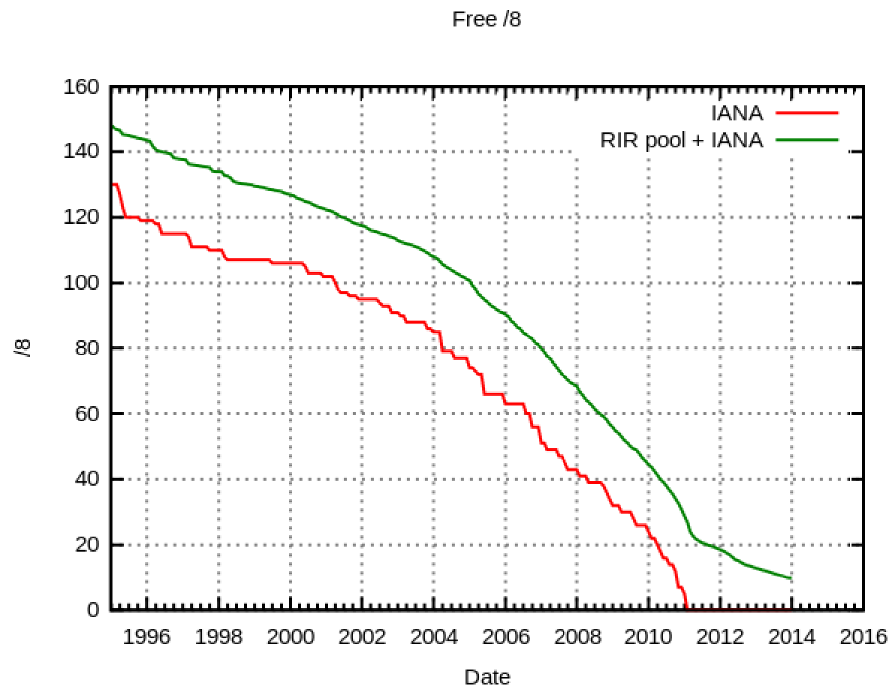
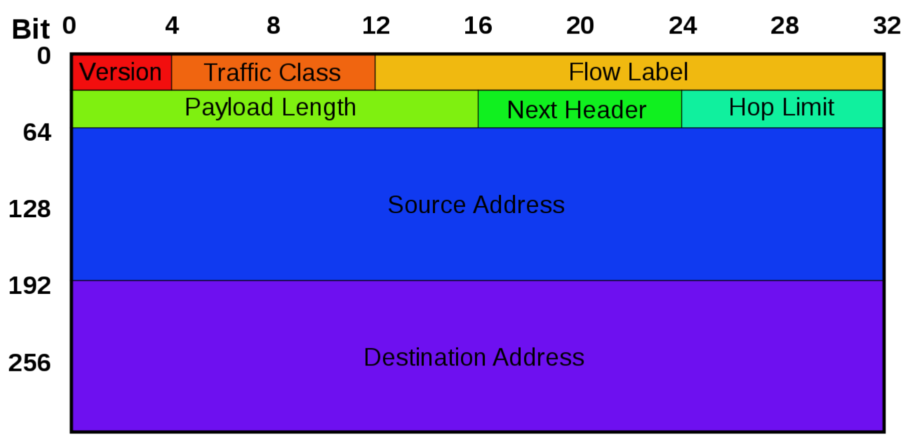
IPv6 over the TSCH mode of IEEE 802.15.4e

Repetition: IPv6

Motivation for IPv6? - Missing IP-addresses:





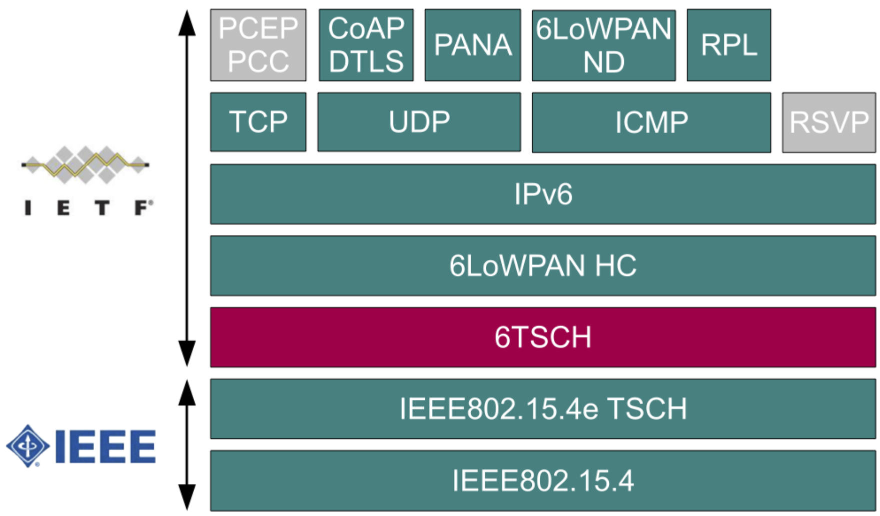
IPv6-Header

As the header format is simplified but also more flexibe the Next Header field contains a number which indicates that there are options coming behind the current header or that the current header was the last header.

Further changes in IPv6:

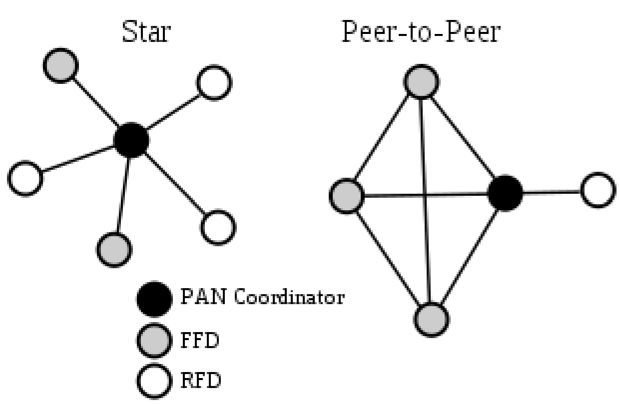
- No more NAT (Network Address Translation)
- Auto-configuration
- No more private address collisions
- Better multicast routing
- Simplified, more efficient routing
- True quality of service (QoS), also called “flow labeling”
- Built-in authentication and privacy support
- Flexible options and extensions
- Easier administration (say good-bye to DHCP)

The Stack

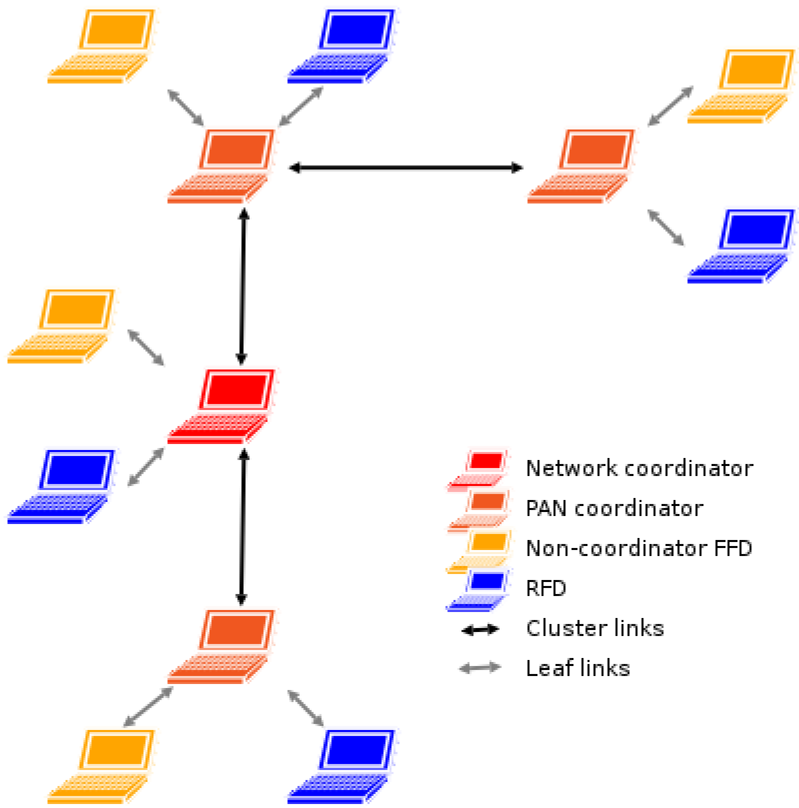


IEEE 802.15.4

A IEEE 802.15.4 network can have different topologies. The basic ones are the star and peer-to-peer topology shown in the following figure:

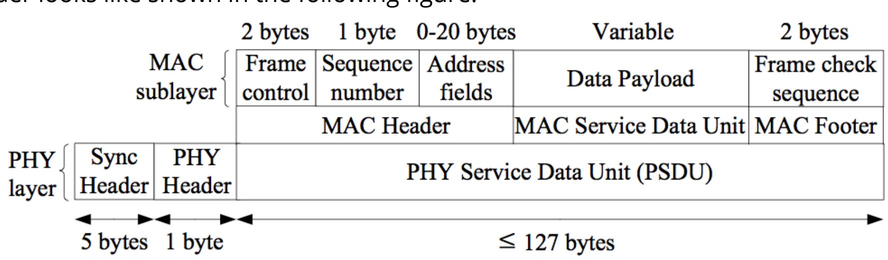


- **RFD**: Is a **Reduced Function Device** which can only communicate with FFD's but therefore can be manifactured with less costs
- The **FFD** is a **Full Function Device** can communicate with all types of devices in the network. They can as a PAN Coordinator and
- **PAN Coordinator**: The Personal Area Network Coordinator sets the PAN-Identifier of the network and routes traffic to other nodes. It is preferably the most powerfull FFD in the network or the current cluster.



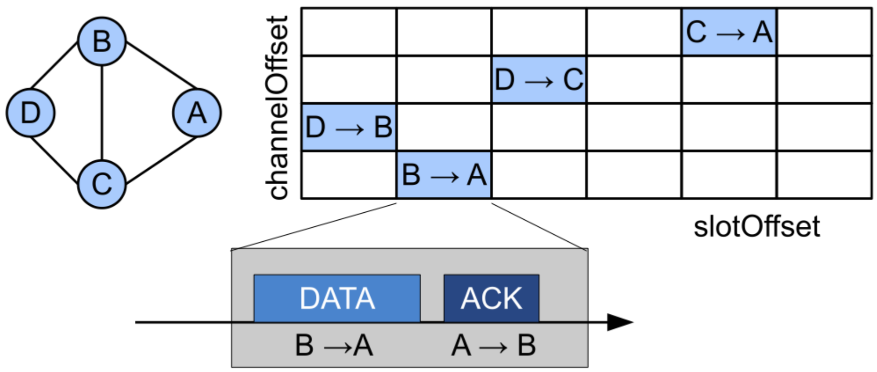
The more advanced topology is a "Cluster Tree". With such a tree and a higher layer routing protocoll (see RPL) it is even possible that RFD's communicate over FFD's with each other. The performance in peer-to-peer scenario's isn't great but therefore the management (time-synchronisation, distribution of optimized scheduling) is simplified.

A typical IEEE 802.15.4 header looks like shown in the following figure:



IEEE 802.15.4e / TSCH

The IEEE 802.15.4e standard introduces TSCH the Time Slotted Channel Hopping where you choose for each slotOffset a different channel resulting from a schedule which is defined in the work of 6TiSCH. But a simple schedule would look like:



Interaction with other protocols

6LoWPAN

The 6LoWPAN or „IPv6 over Low power Wireless Personal Area Network“ Layer is an intermediate layer between IPv6 and IEEE 802.15.4. It takles the problems of small packets and sparse bandwidth as well as routing in possibly fast changing networks.

The set of things described in 6LoWPAN are:

- Header compression
- Packet fragmentation and defragmentation
- Routing

Header compression

A 15.4 packet has a Maximum Transmission Unit (MTU) of 127 bytes where 46 bytes are used for the MAC layer and the AES encryption. Furthermore 48 bytes are needed for the IPv6 and the UDP header leaving only 33 bytes for payload.
With header compression the IPv6 and the UDP header can be compressed done to minimal 7 bytes increasing the possible payload to 74 bytes.

Packet fragmentation and defragmentation

IPv6 has a MTU of 1280 bytes obviously not fitting into a 15.4 packet of 127 bytes.
The first 5 bits of the 6LoWPAN layer describe the type where 11000xxx describes that the packet is the first fragment.
Headers of following fragments start with 11100xxx.

Routing

The 6LoWPAN has to routing approaches:

- Mesh-Routing (mesh-under)
- IP-Routing (route-over)

Mesh-Routing

A node can be adressed via a 8bit long ID which gets assigned during the joining process.
So at mesh routing a mesh header with origin id, destination id and the count of the remaining hops is put in front of possible fragmentation and compression headers.
Most of the mesh-under protocols in 15.4 use some kind of distance vector algorithm.

IP-Routing

IP-Routing is called route-over as it's using the IP protocol to route. The advantage of mesh-under in contrast to route-over is that nodes don't have to traverse the hole stack up to IP.
The 6LoWPAN specific route-over protocol is RPL following in the next chapter.

RPL

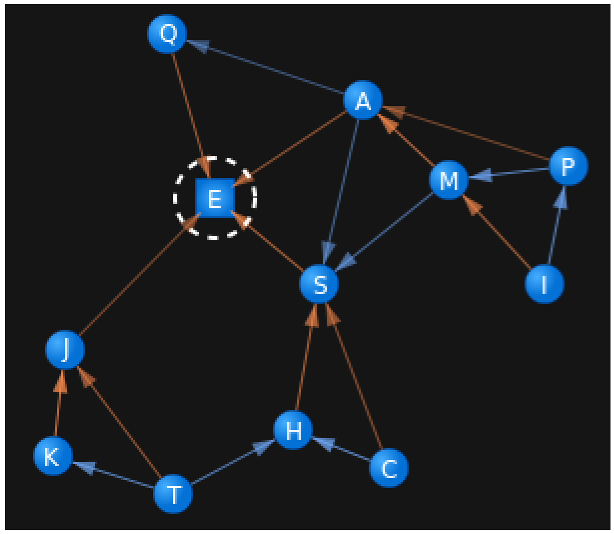
RPL is an routing protocol for Low power and Lossy Networks and is developed by the IETF workgroup ROLL (Routing Over Low power Lossy Networks) as existing routing protocols like OSPF, IS-IS, AODV and OLSR did not satisfy the requirements of such networks.

RPL is based a distance vector algorithm which constructs a destination oriented, directed, acyclic graph (DODAG). An example is shown in the following figure where node E is the destination. Every node has a lokal routing table and sends DODAG Information Objects (DIO) periodically. A DIO holds information like:

- the rank of the node within the DODAG
- the DODAG ID
- the RPL instance ID
- the current iteration of the RPL instance
- the node is the destionation node

With the DIO's from the neighboring nodes every node can choose the best fitting parent node / the next hop directed to the DODAG destination and send the parent node a Destination Advertisement Object (DAO) which contains a list with all the nodes which are reachable over the current node. If a node doesn't have any children (recall Reduced Function Devices) the list contains only the node itself.

Possible routing methods are:
- Multipeer to Multipeer (MP2MP)
- Publish-subscribe (PS, or pub)
- Peer to Peer (P2P) and
- Peer to Multipeer (P2MP)



In incative networks DIO's can be sparse so if a node (e.x. a joining node) needs information from neighbors he can send a DIO with a DODAG Information Solicitation (DIS) request.

6TiSCH

6TiSCH sets a clear focus on mesh networks which are connected via synchronized backbone routers (BBR's). Within the network tracks are established via the schedule to guarantee low jitter and latency. Tracks are fixed routes or multi-hop paths from one node to another node which consist of a deterministic sequence of cells.
The reservation of tracks can be globally optimized trough an Path Computation Element (PCE), which is responsible for building and maintaining the TSCH schedule, when centralized scheduling is used.

Scheduling

The 6TiSCH workgroup defines three models for scheduling:

1. "a centralized route computation that builds and maintains the communication schedule, and distributes it to the nodes. This schedule includes forwarding information associated to time slots; RPL operations only apply to emergency repair actions when the reference topology becomes unusable. A number of existing protocols are going to be extended to push the schedule from the PCE to the device, including the PCE Communication Protocol (PCEP), Forwarding and Control Element Separation (ForCES), Software-Defined Networking (SDN) OpenFlow 6 or even through network management over the Constrained Application Protocol (CoAP)."
2. "a distributed resource reservation and signaling protocol that establishes tracks between source and destination nodes along multi-hop routes identified by RPL. The track may be setup by extensions to the legacy Resource ReSerVation Protocol (RSVP) or the more recent but rather heavy Next Steps in Signaling (NSIS) protocol."
3. "a best effort resource allocation that is used to transport data frames on a per hop basis in the absence of a reservation protocol."[1]

Slotframes, Timeslots, Channels and Cells

Each node can have multiple Slotframes for different activities with different priorities. Each Slotframe is a Matrix of Channels and Timeslots which are forming cells. After all timeslots of the slotframe the slotframe starts from the beginning. The current timeslot number is stored in Absolute Slot Number (ASN).

Because the schedule is deterministic and a timeslot belongs to at most one track / each cell is bound to a node it is know from whom the receiving data are (presumly) coming.

Routing

6TiSCH has three modes for routing:

- mesh-under / track forwarding
- route-over / IPv6 forwarding
- 6LoWPAN Fragment Forwarding

Track forwarding

Track forwarding is an operation not bound to any upper layer. It's making usage of the strict scheduling scheme. So a

IPv6 forwarding

Fragment forwarding

Dealing with time-synchronization problems

Nodes have to stay tightly synchronized to allow the strict time division. Therefore the DODAG of RPL is used. Each node keeps a Join Priority (JP) which takes the value of the DAGRank() as specified in [RFC6550].
When a node joins the network, it advertises itself with the highest possible Join Priority of 255 as it is not part of the DODAG yet. The node listens for Enhanced Beacons from other nodes which contains the JP of the other node and takes the node with the lesser JP as its time parent.

> Written with StackEdit (https://stackedit.io/).

---

1. IETF 6TSCH: Combining IPv6 Connectivity with Industrial Performance "http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=6603730 (http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=6603730)" ↩