GENI Instrumentation & Measurement System

Passive Packet Capture System Design and Implementation

Configure

Parameters configured through GIMS manager (via XML/RPC call to measurement device)

- Device Name: the name of the capture host
- Site Location: the location of the capture host
- Experiment ID: an identifier for the experiment
- VLAN: the experiment's VLAN tag
- Metadata Spec: any user defined metadata
- Capture Spec: network device information and libpcap filter expression
- Transform Spec: sub-levels indicate transformations to the captured packets
  - Sample Spec: how to sample packets, with corresponding parameters
  - Aggregation Spec: how to aggregate packets
  - Anon Spec: how to anonymize packets, with cryptographic key if necessary
- Storage Spec: where and how to transfer captured data, including user credentials

Start

Aggregate

SNMP-like aggregation [RFC 1157]

- Packet and byte counts, periodically exported in text format

Flow aggregation

- Standard IPFIX record export [RFC 3917]
- Uses YAF, libfixbuf, and related libraries from CERT (http://tools.netsa.cert.org/yaf/)
- User could use yafscii tool (or similar) to dump records to text for conversion, or write a custom tool

XML/RPC

Metadata

- Modeled after existing repositories (e.g., CRAWDAD and DatCat)
- Metadata Structure
  - Data:
    - At collection time: trace name, format, start and end times, time zone, geographical location, system configuration, storage type, anonymization key, packet drops
    - Post-processing: MD5 hash of trace, trace size, any relevant time synchronization information
  - Tools:
    - Versioning details for ancillary libraries (e.g., YAF, libpcap)
  - Authors/Creators
    - User-specified metadata

Stop

Packet Capture

- Uses standard libpcap as the underlying packet capture mechanism
  - Can take advantage of software-level optimizations (e.g., PF_RING), or specialized hardware (e.g., Endace DAG cards)

Initial Packet Capture Performance

- PF_RING is an open-source kernel driver for Linux to optimize packet capture
- Testbed experiments with offered load of ~800 Mb/s and full-sized Ethernet frames
  - About 695k packets/sec
  - Zero packet loss with up to 100 bytes of packet header/body captured
  - Significant loss with full frame capture (>60%)

Sample

- Reduce load on measurement system by collecting only some packets
  - Two capabilities: 1-in-n sampling and fixed probabilitistic sampling

Anonymize

- Ensure user privacy while preserving research value of data
- Anonymize source/destination IP addresses in a prefix-preserving fashion
- Current performance: on average, 97 microseconds per address
- References
  - J. Fan, J. Xu, M. Ammar, S. Moon. "Prefix-preserving IP address anonymization." Computer Networks, Vol 46, Issue 2, October 2004.
  - R. Pang, M. Allman, V. Paxson, J. Lee. "The devil and packet trace anonymization." ACM SIGCOMM CCR, Vol 36, Issue 1, January 2006.
    - Code is adapted from open source anonymization tool"tcpmkpub"

Storage Capabilities

Purpose: transfer data across network to user-specific storage location

- Amazon S3
  - Ubiquitous cloud storage service
  - Requires existing user account and credentials
- SFTP
  - Standard secure data transfer protocol
  - Requires existing user account and credentials
- local storage
  - Move data to existing location on capture host
  - Used mainly for testing

Future Work

- Deployment for GENI users
- Implement Transport Layer Security between GIMS manager and capture device
- Create additional tools to decode IPFIX records to decouple traces from libfixbuf
- Enhance metadata to include known outliers and potential system problems, and auxiliary system information
- Investigate additional transformations of packet contents
- Broader study of performance characteristics of capture system

T. Yu, J. Raffensperger, A. Das, J. Sommers
Colgate University
C. Thomas, M. Blodgett, P. Barford
University of Wisconsin
M. Crovella
Boston University