

6TiSCH
Internet-Draft
Intended status: Standards Track
Expires: April 22, 2014

P. Thubert, Ed.
Cisco
T. Watteyne
Linear Technology
RA. Assimi
Centro
November 18, 2013

**An Architecture for IPv6 over the TSCH mode of IEEE 802.15.4e
draft-ietf-6tisch-architecture-00**

Abstract

This document presents an architecture for an IPv6 Multi-Link subnet that is composed of a high speed powered backbone and a number of IEEE802.15.4e TSCH wireless networks attached and synchronized by Backbone Routers. Route Computation may be achieved in a centralized fashion by a Path Computation Element, in a distributed fashion using the Routing Protocol for Low Power and Lossy Networks, or in a mixed mode. The Backbone Routers perform proxy Neighbor Discovery operations over the backbone on behalf of the wireless device, so they can share a same subnet and appear to be connected to the same backbone as classical devices.

Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC 2119](#) [RFC2119].

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of [BCP 78](#) and [BCP 79](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on April 22, 2014.

Copyright Notice

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](#) and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the [Trust Legal Provisions](#) and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1.	Introduction	2
2.	Terminology	3
3.	Applications and Goals	4
4.	Overview and Scope	4
5.	Communication Paradigms and Interaction Models	7
6.	Forwarding Models	8
6.1.	Track Forwarding	8
6.1.1.	Transport Mode	9
6.1.2.	Tunnel Mode	9
6.1.3.	Tunnel Metadata	10
6.2.	Fragment Forwarding	11
6.3.	IPv6 Forwarding	12
7.	TSCH and 6top	12
7.1.	6top	12
7.2.	Network Synchronization	13
7.3.	Slotframes and Priorities	14
7.4.	Packet Marking and Handling	14
8.	Schedule Management Mechanisms	14
8.1.	Minimal Static Scheduling	14
8.2.	Neighbor-to-Neighbor Scheduling	15
8.3.	Remote Monitoring and Schedule Management	15
8.4.	Hop-by-hop Scheduling	16
9.	Centralized vs. Distributed Routing	16
10.	IANA Considerations	17
11.	Security Considerations	17
12.	Acknowledgements	17
13.	References	17
13.1.	Normative References	17
13.2.	Informative References	18
13.3.	External Informative References	19
	Authors' Addresses	19

[1.](#) Introduction

The emergence of radio technology enabled a large variety of new

types of devices to be interconnected, at a very low marginal cost compared to wire, at any range from Near Field to interplanetary distances, and in circumstances where wiring would be less than

practical, for instance rotating devices.

At the same time, a new breed of Time Sensitive Networks is being developed to enable traffic that is highly sensitive to jitter and quite sensitive to latency. Such traffic is not limited to voice and video, but also includes command and control operations such as found in industrial automation or in-vehicle sensors and actuators.

At IEEE802.1, the "Audio/Video Task Group", was renamed TSN for Time Sensitive Networking to address Deterministic Ethernet. The IEEE802.15.4 Medium access Control (MAC) has evolved with IEEE802.15.4e that provides in particular the Timeslotted Channel Hopping (TSCH) mode for industrial-type applications.

Though at a different time scale, both standards provide Deterministic capabilities to the point that a packet that pertains to a certain flow crosses the network from node to node following a very precise schedule, as a train that leaves intermediate stations at precise times along its path. With TSCH, time is formatted into timeslots, and an individual timeslot is allocated to unicast or broadcast communication at the MAC level. The time slotted operation reduces collisions, saves energy, and enables to more closely engineer the network for deterministic properties. The channel hopping aspect is a simple and efficient technique to combat multipath fading and external interference (for example by WiFi emitters).

This document presents an architecture for an IPv6 Multi-Link subnet that is composed of a high speed powered backbone and a number of IEEE802.15.4e TSCH wireless networks attached and synchronized by backbone routers. Route Computation may be achieved in a centralized fashion by a Path Computation Element (PCE), in a distributed fashion using the Routing Protocol for Low Power and Lossy Networks (RPL), or in a mixed mode. The Backbone Routers perform proxy IPv6 Neighbor Discovery (ND) operations over the backbone on behalf of the wireless devices, so they can share a same IPv6 subnet and appear to be connected to the same backbone as classical devices. Timeslots and other device resources are managed by an abstract Network Management Entity (NME) that may cooperate with the PCE in order to minimize the interaction with and the load on the constrained device.

2. Terminology

Readers are expected to be familiar with all the terms and concepts that are discussed in "Neighbor Discovery for IP version 6" [[RFC4861](#)], "IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs): Overview, Assumptions, Problem Statement, and Goals" [[RFC4919](#)], Neighbor Discovery Optimization for Low-power and Lossy Networks [[RFC6775](#)] and "Multi-link Subnet Support in IPv6" [I-D.ietf-

ipv6-multilink-subnets].

Readers may benefit from reading the "RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks" [[RFC6550](#)] specification; "Multi-Link Subnet Issues" [[RFC4903](#)]; "Mobility Support in IPv6" [[RFC6275](#)]; "Neighbor Discovery Proxies (ND Proxy)" [[RFC4389](#)]; "IPv6 Stateless Address Autoconfiguration" [[RFC4862](#)]; "FCFS SAVI: First-Come, First-Served Source Address Validation Improvement for Locally Assigned IPv6 Addresses" [[RFC6620](#)]; and "Optimistic Duplicate Address Detection" [[RFC4429](#)] prior to this specification for a clear understanding of the art in ND-proxying and binding.

The draft uses terminology defined or referenced in [I-D.palattella-6tisch-terminology], [[I-D.chakrabarti-nordmark-6man-efficient-nd](#)], [[I-D.roll-rpl-industrial-applicability](#)], [[RFC5191](#)] and [[RFC4080](#)].

The draft also conforms to the terms and models described in [[RFC3444](#)] and [[RFC5889](#)] and uses the vocabulary and the concepts defined in [[RFC4291](#)] for the IPv6 Architecture.

3. Applications and Goals

The architecture derives from existing industrial standards for Process Control by its focus on Deterministic Networking, in particular with the use of the IEEE802.15.4e TSCH MAC [[IEEE802154e](#)] and the centralized PCE. This approach leverages the TSCH MAC benefits for high reliability against interference, low-power consumption on deterministic traffic, and its Traffic Engineering capabilities. Deterministic Networking applies in particular to open and closed control loops, as well as supervisory control flows and management.

An incremental set of industrial requirements are addressed with the addition of an autonomic and distributed routing operation based on RPL. These use cases include plant setup and decommissioning, as well as monitoring of lots of lesser importance measurements such as corrosion and events. RPL also enables mobile use cases such as mobile workers and cranes.

A Backbone Router is included in order to scale the factory plant subnet to address large deployments, with proxy ND and time synchronization over a high speed backbone.

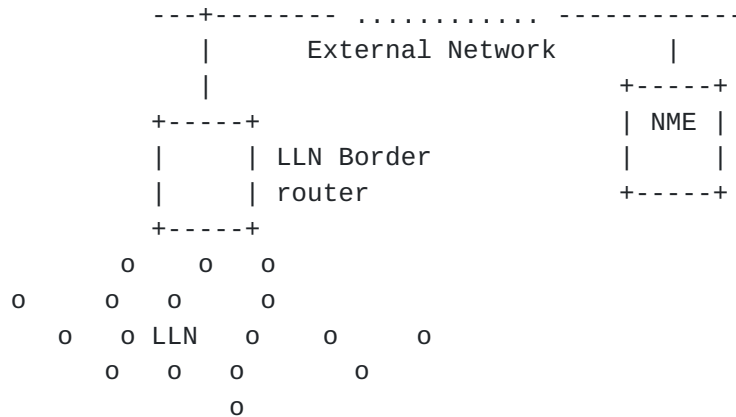
The architecture also applies to building automation that leverage RPL's storing mode to address multipath over a large number of hops, in-vehicle command and control that can be as demanding as industrial applications, commercial automation and asset Tracking with mobile scenarios, home automation and domotics which become more reliable and thus provide a better user experience, and resource management (energy, water, etc.).

4. Overview and Scope

Thubert, Watteyne & AssimExpires April 22, 2014

[Page 4]

The scope of the present work is a subnet that, in its basic configuration, is made of a IEEE802.15.4e Timeslotted Channel Hopping (TSCH) [[I-D.watteyne-6tisch-tsch](#)] MAC Low Power Lossy Network (LLN).

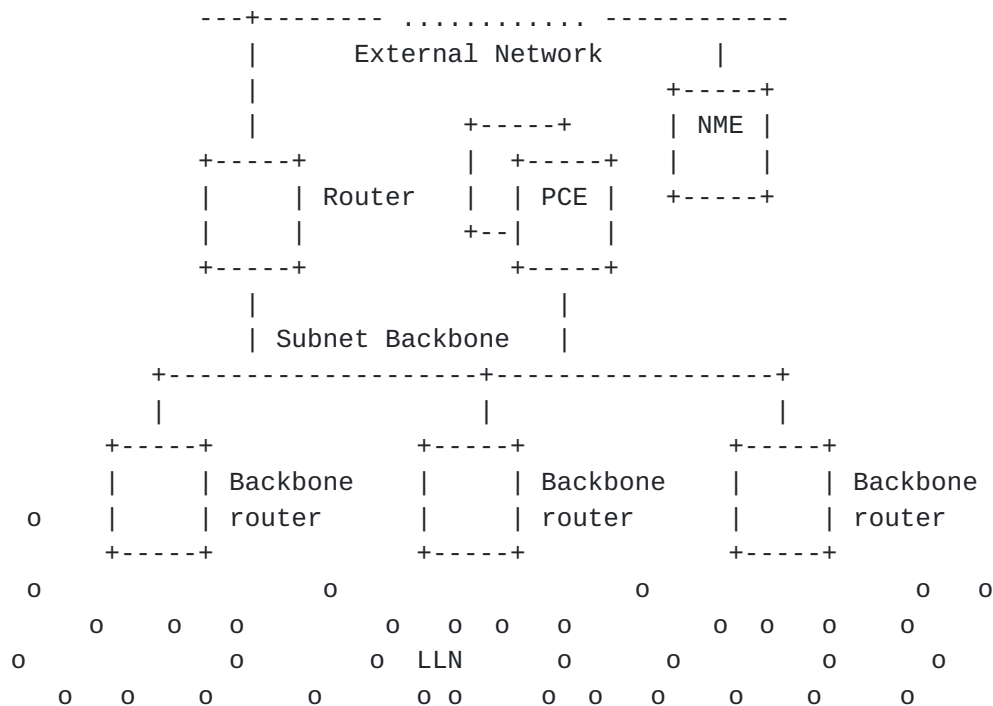


The LLN devices communicate over IPv6 [[RFC2460](#)] using the 6LoWPAN Header Compression (6LoWPAN HC) [[RFC6282](#)]. From the perspective of Layer 3, a single LLN interface (typically an IEEE802.15.4-compliant radio) may be seen as a collection of Links with different capabilities for unicast or multicast services. An IPv6 subnet spans over multiple links, effectively forming a Multi-Link subnet. Within that subnet, Neighbor Devices are discovered with 6LoWPAN Neighbor Discovery (6LoWPAN ND) [[RFC6775](#)]. The Routing Protocol for Low Power and Lossy Networks (RPL) [[RFC6550](#)] enables routing within the LLN, typically within the Multi-Link subnet in the so called Route Over fashion. RPL forms Destination Oriented Directed Acyclic Graphs (DODAGs) within Instances of the protocol, each Instance being associated with an Objective Function (OF) to form a routing topology. A particular LLN device, the LLN Border Router (LBR), acts as RPL root, 6LoWPAN HC terminator, and LLN Border Router (LBR) to the outside. The LBR is usually powered. More on RPL Instances can be found in [[RFC6550](#)], sections "3.1.2. RPL Identifiers" and "3.1.3. Instances, DODAGs, and DODAG Versions".

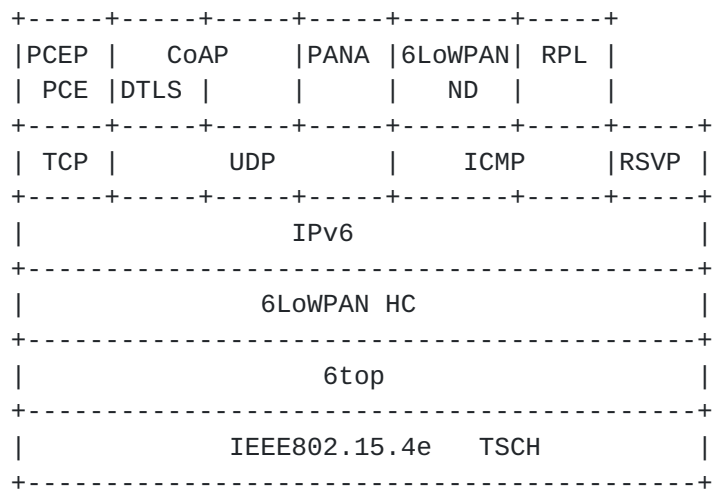
An extended configuration of the subnet comprises multiple LLNs. The LLNs are interconnected and synchronized over a backbone, that can be wired or wireless. The backbone can be a classical IPv6 network, with Neighbor Discovery operating as defined in [[RFC4861](#)] and [[RFC4862](#)]. The backbone can also support Efficiency-aware IPv6 Neighbor Discovery Optimizations [[I-D.chakrabarti-nordmark-6man-efficient-nd](#)] in mixed mode as described in [[I-D.thubert-6lowpan-backbone-router](#)].

Security is often handled at layer 2 and Layer 4. Authentication during the join process can be handled by the Protocol for Carrying Authentication for Network access (PANA) [[RFC5191](#)].

In the extended configuration, the functionality of the LBR is enhanced to that of Backbone Router (BBR). A BBR is an LBR, but also an Energy Aware Default Router (NEAR) as defined in [I-D.chakrabarti-nordmark-6man-efficient-nd]. The BBR performs ND proxy operations between the registered devices and the classical ND devices that are located over the backbone. 6TiSCH BBRs synchronize with one another over the backbone, so as to ensure that the multiple LLNs that form the IPv6 subnet stay tightly synchronized. If the Backbone is Deterministic (such as defined by the Time Sensitive Networking WG at IEEE), then the Backbone Router ensures that the end-to-end deterministic behavior is maintained between the LLN and the backbone.



The main architectural blocks are arranged as follows:



RPL is the routing protocol of choice for LLNs. (TBD RPL) whether there is a need to define a 6TiSCH OF.

(tbd NME) COMAN is working on network Management for LLN. They are considering the Open Mobile Alliance (OMA) Lightweight M2M (LWM2M) Object system. This standard includes DTLS, CoAP (core plus Block and Observe patterns), SenML and CoAP Resource Directory.

(tbd PCE) need to work with PCE WG to define flows to PCE, and define how to accommodate PCE routes and reservation. Will probably look a lot like GMPLS.

(tbd Backbone Router) need to work with 6MAN to define ND proxy. Also need BBR sync sync between deterministic Ethernet and 6TiSCH LLNs.

IEEE802.1TSN: external, maintain consistency. See also AVnu.

IEEE802.15.4: external, (tbd need updates?).

ISA100.20 Common Network Management: external, maintain consistency.

IoT6 European Project: external, maintain consistency.

5. Communication Paradigms and Interaction Models

[I-D.palattella-6tisch-terminology] defines the terms of Communication Paradigms and Interaction Models, which can be placed in parallel to the Information Models and Data Models that are defined in [[RFC3444](#)].

A Communication Paradigms would be an abstract view of a protocol exchange, and would come with an Information Model for the information that is being exchanged. In contrast, an Interaction

Models would be more refined and could point on standard operation
such as a Representational state transfer (REST) "GET" operation and

would match a Data Model for the data that is provided over the protocol exchange.

[I-D.roll-rpl-industrial-applicability] [section 2.1.3](#). and next discusses application-layer paradigms, such as Source-sink (SS) that is a Multipeer to Multipeer (MP2MP) model that is primarily used for alarms and alerts, Publish-subscribe (PS, or pub/sub) that is typically used for sensor data, as well as Peer-to-peer (P2P) and Peer-to-multipeer (P2MP) communications. Additional considerations on Duocast and its N-cast generalization are also provided. Those paradigms are frequently used in industrial automation, which is a major use case for IEEE802.15.4e TSCH wireless networks with [\[ISA100.11a\]](#) and [\[HART\]](#).

This specification focusses on Communication Paradigms and Interaction Models for packet forwarding and TSCH resources (cells) management. Link-layer and Network-layer Packet forwarding interactions are discussed in [Section 6](#), whereas Link-layer (one-hop), Network-layer (multihop along a track), and Application-layer (remote control) management mechanisms for the TSCH schedule are discussed in [Section 8](#).

6. Forwarding Models

6TiSCH supports three different forwarding model, G-MPLS Track Forwarding (TF), 6LoWPAN Fragment Forwarding (FF) and IPv6 Forwarding (6F).

[6.1.](#) Track Forwarding

Track Forwarding is the simplest and fastest. A set of input cells are uniquely bound to a set of output cells, representing a forwarding state that can be used regardless of the upper layer protocol. This model can effectively be seen as a G-MPLS operation in that the information used to switch is not an explicit label, but rather related to other properties of the way the packet was received, a particular cell in the case of 6TiSCH. As a result, as long as the TSCH MAC (and Layer 2 security) accepts a frame, that frame can be switched regardless of the protocol, whether this is an IPv6 packet, a 6LoWPAN fragment, or a frame from an alternate protocol such as WirelessHART or ISA100.11a.

A Track is defined end-to-end as a succession of timeslots. A timeslot belongs to at most one Track. For a given iteration of a Slotframe, the timeslot is associated uniquely with a cell, which indicates the channel at which the timeslot operates for that iteration.

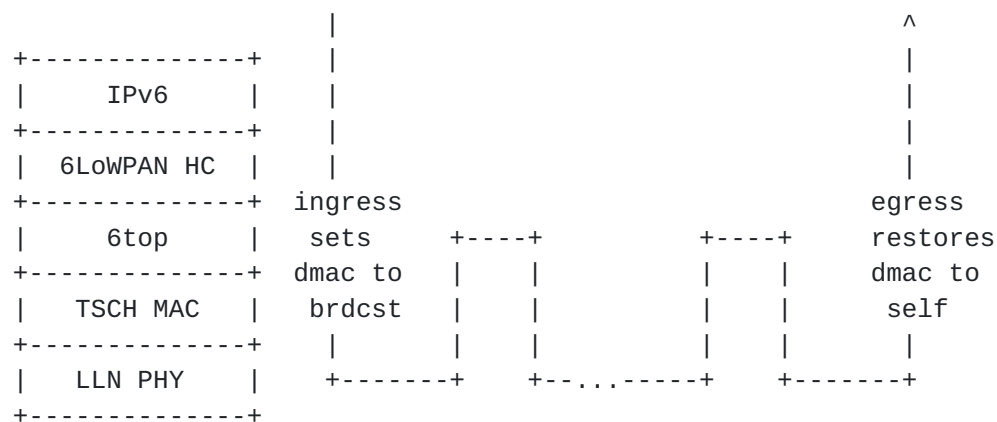
A data frame that is forwarded along a Track has a destination MAC address set to broadcast or a multicast address depending on MAC support. This way, the MAC layer in the intermediate nodes accepts the incoming frame and 6top switches it without incurring a change in the MAC header. In the case of IEEE802.15.4e, this means effectively broadcast, so that along the Track the short address for the destination is set to 0xFFFF.

Conversely, a frame that is received along a Track with a destination MAC address set to this node is extracted from the Track stream and delivered to the upper layer. A frame with an unrecognised MAC address is ignored at the MAC layer and thus is not received at the 6top sublayer.

There are 2 modes for a Track, transport mode and tunnel mode.

6.1.1. Transport Mode

In transport mode, the PDU is associated flow information that refers uniquely to the Track, so the 6top sublayer can place the frame in the appropriate timeslot without ambiguity. In the case of IPv6 traffic, flow identification is transported in the Flow Label of the IPv6 header. Associated with the source IPv6 address, the flow label forms a globally unique identifier for that particular Track that is validated at egress before restoring the destination MAC address (dmac) and punting to the upper layer.

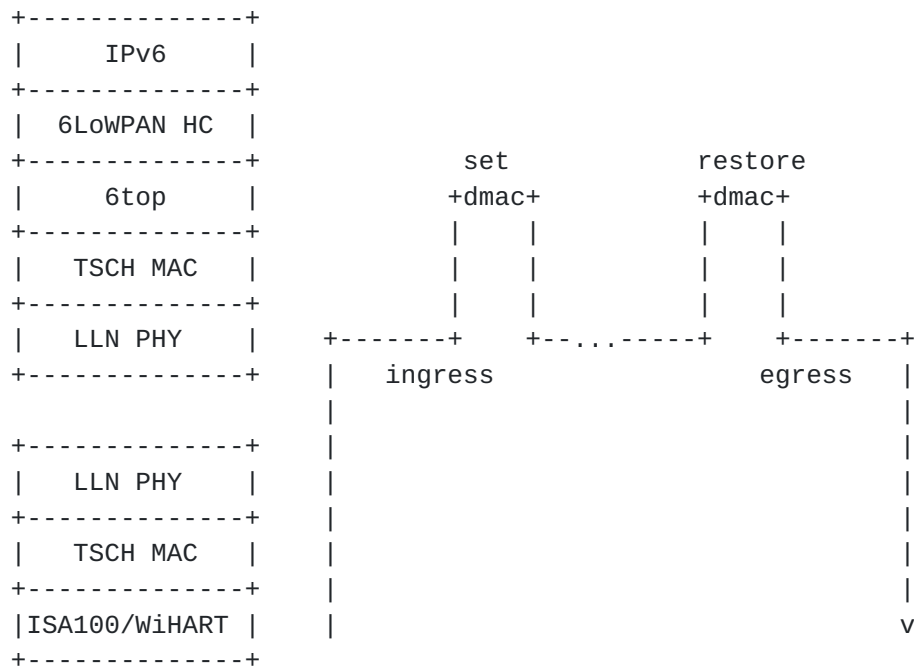


6.1.2. Tunnel Mode

In tunnel mode, the frames originate from an arbitrary protocol over a compatible MAC that may or may not be synchronized with the 6TiSCH network. An example of this would be a router with a dual radio that is capable of receiving and sending WirelessHART or ISA100.11a frames with the second radio, by presenting itself as an access Point or a Backbone Router, respectively.

In that mode, some entity (e.g. PCE) can coordinate with a WirelessHART Network Manager or an ISA100.11a System Manager to

specify the flows that are to be transported transparently over the Track.



In that case, the flow information that identifies the Track is uniquely derived from the information at the receiving end, for instance the incoming timeslots, or an ISA100.11a ContractId. At the ingress 6TiSCH router, the packet destination is recognized as self but the flow information indicates that the frame must be tunnelled over a particular 6top Track so the packet is not punted to upper layer. Instead, it is passed to the 6top sublayer for switching. The 6top sublayer in the ingress router overrides the destination MAC to broadcast and forwards.

At the egress 6top router, the reverse operation occurs. Based on metadata associated to the Track, the frame is passed to the appropriate link layer with the destination MAC restored.

6.1.3. Tunnel Metadata

Metadata coming with the Track configuration is expected to provide the destination MAC address of the egress endpoint as well as the tunnel mode and specific data depending on the mode, for instance a service access point for frame delivery at egress. If the tunnel egress point does not have a MAC address that matches the configuration, the Track installation fails.

In transport mode, if the final layer 3 destination is the tunnel termination, then it is possible that the IPv6 address of the destination is compressed at the 6LoWPAN sublayer based on the MAC address. It is thus mandatory at the ingress point to validate that the MAC address that was used at the 6LoWPAN sublayer for compression matches that of the tunnel egress point. For that reason, the node that injects a packet on a Track checks that the destination is effectively that of the tunnel egress point before it overwrites it to broadcast. The 6top sublayer at the tunnel egress point reverts that operation to the MAC address obtained from the tunnel metadata.

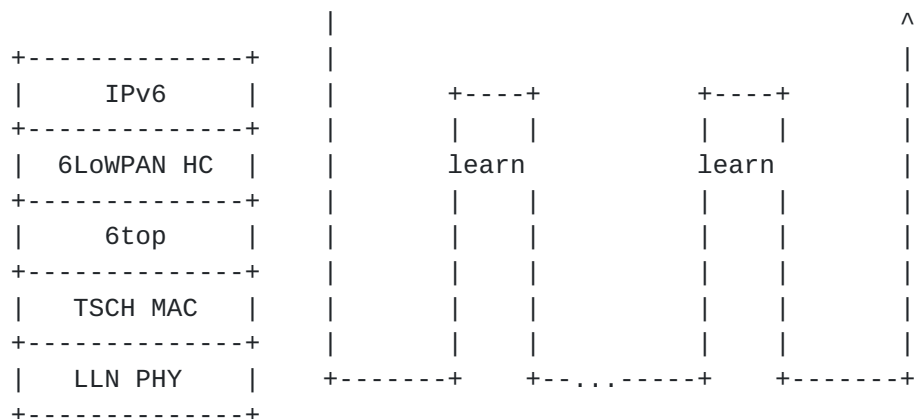
6.2. Fragment Forwarding

Considering that 6LoWPAN packets can be as large as 1280 bytes (the IPv6 MTU), and that the non-storing mode of RPL implies Source Routing that requires space for routing headers, and that a IEEE802.15.4 frame with security may carry in the order of 80 bytes of effective payload, an IPv6 packet might be fragmented into more than 16 fragments at the 6LoWPAN sublayer.

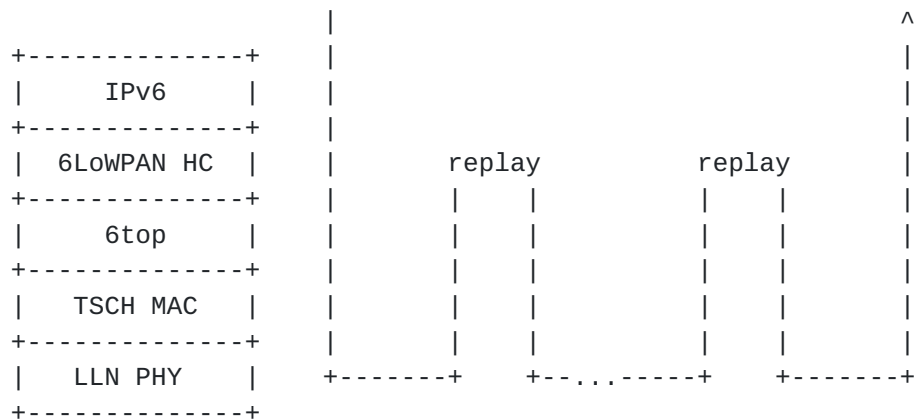
This level of fragmentation is much higher than that traditionally experienced over the Internet with IPv4 fragments, where fragmentation is already known as harmful.

In the case to a multihop route within a 6TiSCH network, Hop-by-Hop recomposition occurs at each hop in order to reform the packet and route it. This creates additional latency and forces intermediate nodes to store a portion of a packet for an undetermined time, thus impacting critical resources such as memory and battery.

[I-D.thubert-roll-forwarding-frags] describes a mechanism whereby the datagram tag in the 6LoWPAN Fragment is used as a label for switching at the 6LoWPAN sublayer. The draft allows for a degree of flow control base on an Explicit Congestion Notification, as well as end-to-end individual fragment recovery.



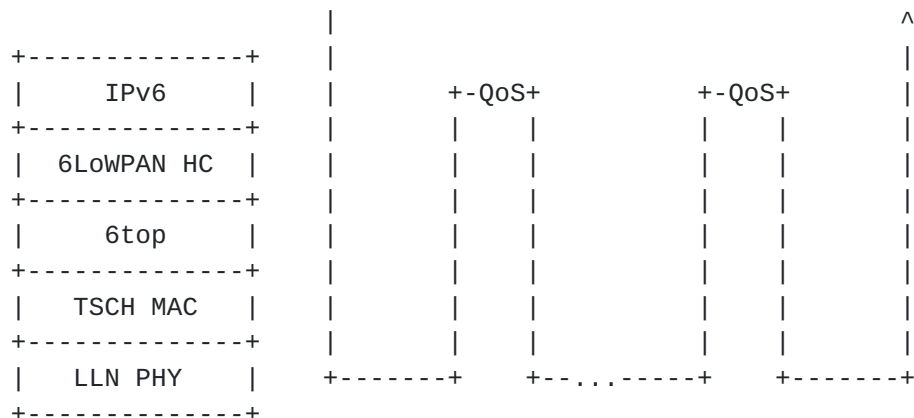
In that model, the first fragment is routed based on the IPv6 header that is present in that fragment. The 6LoWPAN sublayer learns the next hop selection, generates a new datagram tag for transmission to the next hop, and stores that information indexed by the incoming MAC address and datagram tag. The next fragments are then switched based on that stored state.



A bitmap and an ECN echo in the end-to-end acknowledgement enable the source to resend the missing fragments selectively. The first fragment may be resent to carve a new path in case of a path failure. The ECN echo set indicates that the number of outstanding fragments should be reduced.

6.3. IPv6 Forwarding

As the packets are routed at layer 3, traditional QoS and RED operations are expected to prioritize flows with differentiated services. A new class of service for Deterministic Forwarding is being defined to that effect in [I-D.svshah-tsvwg-lln-diffserv-recommendations].



7. TSCH and 6top

[7.1.](#) 6top

Thubert, Watteyne & AssimExpires April 22, 2014

[Page 12]

6top is a sublayer which is the next higher layer to TSCH and which offers a set of commands defining data and management interfaces. The management interface of 6top enables an upper layer to schedule cells and Slotframes in the TSCH schedule. 6top is defined in [I-D.wang-6tisch-6top].

If the scheduling entity explicitly specifies the slotOffset/channelOffset of the cells to be added/deleted, those cells are marked as "hard". 6top cannot move hard cells in the TSCH schedule. Hard cells are for example used by a central PCE.

6top contains a monitoring process which monitors the performance of cells, and can move a cell in the TSCH schedule when it performs bad. This is only applicable to cells which are marked as "soft". To reserve a soft cell, the higher layer does not indicate the exact slotOffset/channelOffset of the cell to add, but rather the resulting bandwidth and QoS requirements. When the monitoring process triggers a cell reallocation, the two neighbor motes communicating over this cell negotiate its new position in the TSCH schedule.

7.2. Network Synchronization

Nodes in a TSCH network must be time synchronized. A node keeps synchronized to its time source neighbor through a combination of frame-based and acknowledgement-based synchronization. In order to maximize battery life and network throughput, it is advisable that RPL ICMP discovery and maintenance traffic (governed by the trickle timer) be somehow coordinated with the transmission of time synchronization packets (especially with enhanced beacons). This could be achieved through an interaction of the 6top sublayer and the RPL objective Function, or could be controlled by a management entity.

Time distribution requires a loop-less structure. Nodes taken in a synchronization loop will rapidly desynchronize from the network and become isolated. It is expected that a RPL DAG with a dedicated global Instance is deployed for the purpose of time synchronization. That Instance is referred to as the Time Synchronization Global Instance (TSGI). The TSGI can be operated in either of the 3 modes that are detailed in RPL [RFC6550] section "3.1.3. Instances, DODAGs, and DODAG Versions". Multiple uncoordinated DODAGs with independent roots may be used if all the roots share a common time source such as the Global Positioning System (GPS). In the absence of a common time source, the TSGI should form a single DODAG with a virtual root. A backbone network is then used to synchronize and coordinate RPL operations between the backbone routers that act as sinks for the LLN.

A node that has not joined the TSGI advertises a MAC level Join Priority of 0xFF to notify its neighbors that it is not capable of serving as time parent. A node that has joined the TSGI advertises a MAC level Join Priority set to its DAGRank() in that Instance, where DAGRank() is the operation specified in [[RFC6550](#)], section "3.5.1. Rank Comparison".

A root is configured or obtains by some external means the knowledge of the RPLInstanceID for the TSGI. The root advertises its DagRank in the TSGI, that MUST be less than 0xFF, as its Join Priority (JP) in its IEEE802.15.4e Extended Beacons (EB). We'll note that the JP is now specified between 0 and 0x3F leaving 2 bits in the octet unused in the IEEE802.15.4e specification. After concertation with IEEE authors, it was asserted that 6TiSCH can make a full use of the octet to carry an integer value up to 0xFF.

A node that reads a Join Priority of less than 0xFF should join the neighbor with the lesser Join Priority and use it as time parent. If the node is configured to serve as time parent, then the node should join the TSGI, obtain a Rank in that Instance and start advertising its own DagRank in the TSGI as its Join Priority in its EBs.

7.3. Slotframes and Priorities

6top uses priority queues to manage concurrent data flows of different priorities. When a packet is received from an higher layer for transmission, the I-MUX module of 6top inserts that packet in the outgoing queue which matches the packet best (DSCP can therefore be used). At each scheduled transmit slot, the MUX module looks for the frame in all the outgoing queues that best matches the cells. If a frame is found, it is given to TSCH for transmission.

7.4. Packet Marking and Handling

reservation Deterministic flow allocation (hard reservation of timeslots) eg centralized RSVP? metrics? Hop-by-hop interaction with 6top. Lazy reservation (use shared slots to transport extra burst and then dynamically (de)allocate) Classical QoS (dynamic based on observation)

8. Schedule Management Mechanisms

6TiSCH uses 4 paradigms to manage the TSCH schedule of the LLN nodes: Static Scheduling, Neighbor-to-Neighbor Scheduling, Multihop Monitoring and Scheduling, and Hop-by-hop Scheduling. Multiple mechanisms are proposed that implement the associated Interaction Models, and can be combined and used in the same LLN. Which mechanism(s) are used depends on application requirements.

8.1. Minimal Static Scheduling

Thubert, Watteyne & AssimExpires April 22, 2014

[Page 14]

A static TSCH schedule can be used to bootstrap a network, as a initial phase during implementation, or as a fall-back mechanism in case of network malfunction. This scheduled can be preconfigured, or learnt by a node when joining the network, but it remains unchanged after the node has joined a network. The Routing Protocol for LLNs (RPL) is used on the resulting network. This "minimal" scheduling mechanism that implements this paradigm is detailed in [I-D.vilajosana-6tisch-minimal].

8.2. Neighbor-to-Neighbor Scheduling

The 6top sublayer [I-D.wang-6tisch-6top] defines a protocol for neighbor nodes to reserve soft cells to one another. Because this reservation is done without global knowledge of the schedule of nodes in the LLN, scheduling collisions are possible. 6top defines a monitoring process which continuously tracks the packet delivery ratio of soft cells. It uses these statistics to trigger the relocation of a soft cell in the schedule, using a negotiation protocol between the neighbors nodes communicating over that cell.

Monitoring and relocation is done in the 6top layer. For the upper layer, the connection between two neighbor node appears as an number of cells. Depending on the traffic requirements, the upper layer can request 6top to add or delete a number of cells scheduled to a particular neighbor, without being responsible for choosing the exact slotOffset/channelOffset of those cells.

8.3. Remote Monitoring and Schedule Management

[I-D.sudhaakar-6tisch-coap] defines an mapping of 6top's set of commands to CoAP resources. This allows an entity to interact with the 6top layer of a node that is multiple hops away. [I-D.sudhaakar-6tisch-coap] defines the CoAP resources and associated methods (GET/PUT/POST/DELETE). The payload of those signalling packets use CBOR to encode the different fields sent and received.

Being able to interact with the 6top sublayer of a node multiple hops away can be used for monitoring, scheduling, or a combination of both. The architecture supports variations on the deployment model, and focuses on the flows rather than the whether there is a proxy or a translational operation on the way.

The entity issuing the CoAP requests can be a central scheduling entity (e.g. a PCE), a node multiple hops away with the authority to modify the TSCH schedule (e.g. the head of a local cluster), or a external device monitoring the overall state of the network (e.g. NME). The architecture allows for different types of interactions between this CoAP client and a node in the network:

Query The CoAP client may retrieve information from a specific node in the network. This is typically a CoAP GET request issued on the appropriate resource on the node.

Report The CoAP client may register for periodic updates from a resource, for example to monitor the state of some statistics maintained by the node. This is typically done through CoAP Observe.

Action The CoAP client may request the node to take some action, for example add a cell to its TSCH schedule. This is typically a CoAP PUT/POST/DELETE request issued on the appropriate resource on the node.

Request The node may issue a request to the client to trigger some action, for example the calculation of a multi-hop route. This is typically a CoAP POST request issued by the node on the appropriate resource on the CoAP client.

Event The node may indicate the occurrence of a specific event to the CoAP client, for example the discovery of a new neighbor. This is typically a CoAP PUT request issued by the node on the appropriate resource on the CoAP client.

[I-D.sudhaakar-6tisch-coap] defines the a basic set of CoAP resources. For cases where extra functionality is needed, the draft also defines the concept of "profiles", as well as a mechanism for a CoAP client to discover the profiles installed on a node.

8.4. Hop-by-hop Scheduling

A node can reserve a track to a destination node multiple hops away by installing soft cells at each intermediate node. This forms a track of soft cells. It is the responsibility of the 6top sublayer of each node on the track to monitor these soft cells and trigger reallocations when needed.

This hop-by-hop reservation mechanism is similar to [RFC2119] and [RFC5974]. The protocol for a node to trigger hop-by-hop scheduling is not defined yet.

9. Centralized vs. Distributed Routing

6TiSCH supports a mixed model of centralized routes and distributed routes. Centralized routes can for example computed by a entity such as a PCE. Distributed routes are computed by the RPL routing protocol.

Both may inject routes in the Routing Tables of the 6TiSCH routers. In either case, each route is associated with a topology that is indexed by an RPLInstanceID, as defined in RPL [RFC6550]. RPL and PCE rely on shared sources to define Global and Local RPLInstanceIDs.

It is possible for centralized and distributed routing to share a same topology. In this case, centralized routes have precedence over distributed routes in case of a conflict.

Inside the 6TiSCH domain, the flow label is used to indicate the topology that must be used for routing. The associated Routing Tables are discussed in [[I-D.thubert-roll-flow-label](#)].

10. IANA Considerations

This specification does not require IANA action.

11. Security Considerations

This specification is not found to introduce new security threat.

12. Acknowledgements

13. References

13.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997.
- [RFC2460] Deering, S.E. and R.M. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", [RFC 2460](#), December 1998.
- [RFC3444] Pras, A. and J. Schoenwaelder, "On the Difference between Information Models and Data Models", [RFC 3444](#), January 2003.
- [RFC4080] Hancock, R., Karagiannis, G., Loughney, J. and S. Van den Bosch, "Next Steps in Signaling (NSIS): Framework", [RFC 4080](#), June 2005.
- [RFC4291] Hinden, R. and S. Deering, "IP Version 6 Addressing Architecture", [RFC 4291](#), February 2006.
- [RFC4861] Narten, T., Nordmark, E., Simpson, W. and H. Soliman, "Neighbor Discovery for IP version 6 (IPv6)", [RFC 4861](#), September 2007.
- [RFC4862] Thomson, S., Narten, T. and T. Jinmei, "IPv6 Stateless Address Autoconfiguration", [RFC 4862](#), September 2007.
- [RFC5191] Forsberg, D., Ohba, Y., Patil, B., Tschofenig, H. and A. Yegin, "Protocol for Carrying Authentication for Network Access (PANA)", [RFC 5191](#), May 2008.
- [RFC5889] Baccelli, E. and M. Townsley, "IP Addressing Model in Ad Hoc Networks", [RFC 5889](#), September 2010.

[RFC5974] Manner, J., Karagiannis, G. and A. McDonald, "NSIS Signaling Layer Protocol (NSLP) for Quality-of-Service Signaling", [RFC 5974](#), October 2010.

- [RFC6282] Hui, J. and P. Thubert, "Compression Format for IPv6 Datagrams over IEEE 802.15.4-Based Networks", [RFC 6282](#), September 2011.
- [RFC6550] Winter, T., Thubert, P., Brandt, A., Hui, J., Kelsey, R., Levis, P., Pister, K., Struik, R., Vasseur, JP. and R. Alexander, "RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks", [RFC 6550](#), March 2012.
- [RFC6775] Shelby, Z., Chakrabarti, S., Nordmark, E. and C. Bormann, "Neighbor Discovery Optimization for IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs)", [RFC 6775](#), November 2012.

13.2. Informative References

- [I-D.chakrabarti-nordmark-6man-efficient-nd]
Chakrabarti, S., Nordmark, E. and M. Wasserman,
"Efficiency aware IPv6 Neighbor Discovery Optimizations",
Internet-Draft [draft-chakrabarti-nordmark-6man-efficient-nd-01](#), November 2012.
- [I-D.ietf-roll-rpl-industrial-applicability]
Phinney, T., Thubert, P. and R. Assimiti, "RPL
applicability in industrial networks", Internet-Draft
[draft-ietf-roll-rpl-industrial-applicability-01](#), September 2013.
- [I-D.ohba-6tisch-security]
Chasko, S., Das, S., Lopez, R., Ohba, Y., Thubert, P. and
A. Yegin, "Security Framework and Key Management Protocol
Requirements for 6TiSCH", Internet-Draft [draft-ohba-6tisch-security-00](#), October 2013.
- [I-D.palattella-6tisch-terminology]
Palattella, M., Thubert, P., Watteyne, T. and Q. Wang,
"Terminology in IPv6 over the TSCH mode of IEEE
802.15.4e", Internet-Draft [draft-palattella-6tisch-terminology-00](#), October 2013.
- [I-D.sudhaakar-6tisch-coap]
Watteyne, T., "6TiSCH Data Model for CoAP", Internet-Draft
[draft-sudhaakar-6tisch-coap-00](#), October 2013.
- [I-D.svshah-tsvwg-lln-diffserv-recommendations]
Shah, S. and P. Thubert, "Differentiated Service Class
Recommendations for LLN Traffic", Internet-Draft [draft-svshah-tsvwg-lln-diffserv-recommendations-00](#), February 2013.

[I-D.thubert-6lowpan-backbone-router]

Thubert, P., "6LoWPAN Backbone Router", Internet-Draft
[draft-thubert-6lowpan-backbone-router-03](#), February 2013.

Thubert, Watteyne & Assim Expires April 22, 2014

[Page 18]

[I-D.thubert-roll-flow-label]

Thubert, P., "Use of the IPv6 Flow Label within an LLN", Internet-Draft [draft-thubert-roll-flow-label-02](#), November 2012.

[I-D.thubert-roll-forwarding-frags]

Thubert, P. and J. Hui, "LLN Fragment Forwarding and Recovery", Internet-Draft [draft-thubert-roll-forwarding-frags-01](#), February 2013.

[I-D.vilajosana-6tisch-minimal]

Vilajosana, X. and K. Pister, "Minimal 6TiSCH Configuration", Internet-Draft [draft-vilajosana-6tisch-minimal-00](#), October 2013.

[I-D.wang-6tisch-6top]

Wang, Q., Vilajosana, X. and T. Watteyne, "6TiSCH Operation Sublayer (6top)", Internet-Draft [draft-wang-6tisch-6top-00](#), October 2013.

[I-D.watteyne-6tisch-tsch]

Watteyne, T., "Using IEEE802.15.4e TSCH in an LLN context: Overview, Problem Statement and Goals", Internet-Draft [draft-watteyne-6tisch-tsch-00](#), October 2013.

[13.3.](#) External Informative References

[HART] www.hartcomm.org, "Highway Addressable Remote Transducer, a group of specifications for industrial process and control devices administered by the HART Foundation", .

[IEEE802.1TSNTG]

IEEE Standards Association, "IEEE 802.1 Time-Sensitive Networks Task Group", March 2013, <<http://www.ieee802.org/1/pages/avbridges.html>>.

[IEEE802154e]

IEEE standard for Information Technology, "IEEE std. 802.15.4e, Part. 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs) Amendment 1: MAC sublayer", April 2012.

[ISA100.11a]

ISA, "ISA100, Wireless Systems for Automation", May 2008, <<http://www.isa.org/Community/SP100WirelessSystemsforAutomation>>.

Pascal Thubert, editor
Cisco Systems, Inc
Building D
45 Allee des Ormes - BP1200
MOUGINS - Sophia Antipolis, 06254
FRANCE

Phone: +33 497 23 26 34
Email: pthubert@cisco.com

Thomas Watteyne
Linear Technology, Dust Networks Product Group
30695 Huntwood Avenue
Hayward, CA 94544
USA

Phone: +1 (510) 400-2978
Email: twatteyne@linear.com

Robert Assimiti
Centero
961 Indian Hills Parkway
Marietta, GA 30068
USA

Phone: +1 404 461 9614
Email: robert.assimiti@centerotech.com

