20 minutes.

The elapsed time from the emission of the GET request from the end node to the reception of the HTTP response by the end node was recorded at each end node for each transaction conducted by the end node. The transaction time data was collected for each end node for each test and analysed.

The initial analysis of the test comprised of plotting the probability that an HTTP transaction initiated from any end node for any length of data response at any time during the test would complete in time less than or equal to some time t. Subsequent analysis examined the same probability but differentiated by response payload length i.e. message type.

5.1 Results for POX Test

For this test 15,777 HTTP transactions with a distribution commensurate with the messaging profile discussed in section 3, were initiated.

At the start of the test the nodes were brought out of reset and with the exception of the RF attenuation introduced on the coaxial cables, were allowed to develop their own routing topology. The final topology is as shown in Figure 4.

5.1.1 Aggregated Results

The following table summarizes the transaction completion probabilities:

200 msec.	56%
1 second	86%
2 second	95%
3 second	97%

Table 1: POX Completion Probability Summary

The table indicates that in 200 milliseconds or less, 56% of all of the HTTP transactions, 8835 transactions, conducted in the test completed. That in less than 1 second 86% completed, in less than 2 seconds 95% completed and in less than 3 seconds 97% completed.

The full test results are shown below in Figure 5 and Figure 6, below

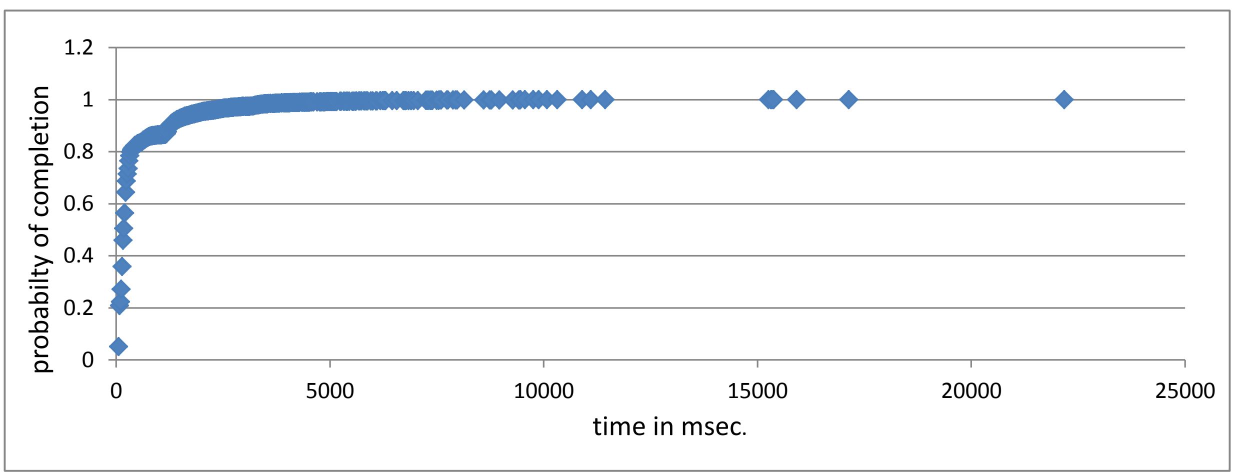

Figure 5: POX Test Results

where each data point represents the completion of at least one transaction.

All transactions completed i.e. no HTTP GET request were made that did not have a commensurate response.

The following figure provides an expanded view of the first 3 seconds of the results shown in Figure 5.

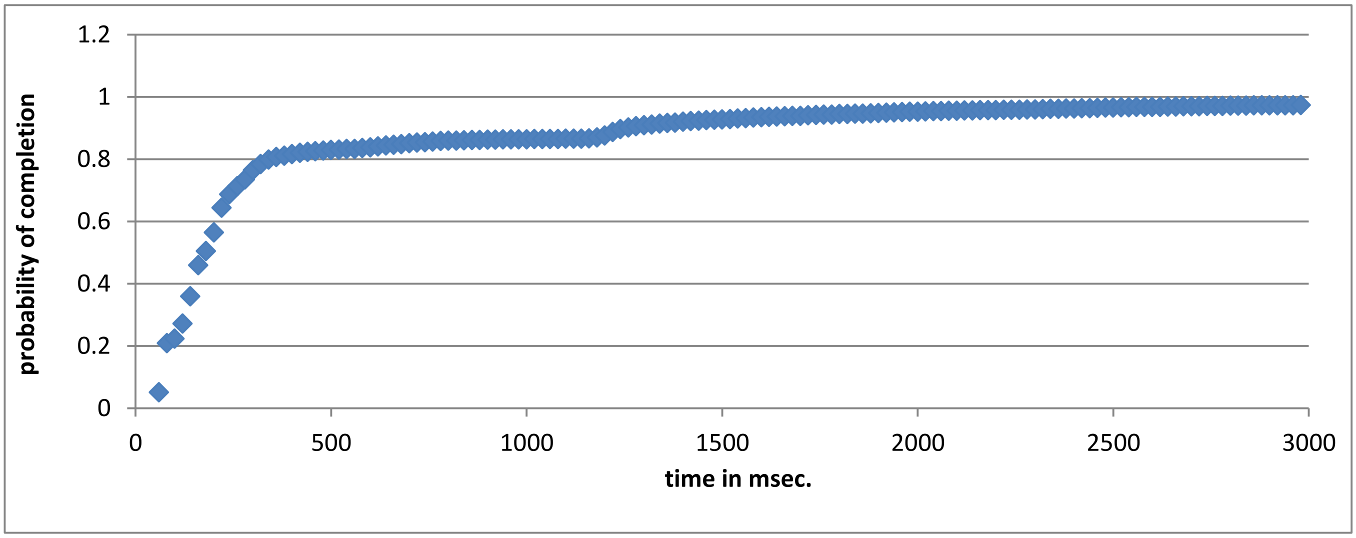

Figure 6: First three seconds of POX Test Results

The distinctive knees in the curves are thought to be due to the TCP retry mechanism.

5.2 Results based on Packet size

In gathering the transaction timing from each of the nodes, the payload size of each of the transactions was recorded. Using that information the resulting completion probabilities can be broken out by payload size.

The following table summarizes the transaction completion probabilities:

	300 bytes	900 bytes	2400 bytes
200 msec.	66%	40%	4%
1 second	90%	82%	64%
2 seconds	98%	93%	75%
3 seconds	99%	96%	89%

Table 2: POX Completion Probabilities by Payload Size

The full test results, as broken out by payload size, are shown below in Figure 7 and Figure 8, below:

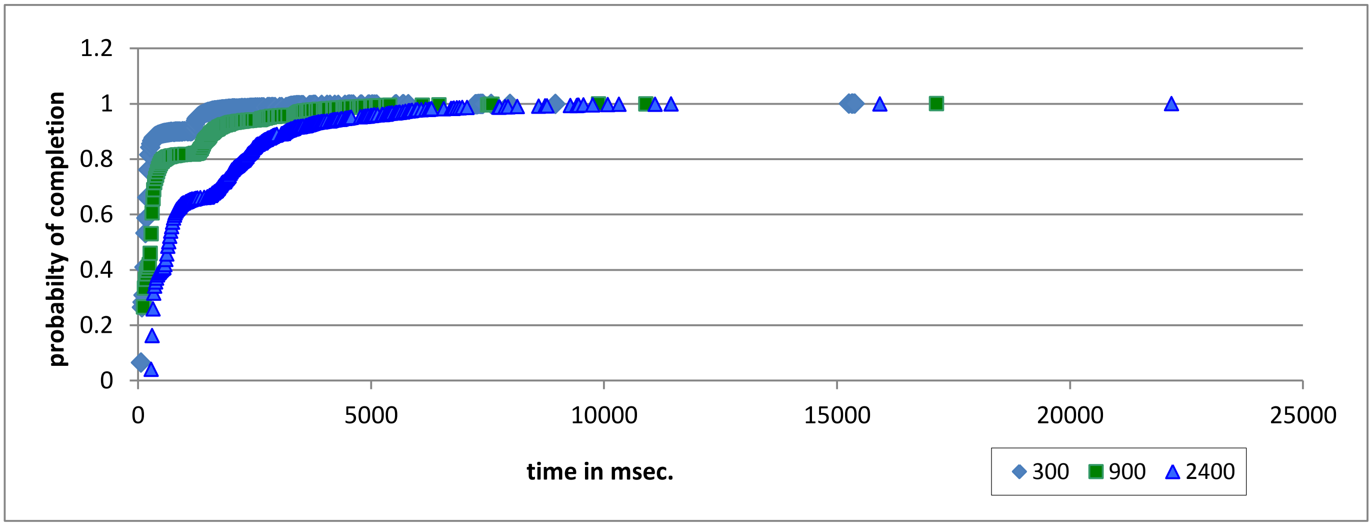

Figure 7: POX Results by Payload Size

The following figure provides an expanded view of the first 3 seconds of the results shown in Figure 7.

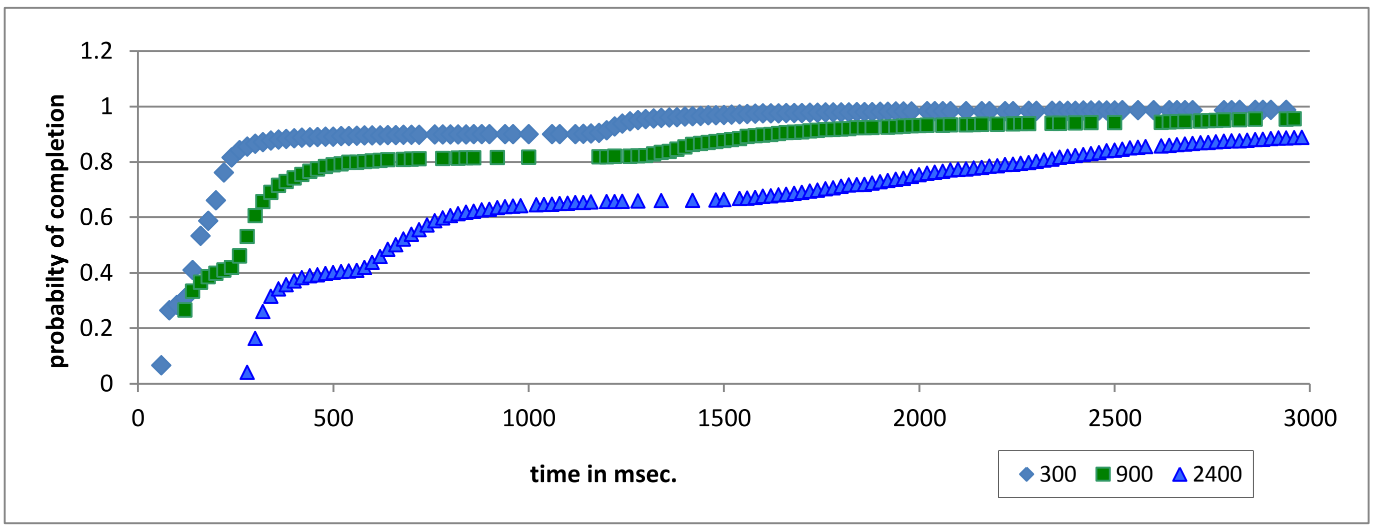

Figure 8: First three seconds POX Results by Payload Size

Clearly the expected transaction completion time is related to the size of the payload being transmitted, with poorer performance being exhibited by larger payloads.

5.3 Results of Simulated EXI Compression Test

For this test 17,572 HTTP transactions with a distribution commensurate with the messaging profile discussed in section, were initiated. The payloads of the transactions were reduced to 20% of their original sizes; the 300 byte payload became 60 bytes, 900 became 180 bytes and 2400 became 480 bytes. It was thought that this reduction was commensurate with the expected reduction to be realized by using grammar based EXI compression on the XML payloads.

The at the start of the test the nodes were brought out of reset and with the exception of the RF attenuation introduced on the coaxial cables, were allowed to develop their own routing topology. The final topology is as shown below:

.

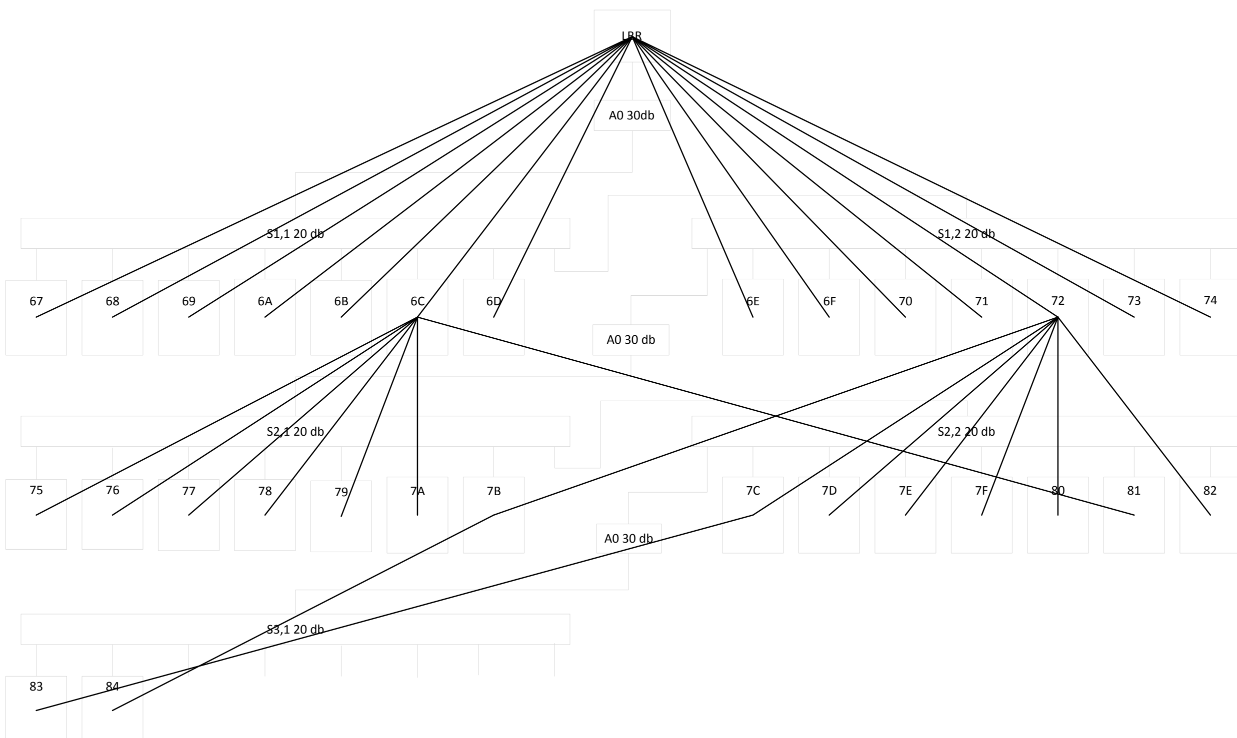

Figure 9: Routing Topology for Test using Simulated Compressed Data

5.4 Aggregated Results

The following table summarizes the transaction completion probabilities:

200 msec	91%
1 second	95%
2 second	99%
3 second	99%

Table 3: Compressed Data Completion Probability Summary

The table indicates that in 200 milliseconds or less, 91% of all of the HTTP transactions conducted in the test completed. That in less than 1 second 95% completed, in less than 2 seconds 99% completed and in less than 3 seconds 99% completed.

The full test results are shown below in Figure 10 and Figure 11, below:

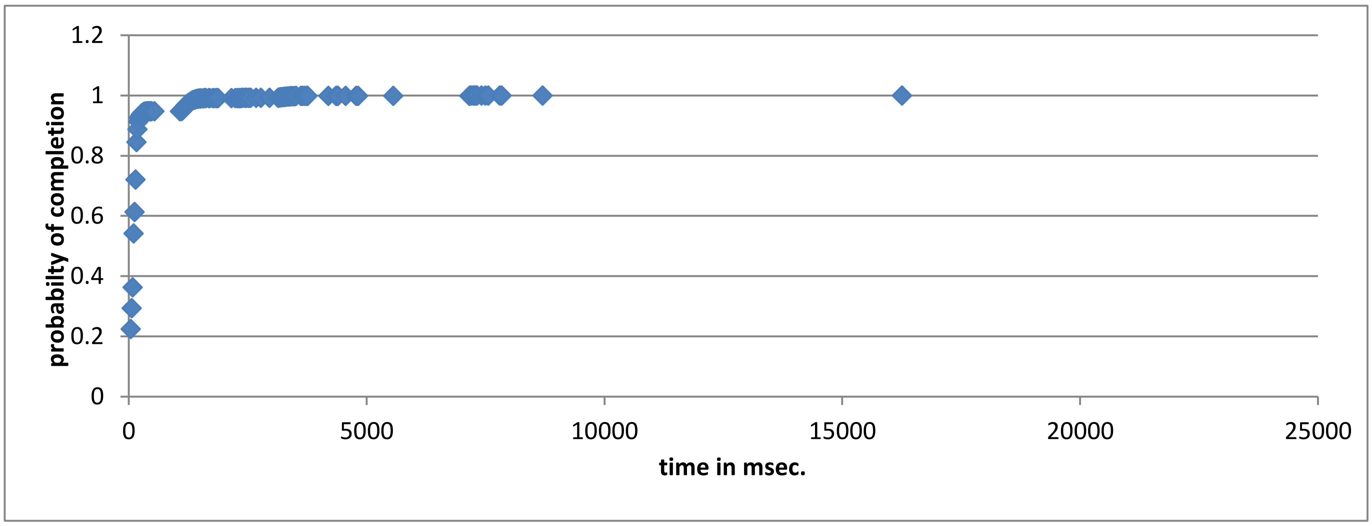

Figure 10: Compressed Data Test Results

where each data point represents the completion of at least one transaction.

All transactions completed i.e. no HTTP GET request were made that did not have a commensurate response.

The following figure provides an expanded view of the first 3 seconds of the results shown in Figure 10.

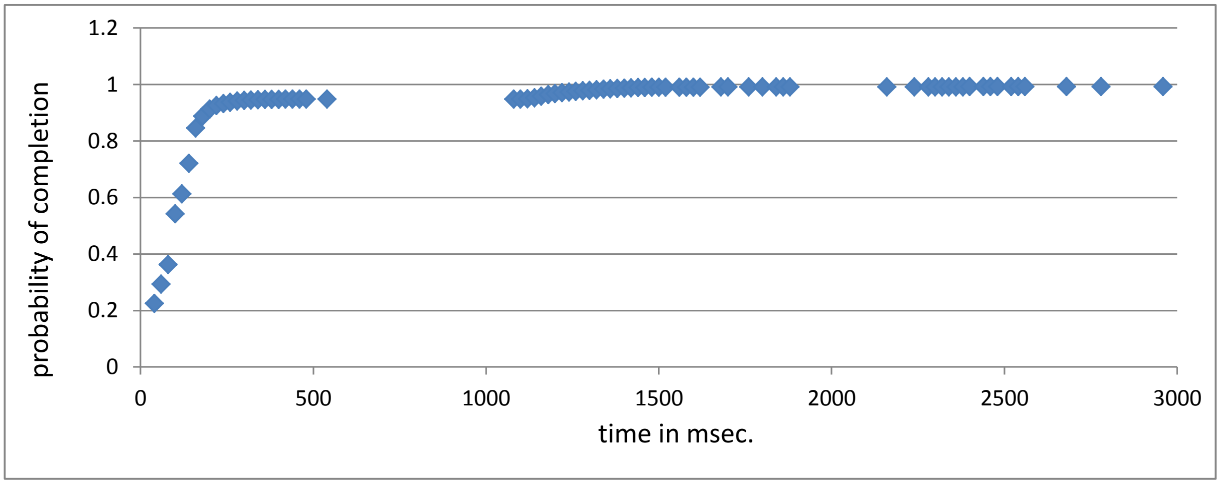

Figure 11: First three seconds of Compressed Data Test Results

5.5 Results based on Packet size

In gathering the transaction timing from each of the nodes, the payload size of each of the transactions was recorded. Using that information the resulting completion probabilities can be broken out by payload size.

The following table summarizes the transaction completion probabilities:

	300 bytes	900 bytes	2400 bytes
200 msec.	91%	94%	69%
1 second	95%	96%	90%
2 seconds	99%	99%	98%
3 seconds	99%	99%	99%

Table 4: Compressed Data Completion Probabilities by Payload Size

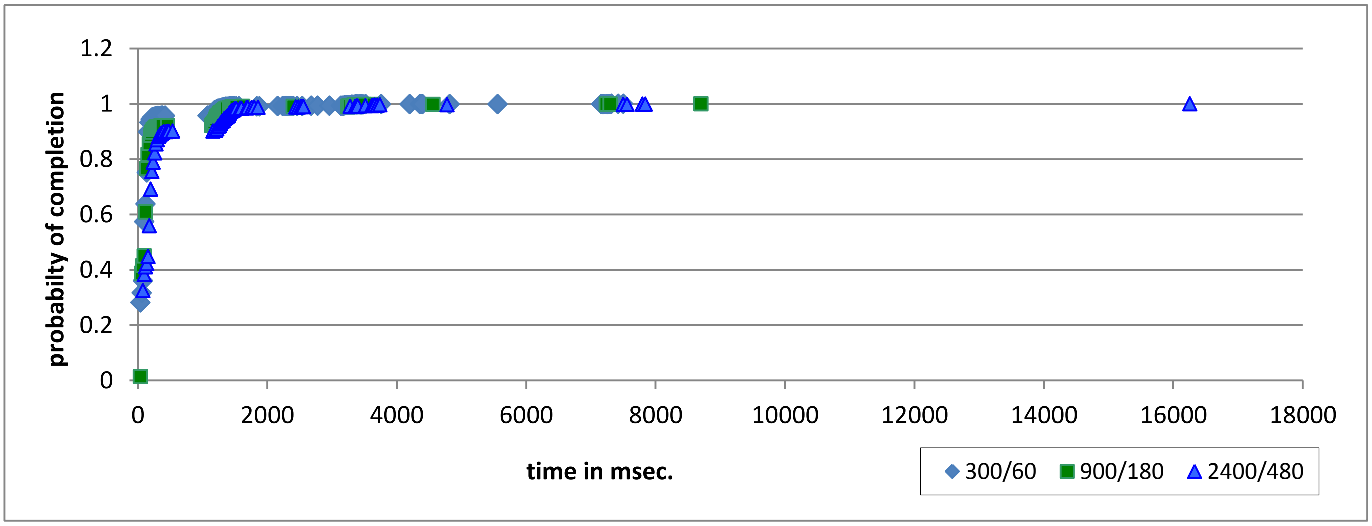

Figure 12: Simulated Compressed Results by Payload Size

The following figure provides an expanded view of the first 3 seconds of the results shown in

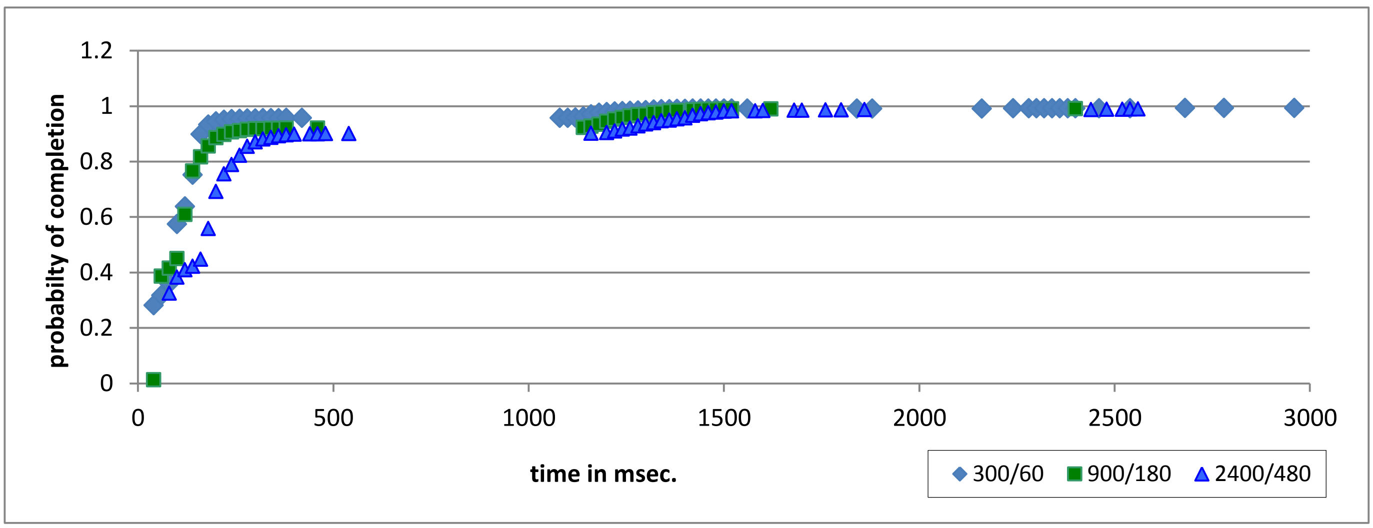

Figure 13: First 3 seconds of Simulated Compressed Results by Payload Size

6 Conclusions

With load exceeding HAN Messaging Profile:

- The metric of 96% success in < 2 secs was met.
- Compression helpful for the larger packets.

Results support SE2/ZIP viability for the typical consumer HAN environment (HTTP based).