doi:10.2498/iti.2013.0588

The Impact of Packet Fragmentation on Internet-of-Things Enabled Systems

James Pope and Robert Simon
Department of Computer Science and C4I Center
George Mason University
Fairfax, VA, 22030, USA
{jpope8, simon}@gmu.edu

Abstract. *The Internet of Things (IoT) refers to a collection of technologies designed to interconnect physical devices with the Internet. Due to device resource constraints IoT connectivity requires the redesign of several basic Internet protocols. This paper studies the impact of packet fragmentation at the data link level on the end-to-end performance of some of these redesigned protocols. Our results show that fragmentation can seriously degrade the performance of a typical IoT device to gateway communication modality. On the other hand, our results also show that with proper design of data broadcast mechanisms, gateway-to-device communication can be maintained at high performance levels. These results can be used as a guide for IoT network engineers.*

Keywords. Internet of Things Communication, Internet Protocols over Embedded Systems, Low-power and Lossy Networks.

1. Introduction

Originally coined to describe systems that connect RFID tags with the Internet, the *Internet of Things* (IoT) now refers to interconnections among a wider range of objects, including sensors, actuators, machines, etc [2]. Due to the need to minimize size, cost and power consumption IoT communication systems require a low power and limited capability protocol stack. At the same time, IoT systems must support two-way and end-to-end IP enabled communication. Running an IP protocol stack in such a resource-constrained environment has received a tremendous amount of attention in recent years [5,6,7]. Work has focused on all layers of the system, including physical standards for wireless transmission, the data link level, the network level and the application level. Despite this attention there remain a number of technical challenges that must be addressed before true end-to-end IP connectivity is achieved for IoT devices.

This paper focuses on one the above challenges, namely the impact of packet fragmentation on IoT systems. Packet fragmentation occurs when a device tries to transmit a packet that is larger then the maximum transmission unit size, or MTU, allowed by the link layer. Under this circumstance the device can either drop the packet or fragment the packet into several smaller packets so that each will be no larger than the MTU. If the packet is fragmented and not dropped then the original packet can be reassembled when all of the packets are received.

Although fragmentation is an effective method to get around the restriction imposed by the MTU, it can have a negative impact on network performance. Fragmentation and reassembly protocols incur additional complexity within the system. Further, since multiple packets now must be transmitted for each original packet, the likelihood of packet loss increases. The impact of packet loss may be particularly severe for wireless IoT systems, where resources are heavily constrained.

We study this issue in the context of a common type of IoT system, a wireless sensor network (WSN) that exhibits a tree-based topology. Tree-based systems have base-stations or gateways that coordinate the activities of a number of devices. There are two fundamental communication flows. The first is *many-to-one,* where each device transmits its packets up the tree to the gateway. The second is *one-to-many*, where the gateway sends a single control or management packet to each of the devices in the system. The many-to-one mode uses unicasting, from individual device to the gateway. The one-to-many mode uses broadcasting, where the packet must be fully received by each of the devices in the system.

Our goal is to precisely quantify the differences in metrics such as Packet Delivery Ratio (PDR) between situations where fragmentation occurs and where it does not occur. For instance, one natural consequence of the tree-based topology is that the devices near the gateway experience higher levels of congestion and loss under the many-to-one mode. This congestion issue could be exacerbated by excess fragmentation.

Another issue is that broadcast applications require that each device fully reassemble each packet, in order to recover the message from the gateway. Reliable broadcasting in an IoT-like wireless system is in general a difficult problem, and involves tradeoffs between message overhead and high delivery rates. To increase the reliability of one-to-many broadcasting we propose *Radiate*, which is based on the use of IPv6 like mechanisms and transmission timers designed to increase packet delivery and decrease congestion.

We conducted experiments comparing fragmentation with no fragmentation, under a number of topologies and packet workload scenarios. Our results indicate that a naive fragmentation strategy has a seriously negative impact on packet delivery ratios under the many-to-one communication mode. We also determined that careful management of wireless broadcasting such as that exemplified by Radiate can eliminate the performance differences between fragmenting and non-fragmenting stratgies. We believe these results will be highly instructive for IoT network engineers.

2. Background and related work

Our work is aimed at IoT devices containing low power communication architectures such as those supported by the IEEE 802.15.4 specification [1]. This technology is designed to support applications such as the Smart Grid, data center power control, industrial networks or building and home automation [2]. These IoT systems are populated by resource constrained devices, have unreliable communication links and low data rates, and are referred to as low-power lossy networks (LLNs) [6].

Traditional IP protocols fail to address the operating characteristics within a LLN environment. To address this problem the IETF has defined several protocols, including 6LoWPAN, that are suitable for a LLN environment [7]. The 6LoWPAN protocol defines a number of addressing, compression and header extension options, including one for packet fragmentation.

Packet fragmentation is the process of breaking packets into smaller fragments, and then resending each fragment as a separate packet. Fragmentation occurs because different networks connected through the Internet can have different maximum sizes or MTUs. For instance, the maximum size of the 802.11 WiFi data payload is 2312 bytes, but the maximum size of the data payload carried by many variants of the 802.15.4 standard is 104 bytes. Packets are fragmented when they arrive on a link that has a smaller maximum size then the packet itself. In IP each fragment is retransmitted in a separate packet, and the entire packet is reassembled at the end-host, once all of the fragments are received.

In 6LoWPAN an adaptation layer sitting between the 802.15.4 link layer and the network layer performs fragmentation. One major difference between fragmentation in IPv4 and IPv6 versus fragmentation in 6LoWPAN is that the latter does *not* copy IP header information into each packet. This means that each device may only be able to reassemble fragments originating from one original packet at a time. Fragments from other packets would need to be dropped.

Although there have been a number of published evaluations for IoT systems in a LLN environment (see, for instance, [4,8,9]) there has been little published work assessing the impact of fragmentation for many-to-one and one-to-many communication modes.

3. Reliable Broadcasting with *Radiate*

Broadcasting is the basic communication operation to support tree-based one-to-many communication. To improve reliability and decrease the overhead associated with a purely flooding strategy, we designed the Radiate protocol. Radiate combines elements from IPv6 with carefully tuned retransmission timers. In particular, Radiate uses the multicast capability of the IPv6 protocol. The sender broadcasts messages using UDP to the all local devices network address ff02::1. Radiate adds a small data structure to all broadcast packets. This structure is defined as:

```
uint32_t sequence_number;
uint16_t source;
```

```
uint16_t length;
uint8_t payload[PAYLOAD_SIZE];
```

When a device has data to send it increments the sequence_number and the data is copied into the payload. The device transmits the message using the networking stack IPv6, 6LoWPAN, and finally IEEE 802.15.4, and then sets a *Trickle* timer to send the message again a some random time in the next second [7]. This is a common technique designed to reduce the chance of collisions. Each instance the timer expires the message is again randomly broadcast, doubling the timer interval (e.g. 1, 2, etc.) until a maximum of 4 is reached. Typically a total of 4 broadcasts are performed for each send execution.

Normally this broadcasting would be excessive, so Radiate mitigates this by increasing the timer if a device overhears the same sequence_number that it is currently retransmitting, canceling the current timer. Thus, if a device broadcasts a message and three neighbors successfully receive and then also broadcast, the device will have increased beyond the maximum and no longer rebroadcast. This heuristic results in devices in dense topologies likely only sending once with devices in sparser (or near the edges of networks) likely sending closer to the maximum rebroadcast.

Devices receiving the broadcast check the sequence_number ensure it is more recent and, if so, start broadcasting in a similar fashion. Eventually all devices will receive the broadcast with the latest sequence number and the re-broadcasting stops.

4. Evaluation Methodology

The goal of our evaluation was to conduct performance studies using a realistic set of network level and application level programs. We constructed a simulation environment using the ContikiOS Simulator, Cooja [3]. For devices we used the Zolertia Z1 series built using a MSP430 processor and CC2420 radio. The system we programmed used RPL routing [7], a simple data reporting application for the many-to-one-mode, and the Radiate protocol carrying periodic control messages for the one-to-many mode.

To evaluate performance under different topologies we used the Unit Disk Graph Medium Distance Loss model with a 50-meter transmission range and a 100 meter interference range. We created three different topologies – 16 devices, 36 devices and 64 devices. The 64 device topology is shown in Figure 1. The topology is based loosely on a grid structure, with an average of 30 meters between devices. Within this structure each device is randomly placed within a proportionally smaller square area. This introduces some variability while still maintaining a connected network. The sink is placed in the middle of the network.

For each topology, a set of experiments was performed to determine the Packet Delivery Ratios (PDR) versus application message rates for both many-to-one (unicast) and one to many (broadcast) communication modes. Note that during this time normal network routing and management traffic continued to be transmitted. Each rate is run for three minutes at which point the rate in increased by three and again run for three minutes up to maximum rate of 30 messages / minute. The rate starts at 3 messages / minute, therefore, the individual experiment runs a total of 30 minutes.

Fragmentation is induced by setting the 6LoWPAN payload threshold to 75 bytes instead of 100 with approximately 25 bytes reserved for lower layer headers. The unicast and broadcast messages are 85 bytes. The fragmentation scenario results in two fragments/transmissions for the sample messages versus only requiring one for the non-fragmentation scenario.

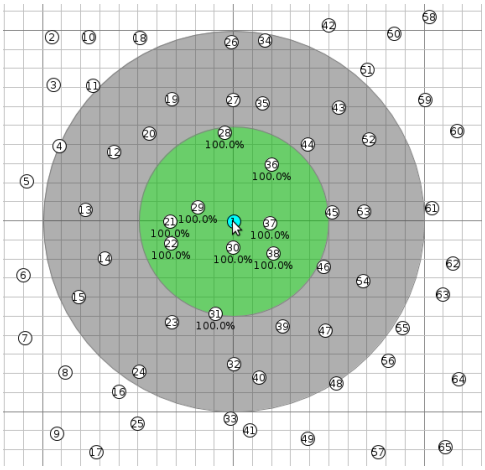

Figure 1: 64 device topology with the gateway in the center

5. Results

The PDR for each device was calculated by taking the number of application messages received divided by the number of application messages sent. The average was then taken over the devices to determine the PDR for the

scenario. The results shown plot this average PDR on the y-axis for the varying rates (in application messages/minute) on the x-axis. The fragmentation scenario is labeled as fragunicast and fragbroadcast; the non-fragmentation scenario is labeled nofragunicast and nofragbroadcast

The unicast scenario results are depicted in Figures 2, 3 and 4. All three graphs clearly show that fragmentation severely degrades the PDR for the different topologies. As the rate increases, the PDR for both scenarios appear to be affected proportionally. Figure 2 shows an increasing difference between fragmentation and non-fragmentation from approximately 5% for a rate of 6 messages / minute to over 50% difference for rate 33 messages / minute.

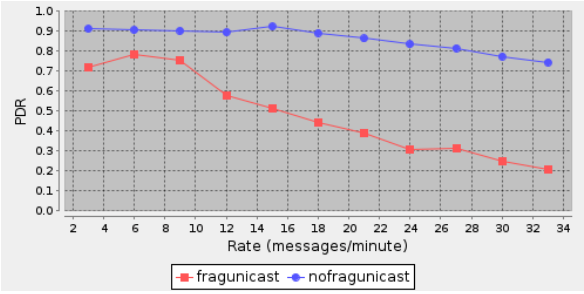

Figure 2 - Unicast 16 Devices

Figure 2 shows that the fragmentation has an initial 10% minor improvement, however, progressively worsens as the rate increases. The non-fragmentation scenario appears to be less affected by the rate increases.

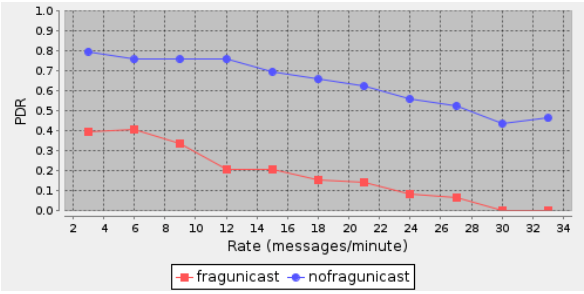

Figure 3 – Unicast 36 Devices

Figures 3 and 4 both show the stress both scenarios experience as the rate increases in larger networks. Figure 3 indicates that the non-fragmentation scenario consistently delivers 40% more messages for all rates. Figure 4 shows that eventually both scenarios essentially fail to deliver any many-to-one messages to the sink. We believe this is due to feeder routes near the sink becoming expectedly congested.

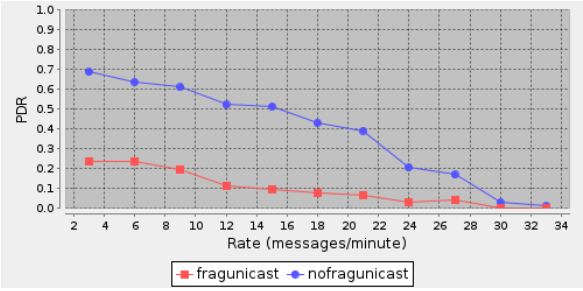

Figure 4 - Unicast 64 Devices

The unicast results clearly show the tremendously negative impact of fragmentation on packet delivery rates. This strongly suggests that network engineers should modify their protocols to either avoid excessive fragmentation or provide additional reliability mechanisms.

The broadcast results, using radiate, are shown in Figures 5, 6, and 7. Surprisingly, it appears that fragmentation has no discernible affect on the PDR.

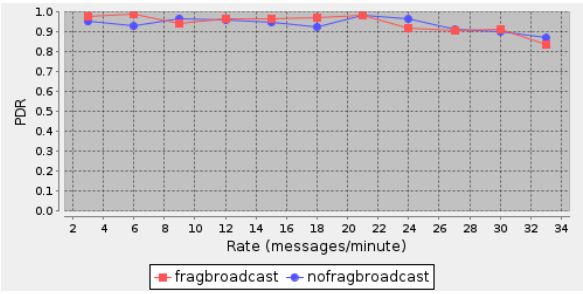

Figure 5 - Broadcast 16 Devices

Figure 5 shows both scenarios successfully deliver 90% or more of the messages for all rates except the last which still achieves approximately 88% at 33 messages / minute.

For larger networks and higher rates, Radiate still delivers good performance. For rates up to 24 messages / minute, the PDR is at or greater than 90%. However, Figures 6 and 7 show both begin to degrade towards 70% for a rate of 33 messages / minute.

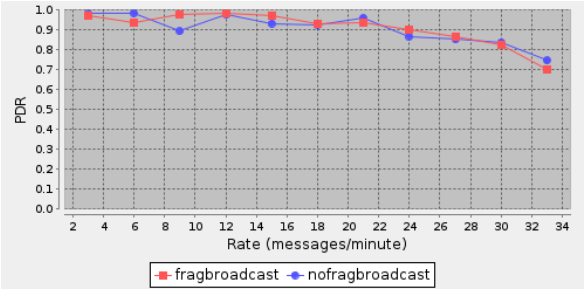

Figure 6 - Broadcast 36 Devices

We investigated the log files for both fragmentation and non-fragmentation scenarios and determined that indeed fragmentation was occurring when expected and messages were being dropped due to reassembly errors.

We believe the difference between the unicast results and the broadcasts results are due to two reasons:

1. Broadcast messages reassembled each hop
2. Neighbors rebroadcast packets allowing correct reception of previously missed fragments

Because the messages are reassembled by the neighboring devices, for each neighbor the full message can then be transmitted again. Even if a message is not received due to a fragmentation reassembly error, the device very likely has several more opportunities to receive the message.

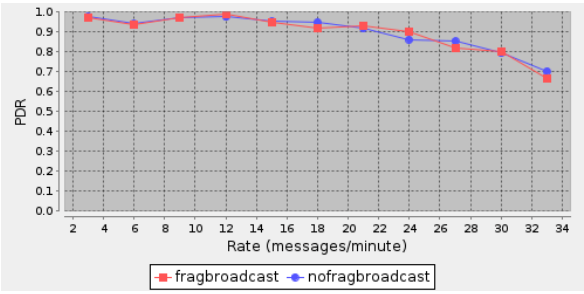

Figure 7 - Broadcast 64 Devices

Finally, we note that the rate appears to have a much greater affect on PDR than the topology. Topology certainly plays a role, however, as Figures 6 and 7 show, there is little difference between their PDR for the varying rates.

6. Conclusions

This paper examined the impact of packet fragmentation at the data link level on the end-to-end performance of IP-based protocols designed to support IoT systems. Using a realistic mixture of applications and network layer routing functions our results show that fragmentation can seriously degrade the performance of the typical IoT device to gateway communication modality. For one-to-many communication we presented the Radiate broadcast protocol. Our results showed that with proper design of data broadcast mechanisms, gateway-to-device communication can maintain high performance levels.

7. Acknowledgements

This work is supported by NSF under grants CNS-1116122 and CNS-1205453.

8. References

[1] 802.15.4e-2012: IEEE Standard for Local and Metropolitan Area Networks – Part 15.4: Low-Rate Wireless Personal Area Networks (LRWPANs) Institute of Electrical and Electronics Engineers Std., 16 April 2012.

[2] L. Atzori, L., Iera, A., and Morabito, G. "The Internet of Things: A survey," Computer Networks, vol. 54, no. 15, pp. 2787–2805, October 2010.

[3] A. Dunkels, "Contiki OS," http://www.sics.se/contiki/wiki/index.php/Main-Page.

[4] Gaddour, O. et al. "Simulation and performance evaluation of DAG construction with RPL", *2012 Third International Conference on Communications and Networking.*

[5] Hui, J.W. and Culler, D.E. , "IPv6 in Low-Power Wireless Networks," Proc. IEEE, vol. 98, no. 11, pp. 1865 – 1878, November 2010.

[6] Jeonggil, K. et al., "Connecting Low-power and Lossy Networks to the Internet," IEEE Communications Magazine, vol. 49, no. 4, pp. 96 – 101, April 2011.

[7] Levis, P., Tavakoli, A. , and S. Dawson-Haggerty, A. Overview of Existing Routing Protocols for Low Power and

Lossy Networks, IETF Draft Report, http://tools.ietf.org/html/draft-ietf-roll-protocols-survey-07, 2009.

[8] Long, N et al, "Comparative performance study of RPL in Wireless Sensor Networks"*, 2012 IEEE 19th Symposium on Communications and Vehicular Technology in the Benelux.*

[9] Tripathi, J. and de Oliveira, J.C. "On adaptive timers for improved RPL operation in low-power and lossy sensor networks," *2013 Fifth International Conference on Communication Systems and Networks.*