Low-power Interoperability for the IPv6-based Internet of Things

Adam Dunkels, Joakim Eriksson, Nicolas Tsiftes
Swedish Institute of Computer Science
{adam,joakime,nvt}@sics.se

1 Introduction

The Internet of Things requires interoperability and low power consumption, but interoperability and low power consumption have thus far been mutually exclusive. This talk outlines the challenges in attaining low power operation for the IPv6-based Internet of Things, how this affects interoperability, and what must be done to combine the two.

Research and standardization has come a long way towards providing efficient protocols and specifications for IPv6 for the Internet of Things. The efforts of the IETF 6lowpan [9] and ROLL [13] working groups and the the IPSO Alliance [3] have resulted in protocols and interoperability testing frameworks for those protocols. One recent result is the IETF RPL IPv6 routing protocol for low-power, lossy networks which was recently moved towards a standard RFC document [12, 13].

The first step towards interoperability for the Internet of Things is interoperability at the IPv6 layer. In a joint project between Cisco, Atmel, and SICS, the Contiki operating system and its uIPv6 stack became the first low-power wireless operating system to provide a certified IPv6 Ready stack [4].

The second step is interoperability at the routing layer. The RPL protocol provides a framework for interoperable routing. Recent versions of Contiki contains ContikiRPL, one of the first implementations of the IETF RPL routing protocol [6, 11]. ContikiRPL has previously been successfully tested for interoperability through the IPSO Alliance interop program, where it was used on three different platforms and ran over two different link layers, IEEE 802.15.4 and the Watteco low-power power-line communication module. In a joint project between Johns Hopkins University, UC Berkeley, and SICS, we have demonstrated interoperability between the RPL implementations in Contiki and TinyOS [6]. Providing interoperability between two different operating systems was not without challenges: our results show that the resulting system performance depends on numerous implementation-specific factors and that interoperability therefore is not necessarily a binary property.

The next step for interoperability is low-power interoperability. Existing protocols such as RPL are designed for running over radio layers such as IEEE 802.15.4. Radios such as IEEE 802.15.4 are simpler and have a lower output power than radios such as WiFi and Bluetooth. To attain a lifetime of years of batteries, however, the radio must be efficiently duty cycled so that it is kept off for most of the time. But radio duty cycling creates a new set of dynamics for which existing protocols have not been designed [2, 7].

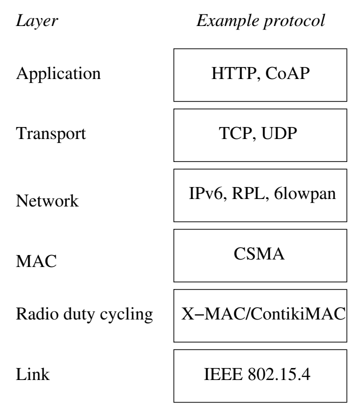

Figure 1. The low-power IPv6 stack consists of the standard IPv6 protocols at the network layer and transport layers, and of new protocols from the network layer and down.

Existing interoperability experiments have not taken power consumption into account, but have been performed with an always-on radio layer. Contiki provides a set of radio duty cycling mechanisms such as ContikiMAC [2], X-MAC [1], and LPP [8]. By running uIPv6 and ContikiRPL over ContikiMAC, we have been able to attain as low power consumption with IPv6/RPL as with specialized sensor network protocols such as Contiki Collect. Our results show that the radio can be kept off more than 99% of the time while attaining full IPv6 communication, providing years of lifetime on batteries. But these low-power results have been achieved in a Contiki-only environment. Achieving full low-power interoperability has yet to be done.

2 IPv6 for Low-Power Wireless

The IPv6 stack for low-power wireless follows the IP traditional architecture but with a set of new protocols from the network layer and down, as shown in Figure 1. Header compression is provided by the 6lowpan adaptation layer. Routing for low-power and lossy networks is provided by the RPL protocol.

The headers of IPv6 packets tend to be large compared to the typical amount of data in low-power wireless networks. The header size adds to the energy required to transmit and receive packets and also increases the probability of bit-errors in transit. To reduce the size of the headers, IP networks traditionally use a technique called header compression. For low-power wireless networks, the IETF 6lowpan group has specified a header compression mechanism for low-power wireless networks based on the IEEE 802.15.4 standard [9]. Because the IEEE 802.15.4 maximum frame size is small (127 bytes), the group also devised a link-layer fragmentation and reassembly mechanism.

Low-power wireless networks tend to be multi-hop since the physical range of each device is small. To reach devices in a multi-hop network, a routing protocol is needed. In the IP architecture, routing occurs at the IP level. For low-power wireless networks, the IETF ROLL group have designed a routing protocol called RPL [12, 13]. RPL is optimized for the many-to-one traffic pattern that is common in many low-power wireless applications but also supports any-to-any routing. In RPL, a root node builds a directed acyclic graph through which IPv6 packets are routed. Since different low-power wireless applications have different demands on the network traffic, RPL supports different metrics by which the graph can be constructed. Likewise, after the graph has been constructed, different parent selection strategies are supported. In RPL, these are called objective functions.

At the MAC, radio duty cycling, and link layers, the IETF does not specify what mechanisms that should be used. These layers are typically defined by other organizations such as the IEEE. For low-power wireless IPv6, the most common is to use CSMA at the MAC layer and IEEE 802.15.4 at the link layer. At the radio duty cycling layer, no standard or default mechanisms have yet been defined.

3 Low-Power Implies Duty Cycling

Radio duty cycling is essential to attaining low power consumption. Without duty cycling, network lifetime is counted in days. To reach a network lifetime of years, duty cycling is needed.

The radio transceiver is the most power-consuming component of many low-power wireless devices. To reduce power consumption and to extend system lifetime, the radio transceiver must be efficiently managed. But the radio transceiver consumes as much power when it is in idle listening mode as it is when actively transmitting messages. Therefore, it is not enough to reduce transmissions: to save power, the radio transceiver must be completely switched off for most of the time. But when the transceiver is switched off, the device cannot receive messages from neighbors, making it difficult to participate in the network.

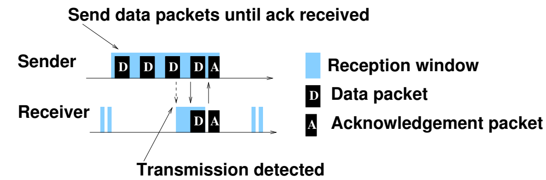

Figure 2. ContikiMAC, from Dunkels et al. [2].

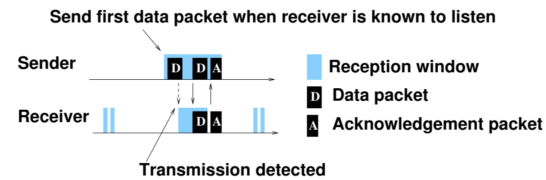

Figure 3. ContikiMAC sender phase-lock.

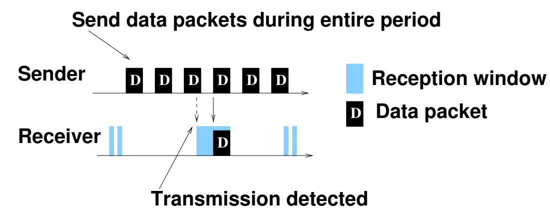

Figure 4. ContikiMAC broadcast.

To allow low-power wireless devices to actively participate in a low-power wireless network while maintaining a low power consumption, the radio transceiver must be duty cycled. With radio duty cycling, the radio is switched off most of the time, but switched on often enough to allow the device to receive transmissions from other nodes. Over the years, many different duty cycling schemes have been designed [1, 2, 5, 10].

To illustrate the concept of duty cycling, we look at ContikiMAC, the default duty cycling mechanism in Contiki [2]. The principles of ContikiMAC is illustrated in Figure 2, Figure 3, and Figure 4. In ContikiMAC, nodes periodically wake up to check for a transmission from a neighbor. To transmit a message, the sender repeatedly transmits the packet until an acknowledgment is received from the receiver. After a successful transmission, the sender has learned the wake-up phase of the receiver, and subsequently needs to send fewer transmissions. A broadcast transmission must wake up all neighbors. The sender therefore extends the packet train for a full wake-up period.

Radio duty cycling gives a low power consumption but both brings costs in terms of reduced bandwidth and introduces new network dynamics [2, 7]. Different types of transmissions have different implications in terms of power consumption and radio interference. Broadcast transmissions typically cost more than unicast transmissions, as shown in Figure 4. Existing protocols such as RPL do not take these dynamics into account. How radio duty cycling affects the behavior and performance of protocols such as RPL is still an area of open research.

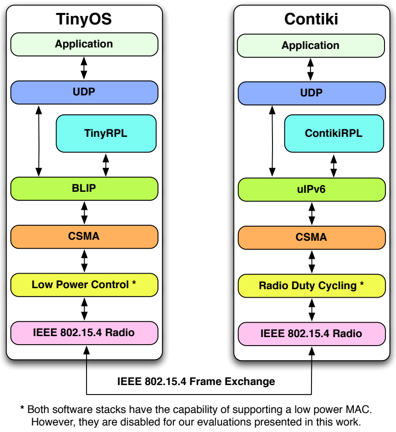

Figure 5. Contiki and TinyOS IPv6 interoperability, from Ko et al. [6]. We demonstrated interoperability at the network layer, the MAC layer, and the link layer, but without radio duty cycling.

4 Low-Power Interoperability

To attain low-power interoperability for IPv6 for the Internet of Things, interoperable radio duty cycling is essential. We have demonstrated interoperability between Contiki and TinyOS [6], but with an always-on radio layer. Our experiments showed that interoperability is not a binary property: two implementations that have good performance on their own can have a suboptimal performance in a mixed network. This is due to subtle variations in implementation choices and low-level details. Our results suggest that implementations of Internet of Things protocols need to be tested not just for correctness but also for performance. Given that interoperability in the simpler case of an always-on radio provides such unexpected results, we have reason to believe that interoperability with duty cycling will provide many unforeseen challenges.

We see at least three challenges in attaining low-power interoperability. First, existing duty cycling mechanisms have not been designed for interoperability. Mechanisms such as ContikiMAC and the TinyOS BoX-MAC protocols are defined by their implementations and no formal specifications have been developed. Standardization within the IEEE 802.15.4e group have taken the first steps in this direction. Second, duty cycling protocols are typically timing-sensitive, making it difficult to develop and test interoperable implementations. Third, traditional interoperability testing practices, which are based on physical meetings that are bounded in time, have not been well-suited for testing interoperability between duty cycling protocols.

The Contiki simulation environment provides a way to develop and test interoperability between duty cycling mechanisms across operating systems. We have already used the Contiki simulation environment to demonstrate interoperability between Contiki and TinyOS. We believe that the Contiki simulation environment is an important tool in addressing the challenges of low-power IPv6 interoperability.

5 Conclusions

IPv6 provides interoperability for the Internet of Things, but attaining low-power interoperability still is an open problem due to at least two issues. Existing protocols for low-power wireless typically have not been designed for duty cycling and existing duty cycling mechanisms have not been designed for interoperability. Solving low-power interoperability is crucial to making the Internet of Things a reality.

Acknowledgments

This work was funded by the Swedish Strategic Research Foundation and the EU Commission.

6 References

[1] M. Buettner, G. V. Yee, E. Anderson, and R. Han. X-MAC: a short preamble MAC protocol for duty-cycled wireless sensor networks. In *Proceedings of the International Conference on Embedded Networked Sensor Systems (ACM SenSys)*, Boulder, Colorado, USA, 2006.

[2] A. Dunkels, L. Mottola, N. Tsiftes, F. Österlind, J. Eriksson, and N. Finne. The announcement layer: Beacon coordination for the sensornet stack. In *Proceedings of the European Conference on Wireless Sensor Networks (EWSN)*, 2011.

[3] A. Dunkels and J-P Vasseur. IP for Smart Objects, September 2008. IPSO Alliance White Paper 1, available from www.ipso-alliance.org.

[4] M. Durvy, J. Abeillé, P. Wetterwald, C. O'Flynn, B. Leverett, E. Gnoske, M. Vidales, G. Mulligan, N. Tsiftes, N. Finne, and A. Dunkels. Making Sensor Networks IPv6 Ready. In *Proceedings of the International Conference on Embedded Networked Sensor Systems (ACM SenSys)*, Raleigh, North Carolina, USA, November 2008.

[5] J. Hui and D. Culler. IP is Dead, Long Live IP for Wireless Sensor Networks. In *Proceedings of the International Conference on Embedded Networked Sensor Systems (ACM SenSys)*, Raleigh, North Carolina, USA, November 2008.

[6] J. Ko, J. Eriksson, N. Tsiftes, S. Dawson-Haggerty, A. Terzis, A. Dunkels, and D. Culler. ContikiRPL and TinyRPL: Happy Together. In *Proceedings of the workshop on Extending the Internet to Low power and Lossy Networks (IP+SN 2011)*, Chicago, IL, USA, April 2011.

[7] M. Lunden and A. Dunkels. The politecast communcation primtive for low-power wireless. *The ACM SIGCOMM Computer Communications Review*, April 2011.

[8] R. Musaloiu-E., C-J. M. Liang, and A. Terzis. Koala: Ultra-Low Power Data Retrieval in Wireless Sensor Networks. In *Proceedings of the International Conference on Information Processing in Sensor Networks (ACM/IEEE IPSN)*, St. Louis, Missouri, USA, 2008.

[9] T. Narten, E. Nordmark, and W. Simpson. Neighbor Discovery for IP Version 6 (IPv6). RFC 2461, December 1998.

[10] Y. Sun, O. Gurewitz, and D. Johnson. RI-MAC: A Receiver-Initiated Asynchronous Duty Cycle MAC Protocol for Dynamic Traffic Loads in Wireless Sensor Networks. In *Proceedings of the International Conference on Embedded Networked Sensor Systems (ACM SenSys)*, Raleigh, NC, USA, 2008.

[11] N. Tsiftes, J. Eriksson, and A. Dunkels. Low-Power Wireless IPv6 Routing with ContikiRPL. In *Proceedings of the International Conference on Information Processing in Sensor Networks (ACM/IEEE IPSN)*, Stockholm, Sweden, April 2010.

[12] J.P. Vasseur and A. Dunkels. *Interconnecting Smart Objects with IP: The Next Internet*. Morgan Kaufmann, 2010.

[13] T. Winter (Ed.), P. Thubert (Ed.), and RPL Author Team. RPL: IPv6 Routing Protocol for Low power and Lossy Networks. Internet Draft draft-ietf-roll-rpl-18, work in progress.