Technical Disclosure Commons

Defensive Publications Series

December 20, 2018

PATH INFORMATION-DRIVEN QUIC PATH_CHALLENGE TO OPTIMIZE PATH SELECTION IN 5G NETWORKS

Sebastian Jeuk

Gonzalo Salgueiro

David Hanes

PATH INFORMATION-DRIVEN QUIC PATH_CHALLENGE TO OPTIMIZE PATH SELECTION IN 5G NETWORKS

AUTHORS:
Sebastian Jeuk
Gonzalo Salgueiro
David Hanes

ABSTRACT

Proposed herein are techniques that make use of an extended Quick UDP Internet Connection (QUIC) PATH_CHALLENGE frame to perform dynamic path selection operations within a 5G Wi-Fi/cellular network to optimize end-to-end performance. The techniques presented herein expands the capabilities of the current PATH_CHALLENGE function, as defined in the Internet Engineering Task Force (IETF) QUIC drafts. The additional information gathered as part of the PATH_CHALLENGE and the PATH_RESPONSE frame are used to understand path characteristics and dynamically optimize traffic flows accordingly.

DETAILED DESCRIPTION

Quick UDP Internet Connection (QUIC) and the QUIC PATH_CHALLENGE functionality is currently being standardized by the Internet Engineering Task Force (IETF). However, there has been application of the QUIC PATH_CHALLENGE within 5G or the extension of the PATH_CHALLENGE frame to incorporate in-band path metadata.

The techniques presented herein leverage the QUIC PATH_CHALLENGE concept within a 5G environment. The 5G QUIC PATH_CHALLENGE functionality as it works today allows gathering path validity information to help select the most appropriate path. QUIC sends out a PATH_CHALLENGE frame with random payload data to a destination to see if the new path is ready/good. A response is sent back from the remote endpoint using a PATH_RESPONSE frame containing the same random data as the PATH_CHALLENGE frame.

In 5G networks, there is a higher density of cellular and Wi-Fi sites for connectivity. The expectation is that mobile and IoT devices will seamlessly switch between various frequencies and networks for both Wi-FI and cellular. In addition to using QUIC

Published by Technical Disclosure Commons, 2018

PATH_CHALLENGE frames to optimize Wi-Fi and cellular network selection, the techniques presented herein expand the PATH_CHALLENGE frame to incorporate information of the traversed path. As a result, endpoints can use the in-band collected data to find optimal paths for their traffic or trigger switchovers between different paths to optimize performance.

Figure 1, below, shows a PATH_CHALLENGE frame with path relevant information.

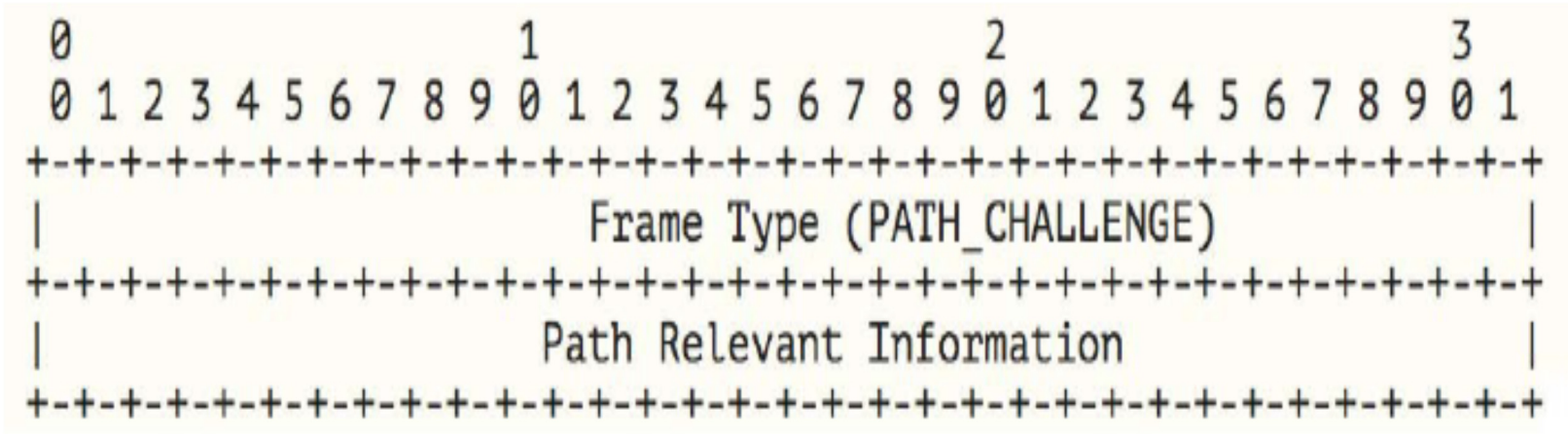

Figure 1

In Figure 1, the path relevant information can be defined similar to that which is collected by in-band operation, administration, and maintenance (iOAM) and, in fact, the PATH_CHALLENGE frame could trigger the same iOAM component to collect relevant information. The following list identifies a set of information that can be inserted into the updated PATH_CHALLENGE frame:

- hop count;
- per-hop device information;
- lowest link bandwidth;
- highest link bandwidth;
- per-hop utilization; and
- end-to-end latency.

It is to be understood that introducing metadata into the PATH_CHALLENGE operation of QUIC is a key element of the techniques presented herein. This introduction of metadata provides QUIC with a way to understand not only whether a path is active or operational, but also the actual characteristics of specific paths. The techniques presented herein include the collection of path information on a per-hop and per-path basis. This information is stored in the PATH_CHALLENGE or PATH_RESPONE header and transmitted back to the source. The source is then able to use the collected information to compute the most optimal path in-band with the actual QUIC connection setup (not relying on outside or additional protocols). For the purpose of 5G, the PATH Verification Offloading system, also proposed herein, can handle the path computations on behalf of the IoT device that often has restricted resources. The PATH Verification Offload system can sit close to the IoT QUIC source to be able to compute the path best to a specific IoT sensor. A possible location is the 5G cell aggregation point, Fog controller, or IoT gateway. Figure 2, below compares and highlights the detailed path view that becomes available with the proposed enhancements to the PATH-CHALLENGE.

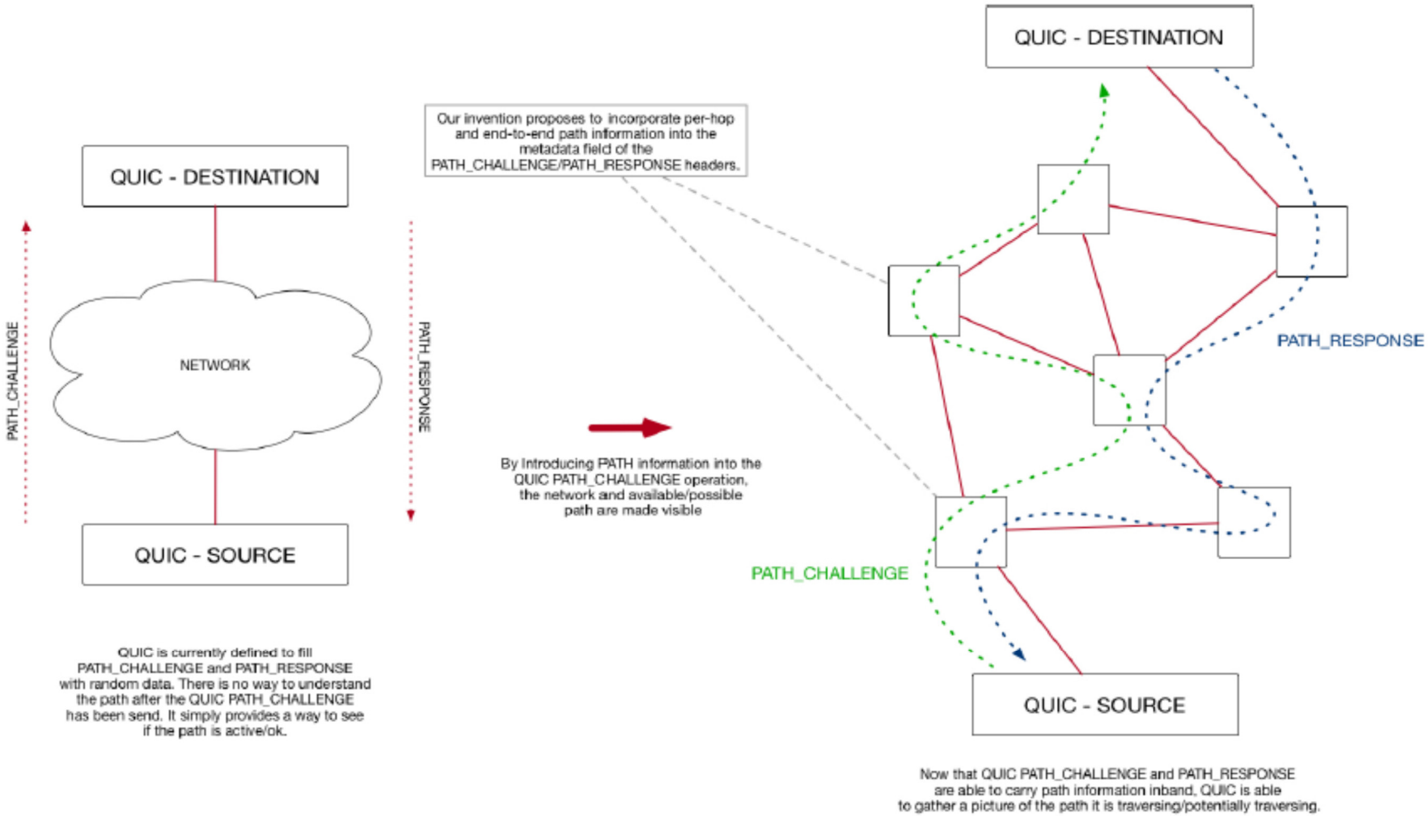
QUIC - DESTINATION
PATH_CHALLENGE
NETWORK
PATH_RESPONSE
QUIC - SOURCE
QUIC is currently defined to fill PATH_CHALLENGE and PATH_RESPONSE with random data. There is no way to understand the path after the QUIC PATH_CHALLENGE has been send. It simply provides a way to see if the path is active/ok.
Our invention proposes to incorporate per-hop and end-to-end path information into the metadata field of the PATH_CHALLENGE/PATH_RESPONSE headers.
By Introducing PATH information into the QUIC PATH_CHALLENGE operation, the network and available/possible path are made visible
QUIC - DESTINATION
PATH_RESPONSE
PATH_CHALLENGE
QUIC - SOURCE
Now that QUIC PATH_CHALLENGE and PATH_RESPONSE are able to carry path information inband, QUIC is able to gather a picture of the path it is traversing/potentially traversing.

Figure 2

Here, it is noted that the techniques presented herein introduce a novel way to understand path characteristics within a QUIC setup phase. This is currently not possible as QUIC only relies on random data unrelated to the actual path it traverses.

Referring to Figure 2, the destination to which the PATH_CHALLENGE frame was sent updates the PATH_RESPONSE frame with the information collected through the PATH_CHALLENGE frame. The origin of the PATH_CHALLENGE will use this information when it is received through the PATH_RESPONSE frame to perform path calculations to be able to select the most optimal path.

In one embodiment, PATH Verification Offloading is proposed. PATH Verification Offloading is a mechanism that is able to proxy a PATH_CHALLENGE frame and the consecutive path calculations for devices/things in the network that have very little processing power. For example, in a 5G cell, IoT sensors might send out a PATH_CHALLENGE frame to determine good/ready paths to a destination within the core of the network. Here, resource usage on a constrained device for calculating path optimization is not feasible. Proposed herein is to have a central device that traffic passes on the way from the 5G cell to the core (or alternatively a proxy) to perform the PATH_CHALLENGE operations on behalf of the IoT sensor.

In certain PATH Verification Offloading embodiments, the proxy splits up the PATH_CHALLENGE operation into two sections, (1) IoT sensor to 5G cell entry point/QUIC PATH_CHALLENGE proxy and (2) QUIC PATH_CHALLENGE proxy and destination. While there are circumstances where the calculation of finding the optimal path for section (1) is necessary, processing collected path information is minimal (i.e., selecting Wi-Fi over cellular or different frequencies within a cell). However, the PATH_CHALLENGE for the section (2) is more resource intense, as it can entail a larger set of hops and hence a larger set of information that needs processing.

In another embodiment, the techniques presented herein are extended by introducing a mechanism that provides grouped path verification and optimization offloading for IoT sensors in 5G networks. Figure 3, below, illustrates an overview of this embodiment.

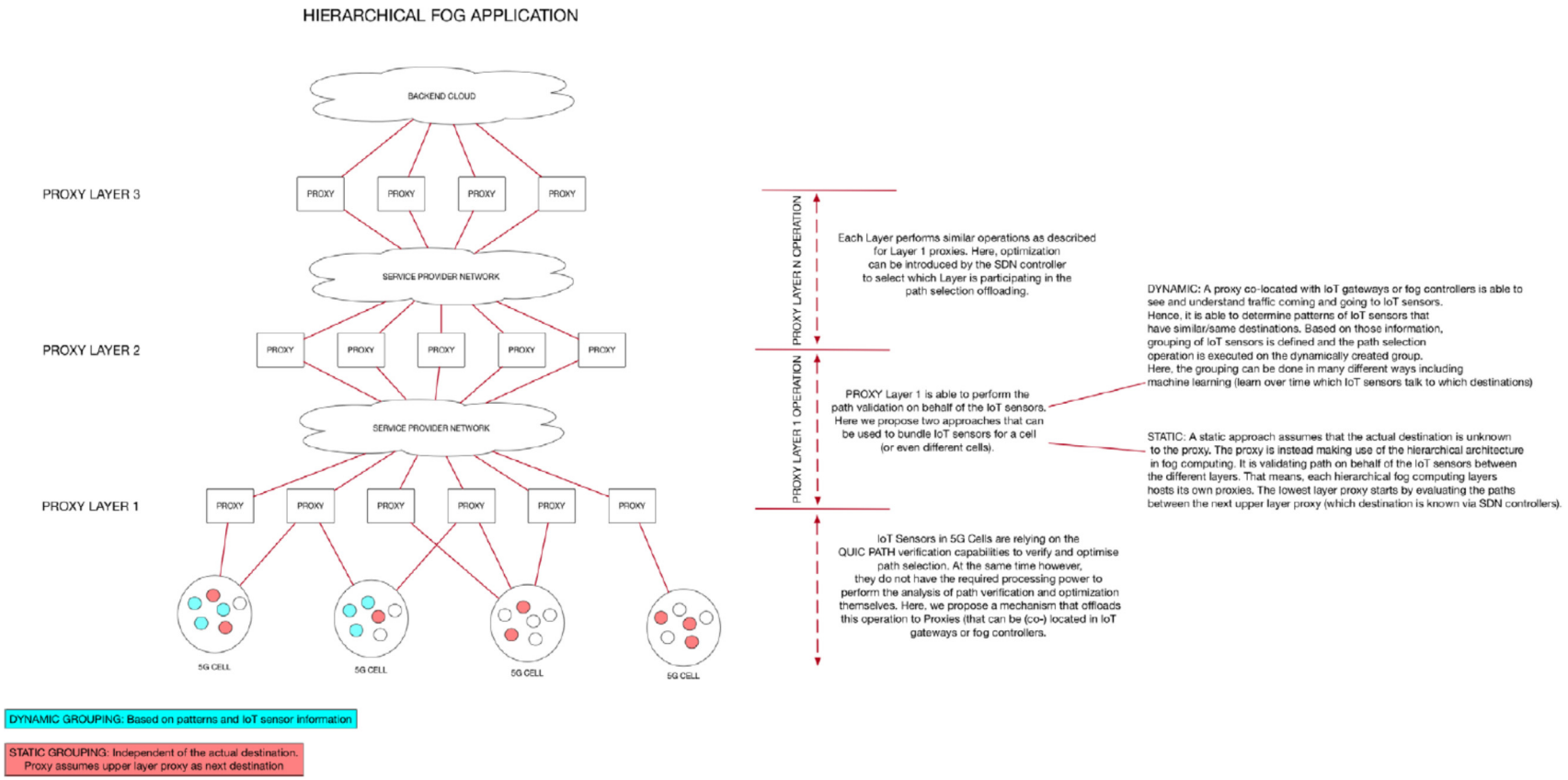
HIERARCHICAL FOG APPLICATION
BACKEND CLOUD
PROXY LAYER 3
PROXY
SERVICE PROVIDER NETWORK
PROXY LAYER 2
SERVICE PROVIDER NETWORK
PROXY LAYER 1
5G CELL
PROXY LAYER N OPERATION
Each Layer performs similar operations as described for Layer 1 proxies. Here, optimization can be introduced by the SDN controller to select which Layer is participating in the path selection offloading.
PROXY LAYER 1 OPERATION
PROXY Layer 1 is able to perform the path validation on behalf of the IoT sensors. Here we propose two approaches that can be used to bundle IoT sensors for a cell (or even different cells).
DYNAMIC: A proxy co-located with IoT gateways or fog controllers is able to see and understand traffic coming and going to IoT sensors. Hence, it is able to determine patterns of IoT sensors that have similar/same destinations. Based on those information, grouping of IoT sensors is defined and the path selection operation is executed on the dynamically created group. Here, the grouping can be done in many different ways including machine learning (learn over time which IoT sensors talk to which destinations)
STATIC: A static approach assumes that the actual destination is unknown to the proxy. The proxy is instead making use of the hierarchical architecture in fog computing. It is validating path on behalf of the IoT sensors between the different layers. That means, each hierarchical fog computing layers hosts its own proxies. The lowest layer proxy starts by evaluating the paths between the next upper layer proxy (which destination is known via SDN controllers).
IoT Sensors in 5G Cells are relying on the QUIC PATH verification capabilities to verify and optimise path selection. At the same time however, they do not have the required processing power to perform the analysis of path verification and optimization themselves. Here, we propose a mechanism that offloads this operation to Proxies (that can be (co-) located in IoT gateways or fog controllers.
DYNAMIC GROUPING: Based on patterns and IoT sensor information
STATIC GROUPING: Independent of the actual destination. Proxy assumes upper layer proxy as next destination

Figure 3

As shown in Figure 3, in a hierarchical fog environment the techniques presented herein use proxies deployed at the different levels. The level closest to the endpoints is performing the path verification/optimization on-behalf of the IoT sensors to not utilize already restricted resources. To be able to perform this in bulk, the techniques presented herein also use a new mechanism on the proxy that allows grouping endpoints. Here, two different approaches are applicable, dynamic and static. In case of dynamically grouping IoT sensors, already existing mechanisms (e.g., pattern recognition or machine learning) can be used, over time, to understand common destinations of different endpoints scattered across different 5G cells. After the grouping is defined, the proxy performs the typical QUIC path verification on behalf of the group (or domain) of IoT sensors and is optimizing path selection accordingly.

Secondly, the techniques presented herein introduce a static approach whereby sensors passing a proxy at a specific layer within the hierarchical fog architecture are grouped together and path verification and selection is performed between the different layers. Here, the actual destination for traffic originating from 5G endpoints can be different. It is replaced by the destination of the next upper layer proxy and path verification and selection is performed between the two proxies on the different layers. This operation can be repeated at different layers depending on deployment size or other demands. Here, the techniques presented herein introduce a SDN controller capable of maintaining a broader view on the proxies deployed within the network that is used to define how layers (and the associated proxies) are selected. Additionally, to prevent a single proxy from being selected (preventing single point of failure for the performed path selection/optimization) anycast IPs can be used on all proxies of a specific layer, allowing the closest to respond.

In another embodiment, PATH_CHALLENGE results can be stored in the network (using previously mentioned proxy or another system with database capabilities) with a time-based decay factor. These results can be used when a new device connects or roams into a 5G cell so that this device can be directed to the most optimal frequency, network, and medium, whether it be Wi-Fi or cellular.

The time decay mechanism will work as follows. The most recent PATH_CHALLENGE results captured from a recent connection has a higher validity than

results from a few seconds or minutes prior. So, as soon as results are captured a decay timer starts so that these results are less meaningful when making decisions about optimal connections for new devices. If, for example, hundreds of devices move through a 5G area, there will be many PATH_CHALLENGE results available. From this data the 5G network could also steer premium customers to better connections. Furthermore, this constant influx of PATH-CHALLENGE data could be correlated with usage, user, and time of day trends and used as learning for an artificial intelligence (AI) system to reliably predict the best paths for devices in a 5G area.

In summary, the techniques presented herein add extended capabilities to the QUIC PATH_CHALLENGE message to provide path optimization and verification for IoT devices in 5G environments. Additionally, this the techniques presented herein propose the additional novel emoluments of PATH Verification Offloading, grouped path verification and offloading, and a time-based decay mechanism for storing and prioritizing paths for newly connected or roaming devices. The novel elements presented herein include:

- Expansion of the PATH_CHALLENGE and PATH_RESPONSE frames to incorporate in-band gathered path information on a per-hop basis;
- Dynamically optimization of 5G network path selections (Wi-Fi vs. cellular, different frequencies, different paths within the 5G NG core) based on path information received within the PATH_RESPONSE frame;
- Dynamic re-use and time-based decay of optimal paths information;
- PATH Verification Offloading to keep computational overhead at IoT devices to a minimum; and
- Grouped path verification and optimization offloading for IoT sensors in 5G networks.