Extending the Internet with TOSKI
(TinyOS Kernel - IPv6)

Jonathan Hui, Chad Metcalf, Alec Woo
Arch Rock Corporation

IPSN 2009
San Francisco, California
April 16th, 2009
Schedule

• Brief Introduction to IPv6/6LoWPAN (20 min)
  • Sensor networking (SerialForwarder vs. IP router)
  • IPv6 Essentials
  • Defining an IP link (route-over vs. mesh-under)
  • Delivering IPv6 datagrams (6LoWPAN adaptation)
  • TOSKI overview

• Hands-on Tutorial (80 min)
  • UDP (20 min)
  • TCP (20 min)
  • Telnet (20 min)
  • HTTP (20 min)

• Open playtime/discussion (20 min)
Sensor Networking Past

- Interconnect:
  - Bridge node, serial forwarder
  - Stateful application gateways

- Networking abstractions
  - Collection, dissemination

- No network architecture
  - No discovery, naming, addressing, configuration, and management
Sensor Internetworking

- **Interconnect:** IP routers/egress points
  - No application state - nodes communicate end-to-end with destination
  - Small number of egress - optimizations (e.g. proxy, cache, translation)

- **Networking abstractions**
  - Unicast, Multicast

- **Network architecture**
  - IP addresses, UDP/TCP ports, ICMPv6, ...
Introduction to IPv6

- IPv6 specified in 1998 - RFC 2460

- Goals of IPv6
  - Scalability
  - Unattended operation
  - Configurability and manageability
  - Flexibility and extensibility

- UDP and TCP largely unchanged
IPv6 Addressing

- 128 bits
  - Multiple IPv6 addresses for every grain of sand on the earth!
  - Types: Unicast, Multicast, Anycast

- Notation
  - Unicast: 2001:0db8:85a3::5678
  - Multicast: FF02::1 (link-local all-nodes)
  - Prefix: 2001:0db8:85a3::/64

- Address Format
  - Prefix: Identifies a subnet
  - IID: Identifies a interface in the subnet - usually derived from L2 address

- Scopes
  - Link-Local: interfaces within a single link only
  - Global: interfaces anywhere in the Internet.
IPv4 Mapped Addresses

- IPv6 addresses can carry IPv4 addresses
  - ::ffff:192.168.0.1
  - ::ffff:c0a8:1

- Allows IPv6-only nodes to communicate with IPv4-only nodes
  - Stateless IP/ICMP Translation (SIIT)
Configuration & Management

- IPv6 Neighbor Discovery
  - Address resolution
  - Neighbor unreachability detection
  - Duplication address detection
  - Default route selection

- Stateless Address Autoconfiguration
  - Prefix advertised by router combined with L2-derived IID

- ICMPv6
  - Reachability
  - Availability
  - ...
Building an IPv6 Sensornet

- Defining the IP link
- Delivering IPv6 datagrams
What is an IP link?

Nodes reachable by a single IP hop
Defining the IP Link

- **Single IP Link**
  - Hide physical connectivity from IP
  - Switched Ethernet and WiFi
  - “Routing/Forwarding” at Layer 2 and Layer 3

- **Multiple IP Links**
  - IP-level visibility into physical connectivity
  - “Routing/Forwarding” only at Layer 3
  - MPLS over ATM
Defining the IP Link

- **IP Link <=> PAN (mesh-under)**
  - Every node is a 15.4 switch
  - Access to link-specific mechanisms
  - Hide radio specifics from IP???

- **IP Link <=> Radio (route-over)**
  - Every node is an IP router
  - Single layer of routing/forwarding
  - IP overlays based on IP
Delivering IPv6 Datagrams

6LoWPAN Adaptation

- Fragmentation
  - IPv6 minimum MTU requirement is 1280 bytes!

- IPv6 Header Compression
  - Basic UDP/IPv6 header is 48 bytes!
6LoWPAN Header Format

- “Stacked-header” format inspired by IPv6 extension headers

- Each header contains:
  - Header Type that identifies the header
  - Header-specific fields
  - Payload

![Diagram of 6LoWPAN Header Format](image-url)
6LoWPAN Fragment Header

- First two bits identify header
- O-bit indicates of Datagram Offset is included
- 11-bit Datagram Size supports up to 2048 bytes
- Datagram Tag identifies fragments for a datagram
IPv6 Header Compression

- **UDP/IPv6/15.4 Header: 58 bytes!**
- **Remove redundant information across layers (20.5 bytes)**
  - Version (0.5): by virtue of using 6LoWPAN
  - Lengths (4): derived from 15.4 or Fragment header
  - Address suffixes (16): derived from 15.4 addresses
- **Compress commonly used values (24.5 bytes)**
  - Traffic Class/Flow Label (3.5): zero
  - Address prefixes (16): network prefix
  - Common multicast addresses (16): all-nodes, etc.
  - UDP ports (3): use one of 16 ports
  - Checksum (2): covered by upper layer
- **Result:**
  - 48-byte UDP/IPv6 header to 6 bytes in best case!
Some Examples

IEEE 802.15.4 Header - 22 bytes

<table>
<thead>
<tr>
<th>Length</th>
<th>FCF</th>
<th>DSN</th>
<th>PAN ID</th>
<th>Source Address (00-17-3B-FF-11-22-33)</th>
<th>Destination Address (00-17-3B-FF-44-55-66)</th>
</tr>
</thead>
</table>

Link-Local Unicast \((fe80::0217:3bff:fe11:2233 \rightarrow fe80::0217:3bff:fe33:4455)\)

- Dispatch
- IPHC
- NHC
- UDP Ports
- UDP Checksum
- 6 bytes

Link-Local Multicast \((fe80::0217:3bff:fe11:2233 \rightarrow ff02::1)\)

- Dispatch
- IPHC
- Mcast Grp
- NHC
- UDP Ports
- Checksum
- 7 bytes


- Dispatch
- IPHC
- CID
- Hop Limit
- Dst IID (0068)
- NHC
- UDP Ports
- UDP Checksum
- 10 bytes
TOSKI Overview

- Get people thinking about IPv6 sensor applications
- TinyOS-Based Network Kernel
  - 6LoWPAN adaptation layer
  - IPv6 network layer
  - UDP/TCP transport layer

Diagram:
- APP: HTTP, Telnet, SNMP, DNS, ..., DHCPv6
- TRN: UDP, TCP
- NET: Forwarder, Forwarding Table, Router, Routing Table, ICMPv6 Discovery, Autoconf
- LNK: Data, Ack, Neighbor Table
- PHY: IEEE 802.15.4
TOSKI Overview

- TOSKI wrapped behind a single component:
  ```
  configuration KernelC {
    provides interface Boot;
    provides interface Leds;
    provides interface Udp as Udp0;
    provides interface Tcp as Tcp0;
  }
  ```

- Single socket for UDP and TCP each

- Included with the evaluation release:
  - Simple Telnet server
  - Simple Http server
  - Simple MSP430 ADC 12 driver
Hands-on

- Now let’s play with TOSKI.