

Osmocom kernel-level GTP implementation

Andreas Schultz + Harald Welte

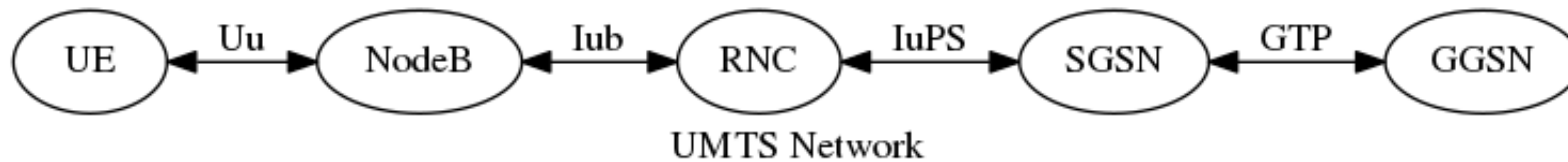
What is GTP / GTP in GPRS

- GTP, the *GPRS Tunneling Protocol* was first introduced with GPRS, the packet-switched add-on to GSM networks
- *PDP contexts* are sessions with an external packet data network (IP) which provide a tunnel between the phone and that external network
- GTP is used between the SGSN and GGSN to tunnel the User-plane (IP) packet data of a PDP Context
- latest definition in (3GPP TS 29.281 version 12.1.0 Release 12)

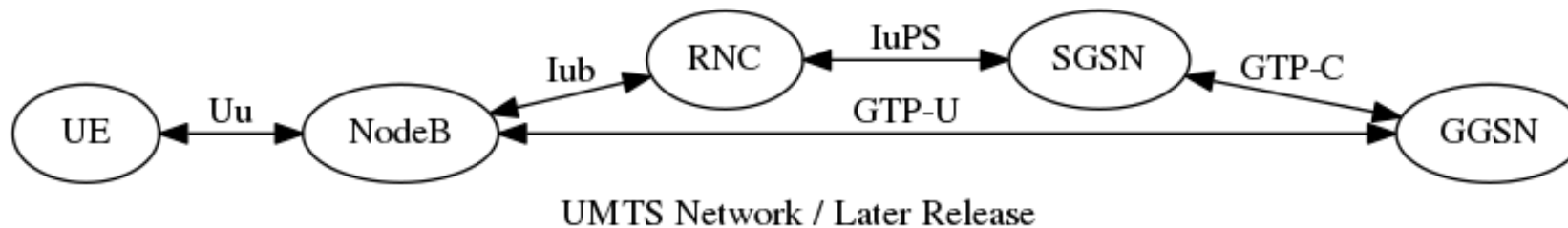


GTP in UMTS

- In UMTS, PDP originally remained between SGSN and GGSN

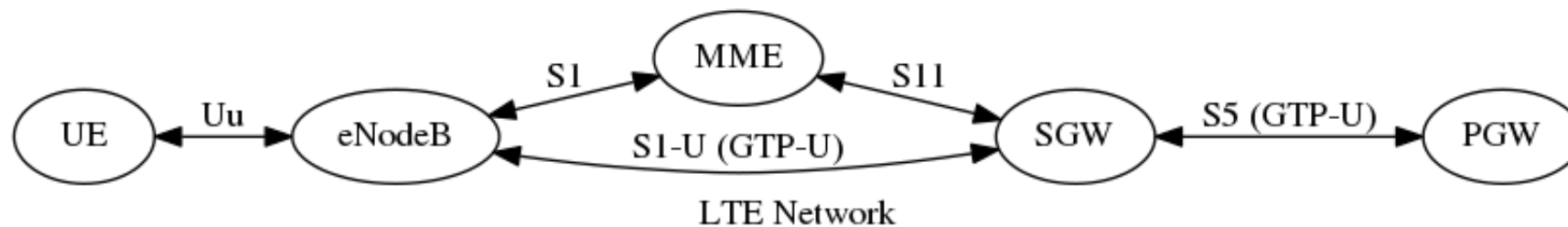


- With HSPA in later releases (Rel7+), user plane is moved out of SGSN and NodeB generates GTP directly
- hNodeB (femtocell) architecture also originates GTP directly in hNodeB



GTP in LTE

- LTE uses GTP at even more interfaces between network elements
- LTE introduces GTP-C Version 2 for control plane, but user plane remains GTP-U v1 like before.
- What used to be the GGSN is now called PDN-GW in LTE



GTP

- control (GTP-C) **and** user (GTP-U) plane protocol
- UDP based on well known ports (3386, 2152)
- per PDP context/tunnel identifiers (TID)
 - one for each direction(!) in GTP v1
 - is the only identifier for tunnel, not IP/Port tuples!
- support for packet sequence numbers and reordering

GTP-C

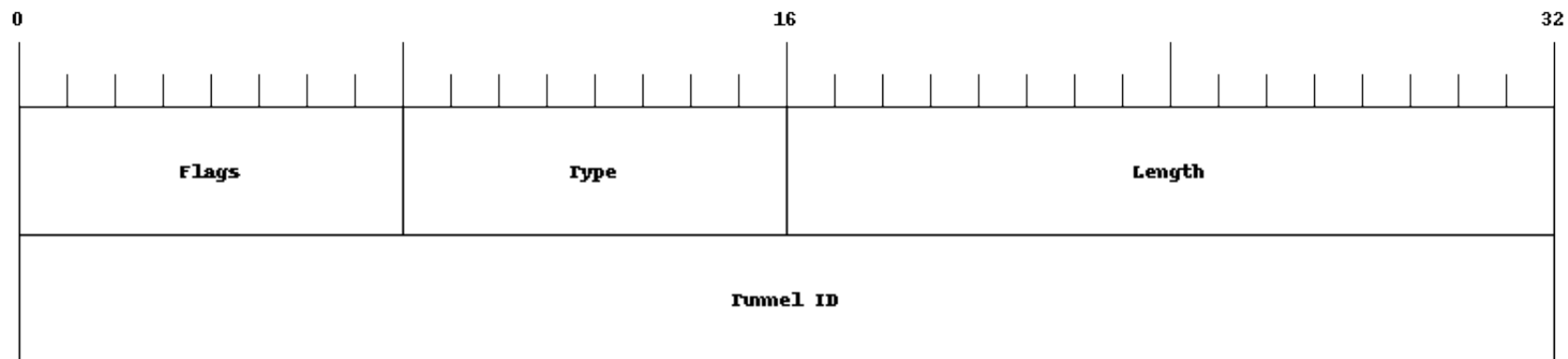
control protocol for:

- exchanging meta data
- assigning parameters to client:
 - IPv4 addresses
 - IPv6 prefixes
 - DNS servers
 - ...
- establishing and removing tunnels
- moving tunnel endpoints to other SGW's

GTP-U

two main jobs:

- per client tunnelling of IP packets
- path maintenance
 - echo requests/replies
 - error reporting



Problems with Userspace GTP

- Userspace GTP was fine with GPRS speeds
 - Packet rates / bandwidths small compared to routing
- With HSPA and particularly LTE, the world has changed
 - Signalling / Control Plane traffic is very limited
 - User Plane traffic is very high bandwidth

Osmocom kernel-level GTP

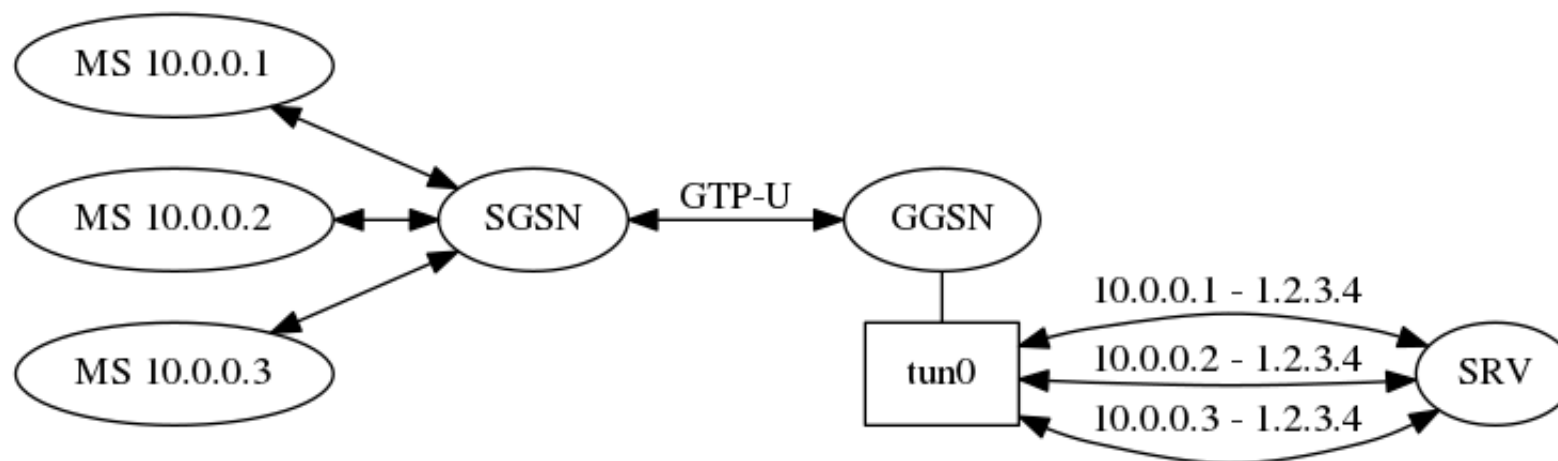
- Move bandwidth-critical User Plane into Linux kernel
- Keep Control Plane stays in Userspace (OpenGGSN)
- Use netlink based API to control user plane in kernel
- Started by Harald Welte + Pablo Neira @ sysmocom in 2012
 - unfortunately abandoned due to customer disappearing
- picked up by Andreas Schultz @ travelping in 2015
 - fixed Harald + Pablo's bugs
 - introduce network namespace support

Design

- netlink API built on top of genl
- initialization:
 - create GTP socket in userspace
 - create tun like network interface
 - bind GTP sockets to it
- for each PDP context, user space adds one record
 - IP address assigned to MS for this PDP context
 - TEIDs to identify mobile-originated packets
 - SGSN IP for GTP header of mobile-terminated packets

Design

- netlink API built on top of genl
- all PDP contexts share one tun device
 - one tun device for all subscribers, not one per subscriber



STATUS

- working
 - GTP-U v0 and v1 for IPv4 over IPv4
 - multiple tunnels supported
 - only single APN (multiple IP ranges not permitted)
- Limitations
 - no IPv6 support
 - no offload support
- existing users (both GPLv2)
 - OpenGGSN (<http://cgit.osmocom.org/openggsn/>)
 - new *ergw* Erlang GGSN / P-GW (<https://github.com/traveling/ergw>)

TODO

- TODO:
 - final round of clean-up, submission for mainline
 - fix or remove IPv6 support
- discussion items:
 - path MTU discovery
 - implications by/for offloading features (csum / gso / ...)
- wishlist
 - light-weight tunneling integration

The End

Questions?