Packet Payload Processing (PPP)

|  |  |
| --- | --- |
| Version | 0.1 |
| Author | Tom Sumardi |
| TODO |  |

Table of Contents

Introduction 1

Abbreviations 1

Conventions/Requirements 1

Summary 1

Components 2

Architecture 2

Component descriptions: 2

# Introduction

This documentation intended to serve the purpose of giving detail level architecture description.

# Abbreviations

TBD

# Conventions/Requirements

* Performance bound
* Bidirectional communication interface from MS to PPP will be through AMQP
* Unidirectional communication interface from PPP to PHS will be through ZeroMQ
* PHS is a service that plugs into PPP component
* Packet outgoing interface will go through different interface than packet incoming interface
* Default PPP plugin will be HTTP processing

# Summary

PPP operation only involves user space and acts as plugin to the IPPS with “HTTP processing” as the default plugin.

PPP can be null operation if no payload parsing to be done on the packet and by default set to process HTTP packet. It parses, dissects and applies regex on the packet payload (L4 payload) at user space level. It performs regex matching based on configuration given and determines if the packet needs to be passed to the PHS or not.

Example of PPP acting as packet injection plugin operation:

* above L4 decoding
* above L4 packet filtering and routing
* session correlation

# Components

null

IPPS Thread

HTTP

plugin

ZeroMQ

Logging

Other…

PHS

Figure 1 (component stack)

* Background Configuration/registration thread
* Plugin
* ZeroMQ communication
* Logging

# Architecture

## Component descriptions:

??