LWIG guidance document
draft-bormann-lwig-guidance-01

Carsten Bormann <cabo@tzi.org>
IETF83, 2011-03-29
Scope

• **Try** doing one single document
  – Could very well “roadmap out” to specific documents
• Implementation guidance only
  – Not developing/changing any protocols or services
    • “The techniques shall [...] not affect conformance to the relevant specifications.”
  – Not software engineering best practices
• Guidance:
  – Focus on a limited number of protocols
  – Configuring a protocol for a constrained system
  – What can be left out — what’s the bare minimum?
2: “Constrained”

• Distinguish 2 rough classes of constrained nodes:
  
  • 10 KiB data/100 KiB code (“quite constrained”)
  • 50 KiB data/250 KiB code (“not so constrained”)

• In each case, make clear which class is being targeted

• (These are a starting point for making sure we discuss from the same requirements, not exact classes.)
Focus protocols

• The group shall focus only on techniques that have been used in actual implementations [...]  

• The topics for this working group will be chosen from these protocols: IPv4, IPv6, UDP, TCP, ICMPv4/v6, MLD/IGMP, ND, DNS, DHCPv4/v6, IPsec, 6LOWPAN, and RPL protocols.
Data Plane protocols

• Application Layer
  – HTTP, CoAP
  – Others? (XMPP, TFTP)

• Transport Layer
  – TCP, UDP
  – Others?

• Network Layer
  – IPv4, IPv6

• Link layer support
  – 6lowpan
3: Data Plane

• 3.1 Link Adaptation Layer
  – 3.1.1: 6LoWPAN fragmentation

• 3.4
  – 3.4.1: CoAP
    • message layer processing, message parsing, storing message IDs
Control Plane protocols

• Application Layer
  – DNS, DHCP, DHCPv6
  – Others? (SIP)

• Transport Layer
  – ?

• Network Layer
  – ICMP, ICMPv6, IGMP/MLD
  – RPL, AODV/DYMO, OLSRv2

• Link Layer support
  – ARP, ND
4: Control Plane

• 4.5.1: SNMP
  – (focusing on the “strncmp” style of implementation)
Security protocols

• TLS, ciphersuites, certificates
• IPsec, IKEv2, transforms, ...
• PANA, EAP, EAP methods
5: Security Protocols

- 5.4.1: PANA
“Wire-visible” constraints

- Checksum
- MTU
- Fragmentation and reassembly
- Options — implications of leaving some out
- Simplified TCP optimized for LLNs
- Out-of-order packets
“Wire-invisible” constraints

- Buffering
- Memory management
- Timers
- Energy efficiency
- API
- Data structures
- Table sizes (somewhat wire-visible)
- Improved error handling due to resource overconsumption
Next steps

• Document is still very sparse
• Looking for additional contributions
  – Input from CoAP interop likely
  – DTLS maybe?

• When are we ready for WG adoption?