2021-07-16 JANOG48 ライトニングトーク

QUICとNATと

Yuya Kawakami, SDN Tech Lead, Enterprise Cloud 2.0

NTT Communications

はじめに

- 個人の**「自主的な研究の成果の発表」**だと受け止めてください
- QUICやNATの専門家ではありません
- 誤りやコメントがあれば是非ご連絡ください、事後資料で訂正します
- 時間が足りないので爆速で話します

きっかけ

Twitterで流れてきたつぶやき

うええ、ルーターのNATテーブルを圧迫してるの、LAN内部に設置したDNSサーバーだけかと思ったら、QUICか！どうりで、googleとかQUICに対応してるサイト使うとトラブル起きる訳だ。。

1:22 PM · Jun 7, 2021 · twitcle plus

450 Retweets **25** Quote Tweets **836** Likes

ですよねー

この発表を聞いて欲しい人

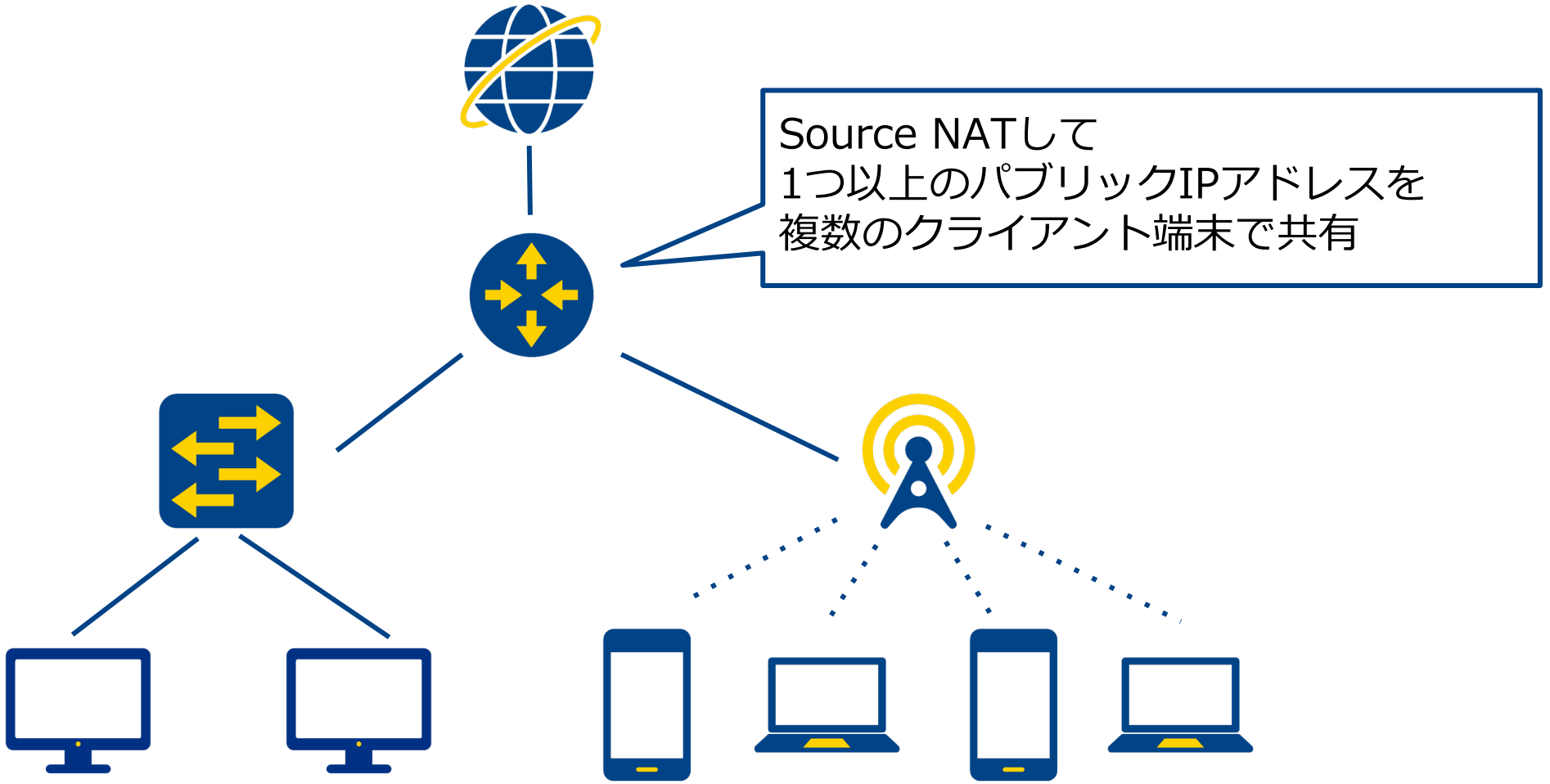
ぜひ聞いて欲しい人

- 中小規模・SOHOのネットワークを構築・運用している人
- カンファレンスやイベントのネットワークを構築・運用している人
- ご家庭のルータを管理・運用する人

もう知ってるでしょという人

- ブロードバンドルータメーカー
- CGNをやる規模のキャリアISPやモバイル事業者

こういうネットワークを作る人



QUICがRFC9000になりました

QUIC: A UDP-Based Multiplexed and Secure Transport

[Search] [txt|html|xml|pdf|bibtex] [Tracker] [WG] [Email] [Diff1] [Diff2]

From: draft-ietf-quic-transport-34 Proposed Standard
Internet Engineering Task Force (IETF) J. Iyengar, Ed.
Request for Comments: 9000 Fastly
Category: Standards Track M. Thomson, Ed.
ISSN: 2070-1721 Mozilla
May 2021

QUIC: A UDP-Based Multiplexed and Secure Transport

Abstract

This document defines the core of the QUIC transport protocol. QUIC provides applications with flow-controlled streams for structured communication, low-latency connection establishment, and network path migration. QUIC includes security measures that ensure confidentiality, integrity, and availability in a range of deployment circumstances. Accompanying documents describe the integration of TLS for key negotiation, loss detection, and an exemplary congestion control algorithm.

QUICはどういうプロトコルなのか？

NATに関係するネットワーク運用者の観点では

1. HTTP/3のためにUDP443番で使われている
2. UDPを使うトランスポートのプロトコルである
3. 最低限以外のヘッダとペイロードがTLSで暗号化されている
4. 複数のHTTPストリームを重畳できる

なぜQUICでNATテーブルが溢れやすいのか

1. HTTP/3のためにUDP443番で使われている
   - ○ Googleなどのサービスやブラウザで既にデプロイされている
   - ○ HTTPのトラフィックはインターネットトラフィックの大半を占める

2. UDPを使うトランスポートのプロトコルである
   - ○ TCPの場合はFINでコネクションの終了を検知してマッピングを消すことができる
   - ○ UDPはコネクションの終了が分からないので能動的にマッピングを消すことができない
   - ○ 途中のルータは**無通信時間を基にコネクション終了を判断するしかない**

3. 最低限以外のQUICヘッダとペイロードがTLSで暗号化されている
   - ○ **コネクションの終了を知らせるフレームも暗号化されている**ため途中のルータでは
     コネクションの終了を知ることはできない
   - ○ そもそもプロトコル仕様上クローズせずにコネクションが終了することもある

一方で溢れにくくなる要素もある

4. 複数のHTTPストリームを重畳できる
   - ○ 従来複数のTCPコネクションを使用していたものが1本のUDPにまとめられる
   - ○ 1つのWebサイトを開くときに必要な**マッピングの数は減る**はず

NATのUDPマッピングのタイムアウトの要件

RFC4787 (BCP)によるとUDPのタイムアウトを**2分未満にしてはならず**、
デフォルト値としては**5分以上が推奨**される

REQ-5: **A NAT UDP mapping timer MUST NOT expire in less than two minutes**, unless REQ-5a applies.

a) For specific destination ports in the well-known port range (ports 0-1023), a NAT MAY have shorter UDP mapping timers that are specific to the IANA-registered application running over that specific destination port.

b) The value of the NAT UDP mapping timer MAY be configurable.

c) A default value of **five minutes or more for the NAT UDP mapping timer** is RECOMMENDED.

NATのUDPマッピングのタイムアウトの例外

とはいえ長くしすぎると短い時間でコネクションを終了するアプリケーションで
マッピングが埋まるので、**特定の宛先ポートのタイムアウトは短くして良い**

➡ **DNSのタイムアウトを短くする**実装は多い

a) Some UDP protocols using UDP use very short-lived connections. There can be very many such connections; keeping them all in a connections table could cause considerable load on the NAT. **Having shorter timers for these specific applications is, therefore, an optimization technique**. It is important that the shorter timers applied to specific protocols be used sparingly, and only for protocols using well-known destination ports that are known to have a shorter timer, and that are known not to be used by any applications for other purposes.

QUICの実装にあたって

Manageability of the QUIC Transport Protocol ではRFC4787の2分の値を参照しつつ、フィールドでは**実際には30秒や60秒でタイムアウトしてしまう**ことに言及している

> [RFC4787] requires a network state timeout that is not less than 2 minutes for most UDP traffic. However, in practice, **a QUIC endpoint can experience lower timeouts, in the range of 30 to 60 seconds**.

QUICの実装の要件

RFC9000 ではタイムアウトを防ぐために**30秒ごとにパケットを送信することが必要である**と述べられている

> Though REQ-5 in [RFC4787] recommends a 2-minute timeout interval, experience shows that sending packets **every 30 seconds is necessary** to prevent the majority of middleboxes from losing state for UDP flows [GATEWAY].

QUICの実装例

Google ChromeのQUICの実装 “QUICHE” では
15秒おきにPING Frameを送信してNATのタイムアウトを防いでいる

master/quic/core/quic_constants.h

```
// Default ping timeout.
const int64_t kPingTimeoutSecs = 15;  // 15 secs.
```

master/quic/core/quic_connection.cc

```
void QuicConnection::SetPingAlarm() {
  if (perspective_ == Perspective::IS_SERVER &&
      initial_retransmittable_on_wire_timeout_.IsInfinite()) {
    // The PING alarm exists to support two features:
    // 1) clients send PINGs every 15s to prevent NAT timeouts,
    // 2) both clients and servers can send retransmittable on the wire PINGs
    // (ROWP) while ShouldKeepConnectionAlive is true and there is no packets in
    // flight.
    return;
  }
```

ブロードバンドルータのNATの実装

国内でよくNATルータとして使われる機器のUDPマッピングの消去方式は
タイマー方式で、**デフォルトのタイムアウト値は300秒が多い**(RFC4787通り)

➡ QUICのためのマッピングが300秒残り続ける

ルータ	方式	タイマーのデフォルト値	設定可否	NATセッション数上限
A	タイマー	300秒	可	2048
B	不明	不明	不可	2048
C	不明	不明	不可	不明
D	タイマー	180秒	不可	不明
E	タイマー	900秒	可	4096-65534
F	タイマー	300秒	可	数千以上(メモリ次第)
G	タイマー	300秒	可	数千以上(メモリ次第)

※上のほうが御家庭用、下のほうが誤家庭用

GCPのCloud NATの実装

GCPのCloud NAT ではTCPについては再利用禁止期間の言及があるが、
UDPについてはタイマーや再利用に関する言及がない

> Cloud NAT ゲートウェイが TCP 接続を終了した後、ゲートウェイが同じ NAT 送信元 IP アドレスと送信元ポートのタプルを同じ宛先（宛先 IP アドレス、宛先ポート、プロトコル）に再利用する前に、Google Cloud は 2 分間の遅延を適用します。

ポートの再利用禁止期間

CGN(Carrier-Grade NAT) の要件を示した RFC6888 では一部の例外を除いて
ポートの再利用禁止期間を2分と定めている

例外としてトラックされた**TCPポートは即時再利用することができる**

> REQ-8: Once an external port is deallocated, **it SHOULD NOT be reallocated to a new mapping until at least 120 seconds have passed**, with the exceptions being:
>
> a. If the CGN tracks TCP sessions (e.g., with a state machine, as in Section 3.5.2.2 of [RFC6146]), **TCP ports MAY be reused immediately**.

※シングルテナンシー用途のブロードバンドルータがこのRFCに従うべきかどうかは別の問題です

対策になりえるのは？

1. NATのUDPタイマーを短くする？
2. IPv6を使う？
3. LRU方式を実装しているNATを使う？
~~4. UDP443を塞いでTCPにフォールバックさせる？~~

QUICの推奨タイマー値

Manageability of the QUIC Transport Protocol では
QUICは30秒のタイムアウトで動くように作られているが、
それでも**最低2分のタイムアウト値**にするように求めている

➡ **2分(120秒)に減らすことは一時的な対処としては有効かもしれないが…**

> Even though QUIC has explicitly been designed to tolerate NAT rebindings, decreasing the NAT timeout is not recommended, as it may negatively impact application performance or incentivize endpoints to send very frequent keep-alive packets.
>
> The recommendation is therefore that, even when lower state timeouts are used for other UDP traffic, **a state timeout of at least two minutes ought to be used for QUIC traffic**.

SPI (Stateful Packet Inspection)

NATのタイムアウトが300秒でも、IPv6を使っていてNATが関係ない場合でも、
実際にはSPIを行うStateful Firewall等の**動的フィルタのエントリーが埋まったり、タイムアウトでサーバ側からのパケットが届かなくなったり**することがあるので、
このタイムアウト値も気にする必要がある

ルータ	方式	タイマーのデフォルト値	設定可否
A	タイマー	300秒	可
B	不明	不明	不可
C	タイマー	**30秒**	可
E	タイマー	**30秒**	可
F	タイマー	なし	可

※OpenStackのSecurity GroupsなどもSPIだよ！

LRU (Least Recently Used)

NATテーブルが溢れたときに、最も古いエントリーを消す方式

➡ この方式にもいくつかの問題がある

- 実装しているルータはそれほど多くない（個人の感想です）
- RFC4787に従って最低2分のタイムアウトを守っている場合は効果が薄い
- 守っていないと**ポートの再利用間隔が2分未満になる**問題がある
  - ○ 特にマルチテナンシー環境では十分に注意する必要がある
- TCPとUDPでマッピングテーブルを共有している場合、Keepaliveの長いTCPベースのプロトコルが割を食う可能性がある
  - ○ Google Cloud MessagingはKeepaliveが15分
- LRUで消されないように新しいプロトコル同士でKeepaliveを短くする競争が激化する可能性もある

まとめ

実施可能な具体的な解決策は状況によるので

1. QUICの特性を理解しておこう
2. 使用するルータのNATのUDPマッピングの挙動を理解しておこう
3. SPIのタイマーも気にしておこう

新しい世代のプロトコルのことを
しっかり理解して
より良いインターネットにしよう

時間足りないので

JANOG49で一緒にプログラム応募して話してくれる人とかいたら是非

- ブロードバンドメーカでQUIC対応してる方
- ネットワーク構築している人でQUICで悩んでる方
- ミドルボックスの開発をしていてQUICについて考えないといけない方

謝辞

夜中まで議論にお付き合いいただいて色々教えていただいた @kazuho さん、
ありがとうございました！