Advantages of using an IPFIX File Format for SIPCLF

draft-niccolini-sipclf-ipfix-00

Saverio Niccolini, Benoit Claise, Brian Trammel, Hadriel Kaplan

Why this draft?

- You need to understand something to be able to like it
  - Who knows IPFIX thinks it is perfect for SIPCLF
  - Who does not know IPFIX can not judge if they are right
- Then… try to educate people about IPFIX
  - Allow folks to have a rational conversation about IPFIX in SIPCLF WG
  - Discuss the main advantages related to the usage of IPFIX file format for SIPCLF
  - Provide an example of how a IPFIX file for SIPCLF would look like

What is IPFIX?

- IPFIX = IP Flow Information eXchange, RFC5101
- Standardized version of NetFlow version 9, RFC3954
- Optimized for the export of repetitive information
  - Template based
    - ➢Contains the information element type and length
  - Data Records
    - ➢Contains the information element value

IPFIX file format for SIPCLF: why? (I)

- Why using IPFIX for SIPCLF makes sense
  - IPFIX already contains a well established Information Model initially populated from [RFC5102] and PSAMP [RFC5476]
    - IANA IPFIX registry with 300 Information Elements
    - IP address, date, etc… already exist
  - IPFIX has a self-describing syntax model
    - that allows the definition of a common set of "standard" fields
    - Using the template
  - IPFIX format has native support for extensibility on top of the “standard” fields
    - Enterprise specific information element
  - Number of applicable tools already parsing IPFIX today
    - ability to reuse these tools for SIPCLF scopes

IPFIX file format for SIPCLF: why? (II)

- Why using IPFIX for SIPCLF makes sense
  - The definition of a protocol mechanism to export the log record to collectors, then filtering controls/config., etc.,
    - IPFIX has it all…
  - IPFIX supports both binary and ascii record field values
    - a binary-capable encoding is necessary to encode the entire SIP message (SIP can contain binary bodies, e.g., ISUP, QSIG)
  - IPFIX records support length encoding
    - enabling a parser to skip past record fields or entire records without parsing their contents
  - IPFIX File Format, RFC5655
    - Store the template and flow records into a file format, for exchange between collectors

Even more reasons (looking at the future)

- The charter and problem statement do not address these points now (but it is worth opening the eyes instead of keeping them close…)
  - SIPCLF correlation with the media related information WILL happen
    - A consistent data model will be required!
    - Otherwise costly proxies!
  - SIPCLF information will have to be transferred (pushed or pulled) in order to do some correlation
    - Choice of IPFIX File transfer or IPFIX export
    - Charter: “Furthermore, these log records can also be used to train anomaly detection systems and feed events into a security event management system.“ => currently done with NetFlow v9/IPFIX

Example

- A request record is described by the following template:

Name	Num	Len	Present?
observationTimeSeconds	322	4	always
sourceIPv4Address	8	4	v4 only
sourceIPv6Address	27	16	v6 only
sipMethod	BBB	1	always
sipAuthUsername	AAA	variable	if authenticated
sipRequestURI	CCC	variable	always
sipFromURI	DDD	variable	always
sipToURI	EEE	variable	always
sipCallId	FFF	variable	always
sipServerTransaction	HHH	variable	always
sipClientTransaction	JJJ	variable	always

- Note: This draft discusses possible information elements for the purpose of providing an example ONLY

Example

- And a response record by the following template:

Name	Num	Len
observationTimeSeconds	322	4
sipMethod	BBB	1
sipResponseStatus	GGG	1
sipServerTransaction	HHH	variable
sipClientTransaction	JJJ	variable
sipToURI	EEE	variable

Let's summarize

- Quote from Dave Harrington on the mailing list
  - "IPFIX already provides a protocol and a data modeling language
  - In addition, [RFC5655] specifies a file format for storing data that has been received in the ipfix file format.
  - The IPFIX File format is designed to facilitate interoperability and reusability among a wide variety of flow storage, processing, and analysis tools."

- In simple words
  - why would we have yet another data modeling language?

Logical Conclusion?

- …choose IPFIX today?
  - Existing information model
  - Exist file format
  - Existing tools
  - Ready for your future requirements

- What do you think?