

SYSTEM AND METHOD FOR MANAGING NETWORK TRAFFIC

TECHNICAL FIELD

The present invention relates to computer networks, and in particular but not exclusively, to systems and methods for managing network traffic for data delivery across
5 multiple networks wirelessly.

BACKGROUND TO THE INVENTION

Wireless networks are deployed for devices to stay connected and communications over geologically wide areas. Due to the proliferation of mobile devices such as laptops, mobile phones and tablets, as well as video and audio streaming services and other cloud
10 services, there is an increasing demand for secure and high-performance communication over various types of wireless networks, which introduces high latency, especially for delivering internet-based services over large areas.

The implementation of multipath delivery for data flows ensures reliable data delivery. Existing systems have adopted the use of multipath data delivery protocols, for instance,
15 the Multipath Transport Control Protocol (MPTCP) and are capable of dividing the traffic over multiple paths. These multipath systems take advantage of alternative paths that exist between source and destination to provide end-to-end reliability and robustness. However, common limitations of such transmission system, such as congestion, interference and network coverage in the wireless network often lead to poor network performance.
20 Furthermore, potential security issues could arise from cross path traffic fragmentation, resulting in connections that are susceptible to attacks along the way towards the target system.

Therefore, the present invention seeks to provide a system and method to overcome at least in part some of the aforementioned disadvantages. In particular, a system and a
25 method for managing network traffic for secure data delivery across a network wirelessly so as to optimize or improve the performance and reliability of network communication by the configurable multi-number of packet scheduling algorithms to distribute the network traffic across available network paths.

SUMMARY OF THE INVENTION

The system for managing network traffic over multiple network paths can integrate multiple network interface devices for delivering network traffic for improving the network interface and maximizes available bandwidth.

5 The system provides seamless processing of ongoing user requests to minimize or eliminate network disruption. This enables smooth and continuous network communication when a primary network experiences slowdown or failure. This is advantageous for deployment in critical systems, particularly for use in remote locations with limited human intervention.

10 The system provides secure transmission of network traffic across multiple network paths by retaining the established connection information to enable secure transmission of authorized data packets when switching network paths.

The method for managing network traffic over multiple network paths between a source node and a target node. The method comprising:

15 establishing a connection record for managing multipath transport layer connections between the source node and the target node, the connection record responsive to establishment of a plurality of network connections for transmitting data between the source node and the target node;

intercepting data based on the connection record;

identifying the plurality of network connections associated with the intercepted data;

20 receiving external information relating to the plurality of network connections;

scheduling the intercepted data based on the connection record, wherein the scheduling comprises selecting network connection or network connections out of the plurality of network connections to schedule the data, wherein the selecting is based on the external information;

25 wherein scheduling the data through the selected network connection or network connections enable managing the multipath transport connections between the source node and the target node.

Preferably, scheduling redirects the intercepted data to at least a second network connection based on the received external information.

Preferably, scheduling comprises switching network paths based on the received external information.

Preferably, scheduling is performed when maintaining the connection record.

Preferably, scheduling comprises accelerating the intercepted data via the selected
5 network connection or network connections.

Preferably, scheduling comprises delivering accelerated traffic over multiple networks and multiple paths. The scheduling of the packets may comprise breaking received byte stream into segments to be transmitted on one of the available sub-flows. The scheduling may comprise arranging the data streams in multiple queues for
10 transmission via the selected tunnel paths.

Preferably, selecting of network connection or network connections comprises one or more of: (i) response time from various network paths; (ii) network signals; (iii) congestion level; (iv) estimated available bandwidth; (v) priority setting for packet delivery; (vi) transport layer session information.

15 Preferably, receiving external information comprises determining a measure relating to the plurality of network connections based on estimated bandwidth and/or average round trip time measured from the plurality of network connections.

Preferably, receiving external information comprises receiving information that one of the plurality of network connections is terminated.

20 Preferably, multipath transport layer connections identifiers are generated for encoding intercepted data for transmission.

Preferably, a plurality of additional connection records is established in relation to a plurality of additional source nodes.

25 There is an apparatus for managing network traffic over multiple network paths using multipath data connections. The apparatus comprises:

a data processor;

a connection manager configured to:

establish a connection record for managing multipath transport layer connections between a source node and a target node, the connection record responsive to establishment of a plurality of network connections for transmitting data between the source node and the target node;

5 intercept data based on the connection record;

identify the plurality of network connections associated with the intercepted data;

a path manager configured to:

receive external information relating to the plurality of network connections;

10 schedule the intercepted data based on the connection record, wherein the scheduling comprises selecting network connection or network connections out of the plurality of network connections to schedule the data, wherein the selecting is based on the external information;

15 wherein scheduling the data through the selected network connection or network connections enable managing the multipath transport connections between the source node and the target node.

Preferably, the path manager performs the scheduling of the intercepted data by redirecting to at least a second network connection based on the received external information.

20 Preferably, the path manager performs the scheduling of the intercepted data by switching network paths based on the received external information.

Preferably, the path manager performs the scheduling when maintaining the connection record.

Preferably, the path manager performs the scheduling by accelerating the intercepted data via the selected network connection or network connections.

25 Preferably, the path manager selects at least one network connection comprising one or more of: (i) response time from various network paths; (ii) network signals; (iii) congestion level; (iv) estimated available bandwidth; (v) priority setting for packet delivery; (vi) transport layer session information.

Preferably, the path manager receives the external information by determining a measure relating to the plurality of network connections based on estimated bandwidth and/or average round trip time measured from the plurality of network connections.

5 Preferably, the path manager receives the external information comprising receiving information that one of the plurality of network connections is terminated.

Preferably, the connection manager generates multipath transport layer connections identifiers for encoding intercepted data for transmission. The connection manager may provide a public key exchange between the source node and the target node in the establishment of the connection record for secure data transmission.

10 Preferably, the connection manager establishes a plurality of additional connection records in relation to a plurality of additional source nodes.

A computer program product, comprising a plurality of data processor executable instructions that when executed by a data processor in a system causes the system to perform the method as detailed in the first aspect above.

15 There is a system for managing network traffic over multiple networks using multipath data connections. The system comprises an intermediary device, a user unit, and a service unit.

BRIEF DESCRIPTION OF DRAWINGS

20 Aspects of the present invention will now be described by way of illustrative example only, with reference to the accompanying drawings, of which:

Figure 1 is a schematic diagram of a system for managing network traffic over multiple network paths in accordance with an embodiment of the present invention.

Figure 2 is a schematic diagram of the system of Figure 1 deploying network switching in accordance with an embodiment of the present invention.

25 Figure 3 is a schematic diagram of the software architecture of the system of Figure 1 in accordance with an embodiment of the present invention.

Figure 4 is a schematic diagram of the software architecture of the system of Figure 2 illustrating an aspect of the intermediary device in accordance with an embodiment of the present invention.

Figure 5 is a schematic diagram of the system providing scheduling decision in accordance with an embodiment of the present invention.

Figure 6 is a schematic diagram of the system of Figure 3 illustrating an implementation of the failover in accordance with an embodiment of the present invention.

5 Figure 7 is a schematic diagram of the software architecture of a system in Figure 1 or 2 deploying network switching in multipath transmission in accordance with an embodiment of the present invention.

Figure 8 is a schematic diagram of an implementation of UDP stitching in accordance with an embodiment of the present invention.

10 **DETAILED DESCRIPTION OF EMBODIMENTS**

Particular embodiments of the present invention will now be described with reference to the accompanying drawings. The terminology user herein is for the purpose of describing particular embodiments only and is not intended to limit the scope of the present invention. Additionally, unless defined otherwise, all technical and scientific terms used herein have
15 the same meanings as commonly understood by one of ordinary skill in the art to which this invention belongs.

The use of the term "network" includes point to point network, broadcast network, local area network, telecommunications network, data communication network, computer network and any other wireless network.

20 In accordance with an embodiment of the invention, there is a system for managing network traffic over multiple network paths for secure network transmission. The system comprises one or more source devices in communication with one or more target devices via one or more networks. Communication between devices may include combinations of wired/fixed network and a wireless network. In some embodiments, the network may be
25 public and/or private.

The system comprises a user device (or source device), a service device (target device) and an intermediary device. The intermediary device integrates multiple network interface devices for delivering network traffic. The intermediary device establishes connection information responsive to a multipath transport layer session for transmitting encrypted
30 packets between the source device and the target device.

The intermediary device may receive data from a source device and provide that data to a target device. The source device is any device that provides data that is received by the intermediary device. The service device is any digital device that receives the data that was provided by the source device via the intermediary device. The user device communicates
5 directly with one or more service devices. In another embodiment, the service device receives requests from the user device.

Figure 1 depicts an embodiment of a network environment deploying multiple interface devices. The user device 102 is in communication with multiple service devices (via STA-BU in Figure 1) via network communication means, and the intermediary device
10 104. The user device 102 connects to the intermediary device 104 via the network communication means. The user device 102 connects to the service device via the network communication means. The network communication means may include cellular or satellite network.

The user device 102 via the communication network and intermediary device 104 may
15 establish at least one connection to request content from the service device over the communication network. The intermediary 104 device may forward a request from the user device 102 to the service device. The intermediary device 104 (or a gateway or proxy) is shown between satellite and cellular networks network communication means in Figure 1. It would be appreciated that the intermediary device 104 may be deployed on one or more
20 networks on any point between a user device 102 and a service device. In some embodiments, one or more intermediary devices 104 may be located at any point in the network or network communications path between a user device 102 and a service device. A first intermediary 104 device may be deployed on a first network and a second intermediary device 104 may be deployed on a second network.

25 The intermediary device 104 accelerates network traffic by multiplexing and encrypting user traffic payload, and decrypting and multiplexing at the receiver side. The intermediary device 104 on a network may communicate data packets between established streams.

The intermediary 104 accelerates transport layer traffic between a user 102 and a
30 service device. The intermediary 104 may deliver the accelerated network traffic via a path manager 105. The intermediary 104 may act as a proxy to provide access to one or more service devices. The intermediary 104 includes a device for accelerating network traffic.

The intermediary device 104 integrates multiple network interface devices for delivering network traffic (via one or more STA-Bus or virtual machines). The intermediary device 104 determines suitable network communication paths for transmitting encrypted packets, which may be validated by performing authentication. In some
5 embodiments, the intermediary device 104 may receive or intercept encrypted packets from the same source or different sources, and directed to the same destination. The intermediary device 104 receives the encrypted packets, and determine authentication to the user's connection to a network. The intermediary device 104 directs the packets to a path manager 105 for transmission to the destination.

10 Figures 2 and 3 illustrate a network environment for delivering and/or operating a computing environment for multipath delivery. The intermediary device 104 includes an accelerator proxy 104-A and a peer proxy 104-B. The accelerator proxy 104-A may include a computer device with installed accelerator proxy software and configurations. The peer proxy 104-B may include a virtual network function with installed peer proxy software and
15 configurations. The intermediary device 104 communicates with the user device 102 to establish a plurality of tunnel paths in relation to the number of network paths. This enables the intermediary device 104 to transfer data over multi-number of network paths simultaneously. The intermediary device 104 manages and tracks one or more connection information between the user device 102 and the service device.

20 For the avoidance of doubt, the use of the term "connection" in this case, is in the context in which the protocol tunnels communicate between nodes of the intermediary 104. For example, the path manager 105 maps each connection request in response to receipt of a communication from a source, having source IP, to one or more data tunnels. The path manager 105 maps the connection request from a user 102 to a data tunnel. The mapped
25 connection may be provided an assigned connection information between the mapped entities. This connection information is preserved by the path manager 105 for TCP and UDP packet authentication for encryption and decryption of communications.

In another example, the path manager 105 may request establishment of a connection from a source IP to a destination IP. The transport layer packets (TCP or UDP) may be
30 communicated between established streams for multipath delivery communication from the source IP to the destination IP.

As shown in Figure 3, the intermediary 104 comprises a DNAT for intercepting and redirecting transport layer connection to a predetermined IP address and port. The intermediary 104 executes transparently to any applications and user device 102. The path manager 105 manages a plurality of connection in a plurality of network paths. The path manager 105 controls and manages the delivery of the network traffic to the user or the service device. The intermediary 104 establishes data queues for queuing and transmitting one or more network traffic for transmission by the intermediary 104. The intermediary 104 manages a number of connections in queues. In an embodiment, the intermediary 104 manages multiple queues between the nodes of the proxies, and sends network traffic for a certain network path to a dedicated queue. The intermediary 104 on a network may communicate data packets between the established streams between nodes of the intermediary 104. The intermediary 104 may be a hardware unit or a software component.

The intermediary 104 comprises a connection manager 106 for mapping user traffic from each source IP into the connection, mapping user traffic from each source IP into the connection, handling incoming user traffic and multiplexing payload data into each connection, sending outgoing user traffic from connection.

The intermediary device 104 establishes data tunnels on receiving a connection request by a user device 102 to connect to the service. The intermediary device 104 may be configured to establish a maximum number of data tunnels over a number of network paths for user device 102. The intermediary 104 device may allow configuration of this predetermined maximum. Incoming packets may be associated with existing data tunnels or newly created data tunnels. The path manager 105 maps an incoming user traffic connection request to one or more data tunnels. Each connection possesses a set of connection identifiers that identify the connection.

Each user IP is considered a mapping key during its lifetime, each user IP is mapped with a data tunnel connection at the accelerator proxy 104-A. The path manager 105 maps the user connection request to a data tunnel having an assigned connection information (e.g. Connection ID). The mapped connection is provided a connection information, which is unique to the mapped connection between source-target entities. Connection IDs allow connections to migrate to a new network path, by directing to another endpoint or forced by a change in a middlebox.

The connection information is preserved by the path manager for TCP and UDP packet authentication for encryption and decryption of communications. The connection information may be an identification number that uniquely indicates specific communication channel for TCP/UDP connections. There is mapping between the
5 identification number and the unique user connection information of the TCP/UDP connections (source IP address, source IP port number, destination IP address, destination port number).

Establishment of Connection

The intermediary 104 identifies a user by its unique user connection information (or
10 source IP address) when a user requests for establishing a connection. The intermediary 104 may identify one or more connections (e.g. TCP or UDP connections) under a connection request. For example, a user device 102 may request establishment of a connection from a source IP address to a destination IP address. On detecting a new user (new source IP address), the intermediary 104 may request establishment of a connection
15 from source IP address to a destination IP address.

During the handshake process of accelerator transport protocol, the connection is issued a unique identification number and new shared key for encryption and decryption of packets for specific communications. The connection is maintained until it expires by the designated time, which is managed by the user. On expiry of the designated time, no packet
20 transmission occurs between nodes of the intermediary (or between intermediaries), without physical network connectivity after the handshake process.

Creation of connection record

The intermediary device 104, via the connection manager 106, establishes a connection record for multipath delivery for a connection request received from the user
25 device 102. The connection manager 106 may be configured to create or establish the connection record, keep track of statuses, and/or update the connection record. The connection manager 106 may associate the established connection record with the user connection request. In response to receiving the connection request, the intermediary 104 checks the IP header to identify the user device 102. The intermediary establishes a plurality
30 of tunnel paths based on the number of network interface configured to deliver outgoing traffic via gateways. The connection request from the same user device 102 will be mapped

to the tunnel paths of the user IP based on a scheduling determined by the path manager 105. The mapping of the user connection request to the tunnel paths of the user IP is created and maintained as the connection record.

Authentication

5 The intermediary 104 receives communication (e.g. data or information), encrypts or decrypts the communication and sends between the user device 102 and the service device, via the path manager 105. The accelerator proxy 104-A intercepts the user traffic, encrypts the user traffic, and sends via a known stream and the established connection. The peer proxy 104-B receives, decrypts and forwards the stream data to the service or destination
10 based on the connection record. The authenticated packets may be transmitted across different networks and multiple network paths, data channel change across different networks and multi-channel UDP packet delivery (referred to as "UDP stitching").

Authentication is achieved with the issued identification number and cryptographic mechanism to minimize packet round trip time for connection establishment. The
15 cryptographic handshake provides an authenticated key exchange using Public Key Infrastructure, or standard format of public key certificates, e.g. X.509. Authentication of the identity of the form using X.509 may be carried out for intermediary proxies 104 having a key pair. Conventional packet authentication in IP based protocols, such as TCP and UDP, involves exchanging ports and IP addresses (source port number, destination port number,
20 source IP address, destination IP address) of the source and destination. As such, this conventional packet authentication method is not able to support mobility/roaming of the user device 102 across different networks, multipath packet delivery using multi-number of networks.

Referring to Figure 2, there is a deployment of the system for network switching. The
25 intermediary 104 manages or handles data via a network address translation (NAT) module. The system may be provided a network address translation module for updating the UDP mapping and connection in real-time. The connection information is updated while the user connection remains unchanged. NAT connection record may be used to store, track and maintain information regarding a socket or socket information (or user connection
30 information).

In an embodiment, with reference to Figure 4, the NAT component modifies the port information, updates the UDP port connection information (4-number-tuple; IP address, port number of user device 102, IP address and port number of the receiving proxy of the intermediary 104). The intermediary 104 may update the port information to enable (1)
5 selective sending of TCP connection or UDP stream from an identified source to a certain path in accordance to load balancing or packet scheduling algorithms, their settings and path manager; and (2) switch TCP connections and UDP stream over to the other available network path due to the connection failure between the accelerator proxy 104-A and the peer proxy 104-B. For example, by network service interruption, or deviation of the
10 measured latency or signal strength of wireless network signal, (e.g. above or below the threshold that may be set by a user, or selected by path manager 105.

In one embodiment, the network observation service of the intermediary 104 may determine the service device for which to distribute a user connection request. If the intermediary 104 determines that the service device is not available or has latency over the
15 predetermined threshold, the intermediary 104 can direct or distribute the user connection requests to another service device.

The connection information may be in the form of a connection ID (or identifier). The connection ID is maintained over the lifetime of the communication or connection between the nodes of the intermediary. This connection ID may be retrieved when re-establishing
20 the mapped connection. In migration initiation, the update of connection ID may update the packet information tuples, such as 4-number-tuple comprising at least two IP addresses and two ports.

The NAT function in the intermediary 104 interfaces with the user device 102, and switches the tunnel backhaul from network path 1 (physical interface ETH_2 configured to
25 be the WAN gateway GW1) to network path 2 (physical interface ETH_3 configured to be the WAN gateway GW2). A first connection ID (STAP_ID01) is established and mapped with the user device 102 in the LAN. The migration process, triggered by a network observation decision at a first intermediary device 104, is transparent to the user application (TCP01). The endpoint of the tunnel path is modified, but retains the stream mapping, user
30 payload frame status and session context connection ID which contains authentication information including shared key that is issued in the handshake process, and related control information.

The path manager 105 monitors, measure, collects, analyses and reports data on the frequency that is determined by the user, or by default. The path manager 105 may invoke network observation service by monitoring the condition of the network paths and selecting suitable network path for general packet delivery, or prioritized packet delivery, or maximum utilization of the available networks capacities. The path manager 105 may decide on a suitable network path by measuring the response time from different network paths, wireless signal strength (e.g, WiFi signal strength, 3G/4G/LTE mobile network signal strength), congestion level, estimated available bandwidth, priority setting and many other factors when transmitting over multiple network pathways or choosing a specific network path for the packet delivery.

The path manager 105 may also detect when a tunnel path is down or disconnected. Advantageously, ongoing requests may be switched seamlessly to remaining tunnel paths with available bandwidth. Switching from one tunnel path to another may be carried out when deploying multiple network pathways for transmission. The retention of the connection information for a mapped user-server side intermediate device connection enables the connection information to be retrieved when this same mapped connection is re-established.

In addition, the path manager 105 may provide real-time update on existing network bandwidth as well as estimating the estimated bandwidth of the network. Advantageously, this enables monitoring the performance for optimizing the bandwidth on multiple network paths in the system.

With reference to Figure 5, the system provides load balancing by distributing network traffic according to a scheduling mechanism. The scheduling mechanism comprises a scheduling algorithm which may be selected from at least one of round robin, hash code by destination IP and port, least connection, weighted round robin, weight least connection, dynamic weighted round robin. The scheduling mechanism may be configured via the web interface and command line interface.

Scheduling Algorithms

- Round Robin: the algorithms map user connection to available tunnel paths based on the round robin method, distributing to each available path once before starting over.

- Hash code by destination IP and port: the algorithms map user connection to available tunnel paths based on the hash code of 2-number-tuple (destination_ip, destination port). This scheduling type may provide user experience consistency while connecting to same web server in some instance.
- 5 • Least connection: the algorithms map user connections to available tunnel paths based on the number of active streams that are currently in use to accelerate traffic. The connection with the least active streams has the most priority for selection.
- 10 • Weighted round robin: the algorithms map user connection to available tunnel paths based on the round robin method with a weighted factor calculated for each path. The weighted factor is calculated based on the estimated bandwidth from the paths.
- 15 • Weighted least connection: the algorithms map user connection to available tunnel paths based on the least connection method with a weighted factor calculated for each path. The weighted factor is calculated based on the estimated bandwidth from the paths.
- 20 • Dynamic weighted round robin: the algorithms map user connection to available tunnel paths based on the weighted round robin method. The weighted factor is calculated based on the estimated bandwidth, also the average measured RTT from the paths.

Network Switching

The system deploys an adaptive network switching functionality when there are multi-number of networks in the distributed system. The distributed system may utilize a single network path. The intermediary provides connection roaming when there are multi-number of networks and multi-number of networks paths to be used simultaneously, for throughput and resilience improvements.

The intermediary 104 enables network switching or “Connection migration” from a primary network to one or more backup networks to provide smooth transition from one network communication. The intermediary 104 may switch network when deploying multiple network pathways. From Figure 6, according to an embodiment, the intermediary

104 may switch from the primary communication via satellite network to the LTE communication as failover if the network monitoring of the intermediary 104 determines that there are network issues, such as satellite malfunction, or when the primary communication network performs below a desired threshold.

5 The intermediary 104 may be configured to provide seamless switching of the networks. By monitoring the present network conditions, the intermediary 104 may determine if network switching is required. If the intermediary 104 determines any changes to the endpoint and the user connection information (e.g. IP address, port numbers), such as changes migrating to a new network, the intermediary 104 may perform migration from
10 the current established connection. The intermediary 104 may migrate to a previously-mapped connection by retrieving the connection information (or Connection ID) and allow re-establishing that mapped connection. In an embodiment, the intermediary 104 may establish a new mapped connection if a connection ID is not available for an endpoint, prior to connection migration. Advantageously, ongoing user requests can be continually
15 processed, thereby eliminating interruption to network services and disruption to user experience.

 According to embodiments, the intermediary 104 may be deployed to enable multi-number of networks and multi-number of network paths concurrently. Figure 7 illustrates the deployment of a system for network switching for multipath delivery. The system may
20 comprise two network paths, and two network devices, user 1 (IP1) and user 2 (IP2). TCP connections originating from IP1 may accelerate via tunnel path 1 – STAP_P1 (TCP1, TCP2) and tunnel path 2 – STAP_P2 (TCP3) simultaneously. TCP connections from IP2 may connect and accelerate via two tunnel paths.

UDP Stitching

25 The established connection has an assigned connection ID that is used in packet encryption and decryption. This method of packet authentication is used in multipath delivery, data channel change among different networks, multi-channel UDP packet delivery (or UDP stitching).

 The intermediary may be configured to accelerate the UDP user service via multiple
30 data tunnels. The intermediary may establish more streams in terms of their source and destination. These established streams may be mapped to a single UDP connection for

delivering data via the data tunnels. The intermediary may provide its scheduling algorithm to select the streams on which the endpoint transmits data.

The intermediary may be configured various settings of Class of Service (CoS) to vary between reliability and latency. The CoS settings manages the received packets in the receiving queue in the server side of the intermediary, which result in differing reliability and delay in final delivery of the packets. The intermediary may be configured for different types of services. The configuration may be performed via the Web UI or management console, or by the users.

In an embodiment, the intermediary may be configured a higher CoS to reduce delay in the final delivery of data to the end point. A higher CoS would be advantageous for real-time service, such as real-time video or audio streaming. This advantageously eliminates the temporary obstruction to traffic delivery (or head of blocking) until the server stream recovers to consume data and provide feedback on data window size due mainly in part to the memory limit when accelerating UDP service via a single stream. The delivery capacity may be maximized by supporting multiple delivery streams in the data tunnel if the traffic acceleration node is a single-tunnel acceleration, for instance VPN-like application, IPSec-Over-UDP application. In addition, delivery capacity of a single UDP connection may extend which is intercepted by its identity tube (source IP address, source port number, Destination IP address, Destination port number), e.g. video UDP streaming applications.

As shown in Figure 8, a user IP1 configures n number of streams for each UDP connection to deliver data and is assigned a designated data tunnel (STAP_IP1). UDP connection (UDP1) which is identified by a tube (src_ip1, src_port1, dst_ip1, dst_port1). The UDP1 is intercepted at the accelerator proxy 104-A and mapped from stream1_1 to stream1_n. Payloads are delivered via these streams at the accelerator proxy 104-A by round robin scheduling method, and in same manner at peer proxy 104-B as the reversed direction.

When deploying on VPN-like application, IPSec-Over-UDP application, there is one UDP connection (UDP1) that takes care of data delivery. With one stream mapped to UDP1 via the data tunnel (STAP_IP1), the tunnel capacity flow is not fully utilized. This enables more streams to deliver UDP1 packets simultaneously, thereby maximising the delivery capacity with flow control.

It is to be understood that the above embodiments have been provided only by way of exemplification of this invention, and that further modifications and improvements thereto, as would be apparent to persons skilled in the relevant art, are deemed to fall within the broad scope and ambit of the present invention described herein. It is further to be
5 understood that features from one or more of the described embodiments may be combined to form further embodiments.