GHC

Carsten Bormann, IETF90 (Toronto, 2014-07-24)

Status

- Technically completed in 6LoWPAN WG
- Presented at 6lo IETF88, 89
- draft-ietf-6lo-ghc-01.txt 2014-06-19
  - adopted as WG document 2013-12-06
  - further simplified in –01

6CIO option:

- IANA considerations: (RFC 5226)
  - “IETF Review or IESG Approval”
- Experimentation bits:
  - reserve bits 0–7 as “for experimentation, not to be used in deployments”.
- in –02 (oops)

Simplified Dictionary

- LZ77 compression is improved by priming it with a **dictionary**
- Dynamic part (32 bytes):
  - –00: was RFC 2460 pseudo header
  - –01: now just SA+DA (main change)
  - Thanks, Thomas Björklund!
- Static part: 16 bytes that *seem useful*

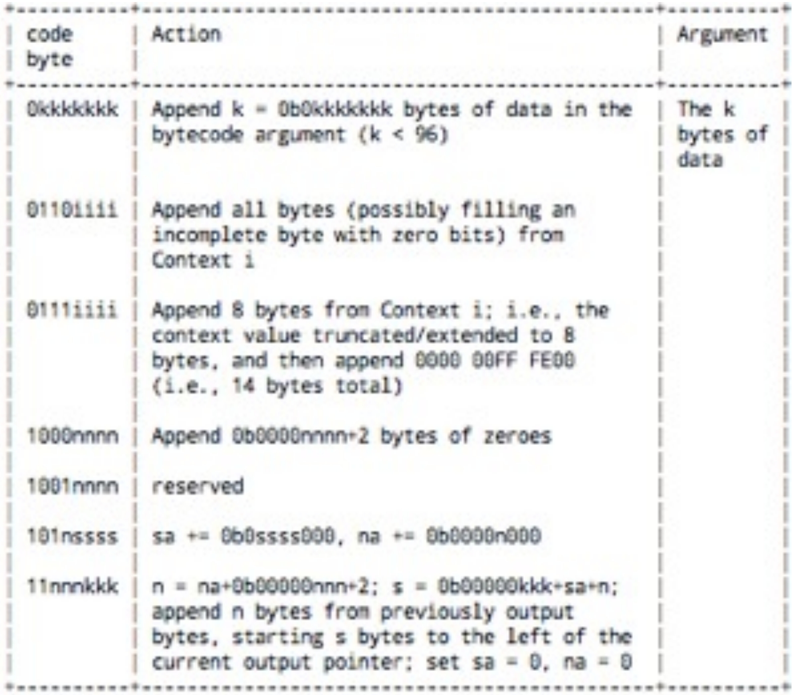
Static Dictionary

- Validated by examples in draft
- We don’t have to get this completely “right”
  - A suboptimal value just misses a few tenths of a percent of optimization
- Ever changing it will require more NHC allocations (and 6CIO bits)
  - Need 17 out of 256 (25 taken before)

Ship it.

6LoWPAN-GHC

- Generic compression of remaining headers and header-like payloads
- draft-ietf-6lo-ghc: simple LZ77 based on **bytecode**
  - **single-page** specification: simple
  - **stateless**
- provides modest compression factors between 1.65 and 1.85 on realistic examples
- fits in 6LoWPAN-HC's NHC



code byte	Action	Argument
0kkkkkkk	Append k = 0b0kkkkkkk bytes of data in the bytecode argument (k < 96)	The k bytes of data
0110iiii	Append all bytes (possibly filling an incomplete byte with zero bits) from Context i	
0111iiii	Append 8 bytes from Context i; i.e., the context value truncated/extended to 8 bytes, and then append 0000 00FF FE00 (i.e., 14 bytes total)	
1000nnnn	Append 0b0000nnnn+2 bytes of zeroes	
1001nnnn	reserved	
101nssss	sa += 0b0ssss000, na += 0b0000n000	
11nnnkkk	n = na+0b00000nnn+2; s = 0b00000kkk+sa+n; append n bytes from previously output bytes, starting s bytes to the left of the current output pointer; set sa = 0, na = 0	

Old:

draft-bormann-6lowpan-ghc-05.txt

Example: ND Neighbor Solicitation

- Payload:

```
 87 00 a7 68 00 00 00 00 fe 80 00 00 00 00 00 00
 02 1c da ff fe 00 30 23 01 01 3b d3 00 00 00 00
 1f 02 00 00 00 00 00 06 00 1c da ff fe 00 20 24
(RFC 2460-style) Pseudoheader:
 20 02 0d b8 00 00 00 00 00 00 00 ff fe 00 3b d3    Source IP address
 fe 80 00 00 00 00 00 00 02 1c da ff fe 00 30 23    Destination IP address
 00 00 00 30 00 00 00 3a                            Length, zeroes, next header
copy: 04 87 00 a7 68
4 nulls: 82
ref(32): fe 80 00 00 00 00 00 00 02 1c da ff fe 00 30 23
 -> ref 101nssss 1 2/11nnnkkk 6 0: b2 f0
copy: 04 01 01 3b d3
4 nulls: 82
copy: 02 1f 02
5 nulls: 83
copy: 02 06 00
ref(24): 1c da ff fe 00 -> ref 101nssss 0 2/11nnnkkk 3 3: a2 db
copy: 02 20 24
Compressed:
 04 87 00 a7 68 82 b2 f0 04 01 01 3b d3 82 02 1f
 02 83 02 06 00 a2 db 02 20 24
Was 48 bytes; compressed to 26 bytes, compression factor 1.85
```

Example: ND Neighbor Solicitation

```
   Payload:
 87 00 a7 68 00 00 00 00 fe 80 00 00 00 00 00 00
 02 1c da ff fe 00 30 23 01 01 3b d3 00 00 00 00
 1f 02 00 00 00 00 00 06 00 1c da ff fe 00 20 24
Dictionary:
 20 02 0d b8 00 00 00 00 00 00 00 ff fe 00 3b d3     Source IP address
 fe 80 00 00 00 00 00 00 02 1c da ff fe 00 30 23     Destination IP address
 16 fe fd 17 fe fd 00 01 00 00 00 00 00 01 00 00     Static Dictionary
copy: 04 87 00 a7 68
4 nulls: 82
ref(40): fe 80 00 00 00 00 00 00 02 1c da ff fe 00 30 23
 -> ref 101nssss 1 3/11nnnkkk 6 0: b3 f0
copy: 04 01 01 3b d3
4 nulls: 82
copy: 02 1f 02
5 nulls: 83
copy: 02 06 00
ref(24): 1c da ff fe 00 -> ref 101nssss 0 2/11nnnkkk 3 3: a2 db
copy: 02 20 24
Compressed:
 04 87 00 a7 68 82 b3 f0 04 01 01 3b d3 82 02 1f
 02 83 02 06 00 a2 db 02 20 24
Was 48 bytes; compressed to 26 bytes, compression factor 1.85
```