Order of Information Elements
Difference Between drafts 01 and 00

Hitoshi Irino, NTT
Difference Between Order of Information Element drafts 01 and 00

- Description about Information Element order in each group is deleted
  - This order is equal to description order in IPFIX-INFO
    - for supporting future extensions of information model.
    - for simplifying order rule.

- Applicability section is added
  - Advantage cases of this order rule
    - Multiple exporters send different Templates containing the same required Information Elements to a collector.
    - Templates are collected to create output that has unified data structure
      - Collector records flow (formed in unified data structure)
      - Mediator sends template to collector or upper mediator

- Draft 01 corresponds to changes in definitions of Information Elements in IPFIX-INFO-15
Basic Length Classification Rule (corresponds to IPFIX-INFO-15)

1. Fixed Length IEs
   1. Multiple of 4 octets
   2. Even Number octets
   3. Odd Number octets

2. Reduced Size Encoding applicable IEs
   - dateTimeSeconds, dateTimeMilliseconds, dateTimeMicroseconds, dateTimeNanoseconds, ipv4Address, ipv6Address, ((un)signed64, (un)signed32, float32)
   - macAddress ((un)signed16)
   - (un)signed8, boolean
   - mplsLabelStackSections

3. Variable Length IEs
   - octetArray, string

"The size of this Information Element is 3 octets." is given in the descriptions in [IPFIX-INFO-15]
Application

- Case where Information Element Order works effectively
  - Exporter sends different templates containing same IEs,
  - Collecting process creates output that has unified data structure.
  - e.g., IPFIX Mediator (Concentrator)

If these templates are formed using same IE order, the Mediator can simplify conversion of data structure. In particular, when these templates contain only required IEs (also these sizes are same as other templates containing IEs), the Mediator does not need to convert the data structure when these are formed using same IE order.
Compare speed of processing of 2 types of Templates

Comparison between best case and worst case (on software simulation)

Input: Data records using
- Templates that contain IEs corresponding to NFv5 fields
  - Case 1: Ordered Template (Best case)
  - Case 2: Unordered Template (Worst case)

Output: Data records using
- Ordered Template

Conversion of Data Structure

Ordered Template
- The number of copy operations decreases.

Unordered Template

Speed of processing Ordered Template is 14%* faster than that of unordered template.

Efficiency increases when any order rule is defined.

*Environment: Xeon 3.0.6GHz, Memory 2GB