QUIC-LB

draft-duke-quic-load-balancers-06
Martin Duke
F5 Networks
IETF 106
Motivation

Defend against:
- Linkability
- Attack against single server
Motivation

Defend against:
Linkability
Attack against single server

DCID

DDoS/Syncookie Offload

Load Balancer

F(DCID)

DCID

SCID
NEW_CONNECTION_ID

Server
HW Crypto

Server
HW Crypto

Server
HW Crypto

Server
HW Crypto

Server
HW Crypto

Server
HW Crypto
Changes...

“QUIC tolerates no mediation by L7 middleboxes”

“QUIC tolerates mediation by *explicitly trusted* L7 middleboxes”
Perfect Linkability
Perfect Unlinkability
Security

Attacks:
(1) Obtain server mapping
(2) Break LB routing
Configuration Schema

uint2 config_rotation_bits;
enum { in_band_config, out_of_band_config } config_method;
select (config_method) {
    case in_band_config: uint64 config_token;
    case out_of_band_config: null;
} config-method

boolean first_octet_encodes_cid_length;
enum { none, non_shared_state, shared_state } retry_service;
select (retry_service) {
    case none: null;
    case non_shared_state: null;
    case shared_state: uint8 key[16];
} retry_service_config;

enum { none, plaintext, obfuscated, stream_cipher, block_cipher } routing_algorithm;
Configuration Schema (cont’d)

```c
select (routing_algorithm) {
    case none: null;
    case plaintext: struct {
        uint8 server_id_length; /* 1..19 */
        uint8 server_id[server_id_length];
    } plaintext_config;
    case obfuscated: struct {
        uint8 routing_bit_mask[19];
        uint16 divisor; /* Must be odd */
        uint16 modulus; /* 0..(divisor - 1) */
    } obfuscated_config;
    case stream_cipher: struct {
        uint8 nonce_length; /* 8..16 */
        uint8 server_id_length; /* 1..(19 - nonce_length) */
        uint8 server_id[server_id_length];
        uint8 key[16];
    } stream_cipher_config;
    case block_cipher: struct {
        uint8 server_id_length;
        uint8 zero_padding_length; /* 0..(16 - server_id_length) */
        uint8 server_id[server_id_length];
        uint8 key[16];
    } block_cipher_config;
} routing_algorithm_config;
```
In-band configuration

“We would never use this”

“Keep it with a few tweaks”

“Find ‘something’ that exists today and use it instead”

“Put it in a different draft”
Discussion Points

• Linkability decisions are made by the server but affect the client. Transport parameter to communicate linkability?
• Retry services are fundamentally version specific but CID parts are not – separate draft?
• Is OCID actually any easier than crypto versions?
• Engagement with cloud load balancer vendors
Next Steps

• Move for adoption
• Start interop of algorithms