Contiki-SNMP
—
Does SNMP fit on constrained devices?

Jürgen Schönwälder, Siarhei Kuryla

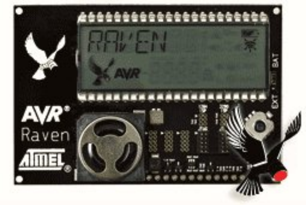
JACOBS UNIVERSITY

AIMS 2011, Nancy, 2011-06-16

AVR Raven Hardware

ATmega1284PV microcontroller:

- runs at 20 MHz
- 16K of RAM
- 128K of ROM (Flash)



Contiki-SNMP

- Contiki is an operating system for embedded devices
- SNMP engine (written in C) for constrained devices
- built on top of the Contiki uIPv6 stack (6LoWPAN)

IEEE 802.15.4

- Small frame size (max frame size = 127 bytes)
- Low power devices (some battery operated)
- Limited memory and processing power
- Low bandwidth (max data rate = 250 kbps)
- Large scale and dense deployments
- Devices and channels tend to be unreliable
- Devices may use sleep schedules to conserve energy

IETF 6LoWPAN

- IPv6 over IEEE 802.15.4 (see RFC 4944)
- General motivation and overview (see RFC 4919)
- RPL (routing), COAP (web transfer protocol), ...

Management of 6LoWPAN Networks

Why 6LoWPAN management?

A: "Autonomic devices won't need management — so don't waste your time on the wrong problem..."

B: Well, no, for the foreseeable future, you will end up managing the autonomic system (it's just one more control loop)

A: But even then, SNMP is clearly the wrong choice since we now have RESTful protocols with modern data encoding

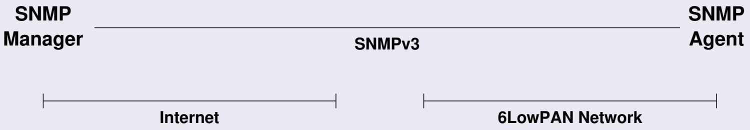
B: So lets do a comparison / competition...

Typical management questions

- How many nodes disappeared during the last night/day?
- How many nodes joined during the last week?
- What is the temperature, pressure, energy usage (add your favorite sensor here) distribution within the network?
- What is wrong with my home automation network?

SNMP and 6LoWPAN: End-to-End

SNMPv3 end-to-end



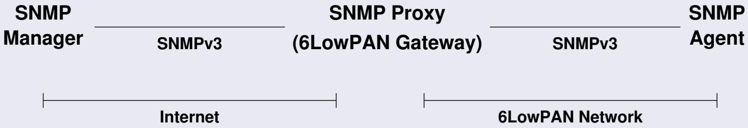
+ Straightforward direct access to individual 6LoWPAN nodes
+ Reuse of existing deployed SNMP-based tools

o End-to-end security, end-to-end key management

- Message size and potential fragmentation issues
- 6LoWPAN nodes must run an SNMP engine
- Trap-directed polling nature of SNMP has high (energy) costs

SNMP and 6LoWPAN: Proxies

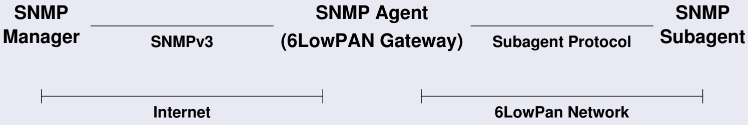
SNMPv3 proxies



+ Indirect access to individual 6LoWPAN nodes
+ Alternate transport encoding can reduce message sizes
o Reuse of existing SNMP-based tools supporting proxies
o Two security domains, different key management schemes
- 6LoWPAN nodes must run an SNMP engine
- Trap-directed polling nature of SNMP has high (energy) costs

SNMP and 6LoWPAN: Subagents

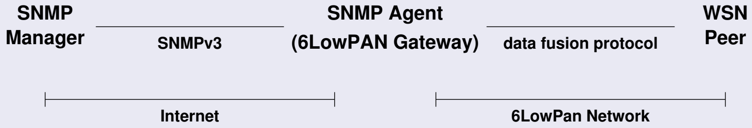
SNMPv3 subagents



+ Indirect access to individual 6LoWPAN nodes
+ Alternate transport encoding can reduce message sizes
- o Reuse of existing SNMP-based tools supporting contexts
- o Two security domains, different key management schemes
- o 6LoWPAN nodes must run an SNMP subagent
- \- Trap-directed polling nature of SNMP has high (energy) costs

SNMP and 6LoWPAN: Data Fusion Protocols

SNMPv3 interfacing to data fusion protocols



+ Indirect access to individual 6LoWPAN nodes
+ Leveraging data fusion protocols (in-network aggregation)
+ SNMP agent acting as a cache, no expensive polling
- o Reuse of existing SNMP-based tools supporting contexts
- o Two security domains, different key management schemes
- ? No direct advantage of 6LoWPAN technology — oops

Contiki-SNMP Overview

General features / limitations

- SNMP messages up to 484-byte length
- `Get`, `GetNext` and `Set` operations
- SNMPv1 and SNMPv3 message processing models
- USM security model, no VACM access control model
- API to define and implement managed objects

USM security algorithms

- HMAC-MD5-96 authentication protocol (RFC 3414)
- CFB128-AES-128 symmetric encryption protocol (RFC 3826)

Implemented MIB Modules and Static Memory Usage

MIB modules

- `SNMPv2-MIB` – SNMP entity information
- `IF-MIB` – network interface information (no 802.14.5 ifType)
- `ENTITY-SENSOR-MIB` – temperature sensor readings

SNMPv1 and SNMPv3 enabled

- 31220 bytes of ROM (around 24% of the available ROM)
- 235 bytes of statically allocated RAM

SNMPv1 enabled

- 8860 bytes of ROM (around 7% of the available ROM)
- 43 bytes of statically allocated RAM

Flash ROM and Static Memory Usage

Memory usage by software module (bytes)

Module	Flash ROM	RAM (static)
snmpd.c	172	2
dispatch.c	1076	26
msg-proc-v1.c	634	6
msg-proc-v3.c	1184	30
cmd-responder.c	302	0
mib.c	1996	6
ber.c	4264	3
usm.c	1160	122
aes_cfb.c	9752	40
md5.c	10264	0
utils.c	416	0

Stack and Heap Usage

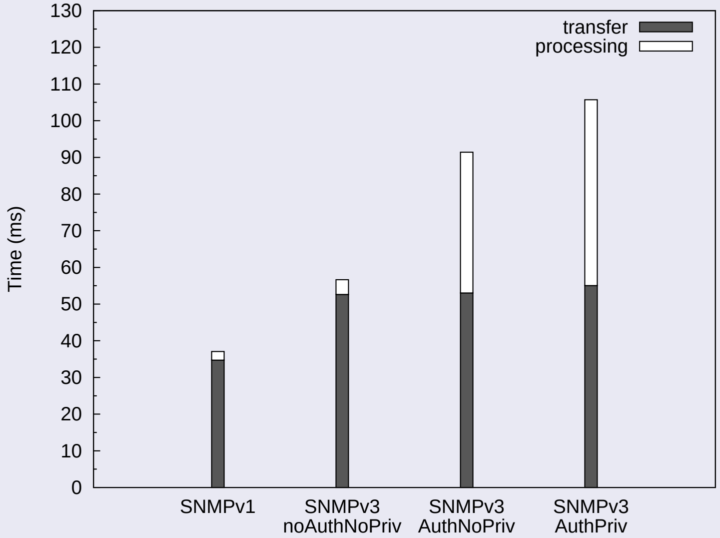
Maximum observed stack usage

Version	Security mode	Max. stack size
SNMPv1	–	688 bytes
SNMPv3	noAuthNoPriv	708 bytes
SNMPv3	authNoPriv	1140 bytes
SNMPv3	authPriv	1144 bytes

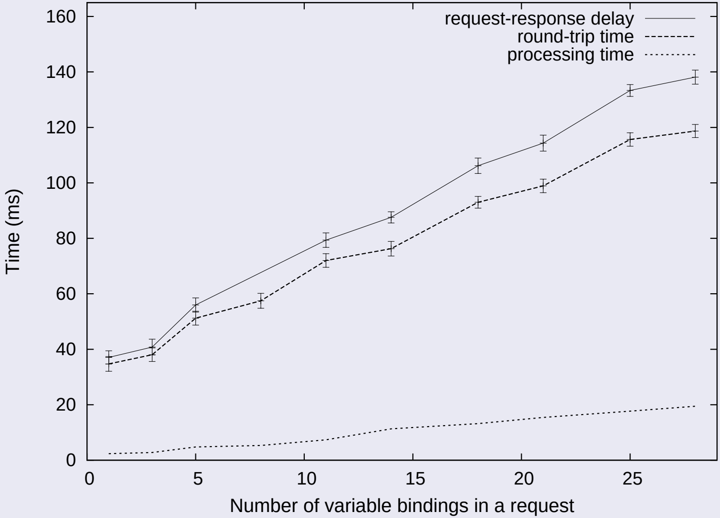
Heap usage

- not more than 910 bytes for storing an SNMPv1 message
- approximately 16 bytes for every managed object in the MIB
- if a managed object is of a string-based type, then additional heap memory is used to store its value

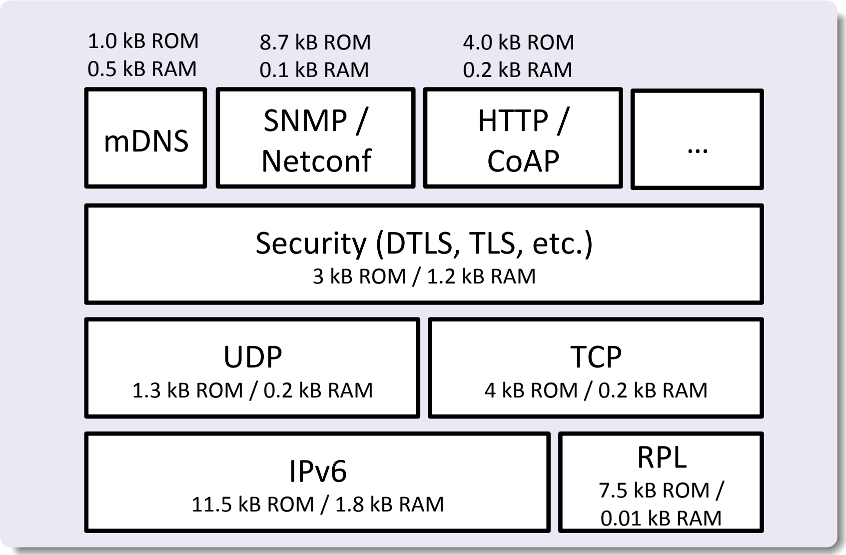
SNMP Request/Response Latency (varying security)



SNMPv1 Request/Response Latency (varying # varbinds)



Bigger Picture (resource requirements of various protocols)



Directly Related Work at Jacobs University

SNMP applicability to constrained devices

- Guidelines how to fit SNMP into constrained devices
- Tricks like making VACM a simple read-only/read-write switch
- <draft-hamid-6lowpan-snmp-optimizations-02.txt>

RPL MIB module specification and implementation

- Definition of a MIB module for the RPL routing protocol
- Implementation and evaluation on Econotags
- <draft-sehgal-roll-rpl-mib-01.txt>

DTLS for constrained devices

- Contiki-SNMP over DTLS (RFC 5590, RFC 5591, RFC 5953)

Other Related Work at Jacobs University

NETCONF Lite implementation and specification

- Profile (subset) of NETCONF 1.1 (RFC 6241)
  - Single session, hence trivial locking
  - No <edit-config>, no <get> / <get-config> filtering
  - No optional capabilities
  - No security (yet) ...
- First prototype shown at the Prague IETF (on AVR Ravens)
- <draft-schoenw-netconf-light-00.txt>

Multicast DNS for network management service discovery

- Managers use mDNS to discover manageable devices
- Devices discover management services via mDNS
- Contiki-mDNS implementation already running
- <draft-schoenw-opsawg-nm-srv-02.txt>

References

J. Ko, A. Terzis, S. Dawson-Haggerty, D. E. Culler, J. W. Hui, and P. Levis.
Connecting Low-Power and Lossy Networks to the Internet.
IEEE Communications Magazine, 49(4):96–101, April 2011.

S. Kuryla and J. Schönwälder.
Evaluation of the Resource Requirements of SNMP Agents on Constrained Devices.
In *Proc. of the 5th International Conference on Autonomous Infrastructure, Management and Security (AIMS 2011)*, LNCS. Springer, June 2011.

J. Schönwälder, H. Mukhtar, S. Joo, and K. Kim.
SNMP Optimizations for Constrained Devices.
Internet Draft <draft-hamid-6lowpan-snmp-optimizations-03.txt>, ETRI, Jacobs University, Ajou University, October 2010.

J. Schönwälder, T. Tsou, and C. Zhou.
DNS SRV Resource Records for Network Management Protocols.
Internet-Draft (work in progress) <draft-schoenw-opsawg-nm-srv-01>, Jacobs University, November 2010.

V. Perelman, J. Schönwälder, and M. Ersue.
Network Configuration Protocol for Constrained Devices (NETCONF Light).
Internet-Draft (work in progress) <draft-schoenw-netconf-light-00>, Jacobs University, Nokia Siemens Networks, June 2011.