

SYSTEM AND METHOD FOR ACCELERATING DATA DELIVERY

FIELD OF THE INVENTION

- 5 The present invention relates to the field of communication systems and networks, and in particular, but not exclusively, to a system and method for accelerating data delivery for providing connectivity between communication devices wirelessly, e.g. providing wireless communications over geologically wide areas.

10 BACKGROUND TO THE INVENTION

- The following discussion of the background to the invention is intended to facilitate an understanding of the present invention. However, it should be appreciated that the discussion is not an acknowledgement that any of the material referred to was published, known or part
15 of the common general knowledge in any jurisdiction as at the priority date of the application.

- The use of mobile devices (e.g. laptops, mobile phones, tablets) and applications (e.g. video and audio streaming services, cloud services) have become commonplace in today's
20 networked environment. These mobile devices and other IoT devices/sensors are highly distributed at the edge of the network along with real time and latency-sensitive service requirements. This has created a demand for secured high-performance communication over various types of wireless networks and in a network environment that spans over large geographical area.

- 25 Mobile devices typically rely on wireless network such as Wi-Fi and mobile network (e.g. LTE and microwave). Moreover, deploying wireless backhaul via satellite provide a very challenging service environment for Internet Service Providers due to the proximity at which the end users/devices are situated. As a consequence, real-time and latency sensitive service
30 requests often endure large round-trip delay, network congestion and service quality degradation.

Most conventional approaches are directed to accelerating content delivery by content caching, compression of protocol header, payload and contents. However, these attempts no longer work when transferring contents via encrypted protocols like HTTPS, TLS, SSL or the like, because the encrypted content cannot be cached. Due to the file size of various media formats, multimedia contents are highly compressed with high-performance compression algorithms, and show higher ratio of compression than real-time compression algorithms that can be used for packet level in real-time. Further compression on the already highly-compressed data using real-time algorithms will only increase the size of data and introduce additional latency in data transmission, and is not suitable for accelerating content delivery.

Furthermore, wireless networks have the tendency towards randomness in terms of availability, latency, capacity of the connectivity. Therefore, accelerating modern web applications, real-time applications and content stream will require a change in the transport layer protocol for better signalling: efficient connection handshake and reliable data transmission in reducing packet loss, improving response time, maximizing available bandwidth, and coupled with the necessity to be non-intrusive with regard to compatibility with existing network equipment.

Therefore, the present invention seeks to provide a system and a method to overcome at least in part some of the aforementioned disadvantages. In particular, to provide a system and a method for accelerating data delivery over wireless network and between remote sites to provide connectivity between communication devices wirelessly.

SUMMARY OF THE INVENTION

Throughout this document, unless otherwise indicated to the contrary, the terms “comprising”, “consisting of”, and the like are to be construed as non-exhaustive, or in other words, as meaning “including, but not limited to”.

Protocol support is one of most essential factors in determining the overall network service performance for wireless communications. In some instances, lack of protocol support that is not optimized for wireless transmission and routing decisions lead to snowballing effects

of intermittent failures, impairing the overall service performance. The invention provides a solution that provides efficient connection handshake and reliable transmission to reduce the number of packets, thereby improving response time. The invention effectively deals with constantly changing situations, to address problems with latency by analysing traffic & routing conditions in real time to find the fastest route between the data source and the destination.

The embodiments of the present invention have at least the following advantages:

1. According to embodiments, there is provided a reliable transport protocol built atop UDP for delivering accelerated data transparently without modifying the user applications to provide higher throughput and lower latency. The transport protocol enables authentication by way of a data and encapsulated packet transportation response for sending and receiving user traffic, including encrypting and decrypting packets, multiple-frames including control and data frames into a packet. This is advantageous for accelerating data delivery in a wireless environment for a plurality of connections on alternative and secured medium.
2. According to embodiments, transport connections in the network traffic are selectively identified to be directed for acceleration based on a predetermined acceleration rule which can be configured at the accelerator proxy. Advantageously, connection requests are selectively accelerated from source to destination, which improves transport layer performance.
3. According to embodiments, optimization of the transport layer performance is provided to address problems associated with TCP's three-way handshake by selectively directing network traffic over the accelerated transport protocol by means of a process mapping of a TCP/UDP connection to a stream. This advantageously improves the RTT handshake process for the transmission and obviates the need to perform TCP long-handshake process via the wireless network.
4. According to embodiments, there is provided an adaptive congestion control that dynamically limit the bandwidth used by a user associated with IP address. This

advantageously controls the amount of data being sent and ensures smooth transfer of data.

In accordance with a first aspect of the present invention, there is provided a method for
5 accelerating data delivery. The method comprises:

establishing a connection for transporting one or more transport layer packets;

configuring a predetermined rule to select a transport layer packet to undergo an acceleration mode, wherein unselected transport layer packets undergo a non-acceleration mode;

10 receiving the transport layer packet selected based on the predetermined rule and retrieving one or more identifiers associated with the transport layer packet;

creating mapping among the one or more identifiers;

determining whether there is mapping between the transport layer packet and the one or more identifiers;

15 directing the transport layer packet through at least one acceleration stream to accelerate the transport layer packet to a destination provided by the one or more identifiers if there is mapping between the transport layer packet and the one or more identifiers.

Preferably, the predetermined rule is configured to select a transport layer packet based on
20 the one or more identifiers.

Preferably, the one or more identifiers comprise Source IP address, Source IP port number, Destination IP address, Destination port number, and stream ID.

25 Preferably, there is provided a connection identifier for authenticating the transport layer packet transported over the connection.

Preferably, placing one or more corresponding payloads of the transport layer packets into the communication buffer associated with the connection identifier.

30

Preferably, when directing the transport layer packet through the at least one acceleration stream, a transparent proxy module performs one or more operations comprising encrypting the transport layer packet, and multiplexing one or more data communication sessions onto the at least one acceleration stream.

Preferably, when accelerating the network traffic to the destination, a remote proxy module performs one or more operations comprising decrypting the encrypted transport layer packet, and demultiplexing the one or more data communications.

- 5 Preferably, the receiving of the transport layer packet utilizes a first transmission protocol, the non-acceleration mode comprises the first transmission protocol, and the acceleration mode comprises a second transmission protocol different from the first transmission protocol.
- 10 Preferably, calculation for limiting bandwidth is based on the following:

$$\text{dl_rate} = \min(\min(\text{estimated_stap_bw}, \text{current_dl_rate} * \text{adaptive_factor}), \text{upper_con_capacity})$$

15 wherein,

- dl_rate: the desired in-coming or out-going speed (bandwidth utilization) in the next phase
- estimated_stap_bw: estimated bandwidth from STAP
- current_dl_rate: the current in-coming or out-going speed (bandwidth utilization)
- 20 • adaptive_factor: to minimize the buffer overrun scenarios, it reduces incoming
- upper_con_capacity: configured upper limit in the bandwidth utilization

In accordance with a second aspect of the present invention, there is provided an apparatus for accelerating data delivery, the apparatus comprises:

25

a transparent proxy module configured to establish a connection to a remote proxy, wherein the connection is for transporting one or more transport layer packets; and configure a predetermined rule to select a transport layer packet to undergo an acceleration mode, wherein unselected transport layer packets undergo a non-

30 acceleration mode;

the transparent proxy module configured to receive the transport layer packet selected based on the predetermined rule and retrieve one or more identifiers associated with the

transport layer packet; create a map among the one or more identifiers; and determine whether there is mapping between the transport layer packet and the one or more identifiers; and direct the transport layer packet through at least one acceleration stream to accelerate the transport layer packet to a destination provided by the one or more
5 identifiers if there is mapping between the transport layer packet and the one or more identifiers.

In accordance with a third aspect of the present invention, there is provided a system for accelerating data delivery. The system comprises a user device, a proxy device and a service
10 device. The system comprises a user device, a proxy device and a service device. The user device connects to the proxy device via a communication means. The proxy device connects to the service device via a communication means. All three devices are operable to perform the method as detailed in accordance with the first aspect of the present invention.

15 Preferably, the proxy device directs any network traffic having the connection identifier to the connection and processed by the nodes.

In accordance with a fourth aspect of the present invention, there is provided a computer program product comprising a plurality of data processor executable instructions that when
20 executed by a data processor in a system causes the system to perform the method as detailed in accordance with the first aspect of the present invention.

In accordance with a fifth aspect of the present invention, there is provided a transport protocol built atop UDP for accelerating data delivery.

25

Other aspects and advantages of the invention will become apparent to those skilled in the art from a review of the ensuing description, which proceeds with reference to the following illustrative drawings of various embodiments of the invention.

30 **BRIEF DESCRIPTION OF DRAWINGS**

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

Figure 1A illustrates a wireless communication system for accelerating data delivery in accordance with an embodiment of the present invention.

5 Figure 1B is a block diagram of the software architecture of the system of Figure 1A.

Figure 1C is a block diagram of the transport protocol of the system of Figure 1B.

Figure 1D is a block diagram of mesh model deployment having a plurality of the system of
10 Figure 1B.

Figure 2 is a block diagram illustrating transparent intercepting of selected packets by the first intermediary device in accordance with an embodiment of the present invention.

15 Figure 3A is a block diagram illustrating multiplexing of TCP/UDP connections in accordance with an embodiment of the present invention.

Figure 3B is a flow chart illustrating an operation of mapping IP address to a predetermined data tunnel in accordance with an embodiment of the present invention.
20

Figure 3C is a flow chart illustrating the demultiplexing and replicating connection performed by the peer proxy of the Figure 3A.

Figure 4 is a block diagram illustrating an operation of circular queue management for
25 handling application data in accordance with an embodiment of the present invention.

Figure 5 is a block diagram illustrating establishment of connection for communication between proxy devices in accordance with an embodiment of the present invention.

30 Figure 6 is a block diagram illustrating an operation of stream multiplexing in accordance with an embodiment of the present invention.

Figure 8A is a flow chart illustrating congestion management in accordance with an embodiment of the present invention.

Figure 8B is a flow chart illustrating an operation for information bundling in ACK frame in accordance with an embodiment of the present invention.

- 5 Figure 9 is a block diagram illustrating an operation for handling multiple protocol of the system of Figure 1B in accordance with an embodiment of the present invention.

Figure 10 is a block diagram illustrating the migration process of the connection ID in accordance with an embodiment of the present invention.

10

DETAILED DESCRIPTION OF THE EMBODIMENTS

Particular embodiments of the present invention will now be described with reference to the accompanying drawings. The terminology used herein is for the purpose of describing
15 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 the same meanings as commonly understood by one of ordinary skill in the art to which this invention belongs.

- 20 The use of the singular forms “a”, “an”, and “the” include both singular and plural referents unless the context clearly indicates otherwise.

The use of the term “STAP connection” refers to a mechanism that enables communication between the first proxy device and the second proxy device. Communication is established
25 between the nodes of the proxy devices.

The use of the term “connection identifier” refers to an identifier that is used to identify a connection between the nodes of two proxy devices.

- 30 The use of the term “wireless” includes 3G, 4G, 5G, Wi-Fi, and any other kinds of wireless connection.

The use of the term “acceleration” may include application acceleration, flow acceleration and other acceleration techniques.

There is provided a transport protocol for accelerating data delivery comprising accelerator
5 transport protocol (STAP) implemented atop UDP for transporting user data.

In accordance with a first aspect of the present invention, there is described a system and method for accelerating data delivery for providing connectivity between communication devices wirelessly and separated by geographical distances. The system comprises a user
10 device 102, a proxy device 104 and a service device 106. The user device 102 connects to the proxy device 104 via a communication means. The proxy device 104 connects to the service device 106 via a communication means. The three devices are connected as described above for the accelerating data from one device to another device, by providing a cryptographic handshake for connection establishment to minimise packet round trip time.

15 The system and method direct selectively network traffic to be transmitted over the accelerated transport protocol based on association of connections identifier. Advantageously, any network traffic identified with the associated connections identifier is directed to the connection and obviates the need to perform TCP long-handshake process
20 via the wireless network, thereby improving the RTT handshake process for IP transmission.

Figure 1A shows a schematic diagram of a system in accordance with an embodiment of the present invention. The system comprises a user device 102, a first proxy 104-A, a second proxy 104-B and a service device 106. The user device 102 connects to the first proxy 104-
25 A via a communication means. The first proxy 104-A connects to the second proxy 104-B via the communication means. The second proxy 104-B connects to the service device 106 via the communication means. The devices and proxies are connected as described as above for the communication of data from one point to another point.

30 The proxy device 104 may be deployed between the user device 102 and the service device 106 connected to a private network or the Internet. The user device 102 may form part of a network that includes one or more clients, routers and proxy devices. The service device 106 may be placed geographically apart from the user device 102. The service device 106 can be connected to a wireless network. The proxy device 104 may comprise a first proxy 104-

A and a second proxy 104-B. The first proxy 104-A and the second proxy 104-B may be in the form of a hardware or software. The second proxy 104-B may also be in the form of a virtual network function (VNF) for increasing network scalability and operational efficiency.

5

In one configuration, a first proxy 104-A may be placed between the user network and the satellite terminal, and a second proxy 104-B may be placed between the user device 102 and the satellite network.

10 In a second configuration, the first proxy 104-A may be placed between the user network and a 3G/4G router, and a second proxy 104-B may be placed between the user device 102 and the 3G/4G network.

Figure 1B illustrates the software architecture of the proxy device 104 as a transparent proxy
15 for accelerating data delivery according to embodiments of the present invention. The transparent proxy includes an accelerator proxy 104-A and a peer proxy 104-B. The accelerator proxy 104 comprises a connection manager 124-A to establish a connection 126 to the peer proxy 104-B via the connection manager 124-B of the peer proxy 104-B. The establishing of the connection 126 enables creating a plurality of interleaving streams to
20 multiplex and demultiplex user payload data onto each stream.

At least some communication between the user device 102 and the service device 106 may be selected to pass through the proxy device 104. The accelerator proxy 104-A may establish a connection 126 with the peer proxy 104-B. The connection 126 established between the
25 nodes of the proxies allows communication to be established. The proxy device 104 uses the connection 126 to accelerate data delivery of the at least some communications between the user device 102 and the service device 106.

Accelerator Proxy Software Architecture

- 30
- Configuration Web UI and management console: responsible for setting the traffic proxying rule and system configurations via Web UI or console, collect and display traffic statistic, operate system commands.

- Service Acceleration Rule: Selecting only packets for the specified source IP address or IP segment which head for the specified destination IP port, which indicates the types of services. e.g. HTTPS, SSH, XMMP and any other application level protocols that used by various commercial and non-commercial applications.
- 5 • Connection Manager 124-A: Establishes connection 126 between two communications devices (nodes) comprising an interleaving multi-stream, which is advantageous for congestion control in the network, efficient buffer management, preventing head-of-blocking situation due to the reliable data delivery feature. Depending on the configuration, the connection manager 124-A contains 1 or many
10 accelerator processes that are responsible for establishing a connection 126 between two communications devices (nodes), mapping user traffic from each source IP into the connection 126, mapping user traffic from each source IP into the connection 126, establishing the connection 126, handling incoming user traffic and multiplexing payload data into each connection 126, sending outgoing user traffic
15 from the connection 126.
- Non-blocking circular ring queues: A response for storing user incoming and outgoing traffic, which is used for congestion control in translating between TCP to an accelerator transport protocol, implementation of packet scheduling for QoS and CoS, Network switching, multi-path delivery / communication channel aggregation
20 and its traffic distribution algorithms and UDP stitching.
- Transport protocol: Accelerator transport protocol over UDP as a data and encapsulated packet transportation tunnel response for sending and receiving user traffic between proxies; it contains techniques of secure connection including encrypting and decrypting packets, multiple-frames including control and data
25 frames into a packet, packet loss detection, reliable delivery, network situation aware flow and congestion control, multi-path delivery, network change among different networks and multi-channel UDP delivery.

After connection establishment between the accelerator proxy 104-A and the peer proxy 104-B, the accelerator proxy 104-A sends destination information to the peer proxy 104-B.

30 The destination information may include the destination IP address and port number. The

peer proxy 104-B receives the destination information, which may be mapped with the TCP or UDP socket number to establish TCP or UDP connection to the destination.

Peer Proxy Software Architecture

5

- Configuration Web UI and management console: responsible for setting system configurations via Web UI or console, collect and display traffic statistic, operate system commands
- Connection Manager 124-B: responsible for de-multiplexing user payload data from the connection 126 and regenerate TCP or/and UDP session to the destination and deliver the data thereto.
- Non-blocking circular ring queues: response for storing user incoming and outgoing traffic, which is used for congestion control in translating between TCP to Accelerator transport protocol, implementation of packet scheduling for QoS and CoS, Network switching, multi-path delivery / communication channel aggregation and its traffic distribution algorithms and UDP stitching.
- Transport protocol: Accelerator transport protocol over UDP as a data and encapsulated packet transportation tunnel response for sending and receiving user traffic between proxies; it contains techniques of secure connection including encrypting and decrypting packets, multiple-frames including control and data frames into a packet, packet loss detection, reliable delivery, network situation aware flow and congestion control, multi-path delivery, network change among different networks and multi-channel UDP delivery.

Figure 1C illustrates the protocol layer of the accelerator transport protocol. The accelerator transport protocol is located atop User Datagram Protocol (UDP), below the application layer by the socket layer. The Connection Manager 124-A handles the proxied inputs of TCP or UDP sockets types from the application layer and sends to the receiver over the accelerator transport protocol. The Connection Manager 124-A translates this to the destination network. Advantageously, the proxy 104-A offers application transparency and does not require modification of the user application source in adopting the accelerator transport protocol. From the perspective of a client and server application, the proxy 104-A

is considered transparent, wherein a user request is redirected without modification to the same. Other non-proxying user traffic are bypassed as the normal protocol layer.

In accordance with an embodiment of the present invention, there is provided a method for
5 accelerating data delivery. The method comprises the following steps:

Step 1: Selection of packets to be intercepted

Step 2: Identifying and mapping each user IP address

10

Step 3: Establishing TCP connection with overall handshake improvement

Step 4: Securing and optimizing data by flow and congestion control

15 **Selection of packets to be intercepted**

Figure 2 is a block diagram illustrating the selection of interested packets from a network traffic to be redirected to the proxy 104 and converted into an accelerator transport protocol.

20 The accelerator proxy 104-A may be deployed on the user side in the existing network to select interested packets from the traffic from user devices and applications. This packet selection may be based on the Service Acceleration Rules, which can be configured to select according to the source information (e.g. source IP address, port number) or destination information (e.g. destination IP address, port number).

25

For example, when a user requests a remote resource such as accessing a website, the server receives credentials from the user. These credentials may include IP addresses, protocol state information, port number. The credentials are captured and stored by the server. In response to the request, the server determines whether the credentials are defined in the Service
30 Acceleration Rule in order to redirect the traffic.

The packets may be redirected to the accelerator proxy 104-A by utilizing Destination Network Address Translation techniques, by updating the IP address and port number of the accelerator proxy 104-A. The accelerator proxy 104-A may capture the original destination

IP address and port number of the packet and transfer this information to the peer proxy 104-B, via a connection 126 established between the modes of the accelerator proxy 104-A and the peer proxy 106-B. The peer proxy 106-B then creates a transport layer connection (TCP or UDP) to the destination and directs the packets to the destination based on the TCP/UDP
5 socket and destination information.

The accelerator proxy 104-A is configured to select packets to be directed towards an acceleration mode by transmitting over the accelerator transport protocol between two nodes of the proxies. Packets that are not selected will not undergo the acceleration mode and will
10 flow as normal.

Identifying and mapping each user IP address

Incoming packets are identified on receipt. New source IP from a new source may be
15 associated with an existing connection or to potentially create a new connection between nodes of the proxies. Each connection may be identified by providing a set of connection identifier (or connection ID). Packets having that connection ID may be routed back to the node and identified by the node upon receipt.

20 New source IPs from a new device 102 which is not registered with or not identified by the accelerator proxy 104-A may be mapped with an existing connection 126. The existing connection 126 having a connection ID may be associated with the source IP by creating a map of the source IP with the connection ID. The accelerator proxy 104-A accepts the new TCP or UDP session.

25 When the accelerator proxy 104-A receives a new TCP session or UDP session from a new user device 102 having new source IPs, the accelerator proxy 104-A creates a new connection 126 to the peer proxy 104-B. This connection 126 having a connection identifier may be created and mapped (<key=IP, value=connection ID>) with the source IP address
30 in the connection manager 124-A, creating an association between the connection 126 and the new source 102.

In other instances, a new connection 126 may be established. The new connection 126 may have a connection ID retrievable from an existing pool of established connection 126.

Connection Identifier

Each connection 126 established between the nodes of the two proxies may be identified by
5 providing a set of connection identifier (or connection ID). The connection ID may be
provided in the form of a 64-bit number which is globally unique in all networks. This
identifier may be included in accelerator protocol header section and exchanged between
accelerator proxy 104-A and peer proxy 104-B to identify the respective connection 126.
The identity may be used to associate with the authentication information for encryption and
10 decryption of the packet during the lifetime of the connection 126.

Reserved STAP connection pool

A reserved pool of connections 126 may comprise a pool of available connections 126
15 established between the nodes of the accelerator proxy 104-A and the peer proxy 104-B.
These connections 126 may be established during the initiation of the accelerator proxy 104-
A and may be maintained to ensure fast delivery of streaming data and real-time application
data, including VOIP, IoT machine data in high latency networks or over long distances
without any waste of time for establishing the connection 126.

20

An established connection 126 in the connection pool remains available and may be
terminated until a configured time-out (e.g. idleness due to no transaction). This value for
time-out may be configured and enabled, regardless of the availability of the physical
network. The timed-out connections may be closed and removed from the pool, and new
25 connections 126 may be inserted into the pool to maintain the activity of the connection
pool.

Once the connection 126 is established between the nodes of the proxies, the encrypted
handshake packet may be sent and decrypted to the peer proxy using a security certificate,
30 such as X.509 certificate. While the X.509 certificate has been described in this case, it
would be appreciated that the security certificate may be modified with the security model.

The handshake packet contains the configuration from the accelerator proxy 104-A at the
user end. The packet may be validated and synchronized in both ends in the handshake

process. If the handshake process is successful, the connection 126 will be accepted by the accelerator process in the peer proxy 104-B at the server end (see Figure 3C and Figure 6). Advantageously, this handshake process is done by single round (1-RTT) of packet exchange.

5

The Connection Manager 124-B is capable of demultiplexing the interleaved stream data and extracting the destination information (e.g. destination IP address and port number) to perform the final data delivery by re-constructing the TCP or UDP session from the user side accelerator proxy 104-A and transferring data by the original protocol, which may be TCP or UDP.

10

Creation of interleaving streams

The connection 126 established between nodes of the proxies comprise a plurality of interleaving streams 128. The streams are identified within the connection 126 by a stream identifier. The stream identifier may be provided as a stream ID. The stream IDs are unique to a stream 128 and can be used to send stream data by a node of the proxy. The streams 128 may be created at either nodes and capable of sending data interleaved with other streams 128 and can be terminated.

20

A new stream 128 for the connection 126 may be created when a new TCP or UDP session is accepted or terminated for the interleaving of data channels for TCP and UDP in the connection 126. A stream 128 may be created by sending data and each stream 128 is identified by a stream ID. Every newly created stream 128 comprises TCP socket number from the OS, source IP address and port number, which may be used for multiplexing and demultiplexing of the connected sessions. (see Figure 3A and Figure 3B).

25

The accelerator proxy 104-A establishes the connection 126 to the peer proxy 104-B, when the accelerator proxy 104-A receives any new TCP session or UDP session including the first session. A stream 128 having a stream ID may be created within the established connection 126, which will be the data channel for the respective TCP or UDP session. There is mapping of the source information, destination information and the stream ID. A map may be created between the Source IP address, Source IP port, Destination IP address, Destination port number, and newly created stream ID.

30

The accelerator proxy 104-A maps the newly created stream ID with the connection identifier and the accepted TCP or UDP session socket number (file-descriptor number) may be mapped with connection identifier, destination IP address and destination port number.

5 The accelerator transport protocol comprising one connection and interleaving multi-stream is advantageous for data transmission, including better flow and congestion control, preventing head-of-blocking in the reliable data delivery and transfer of a multi-number of assets from web applications.

STAP Handshake and Packet Authentication

10

The accelerator transport protocol uses a cryptographic way of 1-RTT handshake to minimize packet round trip time for a connection establishment using Public Key Infrastructure. Both the accelerator proxy 104-A and the peer proxy 104-B have a key pair for the authentication of its identity in the form of X.509.

15

Each connection established between the nodes of the proxies has a unique connection identifier. The connection identifier used in packet encryption and decryption makes use of the newly created encryption key in the handshake process. This advantageously replaces the need for IP address and port number, typically used in TCP and UDP packet authentication. This way of packet authentication enables mobility in the different networks and multi-path delivery, data channel change among different networks, multi-channel UDP packet delivery (called UDP stitching).

20

Establishing TCP Connection (with overall handshake improvement)

25

When TCP connection is terminated between the proxies, the STAP may establish connections at both ends between source and destination which does without the need to establish a new TCP connection to be established over wireless and high latency networks, thereby resulting in overall handshake improvement.

30

As a result, the connection establishment can be done 2 RTTs earlier than a typical TCP handshake, which is advantageous for applications that require short tail transaction and

frequent connections and disconnections, for instance, IoT sensor data reports and ATMs for banking.

Connection management of TCP and STAP pair

5

The connection 126 established between the nodes and TCP connection may be closed when the accelerator proxy 104-A detects events associated with inactivity or broken connections. Such idle events may include inactivity from the user device, broken TCP connections in the network or broken connection. On detecting such events, the accelerator proxy 104-A closes the connection 126 and cleans the related data structure using the connection map.

10

UDP Data reliability

In the process of selecting packets, the server may redirect the traffic to an accelerated transport protocol, by transporting user data via transport protocol implemented atop UDP.

15

Generally, mapping for the UDP user traffic is managed in a similar manner as TCP user traffic as described in previous sections. It would be appreciated that since the accelerator proxy handles user UDP traffic as packet level, there is a possibility that one value endpoint file descriptor mapped with many stream+STAP connection ID key.

20

As an inherent feature of the UDP, the user UDP application has no reliable feature. Therefore, when accelerating by means of the accelerator proxy having reliable property, the ensuing traffic delivery is more reliable to the destination network.

25

Advantageously, the accelerator proxy 104-A may be configured via the Web UI or management console these services with suitable class of service (CoS) for ensuring secure and real-time delivery towards the destination network.

In addition, the accelerator proxy 104-A can advantageously support Peer-to-Peer (P2P) user application with the capability of proxying service port range under a certain rule mapping filter by the specified source and destination IP.

30

Circular Queue for Handling User Traffic

Referring to Figure 4, there is described the flow of incoming and outgoing user TCP/UDP data transmission. The accelerator proxy 104-A stores and processes the data upon certain
5 condition of I/O operation towards user/destination network.

Incoming user traffic may be intercepted at the accelerator proxy 104-A and multiplexed, and in turn encrypted in the data delivery accelerator system. The encrypted data may be sent via a known stream and connection is established. The peer proxy 104-B decrypts
10 multiplexes and forwards stream data to the established destination based on the connection mapping that had been made.

A similar process for sending user traffic by the peer proxy 104-B may be carried out in the reverse direction. The peer proxy 104-B receives the user traffic, decrypts, demultiplexes,
15 and forwards the user data to the correspondent file descriptor from the <stream+STAP connectionID, file descriptor> connection map which may be previously established.

During the above process, I/O operations happen with socket file descriptors which refer to endpoint connections to user socket, destination socket can be in busy state, the accelerator
20 proxy 104-A stores the ongoing traffic into the circular queue for processing in proper state in later time. The circular buffer maximum size per user connection may be configurable depending on the system memory capacity. The accelerator proxy 104-A may apply auto-tuning technique for this circular buffer queue, which drops later over max size packet to ensure the proper working state of the proxy.

25

CoS (Class of Service for UDP acceleration)

Depending on the needs of the user, some UDP applications may require the service reliability via the accelerator proxy 104-A, while other applications may prefer retaining the
30 nature of UDP connection on delivery.

The accelerator transport protocol enables 2 classes of service for the UDP accelerator, enabling high-reliability delivery, wherein the user payload is delivered in the order received with congestion control mechanism of STAP tunnel. In addition, real-time delivery is

enabled, wherein the user payload is delivered as soon as it is received, which results in loss of packets and there will be no retransmission.

QoS (Quality of Service)

5

There is provided an adaptive congestion control that dynamically limits the bandwidth used by a user associated with IP address. Each user device 102 which utilizes the acceleration mode via the accelerator transport protocol may be configured with a bandwidth limitation associated with its IP address. The proxies may be adapted to identify bandwidth limitation by applying quality of service (QoS), which enables detecting and efficient recovery from loss. The accelerator transport protocol accomplishes the QoS with the 'adaptive congestion control' method for controlling the rate of reading the buffer from socket. The adaptive congestion control is determined using the following formula:

$$15 \quad dl_rate = \min(\min(\text{estimated_stap_bw}, \text{current_dl_rate} * \text{adaptive_factor}), \text{upper_con_capacity})$$

wherein,

- dl_rate: the desired in-coming or out-going speed (bandwidth utilization) in the next phase
- estimated_stap_bw: estimated bandwidth from STAP
- current_dl_rate: the current in-coming or out-going speed (bandwidth utilization)
- adaptive_factor: to minimize the buffer overrun scenarios, it reduces incoming
- upper_con_capacity: configured upper limit in the bandwidth utilization

25

Adaptive factor provides that the read rate from user socket is controlled by the configurable limitation, and the estimated bandwidth availability of the connection, measured during the acceleration phase.

$$\text{Adaptive_factor} = \begin{cases} 2, & \text{rqueue in } [0, \frac{1}{4} T] \\ \frac{1}{2}, & \text{rqueue in } (\frac{1}{4} T, \frac{1}{2} T] \\ \frac{1}{4}, & \text{rqueue in } (\frac{1}{2} T, \frac{3}{4} T] \\ \frac{1}{8}, & \text{rqueue in } (\frac{3}{4} T, T] \\ k/\text{current_dl_rate}, & \text{rqueue} > T \end{cases}$$

Wherein,

- k – minimum constant of read rate,
- T – configurable threshold (in bytes)

5

Migration of Connection ID

With reference to Figure 10, in accordance with an embodiment of the invention, there is disclosed a deployment comprising the accelerator proxy 104-A and the peer proxy 104-B and a Network Address Translation (NAT). The UDP port mapping and binding (4-number-tube; subscriber IP, subscriber port, peer IP, peer port) may be changed due to the NAT service. The accelerator proxy 104-A and/or the peer proxy 104-B may rebind or re-establish the mapping to maintain the lifetime of the connection over time having the same identity. This migration updates the 4-number-tube.

15

The migration process switches the tunnel backhaul from network path 1 (physical interface ETH_2 configured to the wan gateway GW1) to network path 2 (physical interface ETH_3 configured to the wan gateway GW2).

20 Connection 126 identity STAP_ID01 may be established and mapped with the user device 102 via the in-path LAN network. The migration process, which is triggered by a network observation decision at accelerator device, is transparent to the user application (TCP01) since it only changes the tunnel endpoint but still keeps the stream mapping, user payload frame status and the session context of connection identity, which contains the authentication information and related control information.

25

It is to be understood that the above embodiments have been provided 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 scope and ambit of the present invention described herein. It is to be understood that features
5 from one or more of the described embodiments may be combined to form further embodiments.