IPFIX interoperability report

Paul Aitken, Elisa Boschi, Lothar Braun, Thomas Dietz, Falko Dressler, Ronny Lampert, Lutz Mark, Jeroen Massar, Gerhard Münz, Carsten Schmoll, Christoph Sommer, Stan Yates

Overview

- Tests overview
- Unclear / ambiguous definitions
- Common implementation mistakes
- Open points

IPFIX Implementations

- CISCO
- Fraunhofer FOKUS
- IBM
- NEC
- Universities of Tübingen and Erlangen

Tests (1/2)

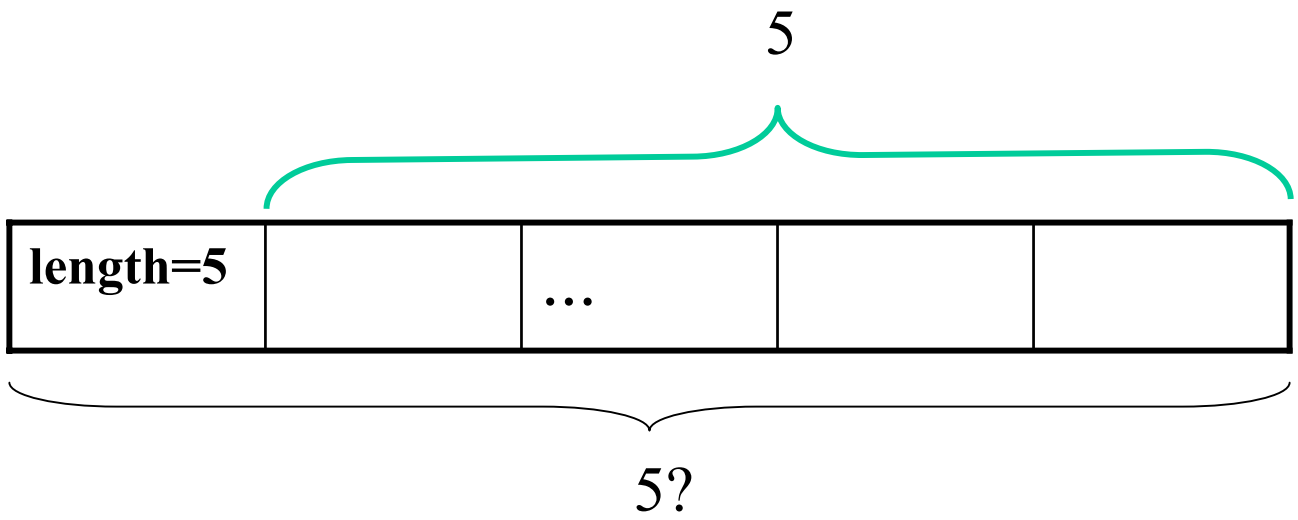
1. IPv4 - connect between exporter/collector from different companies (UDP, TCP, SCTP)
2. Transmission of simple template (few fixed size IEs) + data to different collectors using UDP / TCP / SCTP
3. Transmission of template (with fixed and variable length IEs) + data to different collectors using UDP / TCP / SCTP
4. Temporary network disconnect (i.e. unplug network during export) TCP, SCTP
5. Exporter kill + restart during data transmission (simulates software crash + restart)
6. Collector kill + restart during data transmission (simulates software crash + restart)
7. Transmission of template and data for NetFlow v9 via SCTP
8. Export of non-matching templates and data (wrong number of IEs, single/multiple elements in one template)
9. Big number of records for one template
10. Big templates with a large number of elements (memory stress test)
11. Stress test with multiple exporters active in parallel sending to one collector
12. Export from on exporter to multiple collectors in parallel
13. Multiple use of one field identifier inside one template (successive or with other IEs in between)
14. Incorrect set ID's (2 and 3 are valid)

Tests (2/2)

15. Using any IE's as scope
16. Using multiple scopes
17. Sets with padding, without padding, and with illegal padding
18. paddingOneOctet (IE #210) (a) correctly used, (b) badly extended (length > 1)
19. Enterprise-specific IEs
20. Reduced size encoding of IEs.
21. Template withdrawal message
22. MP stats option template
23. MP reliability option template
24. EP reliability option template
25. Flow keys option template
26. Re-using the same template ID inside the template expiry time (without withdrawing the template) for the same or for different data.
27. Re-using the same template ID after the template expiry time without withdrawing the template.

Unclear or ambiguous definitions

- **Variable length IEs**
  - One field specifies the length
  - Is the length field included in the length?
  - There‘s no example in the protocol draft
  - The protocol draft will be clarified



Unclear or ambiguous definitions

- **Compression of data types**
  - It is not clear which data can use reduced size encoding and which ones cannot
  - Can ***dateTimeSeconds*** and ***dateTimeMilliseconds*** be reduced?
  - All integer-based types can be reduced unless explicitly said it is not
  - Nanoseconds and microseconds cannot
  - As a consequence, don't limit the reduced size to 4, 2, 1 bytes"

    → can use 7, 6, 5, 4, 3, 2, 1 bytes
  - e.g. You need at least 41 bits for the milliseconds today
  - The protocol draft should be updated

Unclear or ambiguous definitions

- At which interval do you need to **resend** templates with **UDP**?
  - The draft says it should be at fixed interval and configurable
  - We need to specify a guideline on the minimum range of the resend time
    - Lower and upper bound
    - Granularity
  - Another idea is to send a new option indicating the template lifetime
  - The draft doesn't exclude to resend depending on the number of packets sent.
  - Item for the Implementation Guidelines Draft

Unclear or ambiguous definitions

- **Template ID data type**
- IPFIX-INFO
  - Abstract data type = unsigned32
- IPFIX-PROTOCOL
  - Unsigned16
- Proposal: change IPFIX-INFO to unsigned16

Common implementation mistakes

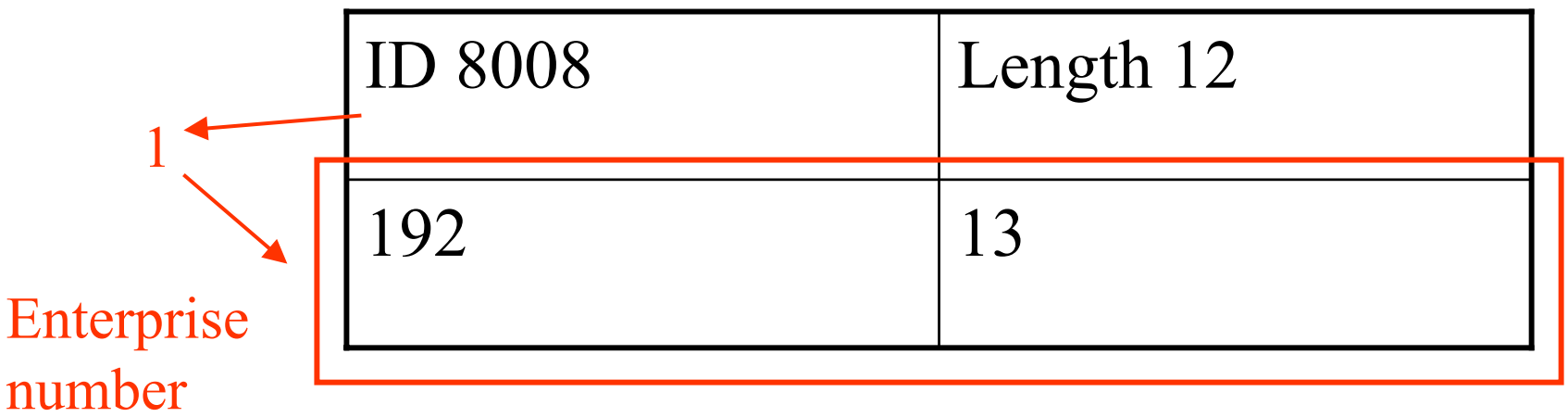
- **Netflow v9 to IPFIX**

- **Option templates**: When you send a option template you have a scope. In IPFIX there must be some scope (it is not allowed to be zero length)
- (that wasn‘t the case in v9)
- No scope is illegal traffic

- **Length**: In v9 you put the number of records, in IPFIX the length

Common implementation mistakes

- **ID numbers**
- When the ID is such that the first bit is 1 the collector interpretes it as ENTERPRISE N.
- Even if it wasn‘t meant to
- No way to tell it‘s wrong



Common implementation mistakes

- **Padding of the data set**

- Padding has to be *shorter* than the length of the record
- If it‘s equal or longer, it is interpreted as another record

Open points

- **PaddingOneOctet**

- If the record is very small, less than 4 octets (e.g. TOS is 1 octet) and you want to export it with padding then
  - 3 times PaddingOneOctet in your template
  - We suggest to have PaddingOctets and the length is the number of octets

  - Change the name and remove the restriction in the draft

Open points

- **Template or Option template withdraw messages**
- Can set contain template withdraws and new definitions?
  - No?
  - Protocol draft?

- Template or option template withdraw for all templates should look like in figure U/V and contain no other templates?
  - Yes?
  - Guidelines?

- Should the collecting process interpret a withdraw message for a template it has not received as an error and close the association?
  - Yes (for TCP and SCTP)?
  - Protocol?

Open points

- **UDP template refresh**
- Consider a template refresh as an error if the received and stored templates differ?
  - Yes?
  - The collector should issue a warning
  - Protocol?

Open points

- **Length**
- What do we do if we receive a template of 0 size?
  - Close the connection
- What about fields of size 0?
  - Same
- Variable size fields
  - In this case it should be ok
- Guidelines or protocol draft?

- **Reserved / unknown set IDs**
- Ignore sets with reserved / unknown set IDs?
  - Yes?
  - Guidelines?

Open points

- How to handle multiple information elements of the same field type?
  - Semantics need to be defined
  - having multiple similar I.E.s in the scope is possible
  - don't limit the protocol specifications. Maybe needed in the future
  - [implementations guideline for IPFIX draft]
    - Difference between scope and non scope elements
    - The exporter should prevent multiple similar IEs
    - the Collecting process should log a warning and may accept the IEs using a first match semantic

Conclusions

- Ipfix-interop mailing list
- Remote interoperability tests
- Implementation Guidelines

Integer Data Types

- From IPFIX-PROTO: *"If reduced sizing is used, it MUST be applied only to following integer types: unsignedLong, long, unsignedInt, int, unsignedShort, short."*

- unsignedLong → 64?
- unsignedShort → 32?