# CoAP/DTLS feedback

Running LWM2M in a IPv4/NATed wireless network

# NAT timeout

On cellular network it can be as bad as 20sec

Combine it with packet loss: welcome to hell!

# Trying to send notification stream

Device strategy for sending consecutive notifications:

Open a DTLS session

Send the notification, wait for the ACK

20 sec after you want to send a second notification, you can:

- abbreviated handshake or new handshake (850bytes in PSK, in ECDSA you die)

- try to send the notification and if you never receive the ACK, then try #1

# Forbidden by RFC7641!

All notifications resulting from a GET
request with an Observe Option MUST be returned within the same epoch of the same connection as the request.

# Request/Response Matching in general

RFC 7252 (CoAP) Section 1.2:

"Endpoint

An entity participating in the CoAP protocol. Colloquially, an endpoint lives on a "Node", although "Host" would be more consistent with Internet standards usage, and is **further identified by transport-layer multiplexing information that can include a UDP port number and a security association** (Section 4.1)."

Section 4.1:

"A CoAP endpoint is the source or destination of a CoAP message. The **specific definition of an endpoint depends on the transport being used for CoAP**. For the transports defined in this specification, the endpoint is identified depending on the security mode used (see Section 9): **With no security, the endpoint is solely identified by an IP address and a UDP port number**. With **other security modes, the endpoint is identified as defined by the security mode**."

# ...and with DTLS

Section 9.1.1:

"The following rules are **added** for matching an Acknowledgement message or Reset message to a Confirmable message, or a Reset message to a Non-confirmable message: The DTLS **session MUST be the same**, and the **epoch MUST be the same.**"

Section 9.1.2:

"The following rules are **added** for matching a response to a request: The DTLS **session MUST be the same**, and the **epoch MUST be the same**."

# What this means in Practice

Assuming that "in addition" means "in addition to matching source - destination IP address:port", then

- any notification received after a CoAP server's IP:port has changed, MUST be discarded.
- all observations need to be re-established after a CoAP server's IP:port has changed.

With constrained devices on NATed IPv4 networks, every time a device wakes up after some time

- the device needs to perform an (abbreviated) handshake to create a new DTLS session or re-associate the existing DTLS session with its new IP:port
- the CoAP client needs to **re-establish all observations** on the device

# Potential Solution

https://tools.ietf.org/html/draft-fossati-tls-iot-optimizations-00#section-4.2

‘Connection ID’ on the DTLS packet

DTLS 1.3 draft?

# LWM2M Object observation

If you observe an object level like /9 (application), every time a resource change you are supposed to send the whole list of objects again.

Device to Cloud notification are a common pattern (telemetry, configuration sync)

Implementers already circumvent this limitation (sending only changes resources)

Would be nice to have “patch” notifications

# SMS wakeup fun

You have a device with a phone number, you send a binary SMS.

Doesn’t work.. The end SMS is encoded in 7bits

You discover your device is roaming in Ecuador,

You do a 3 party meeting with the home operator, the roaming operator and you to find who scramble the SMS

Then you stop using binary SMS and encode it as Base64

# DTLS over SMS

For using DTLS over SMS you need a two way communication

You start sending a lot of SMS for initiating DTLS handshake

Operators flag you as spammer and start throttling you

Don’t use DTLS, don’t use two way SMS

COSE/OSCOAP?

# Mobile Network Operator only

Two way and binary SMS are impossible to use in production if you are not an MNO

And without too much roaming…

We need another secure and standard way to wake-up an offline device

# High scalability CoAP/DTLS

Not to have the maximum performance for a single machine

Be able to add machine to a cluster of CoAP client/server machines

https://github.com/eclipse/leshan/wiki/Cluster

# DTLS, Observation and clustering

A device initiate a communication with a LWM2M server

Open a DTLS session, this server is owning the session key

The front load-balancer must send all the DTLS packet of this session to the same server

Based on IP source/port

NAT is kicking in again!