May 31, 2018

TECHNIQUES FOR VISUALIZATION OF DEVICE INTERNAL PACKET PROCESSING AND NETWORK EVENTS

Jay Johnston

Magnus Mortensen

David White

Michael Robertson

TECHNIQUES FOR VISUALIZATION OF DEVICE INTERNAL PACKET PROCESSING AND NETWORK EVENTS

AUTHORS:
Jay Johnston
Magnus Mortensen
David White
Michael Robertson

ABSTRACT

The techniques presented herein provide an easy to understand animation representing how traffic is entering, being processed, and exiting a device. In certain examples, events are represented by flying icons that enter an interface of a device. As they are processed by different features their "tails" get longer and gain segments that represent what feature acted on them. Any packet drops may be shown, for example, by the icons falling down from the device onto a "floor" that represents the drop reason and piling up to show the scale of the drops.

DETAILED DESCRIPTION

Networking devices (and all computing devices) become vastly more complex each generation and it is increasingly difficult for humans to effectively understand their operation. For complicated products such as networking equipment, it can require years of experience, certifications, and book study to understand how to interpret the running state of a networking device.

Conventional arrangements make it very difficult for humans to observe a system with many interfaces and easily answer questions such as:

- How much traffic is the device processing, and what ratio of packets are dropped vs allowed through the device?
- How much traffic is flowing between the different interfaces of the device?
- What specific operations within the packet-processing system are dropping the packets

Fundamentally, humans have a difficult time parsing raw numbers looking for patterns and anomalies. However, humans are fundamentally adept at visually observing patterns and motion, especially motion patterns that map to familiar concepts that they observe often in the real world, such as moving car traffic patterns, flocks of birds in the sky, flowing water, etc. Therefore, what is needed is a method to map complex statistical information into an easy to understand visual representation that humans can quickly understand and draw useful conclusions therefrom.

Presented herein are novel methods for visually representing the events occurring within a networking device, as well as the network traffic passing through the device. The methods are useful on any computing system and naturally extend outside the scope of networking.

Animated 2-Dimensional Visualization of Network Device Operating State

First, the techniques presented herein illustrate the operation of a networking device in two dimensions. Shown in FIG. 1, below, is the use of the techniques presented herein for a firewall device having complicated packet processing pipelines in which there are hundreds of different reasons why a packet might be dropped, such as the Access list (Access-policy), NAT Failures, or Threat Defense reasons.

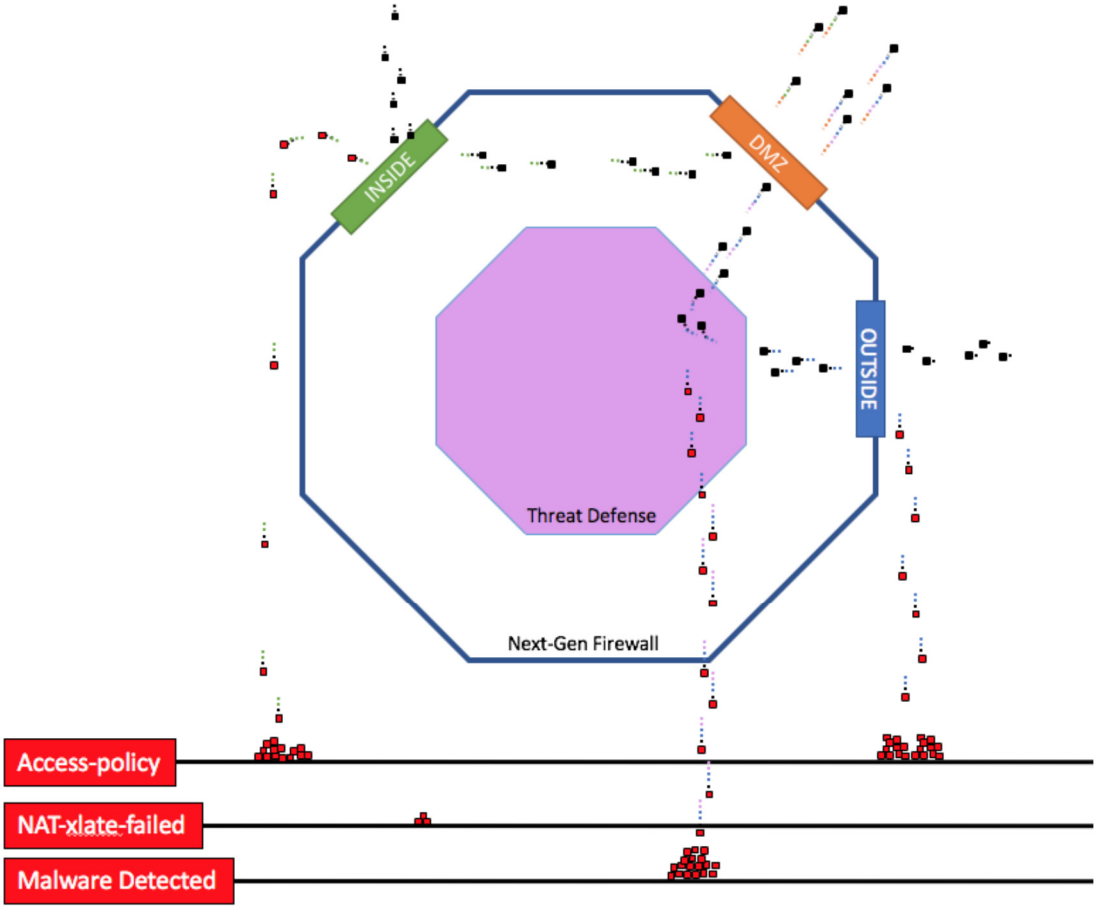

FIG. 1

Operation of the System

- New events such as packets or new connections are represented by simple flying squares. The squares are animated and "fly" in by entering an interface (such as "inside") and pass through the various packet processing stages of the device, and then exit through another interface to be passed on to the rest of the network.
- The tail of the events/units have a color that indicates the feature path that the event took through the system. This provides a simple visualization that could be used to identify things such as a problem area in the system or an opportunity for improving the efficiency of packet drops based on which feature set is handling them.
- The tails get longer as more processing steps are done on the packet/connection. As the event passes through subsystems it picks up an addition on the tail, making

the tail grow longer. This allows a user to easily see the amount of processing that was done on the packet/connection.

- Packet processing functions are represented as shapes inside the device that the events fly through. For example, as a packet is processed by "Threat Defense" function it enters that octagon and if it is not dropped, it exits the octagon to move to the next processing step, and an entry is added to its tail.
- Drop reasons are exposed by showing packets falling down on different "drop planes". As a packet is dropped in a processing layer it falls down to the associated drop layer below. The lower it falls, the more processing must have been done, so it is optimal for administrators to move packets that must be dropped to a higher layer, by modifying their security policy.

Rendering is easily accomplished using HTML5 and JavaScript Canvas, or other browser animation technologies. This system can be easily plugged in to a web-based management interface. Shown below is the system in action with three scenarios on a firewall device. Note that in this visualization the packets/events do not have "tails" but rather use concentric rings that denote the different phases of the packet processing pipeline. As the packet moves through processing functions, the packet gains rings indicating those functions.

Normal Operation

Shown in FIG. 2, below, is a representation of a firewall device under normal operation. Packets are entering the inside interface and most leave the outside interface, while some are routed out the DMZ interface.

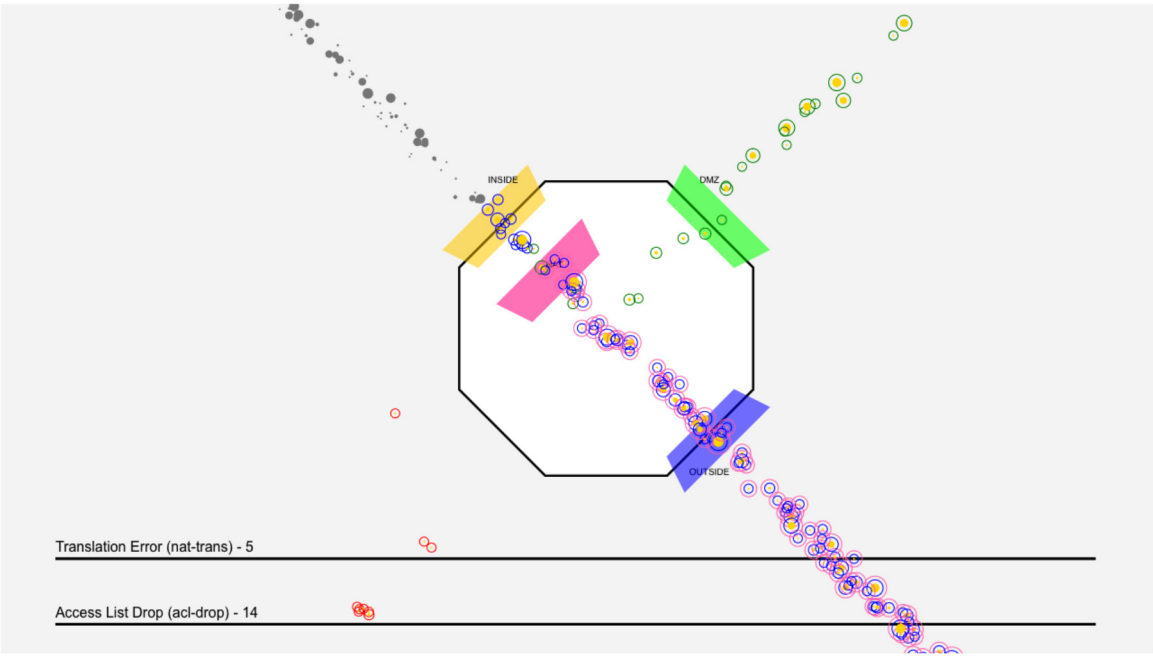

FIG. 2

Attack in Progress

Shown in FIG. 3, below, is a representation of a firewall device under attack. An inside user is hacked and sending malicious traffic which is being blocked by the inside interface access-list.

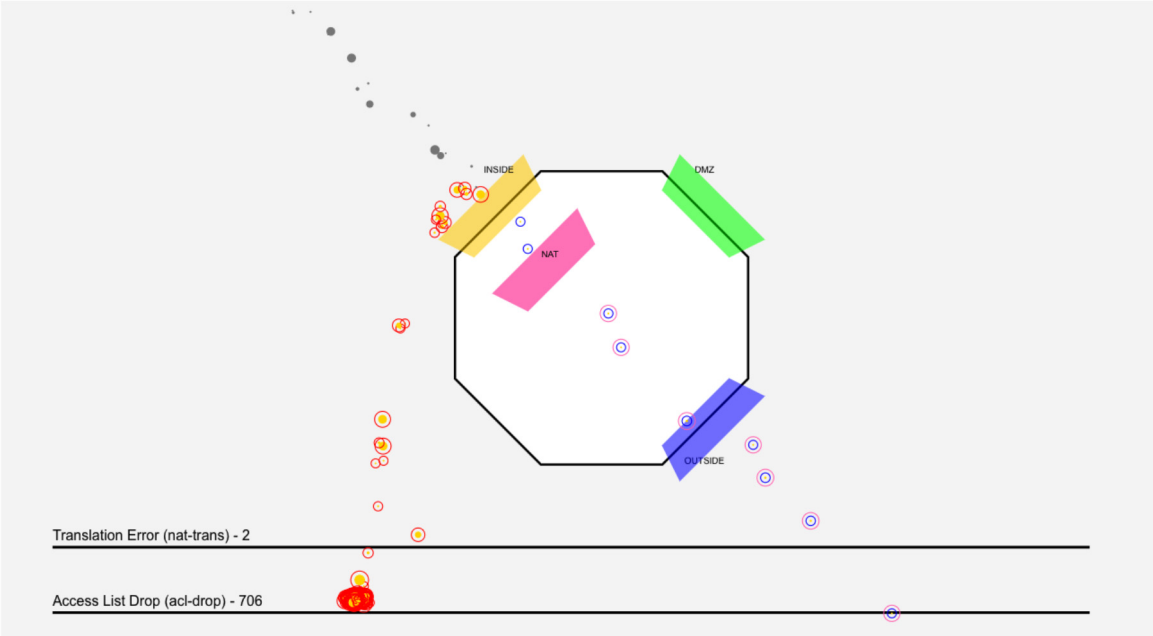

FIG. 3

NAT Misconfiguration Causing Drops

Shown in FIG. 4, below, is a representation of a firewall device that has been improperly configured in the "NAT" policy sub-system. The NAT system is dropping a large percentage of traffic, and the administrator can easily identify that this is the cause of the problem.

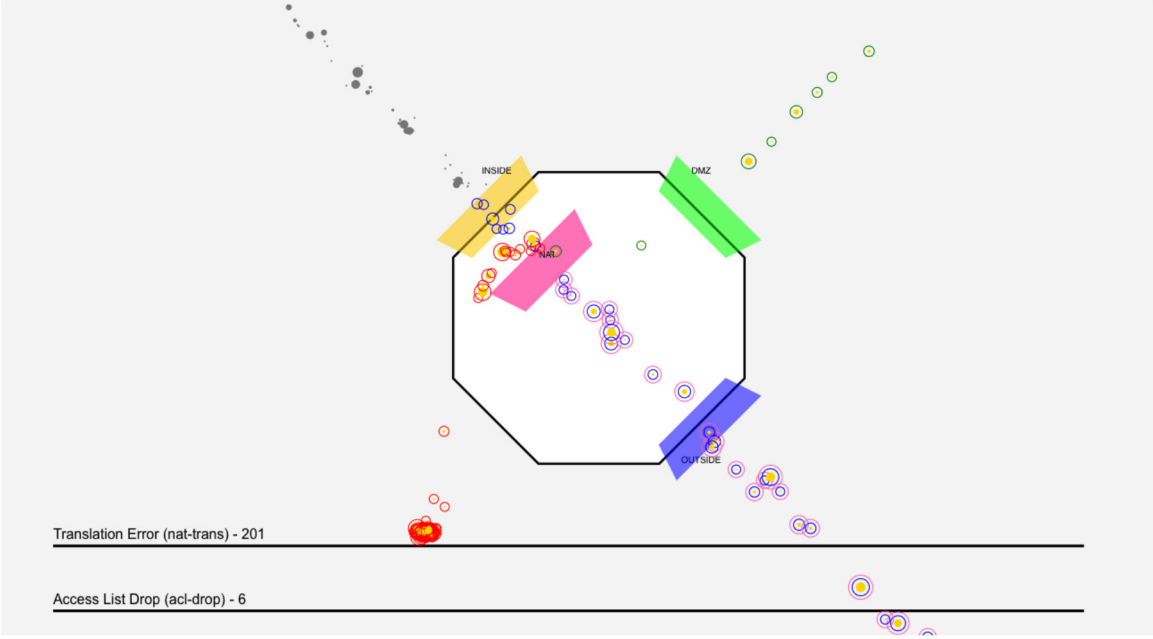

FIG. 4

Shown in FIG. 5, below, is a portion of FIG. 1 again, with annotations:

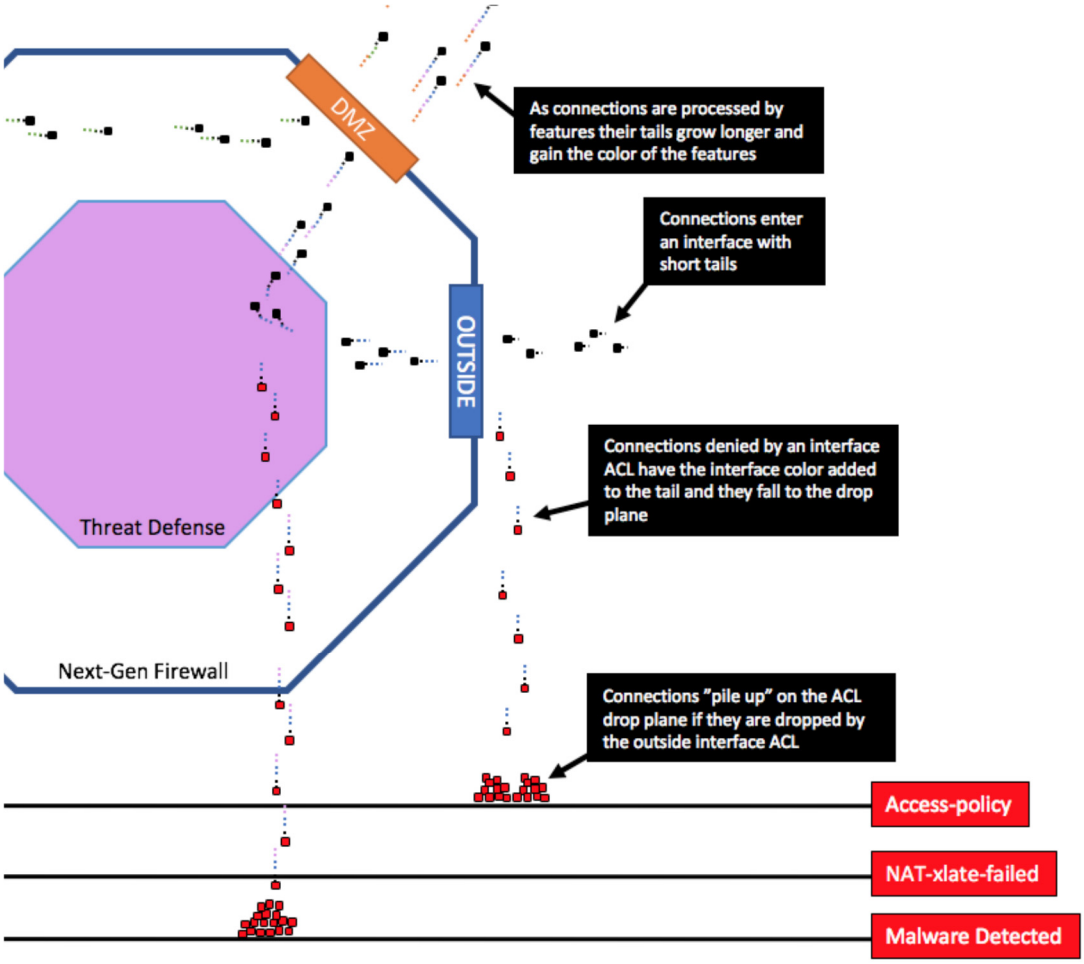

FIG. 5

In certain examples, an explanation of how the tail of the event (packet) gets appended, as the event is acted upon by different components of the system, can be visually represented. One such example is shown in FIG. 6, below.

FIG. 6

Inbound Attack

FIG. 7, below, is a diagram showing an inbound attack against the device. It is very easy for a human to understand that a large amount of the traffic hitting the outside interface is being dropped, and this is an anomalous situation.

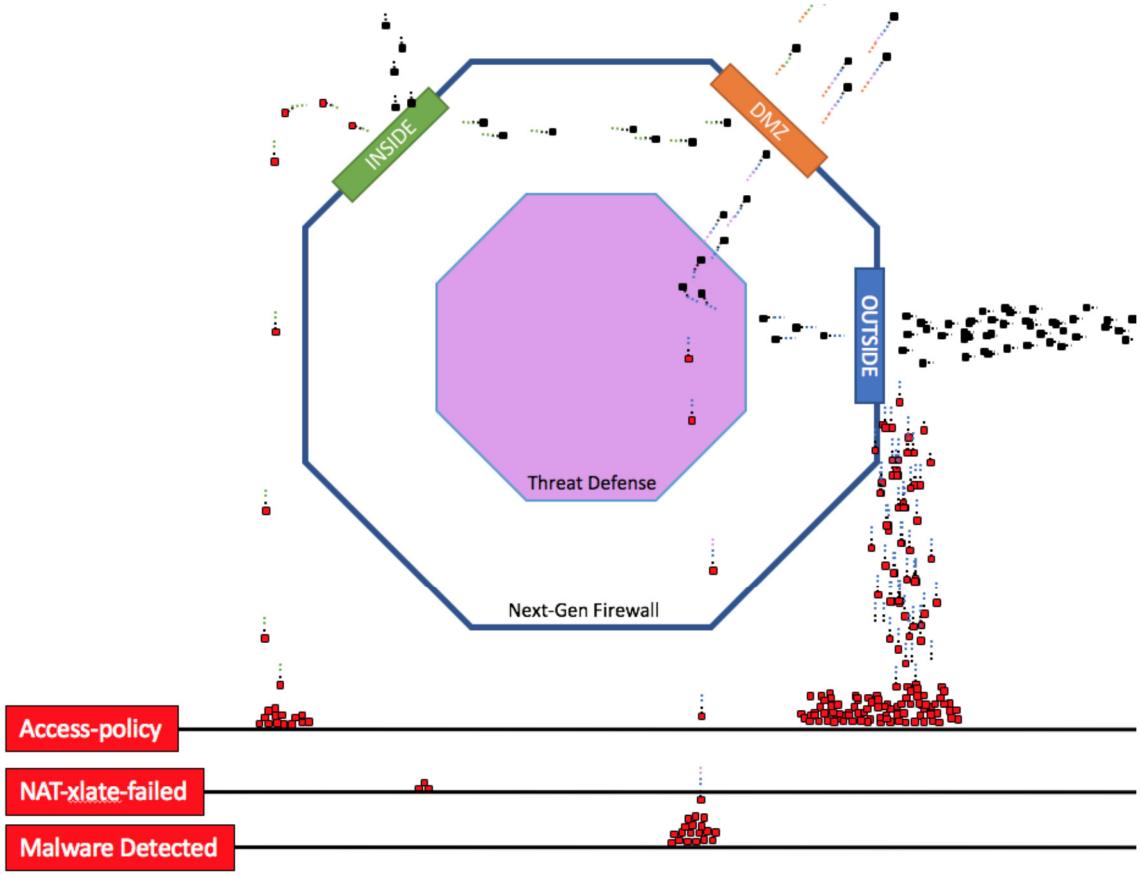

FIG. 7

Extensions to visualize Egress Interface Policing, Marking, Queuing and Shaping

In another embodiment, the techniques presented herein can be extended to cover egress interface Policing, Marking, Queuing/Scheduling and Shaping. As can be seen in FIG. 8, below, being able to visualize proportional amounts of traffic which are getting dropped by the policer is important. Likewise, an indication of how full the queues are (in a visual way), and how much traffic overflowed the queue causing drops, is also important.

With this extension, one can set thresholds for visualizing packet rates, meaning a single image of a packet may represent multiple (e.g., 500 packets). So that the visual representation will not appear too busy/cluttered the user (and tool) can auto-scale based on rates through the interface, while still showing proportionally the amount of traffic going through the various stages and being dropped.

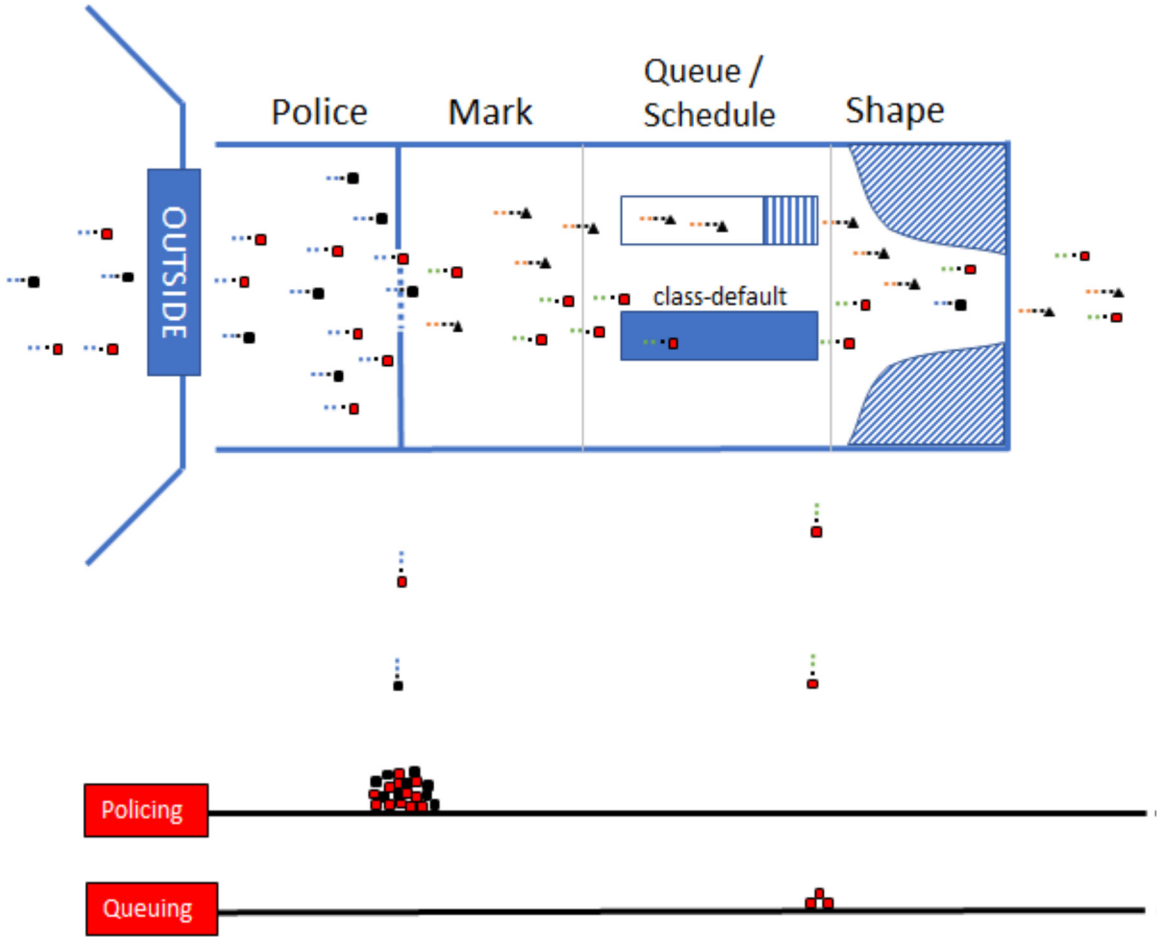

FIG. 8

Animated 3-D visualization of network device operating state

The system works well in three dimensions as well. Shown in FIG. 9, below, is a half-dome floating in space, with the exterior surfaces being the external interfaces of the device. Just like in the two-dimension version, the packets enter and flow through the device and the packets drop down and fall on the floor layers below.

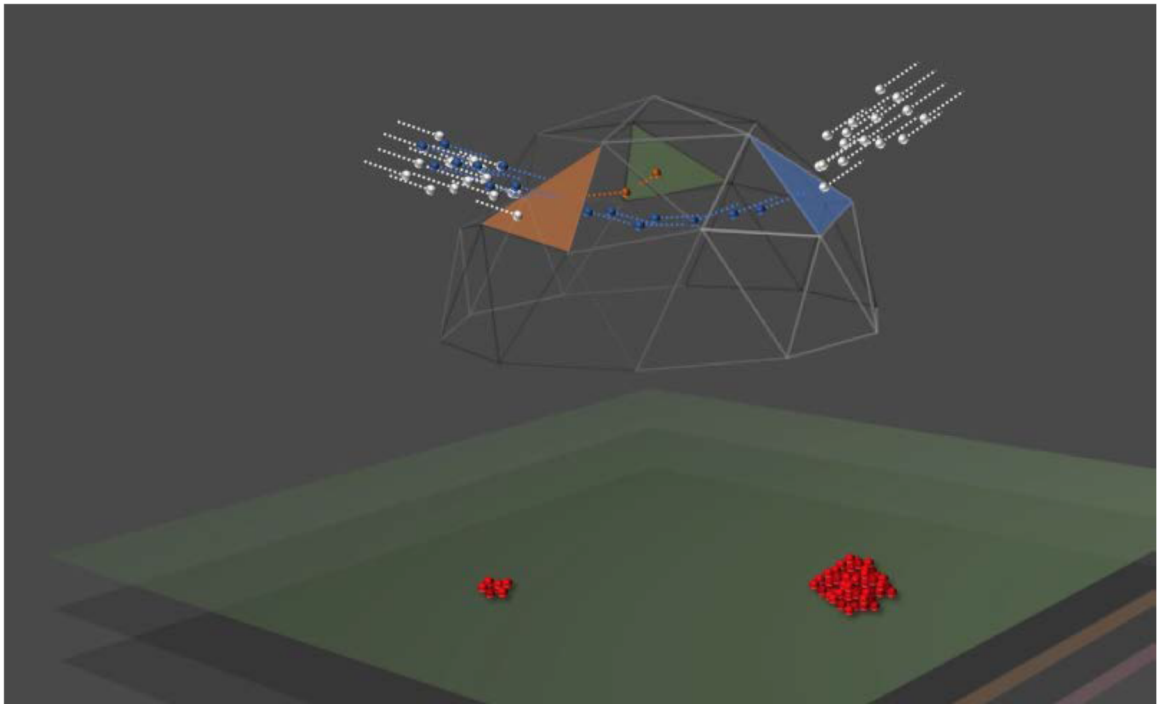

FIG. 9

The three-dimensional view also exposes the feature packet processing order. For example, either a flow, or a pair of interfaces, and the processing layers maybe superimposed inside the dome positioned between the two interfaces, and it is possible to see packets flowing through them or being dropped.

Further Embodiments

- Instead of a color, the tail units could be a symbol (square for NAT, triangle for ACL).
- A number of different processing steps can be generalized to just a few, allowing a user to "zoom out" and see a high-level view of the components, or zoom in to show the specific details that comprise them.
- A user can 'freeze' (or pause) a view to then zoom in on a given flow. Hovering over parts of a tail reveal what processing step occurred.