System Architecture Doc for Data Artifex

# Overview

Data Artifex is network mid-appliance software.

## Purpose

This document lays out the ideas behind Data Artifex and proposes an architecture for its construction.

## Scope

Develop a prototype to prove the merits of the design. Demonstrate traffic optimization in both single-ended and two-way data flows.

# Capabilities View

## Requirements

|  |  |
| --- | --- |
| R-1 | Optimize network traffic to external networks and internally connected devices |
| R-2 | Handle a variety of network traffic sources such as libpcap, netlink, DPDK, sockets, files, etc. |
| R-3 | Analyze network traffic flowing through the system |
| R-4 | Provide an overlay network when operating in two-way mode |
| R-5 | Act as a proxy |
| R-6 | Facilitate routing and forwarding of network traffic |
| R-7 | Act as a gateway |
|  |  |

## Constraints

|  |  |
| --- | --- |
| Co-1 | Must be capable of optimizing network traffic in single-ended and two-way operating modes |
| Co-2 | Must provide at least 40% optimization |
|  |  |

# Operational View

## 

Figure . OV-1

|  |  |
| --- | --- |
| Data Source | The Data Source produces a stream of un-optimized data |
| Data Artifex Sys. 1 | Data Artifex Sys. 1 analyzes data streams from the Data Source. It re-broadcasts or modifies in-line multiple output data streams optimized for different consumers |
| Int. Data Consumer | The internal data consumer receives a stream of data from the Data Artifex sys., optimized for a tablet device connected to a local wireless network, e.g. Unicast UDP with large packet/frame sizes |
| Optimized Data | Data Artifex Sys. 1 Transforms streams of data to enable higher throughput, higher quality, lower latency, better bandwidth utilization, fewer errors, and lower numbers of packets transmitted. |
| Field Site | A location with low quality networks |
| Ext. Net. | The internet or a larger area network with a mixture of different network types |
| Cell. Net. | A 3G or 4G medium-quality cellular network |
| Remote Data Consumer #1 | A smartphone or tablet connected to a cell. Net., consuming optimized data streams from Data Artifex Sys. #1 |
| Remote Data Center | A data center connected to a high-quality network with one or more data consumers looking for high-quality data streams |
| Data Artifex Sys. 2 | A DA sys. Installed in the data center sharing a tunnel with DA sys. #1, used to receive high-quality, low latency optimized data streams via special tunnel/overlay network |
| Remote Data Consumer #2 | A server or appliance that consumes a high-quality data stream from the data source, optimized by DA sys #1 |

## Concept of Operations

* Modes
  + Gateway (optionally transparent)
  + Proxy (optionally transparent)
* Deployment Options
  + Software library
  + Software application
  + Virtual appliance
  + Physical appliance
* Optimization Methods
  + Single-ended
    - Transcoding
      * Bitrate/Framerate reduction
      * Quality reduction
      * Interval modification
      * Data packing
  + Two-way
    - Compression
      * Headers
      * Content
    - Overlay network

## Environment

* Field Site
* Data Center
* Consumer Device
* Cloud

## Use Cases

|  |  |
| --- | --- |
| UC-1 | Optimize data streams generated by a legacy or non-interoperable data source, such as an IP camera/encoder/sensor |
| UC-2 | Optimize bandwidth utilization by caching data requests such as DNS |
| UC-3 | Optimize data streams consumed by devices on lower quality networks such as cellular or satellite |
|  |  |

## Users

* Consumers
* Field Technicians
* Network/Data Center Technicians
* Software Developers

# System View



Figure . Basic Component Diagram

|  |  |
| --- | --- |
| Network Traffic Source | Source of network traffic for the system |
| Network Traffic Cache | Stores packets. |
| Network Traffic Analyzer | Analyzes and classifies network traffic, storing the information in the Network Traffic Information datastore |
| Network Traffic Information | Contains information about network traffic to be processed by the system |
| Network Traffic Processor | Examines rules provided by the System Manager, matches them to available network traffic, and modifies matching network traffic |
| Network Traffic Sink | Destination for processed network traffic |
| System Manager | Tracks system state; Configures processing rules based on intents sent via the management interface. |
| Event Log | All sub-systems post events to the event log. |
| Processing Rules | Contains a set of match criteria and set of actions to take when the match criteria are met |
| System Interface | Enables configuration and monitoring of the system. |

## Components and Functions

* Traffic sources copy packets into queues for processing. Classifiers pull traffic off of the queues, and add labels to them. Directors, examine labels, and publish processing requests events to processors informing them of a possibly interesting packet. Processors ‘claim’ packets.
* The classification system for network traffic analyzes the individual packets, streams, sources, and destination, and labels individual frames for further processing
* Processing systems have lists of patterns and rules that receive traffic based on subscribing to traffic containing a matching set of labels.

### Network Traffic Source

[component diagram]

* Network Traffic Sources (NTS) copy traffic into queues (see Network Traffic Queues) for analysis and processing by the system.
* Network Traffic Sources obtain traffic from one of the following types of sources:
  + A PCAP source
  + A file
  + A socket
  + A kernel interface such as NF\_QUEUE, or DPDK

### Network Traffic Cache

[component diagram]

* Uses

### Network Traffic Analyzer

[component diagram]

* Network Traffic Analyzers (NTA) grab packets as directed from the NTQs associated with NTSs
* NTAs parse packets, storing information about them in Network Traffic Information (NTI) tables
* Upon parsing a new packet, NTAs update session/flow information in the NTI tables.
* On completing analysis, NTAs mark the packet state as analyzed in the NTQ

### Network Traffic Processor

[component diagram]

* Network Traffic Processors (NTP) modify packets that match Processing Rules assigned to them by the System Manager.

### Network Traffic Sink

* Network Traffic Sinks (NTK) inject or transmit traffic for the System.

### System Manager

* The System Manager (SM) keeps track of the system state.
* The System Manager manages pools of NTAs and NTPs as well as hand-off from/to NTSs and NTKs

## Data Flows

### Network Traffic State

* New
* Analyzed
* Processed
* Completed
* Error
* Unknown

### Network Traffic Flow

[Packet flow diagram]

1. Packet received/captured by NTS
2. Packet copied to a NTQ
3. System Manager hands reference to packet to the next available NTA
4. NTA obtains reference to packet in NTQ
5. NTA analyzes packet and updates NTI tables
6. NTA ‘gives back’ reference to packet to System Manager
7. System Manager broadcasts reference to packet data found in NTI tables to available NTPs.
8. NTPs receive broadcast about packet.
9. NTPs check packet attributes against rules.
10. NTPs with no matching rules ignore packet.
11. NTPs with matching rules request reference from System Manager.
12. System Manager hands reference to first matching NTP.
13. NTP obtains reference to packet.
14. NTP modifies packet.
15. NTP hands packet back to system manager.
16. System manager hands reference to packet to appropriate NTK.
17. NTK receives reference to packet.
18. NTK transmits/injects packet.
19. NTK notifies System Manager of successful injection/transmission.
20. System Manager removes packet from queue.

## Interfaces

### System Interface

* External systems interact with the SM via the System Interface (SI) using messages

# Data View

### Tables

### Network Traffic Queue

* Network Traffic Sources copy network traffic into Network Traffic Queues (NTQ)
* Network Traffic Queues are single-producer, single-consumer lock-free queue data structures
* Network Traffic Queues contain Packet Information entries

### Network Traffic Information

* NTI table contents will be persisted in a database
* NTI table contents will be stored in a Key/Value store at runtime.

#### Packet Label Table

* When NTAs determine that a packet contains a header/frame from a particular protocol, they add a label entry for that packet to the Packet Label Table (PLT).
* Packet Label Table Entries contain a reference to the packet and the label associated with it.

### Processing Rule Table

* Processing Rules consist of a set of match criteria and a set of actions to perform on matching traffic
* Processing Rules will be persisted to both a database and a file.
* Processing Rules will be accessed from a database at runtime

### Event Log

* All system components log all events in syslog format to the event log
* Events will be atomically stored in a key/value store
* Events will be persisted in a database

## Data Structures

### Packet Information

* Stores a copy of the packet in a static array
* Contains an ID for the packet that is a hash of the packet plus a salt based on the system time the packet was received/captured.
* Contains a timestamp indicating the time the packet was captured/received

# Standards View

TBD

# Security View

TBD

# Glossary

TBD