# sysmocom

sysmocom - s.f.m.c. GmbH



# OsmoHNBGW User Manual

by Harald Welte

DRAFT

Copyright © 2019 sysmocom - s.f.m.c. GmbH

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with the Invariant Sections being just 'Foreword', 'Acknowledgements' and 'Preface', with no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".

The Asciidoc source code of this manual can be found at http://git.osmocom.org/osmo-hnbgw/ and of the common chapters at http://git.osmocom.org/osmo-gsm-manuals/

| HISTORY | | | |
|---|---|---|---|
| NUMBER | DATE | DESCRIPTION | NAME |
| 1 | November 30th, 2019 | Initial version | HW |

Copyright © 2019 sysmocom - s.f.m.c. GmbH 

# Contents

**1 Foreword** **1**
1.1 Acknowledgements . . . 1
1.2 Endorsements . . . 2

**2 Preface** **2**
2.1 FOSS lives by contribution! . . . 2
2.2 Osmocom and sysmocom . . . 3
2.3 Corrections . . . 3
2.4 Legal disclaimers . . . 3
2.4.1 Spectrum License . . . 3
2.4.2 Software License . . . 3
2.4.3 Trademarks . . . 3
2.4.4 Liability . . . 4
2.4.5 Documentation License . . . 4

**3 Introduction** **4**
3.1 Required Skills . . . 4
3.2 Getting assistance . . . 5

**4 Overview** **5**
4.1 About OsmoHNBGW . . . 5
4.2 The Iuh interface . . . 5

**5 Running OsmoHNBGW** **6**
5.1 SYNOPSIS . . . 6
5.2 OPTIONS . . . 6
5.3 Multiple instances . . . 6
5.4 Configure limits . . . 7
5.5 Configuring Primary Links . . . 7
5.5.1 Configure SCCP/M3UA to connect to an MSC's *IuCS* and an SGSN's *IuPS* interface . . . 7
5.5.2 Configure RUA to accept Iuh connections from hNodeB . . . 8
5.5.3 Configure co-located media gateway . . . 8
5.5.4 Configure co-located User Plane Function . . . 8

Copyright © 2019 sysmocom - s.f.m.c. GmbH 

**6 The Osmocom VTY Interface** **9**
6.1 Accessing the telnet VTY . . . 10
6.2 VTY Nodes . . . 10
6.3 Interactive help . . . 11
6.3.1 The question-mark (`?`) command . . . 11
6.3.2 TAB completion . . . 12
6.3.3 The `list` command . . . 12
6.3.4 The attribute system . . . 14
6.3.5 The expert mode . . . 15

**7 libosmocore Logging System** **16**
7.1 Log categories . . . 16
7.2 Log levels . . . 17
7.3 Log printing options . . . 17
7.4 Log filters . . . 18
7.5 Log targets . . . 18
7.5.1 Logging to the VTY . . . 18
7.5.2 Logging to the ring buffer . . . 18
7.5.3 Logging via gsmtap . . . 19
7.5.4 Logging to a file . . . 20
7.5.5 Logging to syslog . . . 21
7.5.6 Logging to systemd-journal . . . 22
7.5.7 Logging to stderr . . . 23

**8 Signaling Networks: SS7 and SIGTRAN** **23**
8.1 Physical Layer . . . 24
8.2 Message Transfer Part (MTP) . . . 24
8.2.1 Point Codes . . . 24
8.3 Higher-Layer Protocols . . . 24
8.4 Signaling Connection Control Part (SCCP) . . . 24
8.4.1 SCCP Addresses . . . 25
8.4.2 Global Titles . . . 25
8.4.3 Global Title Translation (GTT) . . . 25
8.4.4 Peculiarities of Connection Oriented SCCP . . . 26
8.5 SIGTRAN - SS7 over IP Networks . . . 26
8.5.1 SIGTRAN Concepts / Terminology . . . 26
8.5.1.1 Signaling Gateway (SG) . . . 27
8.5.1.2 Application Server (AS) . . . 27
8.5.1.3 Application Server Process (ASP) . . . 27

Copyright © 2019 sysmocom - s.f.m.c. GmbH 

8.5.2 SIGTRAN variants / stackings . . . 27
8.5.2.1 MTP3 User Adaptation (M3UA) . . . 27
8.5.2.2 SCCP User Adaptation (SUA) . . . 27
8.5.2.3 MTP2 User Adaptation (M2UA) . . . 27
8.5.2.4 MTP2-User Peer-to-Peer Adaptation (M2PA) . . . 28
8.5.3 SIGTRAN security . . . 28
8.5.4 IPv6 support . . . 28
8.5.5 SCTP multi-homing in SIGTRAN . . . 28
8.5.6 SCTP Primary Address . . . 28
8.5.7 SCTP Peer Primary Address . . . 29
8.5.8 SCTP INIT Parameters . . . 29
8.5.9 SCTP role . . . 30
8.5.10 M3UA/SUA role . . . 30
8.5.11 Traffic Modes in SIGTRAN . . . 30

**9 Configure SCCP/M3UA** **31**
9.1 Connect to STP Instance . . . 32
9.2 Local Point-Code . . . 32
9.3 Remote Point-Code . . . 33
9.4 Point-Code Format . . . 34
9.5 AS and ASP . . . 34
9.6 Subsystem Number (SSN) . . . 35
9.7 Routing Context / Routing Key . . . 35
9.7.1 M3UA without Routing Context IE / Routing Context *0* . . . 35
9.7.2 Example: Static Routing . . . 36

**10 MGW Pooling** **38**
10.1 MGW pool VTY configuration . . . 38
10.2 MGW pool management . . . 39
10.2.1 MGW pool status . . . 39
10.2.2 Adding an MGW / MGCP-Client to the MGW pool . . . 39
10.2.3 Reconnecting an MGW / MGCP-Client . . . 40
10.2.4 Monitor MGCP link (keepalive) / MGCP-Client . . . 41
10.2.5 Blocking an MGW / MGCP-Client . . . 41
10.2.6 Removing an MGW / MGCP-Client . . . 42

**11 Glossary** **42**

**A Osmocom TCP/UDP Port Numbers** **51**

Copyright © 2019 sysmocom - s.f.m.c. GmbH 

**B Bibliography / References** **52**

B.0.0.0.1 References . . . . . 52

**C GNU Free Documentation License** **56**

C.1 PREAMBLE . . . . . 57

C.2 APPLICABILITY AND DEFINITIONS . . . . . 57

C.3 VERBATIM COPYING . . . . . 58

C.4 COPYING IN QUANTITY . . . . . 58

C.5 MODIFICATIONS . . . . . 58

C.6 COMBINING DOCUMENTS . . . . . 59

C.7 COLLECTIONS OF DOCUMENTS . . . . . 60

C.8 AGGREGATION WITH INDEPENDENT WORKS . . . . . 60

C.9 TRANSLATION . . . . . 60

C.10 TERMINATION . . . . . 60

C.11 FUTURE REVISIONS OF THIS LICENSE . . . . . 61

C.12 RELICENSING . . . . . 61

C.13 ADDENDUM: How to use this License for your documents . . . . . 61

Copyright © 2019 sysmocom - s.f.m.c. GmbH 

# 1 Foreword

Digital cellular networks based on the GSM specification were designed in the late 1980s and first deployed in the early 1990s in Europe. Over the last 25 years, hundreds of networks were established globally and billions of subscribers have joined the associated networks.

The technological foundation of GSM was based on multi-vendor interoperable standards, first created by government bodies within CEPT, then handed over to ETSI, and now in the hands of 3GPP. Nevertheless, for the first 17 years of GSM technology, the associated protocol stacks and network elements have only existed in proprietary *black-box* implementations and not as Free Software.

In 2008 Dieter Spaar and I started to experiment with inexpensive end-of-life surplus Siemens GSM BTSs. We learned about the A-bis protocol specifications, reviewed protocol traces and started to implement the BSC-side of the A-bis protocol as something originally called `bs11-abis`. All of this was *just for fun*, in order to learn more and to boldly go where no Free Software developer has gone before. The goal was to learn and to bring Free Software into a domain that despite its ubiquity, had not yet seen any Free / Open Source software implementations.

`bs11-abis` quickly turned into `bsc-hack`, then *OpenBSC* and its *OsmoNITB* variant: A minimal implementation of all the required functionality of an entire GSM network, exposing A-bis towards the BTS. The project attracted more interested developers, and surprisingly quickly also commercial interest, contribution and adoption. This allowed adding support for more BTS models.

After having implemented the network-side GSM protocol stack in 2008 and 2009, in 2010 the same group of people set out to create a telephone-side implementation of the GSM protocol stack. This established the creation of the Osmocom umbrella project, under which OpenBSC and the OsmocomBB projects were hosted.

Meanwhile, more interesting telecom standards were discovered and implemented, including TETRA professional mobile radio, DECT cordless telephony, GMR satellite telephony, some SDR hardware, a SIM card protocol tracer and many others.

Increasing commercial interest particularly in the BSS and core network components has lead the way to 3G support in Osmocom, as well as the split of the minimal *OsmoNITB* implementation into separate and fully featured network components: OsmoBSC, OsmoMSC, OsmoHLR, OsmoMGW and OsmoSTP (among others), which allow seamless scaling from a simple "Network In The Box" to a distributed installation for serious load.

It has been a most exciting ride during the last eight-odd years. I would not have wanted to miss it under any circumstances.

— Harald Welte, Osmocom.org and OpenBSC founder, December 2017.

## 1.1 Acknowledgements

My deep thanks to everyone who has contributed to Osmocom. The list of contributors is too long to mention here, but I'd like to call out the following key individuals and organizations, in no particular order:

- Dieter Spaar for being the most amazing reverse engineer I've met in my career
- Holger Freyther for his many code contributions and for shouldering a lot of the maintenance work, setting up Jenkins - and being crazy enough to co-start sysmocom as a company with me ;)
- Andreas Eversberg for taking care of Layer2 and Layer3 of OsmocomBB, and for his work on OsmoBTS and OsmoPCU
- Sylvain Munaut for always tackling the hardest problems, particularly when it comes closer to the physical layer
- Chaos Computer Club for providing us a chance to run real-world deployments with tens of thousands of subscribers every year
- Bernd Schneider of Netzing AG for funding early ip.access nanoBTS support
- On-Waves ehf for being one of the early adopters of OpenBSC and funding a never ending list of features, fixes and general improvement of pretty much all of our GSM network element implementations
- sysmocom, for hosting and funding a lot of Osmocom development, the annual Osmocom Developer Conference and releasing this manual.

Copyright © 2019 sysmocom - s.f.m.c. GmbH 

- Jan Luebbe, Stefan Schmidt, Daniel Willmann, Pablo Neira, Nico Golde, Kevin Redon, Ingo Albrecht, Alexander Huemer, Alexander Chemeris, Max Suraev, Tobias Engel, Jacob Erlbeck, Ivan Kluchnikov
- NLnet Foundation, for providing funding for a number of individual work items within the Osmocom universe, such as LTE support in OsmoCBC or GPRS/EGPRS support for Ericsson RBS6000.
- WaveMobile Ltd, for many years of sponsoring.

May the source be with you!

— Harald Welte, Osmocom.org and OpenBSC founder, January 2016.

### 1.2 Endorsements

This version of the manual is endorsed by Harald Welte as the official version of the manual.

While the GFDL license (see Appendix C) permits anyone to create and distribute modified versions of this manual, such modified versions must remove the above endorsement.

## 2 Preface

First of all, we appreciate your interest in Osmocom software.

Osmocom is a Free and Open Source Software (FOSS) community that develops and maintains a variety of software (and partially also hardware) projects related to mobile communications.

Founded by people with decades of experience in community-driven FOSS projects like the Linux kernel, this community is built on a strong belief in FOSS methodology, open standards and vendor neutrality.

### 2.1 FOSS lives by contribution!

If you are new to FOSS, please try to understand that this development model is not primarily about "free of cost to the GSM network operator", but it is about a collaborative, open development model. It is about sharing ideas and code, but also about sharing the effort of software development and maintenance.

If your organization is benefiting from using Osmocom software, please consider ways how you can contribute back to that community. Such contributions can be many-fold, for example

- sharing your experience about using the software on the public mailing lists, helping to establish best practises in using/operating it,
- providing qualified bug reports, workarounds
- sharing any modifications to the software you may have made, whether bug fixes or new features, even experimental ones
- providing review of patches
- testing new versions of the related software, either in its current "master" branch or even more experimental feature branches
- sharing your part of the maintenance and/or development work, either by donating developer resources or by (partially) funding those people in the community who do.

We're looking forward to receiving your contributions.

Copyright © 2019 sysmocom - s.f.m.c. GmbH 

## 2.2 Osmocom and sysmocom

Some of the founders of the Osmocom project have established *sysmocom - systems for mobile communications GmbH* as a company to provide products and services related to Osmocom.

sysmocom and its staff have contributed by far the largest part of development and maintenance to the Osmocom mobile network infrastructure projects.

As part of this work, sysmocom has also created the manual you are reading.

At sysmocom, we draw a clear line between what is the Osmocom FOSS project, and what is sysmocom as a commercial entity. Under no circumstances does participation in the FOSS projects require any commercial relationship with sysmocom as a company.

## 2.3 Corrections

We have prepared this manual in the hope that it will guide you through the process of installing, configuring and debugging your deployment of cellular network infrastructure elements using Osmocom software. If you do find errors, typos and/or omissions, or have any suggestions on missing topics, please do take the extra time and let us know.

## 2.4 Legal disclaimers

### 2.4.1 Spectrum License

As GSM and UMTS operate in licensed spectrum, please always double-check that you have all required licenses and that you do not transmit on any ARFCN or UARFCN that is not explicitly allocated to you by the applicable regulatory authority in your country.



**Warning**
Depending on your jurisdiction, operating a radio transmitter without a proper license may be considered a felony under criminal law!

### 2.4.2 Software License

The software developed by the Osmocom project and described in this manual is Free / Open Source Software (FOSS) and subject to so-called *copyleft* licensing.

Copyleft licensing is a legal instrument to ensure that this software and any modifications, extensions or derivative versions will always be publicly available to anyone, for any purpose, under the same terms as the original program as developed by Osmocom.

This means that you are free to use the software for whatever purpose, make copies and distribute them - just as long as you ensure to always provide/release the *complete and corresponding* source code.

Every Osmocom software includes a file called `COPYING` in its source code repository which explains the details of the license. The majority of programs is released under GNU Affero General Public License, Version 3 (AGPLv3).

If you have any questions about licensing, don't hesitate to contact the Osmocom community. We're more than happy to clarify if your intended use case is compliant with the software licenses.

### 2.4.3 Trademarks

All trademarks, service marks, trade names, trade dress, product names and logos appearing in this manual are the property of their respective owners. All rights not expressly granted herein are reserved.

For your convenience we have listed below some of the registered trademarks referenced herein. This is not a definitive or complete list of the trademarks used.

Copyright © 2019 sysmocom - s.f.m.c. GmbH 

*Osmocom*® and *OpenBSC*® are registered trademarks of Holger Freyther and Harald Welte.

*sysmocom*® and *sysmoBTS*® are registered trademarks of *sysmocom - systems for mobile communications GmbH*.

*ip.access*® and *nanoBTS*® are registered trademarks of *ip.access Ltd.*

#### 2.4.4 Liability

The software is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the License text included with the software for more details.

#### 2.4.5 Documentation License

Please see Appendix C for further information.

# 3 Introduction

## 3.1 Required Skills

Please note that even while the capital expenses of running mobile networks has decreased significantly due to Osmocom software and associated hardware like sysmoBTS, GSM networks are still primarily operated by large GSM operators.

Neither the GSM specification nor the GSM equipment was ever designed for networks to be installed and configured by anyone but professional GSM engineers, specialized in their respective area like radio planning, radio access network, back-haul or core network.

If you do not share an existing background in GSM network architecture and GSM protocols, correctly installing, configuring and optimizing your GSM network will be tough, irrespective whether you use products with Osmocom software or those of traditional telecom suppliers.

GSM knowledge has many different fields, from radio planning through site installation to core network configuration/administration.

The detailed skills required will depend on the type of installation and/or deployment that you are planning, as well as its associated network architecture. A small laboratory deployment for research at a university is something else than a rural network for a given village with a handful of cells, which is again entirely different from an urban network in a dense city.

Some of the useful skills we recommend are:

- general understanding about RF propagation and path loss in order to estimate coverage of your cells and do RF network planning.
- general understanding about GSM network architecture, its network elements and key transactions on the Layer 3 protocol
- general understanding about voice telephony, particularly those of ISDN heritage (Q.931 call control)
- understanding of GNU/Linux system administration and working on the shell
- understanding of TCP/IP networks and network administration, including tcpdump, tshark, wireshark protocol analyzers.
- ability to work with text based configuration files and command-line based interfaces such as the VTY of the Osmocom network elements

### 3.2 Getting assistance

If you do have a support package / contract with sysmocom (or want to get one), please contact support@sysmocom.de with any issues you may have.

If you don't have a support package / contract, you have the option of using the resources put together by the Osmocom community at https://projects.osmocom.org/, checking out the wiki and the mailing-list for community-based assistance. Please always remember, though: The community has no obligation to help you, and you should address your requests politely to them. The information (and software) provided at osmocom.org is put together by volunteers for free. Treat them like a friend whom you're asking for help, not like a supplier from whom you have bought a service.

If you would like to obtain professional/commercial support on Osmocom CNI, you can always reach out to sales@sysmocom.de to discuss your support needs. Purchasing support from sysmocom helps to cover the ongoing maintenance of the Osmocom CNI software stack.

## 4 Overview

### 4.1 About OsmoHNBGW

OsmoHNBGW implements the Home NodeB Gateway function in the 3G network architecture. It serves as a gateway between the classic 3G core network (CN) domain with its IuCS and IuPS interface and the femtocell based RAN.

A typical 3G network consisting of Osmocom components is illustrated in the following diagram:

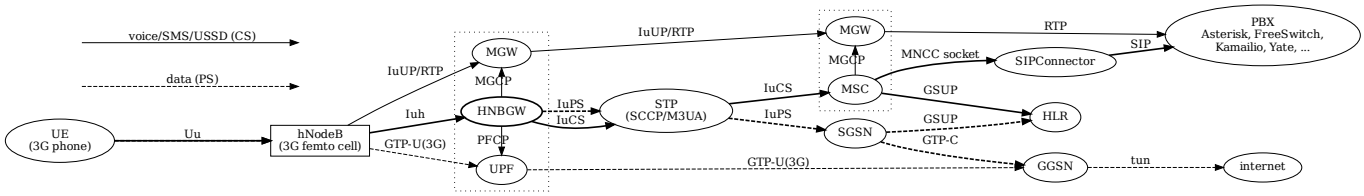



Figure 1: Typical 3G network architecture used with OsmoHNBGW

The HNB-GW performs a translation interface between the IuCS/IuPS interfaces on the one hand side, and the Iuh interface on the or ther hand:

```
                      Iuh                          IuCS/IuPS
                                                   +----+----+
                                                   | CC | MM |
NAS                   +----+----+           .  .  +----+----+
Non-Access Stratum    | CC | MM |         .       | RANAP   |
- - - - - - - - - - - +----+----+-------+         +---------+
                      | RANAP   |       |    H    | SCCP    |
Access Stratum        +---------+ HNBAP |    N    +---------+
                      | RUA     |       |    B    | M3UA    |
                      +---------+-------+    -    +---------+
                      |         SCTP      |    G    | SCTP    |
                      +-------------------+    W    +---------+
                      |         IP        |         | IP      |
                      +-------------------+         +---------+
```

On the femtocell (Home NodeB) side, OsmoHNBGW implements the Iuh interface as specified by 3GPP.

### 4.2 The Iuh interface

Iuh consists of the following sub-layers:

Copyright © 2019 sysmocom - s.f.m.c. GmbH 

- HNBAP (Home NodeB Application Part)
- RUA (RANAP User Adaptation, between RANAP and SCTP

# 5 Running OsmoHNBGW

The OsmoHNBGW executable (`osmo-hnbgw`) offers the following command-line arguments:

## 5.1 SYNOPSIS

**osmo-hnbgw** [-hl-V] [-d *DBGMASK*] [-D] [-c *CONFIGFILE*] [-s] [-T] [-e *LOGLEVEL*]

## 5.2 OPTIONS

**-h, --help**
Print a short help message about the supported options

**-V, --version**
Print the compile-time version number of the OsmoHNBGW program

**-d, --debug *DBGMASK,DBGLEVELS***
Set the log subsystems and levels for logging to stderr. This has mostly been superseded by VTY-based logging configuration, see Section 7 for further information.

**-D, --daemonize**
Fork the process as a daemon into background.

**-c, --config-file *CONFIGFILE***
Specify the file and path name of the configuration file to be used. If none is specified, use `osmo-msc.cfg` in the current working directory.

**-s, --disable-color**
Disable colors for logging to stderr. This has mostly been deprecated by VTY based logging configuration, see Section 7 for more information.

**-T, --timestamp**
Enable time-stamping of log messages to stderr. This has mostly been deprecated by VTY based logging configuration, see Section 7 for more information.

**-e, --log-level *LOGLEVEL***
Set the global log level for logging to stderr. This has mostly been deprecated by VTY based logging configuration, see Section 7 for more information.

## 5.3 Multiple instances

Running multiple instances of `osmo-hnbgw` on the same computer is possible if all interfaces (VTY, CTRL, Iuh) are separated using the appropriate configuration options. The IP based interfaces are binding to local host by default. In order to separate the processes, the user has to bind those services to specific but different IP addresses and/or ports.

The VTY and the Control interface can be bound to IP addresses from the loopback address range, for example:

```
line vty
 bind 127.0.0.2
ctrl
 bind 127.0.0.2
```

Copyright © 2019 sysmocom - s.f.m.c. GmbH 

The Iuh interface can be bound to an individual port:

```
hnbgw
 iuh
  local-ip 0.0.0.0
  local-port 29169
```

For the following links, OsmoHNBGW acts as a client and does not listen/bind to a specific interface, and will hence not encounter conflicts for multiple instances running on the same interface:

- The SCCP/M3UA links are established by OsmoHNBGW contacting an STP.
- The MGCP link is established by OsmoHNBGW contacting an MGW.
- The PFCP link is established by OsmoHNBGW contacting a UPF.

To run multiple OsmoHNBGW instances on the same SCCP routing, each HNBGW has to configure a distinct point-code, see Section 5.5.1.

## 5.4 Configure limits

When connecting hundreds of HNB to OsmoHNBGW, it may be necessary to adjust the operating system's limit on open file descriptors for the osmo-hnbgw process. A typical default limit imposed by operating systems is 1024; this would be exceeded by, for example, about 1024 HNB on Iuh, sockets for other interfaces not considered yet.

It should be ok to set an OS limit on open file descriptors as high as 65536 for osmo-hnbgw, which practically rules out failure from running out of file descriptors anywhere (<50,000 HNB).

When using systemd, the file descriptor limit may be adjusted in the service file by the `LimitNOFILE` setting ("Number of Open FILE descriptors"). OsmoHNBGW ships a systemd service file with a high LimitNOFILE setting.

## 5.5 Configuring Primary Links

### 5.5.1 Configure SCCP/M3UA to connect to an MSC's *IuCS* and an SGSN's *IuPS* interface

OsmoHNBGW acts as client to contact an STP instance and establish an SCCP/M3UA link.

An example configuration of OsmoHNBGW's SCCP link:

```
cs7 instance 0
 point-code 0.23.5
 asp asp-clnt-OsmoHNBGW 2905 0 m3ua
  remote-ip 127.0.0.1
  sctp-role client
 sccp-address msc
  routing-indicator PC
  point-code 0.23.1
 sccp-address sgsn
  routing-indicator PC
  point-code 0.23.2
hnbgw
 iucs
  remote-addr msc
 iups
  remote-addr sgsn
```

This configuration is explained in detail in Section 9.

Copyright © 2019 sysmocom - s.f.m.c. GmbH 

#### 5.5.2 Configure RUA to accept Iuh connections from hNodeB

OsmoHNBGW acts as server to accept Iuh connections from hNodeB devices.

An example configuration for OsmoHNBGW's RUA server:

```
hnbgw
 iuh
  local-ip 10.9.8.7
  local-port 29169
```

#### 5.5.3 Configure co-located media gateway

OsmoHNBGW requires a co-located OsmoMGW instance. The purpose of the co-located media gateway is to relay the RTP traffic between hNodeB and the core network.

For security reasons the RAN network is kept separate and isolated from the core network. Both networks will usually have no transparent routing in between them. The co-located media gateway provides an interface between hNodeB and core network across this boundary.

The configuration is done under the hnbgw node along with `iucs` and `iups`.

An example configuration for OsmoHNBGW's MGCP client:

```
hnbgw
 mgw 0
  remote-ip 127.0.0.1
  remote-port 2427
  reset-endpoint rtpbridge/* ❶
```

❶ The *reset-endpoint* setting instructs the OsmoMGW to send a wildcarded DLCX to the media gateway. This helps to clear lingering calls from the media gateway when the OsmoHNBGW is restarted.

OsmoHNBGW is also able to handle a pool of media gateways for load distribution. See also Section 10.

**Note**
Previous versions of OsmoHNBGW didn't have the *mgw* VTY node and hence didn't support the MGW pooling feature. Therefore, historically the MGW related commands where placed under the `mgcp` VTY node. The MGW related commands under the `mgcp` VTY are still parsed and used but its use is deprecated and hence discouraged in favour of the new `mgw` node. Writing the config to a file from within OsmoHNBGW will automatically convert the config to use the new `mgw` node.

#### 5.5.4 Configure co-located User Plane Function

OsmoHNBGW optionally supports relaying the GTP user plane via a co-located UPF, which is controlled by the PFCP protocol.

PFCP support is optional at compile time, as well as run time. To use a co-located UPF,

- osmo-hnbgw needs to be compiled with *configure --enable-pfcp*,
- and osmo-hnbgw.cfg needs to configure a *pfcp* / *remote-addr* and *local-addr*.

The following example configures OsmoHNBGW to associate via PFCP with a UPF listening on UDP 127.0.0.2:8805, ready to setup GTP tunnel relays.

```
hnbgw
 pfcp
  remote-addr 127.0.0.2
  local-addr 127.0.0.1
```

Copyright © 2019 sysmocom - s.f.m.c. GmbH 

3GPP TS 29.244 4.2.2 specifies that PFCP Request messages shall be sent to UDP port 8805, i.e. the PFCP port is fixed as 8805 and currently not configurable in osmo-hnbgw.

Setting a *local-addr* is required: the PFCP protocol features a Node ID, which uniquely identifies PFCP peers across different interfaces. According to the PFCP specification, the Node ID can be a fully-qualified domain name (FQDN) or an IP address. Currently, osmo-hnbgw has no support for using an FQDN as Node ID, and so far uses the *local-addr* as local Node ID — hence the *local-addr* must not be "0.0.0.0", which is an unfortunate consequence. This is likely to improve in the future, see https://osmocom.org/issues/5647 .

# 6 The Osmocom VTY Interface

All human interaction with Osmocom software is typically performed via an interactive command-line interface called the *VTY*.

**Note**
Integration of your programs and scripts should **not** be done via the telnet VTY interface, which is intended for human interaction only: the VTY responses may arbitrarily change in ways obvious to humans, while your scripts' parsing will likely break often. For external software to interact with Osmocom programs (besides using the dedicated protocols), it is strongly recommended to use the Control interface instead of the VTY, and to actively request / implement the Control interface commands as required for your use case.

The interactive telnet VTY is used to

- explore the current status of the system, including its configuration parameters, but also to view run-time state and statistics,
- review the currently active (running) configuration,
- perform interactive changes to the configuration (for those items that do not require a program restart),
- store the current running configuration to the config file,
- enable or disable logging; to the VTY itself or to other targets.

The Virtual Tele Type (VTY) has the concept of *nodes* and *commands*. Each command has a name and arguments. The name may contain a space to group several similar commands into a specific group. The arguments can be a single word, a string, numbers, ranges or a list of options. The available commands depend on the current node. there are various keyboard shortcuts to ease finding commands and the possible argument values.

Configuration file parsing during program start is actually performed the VTY's CONFIG node, which is also available in the telnet VTY. Apart from that, the telnet VTY features various interactive commands to query and instruct a running Osmocom program. A main difference is that during config file parsing, consistent indenting of parent vs. child nodes is required, while the interactive VTY ignores indenting and relies on the *exit* command to return to a parent node.

**Note**
In the *CONFIG* node, it is not well documented which commands take immediate effect without requiring a program restart. To save your current config with changes you may have made, you may use the `write file` command to **overwrite** your config file with the current configuration, after which you should be able to restart the program with all changes taking effect.

This chapter explains most of the common nodes and commands. A more detailed list is available in various programs' VTY reference manuals, e.g. see [vty-ref-osmomsc].

There are common patterns for the parameters, these include IPv4 addresses, number ranges, a word, a line of text and choice. The following will explain the commonly used syntactical patterns:

Copyright © 2019 sysmocom - s.f.m.c. GmbH 

Table 1: VTY Parameter Patterns

| Pattern | Example | Explanation |
|---|---|---|
| `A.B.C.D` | `127.0.0.1` | An IPv4 address |
| `A.B.C.D/M` | `192.168.1.0/24` | An IPv4 address and mask |
| `X:X::X:X` | `::1` | An IPv6 address |
| `X:X::X:X/M` | `::1/128` | An IPv6 address and mask |
| `TEXT` | `example01` | A single string without any spaces, tabs |
| `.TEXT` | `Some information` | A line of text |
| `(OptionA\|OptionB\|OptionC)` | `OptionA` | A choice between a list of available options |
| `<0-10>` | `5` | A number from a range |

## 6.1 Accessing the telnet VTY

The VTY of a given Osmocom program is implemented as a telnet server, listening to a specific TCP port.

Please see Appendix A to check for the default TCP port number of the VTY interface of the specific Osmocom software you would like to connect to.

As telnet is insecure and offers neither strong authentication nor encryption, the VTY by default only binds to localhost (127.0.0.1) and will thus not be reachable by other hosts on the network.



**Warning**
By default, any user with access to the machine running the Osmocom software will be able to connect to the VTY. We assume that such systems are single-user systems, and anyone with local access to the system also is authorized to access the VTY. If you require stronger security, you may consider using the packet filter of your operating system to restrict access to the Osmocom VTY ports further.

## 6.2 VTY Nodes

The VTY by default has the following minimal nodes:

**VIEW**
When connecting to a telnet VTY, you will be on the *VIEW* node. As its name implies, it can only be used to view the system status, but it does not provide commands to alter the system state or configuration. As long as you are in the non-privileged *VIEW* node, your prompt will end in a > character.

**ENABLE**
The *ENABLE* node is entered by the `enable` command, from the *VIEW* node. Changing into the *ENABLE* node will unlock all kinds of commands that allow you to alter the system state or perform any other change to it. The *ENABLE* node and its children are signified by a # character at the end of your prompt.
You can change back from the *ENABLE* node to the *VIEW* node by using the `disable` command.

**CONFIG**
The *CONFIG* node is entered by the `configure terminal` command from the *ENABLE* node. The config node is used to change the run-time configuration parameters of the system. The prompt will indicate that you are in the config node by a `(config)#` prompt suffix.
You can always leave the *CONFIG* node or any of its children by using the `end` command.
This node is also automatically entered at the time the configuration file is read. All configuration file lines are processed as if they were entered from the VTY *CONFIG* node at start-up.

**Other**
Depending on the specific Osmocom program you are running, there will be few or more other nodes, typically below the *CONFIG* node. For example, the OsmoBSC has nodes for each BTS, and within the BTS node one for each TRX, and within the TRX node one for each Timeslot.

Copyright © 2019 sysmocom - s.f.m.c. GmbH 

## 6.3 Interactive help

The VTY features an interactive help system, designed to help you to efficiently navigate is commands.

**Note**
The VTY is present on most Osmocom GSM/UMTS/GPRS software, thus this chapter is present in all the relevant manuals. The detailed examples below assume you are executing them on the OsmoMSC VTY. They will work in similar fashion on the other VTY interfaces, while the node structure will differ in each program.

### 6.3.1 The question-mark (?) command

If you type a single `?` at the prompt, the VTY will display possible completions at the exact location of your currently entered command.

If you type `?` at an otherwise empty command (without having entered even only a partial command), you will get a list of the first word of all possible commands available at this node:

**Example: Typing ? at start of OsmoMSC prompt**

```
OsmoMSC> ❶
  show         Show running system information
  list         Print command list
  exit         Exit current mode and down to previous mode
  help         Description of the interactive help system
  enable       Turn on privileged mode command
  terminal     Set terminal line parameters
  who          Display who is on vty
  logging      Configure logging
  no           Negate a command or set its defaults
  sms          SMS related commands
  subscriber   Operations on a Subscriber
```

❶ Type `?` here at the prompt, the `?` itself will not be printed.

If you have already entered a partial command, `?` will help you to review possible options of how to continue the command. Let's say you remember that `show` is used to investigate the system status, but you don't remember the exact name of the object. Hitting `?` after typing `show` will help out:

**Example: Typing ? after a partial command**

```
OsmoMSC> show ❶
  version           Displays program version
  online-help       Online help
  history           Display the session command history
  cs7               ITU-T Signaling System 7
  logging           Show current logging configuration
  alarms            Show current logging configuration
  talloc-context    Show talloc memory hierarchy
  stats             Show statistical values
  asciidoc          Asciidoc generation
  rate-counters     Show all rate counters
  fsm               Show information about finite state machines
  fsm-instances     Show information about finite state machine instances
  sgs-connections   Show SGs interface connections / MMEs
  subscriber        Operations on a Subscriber
  bsc               BSC
  connection        Subscriber Connections
  transaction       Transactions
```

Copyright © 2019 sysmocom - s.f.m.c. GmbH 

```
  statistics       Display network statistics
  sms-queue        Display SMSqueue statistics
  smpp          SMPP Interface
```

❶ Type `?` after the `show` command, the `?` itself will not be printed.

You may pick the `bsc` object and type `?` again:

**Example: Typing ? after show bsc**

```
OsmoMSC> show bsc
  <cr>
```

By presenting `<cr>` as the only option, the VTY tells you that your command is complete without any remaining arguments being available, and that you should hit enter, a.k.a. "carriage return".

### 6.3.2 TAB completion

The VTY supports tab (tabulator) completion. Simply type any partial command and press `<tab>`, and it will either show you a list of possible expansions, or completes the command if there's only one choice.

**Example: Use of <tab> pressed after typing only s as command**

```
OsmoMSC> s❶
show       sms        subscriber
```

❶ Type `<tab>` here.

At this point, you may choose `show`, and then press `<tab>` again:

**Example: Use of <tab> pressed after typing show command**

```
OsmoMSC> show ❶
version    online-help history    cs7        logging    alarms
talloc-context stats      asciidoc   rate-counters fsm        fsm-instances
sgs-connections subscriber bsc        connection transaction statistics
sms-queue smpp
```

❶ Type `<tab>` here.

### 6.3.3 The `list` command

The `list` command will give you a full list of all commands and their arguments available at the current node:

**Example: Typing list at start of OsmoMSC VIEW node prompt**

```
OsmoMSC> list
  show version
  show online-help
  list
  exit
  help
  enable
  terminal length <0-512>
  terminal no length
  who
  show history
  show cs7 instance <0-15> users
```

Copyright © 2019 sysmocom - s.f.m.c. GmbH 

```
show cs7 (sua|m3ua|ipa) [<0-65534>]
show cs7 instance <0-15> asp
show cs7 instance <0-15> as (active|all|m3ua|sua)
show cs7 instance <0-15> sccp addressbook
show cs7 instance <0-15> sccp users
show cs7 instance <0-15> sccp ssn <0-65535>
show cs7 instance <0-15> sccp connections
show cs7 instance <0-15> sccp timers
logging enable
logging disable
logging filter all (0|1)
logging color (0|1)
logging timestamp (0|1)
logging print extended-timestamp (0|1)
logging print category (0|1)
logging print category-hex (0|1)
logging print level (0|1)
logging print file (0|1|basename) [last]
logging set-log-mask MASK
logging level (rll|cc|mm|rr|mncc|pag|msc|mgcp|ho|db|ref|ctrl|smpp|ranap|vlr|iucs|bssap| ↩
    sgs|lglobal|llapd|linp|lmux|lmi|lmib|lsms|lctrl|lgtp|lstats|lgsup|loap|lss7|lsccp|lsua ↩
    |lm3ua|lmgcp|ljibuf|lrspro) (debug|info|notice|error|fatal)
logging level set-all (debug|info|notice|error|fatal)
logging level force-all (debug|info|notice|error|fatal)
no logging level force-all
show logging vty
show alarms
show talloc-context (application|all) (full|brief|DEPTH)
show talloc-context (application|all) (full|brief|DEPTH) tree ADDRESS
show talloc-context (application|all) (full|brief|DEPTH) filter REGEXP
show stats
show stats level (global|peer|subscriber)
show asciidoc counters
show rate-counters
show fsm NAME
show fsm all
show fsm-instances NAME
show fsm-instances all
show sgs-connections
show subscriber (msisdn|extension|imsi|tmsi|id) ID
show subscriber cache
show bsc
show connection
show transaction
sms send pending
sms delete expired
subscriber create imsi ID
subscriber (msisdn|extension|imsi|tmsi|id) ID sms sender (msisdn|extension|imsi|tmsi|id)  ↩
    SENDER_ID send .LINE
subscriber (msisdn|extension|imsi|tmsi|id) ID silent-sms sender (msisdn|extension|imsi| ↩
    tmsi|id) SENDER_ID send .LINE
subscriber (msisdn|extension|imsi|tmsi|id) ID silent-call start (any|tch/f|tch/any|sdcch)
subscriber (msisdn|extension|imsi|tmsi|id) ID silent-call stop
subscriber (msisdn|extension|imsi|tmsi|id) ID ussd-notify (0|1|2) .TEXT
subscriber (msisdn|extension|imsi|tmsi|id) ID ms-test close-loop (a|b|c|d|e|f|i)
subscriber (msisdn|extension|imsi|tmsi|id) ID ms-test open-loop
subscriber (msisdn|extension|imsi|tmsi|id) ID paging
show statistics
show sms-queue
logging filter imsi IMSI
show smpp esme
```

Copyright © 2019 sysmocom - s.f.m.c. GmbH 

**Tip**
Remember, the list of available commands will change significantly depending on the Osmocom program you are accessing, its software version and the current node you're at. Compare the above example of the OsmoMSC *VIEW* node with the list of the OsmoMSC *NETWORK* config node:

**Example: Typing list at start of OsmoMSC NETWORK config node prompt**

```
OsmoMSC(config-net)# list
  help
  list
  write terminal
  write file
  write memory
  write
  show running-config
  exit
  end
  network country code <1-999>
  mobile network code <0-999>
  short name NAME
  long name NAME
  encryption a5 <0-3> [<0-3>] [<0-3>] [<0-3>]
  authentication (optional|required)
  rrlp mode (none|ms-based|ms-preferred|ass-preferred)
  mm info (0|1)
  timezone <-19-19> (0|15|30|45)
  timezone <-19-19> (0|15|30|45) <0-2>
  no timezone
  periodic location update <6-1530>
  no periodic location update
```

#### 6.3.4 The attribute system

The VTY allows to edit the configuration at runtime. For many VTY commands the configuration change is immediately valid but for some commands a change becomes valid on a certain event only. In some cases it is even necessary to restart the whole process.

To give the user an overview, which configuration change applies when, the VTY implemets a system of attribute flags, which can be displayed using the `show` command with the parameter `vty-attributes`

**Example: Typing show vty-attributes at the VTY prompt**

```
OsmoBSC> show vty-attributes
  Global attributes:
    ^  This command is hidden (check expert mode)
    !  This command applies immediately
    @  This command applies on VTY node exit
  Library specific attributes:
    A  This command applies on ASP restart
    I  This command applies on IPA link establishment
    L  This command applies on E1 line update
  Application specific attributes:
    o  This command applies on A-bis OML link (re)establishment
    r  This command applies on A-bis RSL link (re)establishment
    l  This command applies for newly created lchans
```

The attributes are symbolized through a single ASCII letter (flag) and do exist in three levels. This is more or less due to the technical aspects of the VTY implementation. For the user, the level of an attribute has only informative purpose.

Copyright © 2019 sysmocom - s.f.m.c. GmbH 

The global attributes, which can be found under the same attribute letter in every osmocom application, exist on the top level. The Library specific attributes below are used in various osmocom libraries. Like with the global attributes the attribute flag letter stays the same throughout every osmocom application here as well. On the third level one can find the application specific attributes. Those are unique to each osmocom application and the attribute letters may have different meanings in different osmocom applications. To make the user more aware of this, lowercase letters were used as attribute flags.

The `list` command with the parameter `with-flags` displays a list of available commands on the current VTY node, along with attribute columns on the left side. Those columns contain the attribute flag letters to indicate to the user how the command behaves in terms of how and when the configuration change takes effect.

**Example: Typing list with-flags at the VTY prompt**

```
OsmoBSC(config-net-bts)# list with-flags
  . ...  help
  . ...  list [with-flags]
  . ...  show vty-attributes
  . ...  show vty-attributes (application|library|global)
  . ...  write terminal
  . ...  write file [PATH]
  . ...  write memory
  . ...  write
  . ...  show running-config ❶
  . ...  exit
  . ...  end
  . o..  type (unknown|bs11|nanobts|rbs2000|nokia_site|sysmobts) ❷
  . ...  description .TEXT
  . ...  no description
  . o..  band BAND
  . .r.  cell_identity <0-65535> ❸
  . .r.  dtx uplink [force]
  . .r.  dtx downlink
  . .r.  no dtx uplink
  . .r.  no dtx downlink
  . .r.  location_area_code <0-65535>
  . o..  base_station_id_code <0-63>
  . o..  ipa unit-id <0-65534> <0-255>
  . o..  ipa rsl-ip A.B.C.D
  . o..  nokia_site skip-reset (0|1)
  ! ...  nokia_site no-local-rel-conf (0|1) ❹
  ! ...  nokia_site bts-reset-timer  <15-100> ❺
```

❶ This command has no attributes assigned.

❷ This command applies on A-bis OML link (re)establishment.

❸ This command applies on A-bis RSL link (re)establishment.

❹, ❺ This command applies immediately.

There are multiple columns because a single command may be associated with multiple attributes at the same time. To improve readability each flag letter gets a dedicated column. Empty spaces in the column are marked with a dot (".")

In some cases the listing will contain commands that are associated with no flags at all. Those commands either play an exceptional role (interactive commands outside "configure terminal", vty node navigation commands, commands to show / write the config file) or will require a full restart of the overall process to take effect.

#### 6.3.5 The expert mode

Some VTY commands are considered relatively dangerous if used in production operation, so the general approach is to hide them. This means that they don't show up anywhere but the source code, but can still be executed. On the one hand, this approach

Copyright © 2019 sysmocom - s.f.m.c. GmbH 

reduces the risk of an accidental invocation and potential service degradation; on the other, it complicates intentional use of the hidden commands.

The VTY features so-called *expert* mode, that makes the hidden commands appear in the interactive help, as well as in the XML VTY reference, just like normal ones. This mode can be activated from the *VIEW* node by invoking the `enable` command with the parameter `expert-mode`. It remains active for the individual VTY session, and gets disabled automatically when the user switches back to the *VIEW* node or terminates the session.

A special attribute in the output of the `list with-flags` command indicates whether a given command is hidden in normal mode, or is a regular command:

**Example: Hidden commands in the output of the list with-flags command**

```
OsmoBSC> enable expert-mode ❶
OsmoBSC# list with-flags
  ...
  ^   bts <0-255> (activate-all-lchan|deactivate-all-lchan) ❷
  ^   bts <0-255> trx <0-255> (activate-all-lchan|deactivate-all-lchan) ❸
  .   bts <0-255> trx <0-255> timeslot <0-7> sub-slot <0-7> mdcx A.B.C.D <0-65535> ❹
  ^   bts <0-255> trx <0-255> timeslot <0-7> sub-slot <0-7> (borken|unused) ❺
  .   bts <0-255> trx <0-255> timeslot <0-7> sub-slot <0-7> handover <0-255> ❻
  .   bts <0-255> trx <0-255> timeslot <0-7> sub-slot <0-7> assignment ❼
  .   bts <0-255> smscb-command (normal|schedule|default) <1-4> HEXSTRING ❽
  ...
```

❶ This command enables the *expert* mode.

❷, ❸, ❺ This is a hidden command (only shown in the *expert* mode).

❹, ❻, ❼, ❽ This is a regular command that is always shown regardless of the mode.

# 7 libosmocore Logging System

In any reasonably complex software it is important to understand how to enable and configure logging in order to get a better insight into what is happening, and to be able to follow the course of action. We therefore ask the reader to bear with us while we explain how the logging subsystem works and how it is configured.

Most Osmocom Software (like `osmo-bts`, `osmo-bsc`, `osmo-nitb`, `osmo-sgsn` and many others) uses the same common logging system.

This chapter describes the architecture and configuration of this common logging system.

The logging system is composed of

- log targets (where to log),
- log categories (who is creating the log line),
- log levels (controlling the verbosity of logging), and
- log filters (filtering or suppressing certain messages).

All logging is done in human-readable ASCII-text. The logging system is configured by means of VTY commands that can either be entered interactively, or read from a configuration file at process start time.

## 7.1 Log categories

Each sub-system of the program in question typically logs its messages as a different category, allowing fine-grained control over which log messages you will or will not see. For example, in OsmoBSC, there are categories for the protocol layers `rsl`, `rr`, `mm`, `cc` and many others. To get a list of categories interactively on the vty, type: `logging level ?`

Copyright © 2019 sysmocom - s.f.m.c. GmbH 

## 7.2 Log levels

For each of the log categories (see Section 7.1), you can set an independent log level, controlling the level of verbosity. Log levels include:

**fatal**
Fatal messages, causing abort and/or re-start of a process. This *shouldn't happen*.

**error**
An actual error has occurred, its cause should be further investigated by the administrator.

**notice**
A noticeable event has occurred, which is not considered to be an error.

**info**
Some information about normal/regular system activity is provided.

**debug**
Verbose information about internal processing of the system, used for debugging purpose. This will log the most.

The log levels are inclusive, e.g. if you select *info*, then this really means that all events with a level of at least *info* will be logged, i.e. including events of *notice*, *error* and *fatal*.

So for example, in OsmoBSC, to set the log level of the Mobility Management category to info, you can use the following command: `log level mm info`.

There is also a special command to set all categories as a one-off to a desired log level. For example, to silence all messages but those logged as notice and above issue the command: `log level set-all notice`

Afterwards you can adjust specific categories as usual.

A similar command is `log level force-all <level>` which causes all categories to behave as if set to log level <level> until the command is reverted with `no log level force-all` after which the individually-configured log levels will again take effect. The difference between `set-all` and `force-all` is that `set-all` actually changes the individual category settings while `force-all` is a (temporary) override of those settings and does not change them.

## 7.3 Log printing options

The logging system has various options to change the information displayed in the log message.

**log color 1**
With this option each log message will log with the color of its category. The color is hard-coded and can not be changed. As with other options a *0* disables this functionality.

**log timestamp 1**
Includes the current time in the log message. When logging to syslog this option should not be needed, but may come in handy when debugging an issue while logging to file.

**log print extended-timestamp 1**
In order to debug time-critical issues this option will print a timestamp with millisecond granularity.

**log print category 1**
Prefix each log message with the category name.

**log print category-hex 1**
Prefix each log message with the category number in hex (<*000b*>).

**log print level 1**
Prefix each log message with the name of the log level.

**log print file 1**
Prefix each log message with the source file and line number. Append the keyword `last` to append the file information instead of prefixing it.

Copyright © 2019 sysmocom - s.f.m.c. GmbH 

## 7.4 Log filters

The default behavior is to filter out everything, i.e. not to log anything. The reason is quite simple: On a busy production setup, logging all events for a given subsystem may very quickly be flooding your console before you have a chance to set a more restrictive filter.

To request no filtering, i.e. see all messages, you may use: `log filter all 1`

In addition to generic filtering, applications can implement special log filters using the same framework to filter on particular context.

For example in OsmoBSC, to only see messages relating to a particular subscriber identified by his IMSI, you may use: `log filter imsi 262020123456789`

## 7.5 Log targets

Each of the log targets represent certain destination for log messages. It can be configured independently by selecting levels (see Section 7.2) for categories (see Section 7.1) as well as filtering (see Section 7.4) and other options like `logging timestamp` for example.

### 7.5.1 Logging to the VTY

Logging messages to the interactive command-line interface (VTY) is most useful for occasional investigation by the system administrator.

Logging to the VTY is disabled by default, and needs to be enabled explicitly for each such session. This means that multiple concurrent VTY sessions each have their own logging configuration. Once you close a VTY session, the log target will be destroyed and your log settings be lost. If you re-connect to the VTY, you have to again activate and configure logging, if you wish.

To create a logging target bound to a VTY, you have to use the following command: `logging enable` This doesn't really activate the generation of any output messages yet, it merely creates and attaches a log target to the VTY session. The newly-created target still doesn't have any filter installed, i.e. *all log messages will be suppressed by default*

Next, you can configure the log levels for desired categories in your VTY session. See Section 7.1 for more details on categories and Section 7.2 for the log level details.

For example, to set the log level of the Call Control category to debug, you can use: `log level cc debug`

Finally, after having configured the levels, you still need to set the filter as it's described in Section 7.4.

---

**Tip**

If many messages are being logged to a VTY session, it may be hard to impossible to still use the same session for any commands. We therefore recommend to open a second VTY session in parallel, and use one only for logging, while the other is used for interacting with the system. Another option would be to use different log target.

---

To review the current vty logging configuration, you can use: `show logging vty`

### 7.5.2 Logging to the ring buffer

To avoid having separate VTY session just for logging output while still having immediate access to them, one can use `alarms` target. It lets you store the log messages inside the ring buffer of a given size which is available with `show alarms` command.

It's configured as follows:

```
OsmoBSC> enable
OsmoBSC# configure terminal
OsmoBSC(config)# log alarms 98
OsmoBSC(config-log)#
```

In the example above 98 is the desired size of the ring buffer (number of messages). Once it's filled, the incoming log messages will push out the oldest messages available in the buffer.

Copyright © 2019 sysmocom - s.f.m.c. GmbH 

### 7.5.3 Logging via gsmtap

GSMTAP is normally a pseudo-header format that enables the IP-transport of GSM (or other telecom) protocols that are not normally transported over IP. For example, the most common situation is to enable GSMTAP in OsmoBTS or OsmoPCU to provide GSM-Um air interface capture files over IP, so they can be analyzed in wireshark.

GSMTAP logging is now a method how Osmocom software can also encapsulate its own log output in GSMTAP frames. We're not trying to re-invent rsyslog here, but this is very handy When debugging complex issues. It enables the reader of the pcap file containing GSMTAP logging together with other protocol traces to reconstruct exact chain of events. A single pcap file can then contain both the log output of any number of Osmocom programs in the same timeline of the messages on various interfaces in and out of said Osmocom programs.

It's configured as follows:

```
OsmoBSC> enable
OsmoBSC# configure terminal
OsmoBSC(config)# log gsmtap 192.168.2.3 ❶
OsmoBSC(config-log)#
```

❶ Configure a GSMTAP logging target recording to 192.168.2.3

The hostname/ip argument is optional: if omitted the default 127.0.0.1 will be used. The log strings inside GSMTAP are already supported by Wireshark. Capturing for `port 4729` on appropriate interface will reveal log messages including source file name and line number as well as application. This makes it easy to consolidate logs from several different network components alongside the air frames. You can also use Wireshark to quickly filter logs for a given subsystem, severity, file name etc.

The GSMTAP logging target also has different implementation I/O modes, which can be configured during the VTY "log gmstap" VTY command:

**nonblocking-io**
: Use non-blocking, synchronous I/O. This is the default mode. It will set the GSMTAP source socket as non-blocking, and attempt synchronous write to the UDP socket, discarding a message if the UDP socket's send buffer is full during write.

**blocking-io**
: Use blocking, synchronous I/O. It will use the GSMTAP source socket in blocking mode, and attempt synchronous write to the UDP socket, blocking the entire thread if the UDP socket's send buffer is full during write. This I/O mode should not be used in any osmocom program driven by an event loop (which includes any program using a VTY), since it may block the thread for an unspecified (potentially long or even infinite) amount of time and hence prevent the event loop to run and process events. This I/O mode should only be used under specific circumstances, like debugging a crash, in which case it's interesting to get the logging output to the UDP socket and hence recorded before the program crashes.

**wq**
: Workqueue mode: Use non-blocking, asynchronous I/O. It will set the GSMTAP source socket as non-blocking, and attempt asynchronous write to the UDP socket by means of a work queue driven by the process poll event loop, delaying writing to the UDP socket until there's enough free space in its send buffer. Since the workqueue has a maximum size, messages will be discarded similar to *nonblocking-io* once the UDP socket's send buffer **and** the GSMTAP log target workqueue are full. This I/O mode provides some extra buffering space in the userspace workqueue, at the expense of extra overhead and delay to transmit messages asynchronously waiting for POLLOUT event from the UDP socket.

In any of the I/O modes above, the UDP socket send buffer size of the GSMTAP log target influences where messages may be dropped (*nonblocking-io*\|*wq*) or the process blocked (*blocking-io*) whenever a high number of messages or test are transmitted in a short time. For that reason, the GSMTAP log target will attempt to set the `SO_SNDBUF` sockopt of the UDP socket to 4MB. In Linux, the `SO_SNDBUF` value being set on the socket will be trimmed by the kernel to a maximum allowed by sysctl `net.core.wmem_max`. Hence, if you want to be on a safer side minimizing dropping messages or blocking the process (depending on the I/O mode set), make sure sysctl `net.core.wmem_max` on your system is set to `4194304` (4MB) or more.

---

**Note**
Linux Kernel used to have a `net.core.wmem_max` of `212992`(208KB) until recently (2025, a6d4f25888b83b8300aef28d9ee22765c1cc9b34), where it was increased to `4194304` (4MB) .

---

Copyright © 2019 sysmocom - s.f.m.c. GmbH 

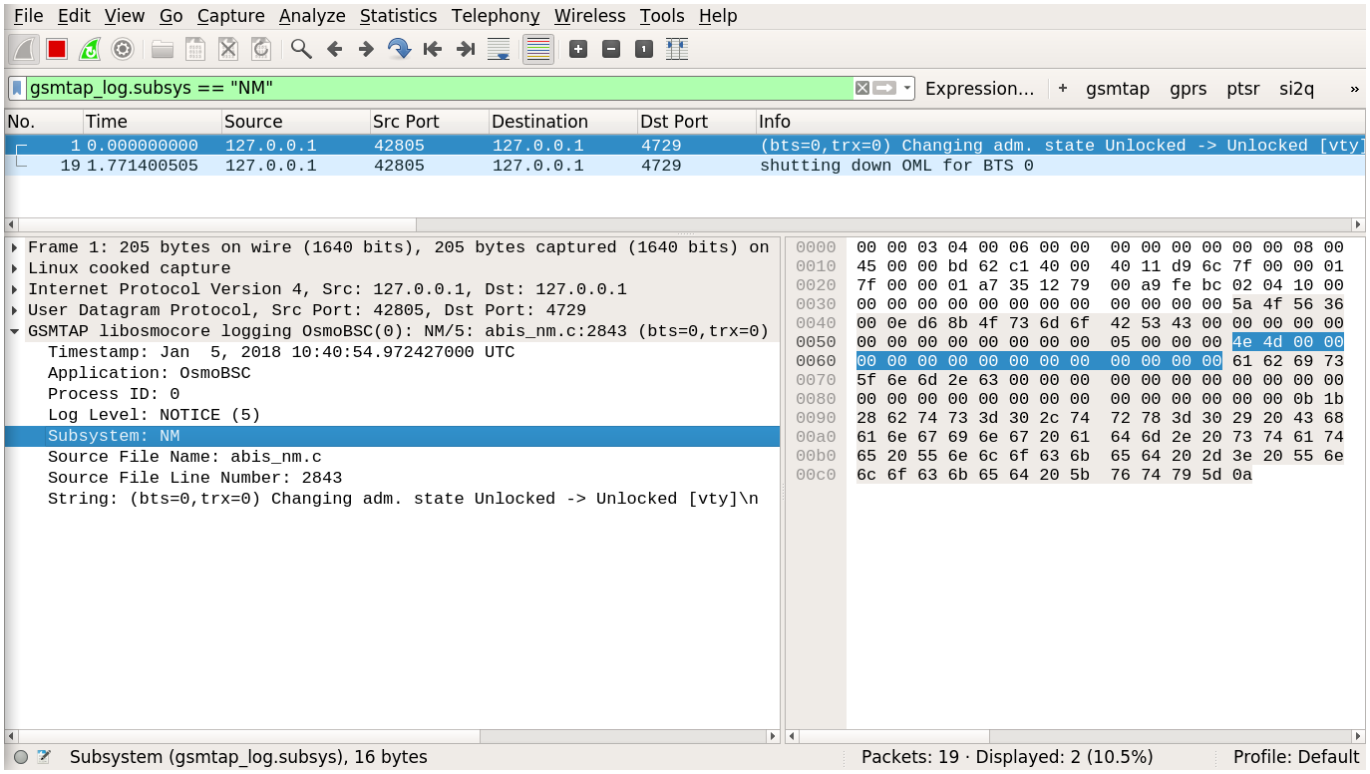



Figure 2: Wireshark with logs delivered over GSMTAP

**Note**

Every time you generate GSMTAP messages and send it to a unicast (non-broadcast/multicast) IP address, please make sure that the destination IP address actually has a socket open on the specified port, or drops the packets in its packet filter. If unicast GSMTAP messages arrive at a closed destination UDP port, the operating system will likely generate ICMP port unreachable messages. Those ICMP messages in turn will, when arriving at the source (the host on which you run the Osmocom software sending GSMTAP), suppress generation of further GSMTAP messages for some time, resulting in incomplete files. In case of doubt, either send GSMTAP to multicast IP addresses, or run something like `nc -l -u -p 4729 > /dev/null` on the destination host to at the GSMTAP port and discard anything arriving at it. If the GSMTAP recording address is identified as a local address, upon creation of the GSMTAP log target, an attempt will be made to create a dummy sink socket to avoid such problems automatically.

### 7.5.4 Logging to a file

As opposed to Logging to the VTY, logging to files is persistent and stored in the configuration file. As such, it is configured in sub-nodes below the configuration node. There can be any number of log files active, each of them having different settings regarding levels / subsystems.

To configure a new log file, enter the following sequence of commands:

```
OsmoBSC> enable
OsmoBSC# configure terminal
OsmoBSC(config)# log file /path/to/my/file ❶
OsmoBSC(config-log)#
```

❶ Configure a file logging target writing to /path/to/my/file

This leaves you at the config-log prompt, from where you can set the detailed configuration for this log file. The available commands at this point are identical to configuring logging on the VTY, they include `logging filter`, `logging level` as well as `logging color` and `logging timestamp`.

The file logging target will by default use a synchronous non-blocking I/O write. If writing cannot be performed synchronously (would block), the message is stored in a workqueue where asynchronous write (driven by poll event loop) will be performed at a later stage. The file logging target can be configured during the VTY `log file` VTY command to perform blocking I/O with the `blocking-io` parameter. This I/O mode should not be used in any osmocom program driven by an event loop (which includes any program using a VTY), since it may block the thread for an unspecified (potentially long or even infinite) amount of time and hence prevent the event loop to run and process events. This I/O mode should only be used under specific circumstances, like debugging a crash, in which case it's interesting to get the logging output to the UDP socket and hence recorded before the program crashes.

**Tip**
Don't forget to use the `copy running-config startup-config` (or its short-hand `write file`) command to make your logging configuration persistent across application re-start.

**Note**
libosmocore provides file close-and-reopen support by SIGHUP, as used by popular log file rotating solutions such as https://github.com/logrotate/logrotate found in most GNU/Linux distributions.

### 7.5.5 Logging to syslog

syslog is a standard for computer data logging maintained by the IETF. Unix-like operating systems like GNU/Linux provide several syslog compatible log daemons that receive log messages generated by application programs.

libosmocore based applications can log messages to syslog by using the syslog log target. You can configure syslog logging by issuing the following commands on the VTY:

```
OsmoBSC> enable
OsmoBSC# configure terminal
OsmoBSC(config)# log syslog daemon
OsmoBSC(config-log)#
```

This leaves you at the config-log prompt, from where you can set the detailed configuration for this log file. The available commands at this point are identical to configuring logging on the VTY, they include `logging filter`, `logging level` as well as `logging color` and `logging timestamp`.

**Tip**
Syslog daemons will normally automatically prefix every message with a time-stamp, so you should disable the libosmocore time-stamping by issuing the `logging timestamp 0` command.

**Note**
Typical syslog implementations (such as glibc's one) write to *AF_UNIX* socket in `/dev/log` in a synchronous blocking I/O fashion, which means transmitting a big amount of messages or text over it in a short amount of time (before syslogd has time to read it) could potentially block the osmocom program's thread issuing the logging. It is hence advised avoiding use of this log target to log notable amounts of information in any osmocom program driven by an event loop (which includes any program using a VTY), since it may block the thread for an unspecified (potentially long or even infinite) amount of time and hence prevent the event loop to run and process events.

Copyright © 2019 sysmocom - s.f.m.c. GmbH 

### 7.5.6 Logging to systemd-journal

systemd has been adopted by the majority of modern GNU/Linux distributions. Along with various daemons and utilities it provides systemd-journald [1] - a daemon responsible for event logging (syslog replacement). libosmocore based applications can log messages directly to systemd-journald.

The key difference from other logging targets is that systemd based logging allows to offload rendering of the meta information, such as location (file name, line number), subsystem, and logging level, to systemd-journald. Furthermore, systemd allows to attach arbitrary meta fields to the logging messages [2], which can be used for advanced log filtering.

[1] https://www.freedesktop.org/software/systemd/man/systemd-journald.service.html [2] https://www.freedesktop.org/software/-systemd/man/systemd.journal-fields.html

It was decided to introduce libsystemd as an optional dependency, so it needs to be enabled explicitly at configure/build time:

```
$ ./configure --enable-systemd-logging
```

---

**Note**
Recent libosmocore packages provided by Osmocom for Debian and CentOS are compiled **with** libsystemd (https://gerrit.osmocom.org/c/libosmocore/+/22651).

---

You can configure systemd based logging in two ways:

**Example: systemd-journal target with offloaded rendering**

```
log systemd-journal raw ❶
 logging filter all 1
 logging level set-all notice
```

❶ `raw` logging handler, rendering offloaded to systemd.

In this example, logging messages will be passed to systemd without any meta information (time, location, level, category) in the text itself, so all the printing parameters like `logging print file` will be ignored. Instead, the meta information is passed separately as *fields* which can be retrieved from the journal and rendered in any preferred way.

```
# Show Osmocom specific fields
$ journalctl --fields | grep OSMO

# Filter messages by logging subsystem at run-time
$ journalctl OSMO_SUBSYS=DMSC -f

# Render specific fields only
$ journalctl --output=verbose \
        --output-fields=SYSLOG_IDENTIFIER,OSMO_SUBSYS,CODE_FILE,CODE_LINE,MESSAGE
```

See `man 7 systemd.journal-fields` for a list of default fields, and `man 1 journalctl` for general information and available formatters.

**Example: systemd-journal target with libosmocore based rendering**

```
log systemd-journal ❶
 logging filter all 1
 logging print file basename
 logging print category-hex 0
 logging print category 1
 logging print level 1
 logging timestamp 0 ❷
 logging color 1 ❸
 logging level set-all notice
```

Copyright © 2019 sysmocom - s.f.m.c. GmbH 

❶ Generic logging handler, rendering is done by libosmocore.

❷ Disable timestamping, systemd will timestamp every message anyway.

❸ Colored messages can be rendered with `journalctl --output=cat`.

In this example, logging messages will be pre-processed by libosmocore before being passed to systemd. No additional fields will be attached, except the logging level (`PRIORITY`). This mode is similar to *syslog* and *stderr*.

**Note**
systemd-journal logging target uses libsystemd, which implements *Native Journal Protocol* (*AF_UNIX _DATAGRAM* socket). That socket is internally configured in synchronous blocking I/O mode, which means transmitting a big amount of messages or text over it in a short amount of time (before systemd-journald has time to read it) could potentially block the osmocom program's thread issuing the logging. libsystemd currently sets the *AF_UNIX* socket `SO_SNDBUF` to 8MB. In Linux, the `SO_SNDBUF` value being set on the socket will be trimmed by the kernel to a maximum allowed by sysctl `net.core.wmem_max`. Hence, if you want to be on a safer side minimizing dropping messages or blocking the process (depending on the I/O mode set), make sure sysctl `net.core.wmem_max` on your system is set to `8388608` (8MB) or more. Besides, the kernel limits the amount of messages queued in *AF_UNIX* sockets to value set in sysctl `net.unix.max_dgram_qlen`, so you may also want to increase that one to be on the safe side if planning to log with a high frequency.

**Note**
Linux Kernel used to have a `net.core.wmem_max` of `212992`(208KB) until recently (2025, a6d4f25888b83b8300aef28d9ee22765c1cc9b34), where it was increased to `4194304` (4MB) .

### 7.5.7 Logging to stderr

If you're not running the respective application as a daemon in the background, you can also use the stderr log target in order to log to the standard error file descriptor of the process.

In order to configure logging to stderr, you can use the following commands:

```
OsmoBSC> enable
OsmoBSC# configure terminal
OsmoBSC(config)# log stderr
OsmoBSC(config-log)#
```

The stderr logging target can be configured during the VTY `log stderr` VTY command to perform blocking I/O with the `blocking-io` parameter. See Section 7.5.4 for more information.

# 8 Signaling Networks: SS7 and SIGTRAN

Classic digital telephony networks (whether wired or wireless) use the ITU-T SS7 (Signaling System 7) to exchange signaling information between network elements.

Most of the ETSI/3GPP interfaces in the GSM and UMTS network are also based on top of [parts of] SS7. This includes, among others, the following interfaces:

- *A* interface between BSC and MSC
- *IuCS* interface between RNC (or HNB-GW) and MSC
- *IuPS* interface between RNC (or HNB-GW) and SGSN

**Note**
This does not include the A-bis interface between BTS and BSC. While Abis traditionally is spoken over the same physical TDM circuits as SS7, the protocol stack from L2 upwards is quite different (Abis uses LAPD, while SS7 uses MTP)!

Copyright © 2019 sysmocom - s.f.m.c. GmbH 

## 8.1 Physical Layer

The traditional physical layer of SS7 is based on TDM (time division multiplex) links of the PDH/SDH family, as they were common in ISDN networks. Some people may know their smallest incarnation as so-called E1/T1 links. It can run either on individual 64kBps timeslots of such a link, or on entire 2Mbps/1.5MBps E1/T1 links.

There are also specifications for SS7 over ATM, though it is unclear to the author if this is actually still used anywhere.

On top of the Physical Layer is the Message Transfer Part (MTP).

## 8.2 Message Transfer Part (MTP)

MTP is the lower layer of the SS7 protocol stack. It is comprised of two sub-layers, called MTP2 and MTP3.

Nodes in a MTP network are addressed by their unique PC (Point Code).

A *MTP Routing Label* is in the MTP header and indicates the *Originationg Point Code* (OPC) as well as the *Destination Point Code* (DPC) and the *Service Indicator Octet* (SIO). The SIO is used to de-multiplex between different upper-layer protocol such as ISUP, TUP or SCCP.

Routing is performed by means of routers with routing tables, similar to routing is performed in IP networks. Even the concept of a *point code mask* analogous to the *netmask* exists.

Routers are connected with one another over one or more *Link Sets*, each comprised of one or multiple *Links*. Multiple Links in a Linkset exist both for load sharing as well as for fail over purposes.

### 8.2.1 Point Codes

The length of point codes depends on the particular MTP dialect that is used. In the 1980s, when international telephony signaling networks were established, most countries had their own national dialects with certain specifics.

Today, mostly the ITU and ANSI variants survive. The ITU variant uses 14bit point codes, while the ANSI variant uses 24 bit point code length.

Point Codes can be represented either as unsigned integers, or grouped. Unfortunately there is no standard as to their representation. In ITU networks, the *3.8.3* notation is commonly used, i.e. one decimal for the first 3 bits, followed by one decimal for the center 8 bits, followed by another decimal for the final 3 bits.

**Example**
The Point Code **1.5.3** (in 3.8.3 notation) is $1*2^{11} + 5*2^3 + 3$ = **2091 decimal**.

## 8.3 Higher-Layer Protocols

There are various higher-layer protocols used on top of MTP3, such as TUP, ISUP, BICC as well as SCCP. Those protocols exist side-by-side on top of MTP3, similar to e.g. ICMP, TCP and UDP existing side-by-side on top of IP.

In the context of cellular networks, SCCP is the most relevant part.

## 8.4 Signaling Connection Control Part (SCCP)

SCCP runs on top of MTP3 and creates something like an overlay network on top of it. SCCP communication can e.g. span multiple different isolated MTP networks, each with their own MTP dialect and addressing.

SCCP provides both connectionless (datagram) and connection-oriented services. Both are used in the context of cellular networks.

Copyright © 2019 sysmocom - s.f.m.c. GmbH 

### 8.4.1 SCCP Addresses

SCCP Addresses are quite complex. This is due to the fact that it is not simply one address format, but in fact a choice of one or multiple different types of addresses.

SCCP Addresses exist as *Calling Party* and *Called Party* addresses. In the context of connectionless datagram services, the sender is always the Calling Party, and the receiver the Called Party. In connection-oriented SCCP, they resemble the initiator and recipient of the connection.

Table 2: SCCP Address Parts

| Acronym | Name | Description |
|---|---|---|
| SSN | Sub-System Number | Describes a given application such as e.g. a GSM MSC, BSC or HLR. Can be compared to port numbers on the Internet |
| PC | Point Code | The Point Code of the underlying MTP network |
| GT | Global Title | What most people would call a "phone number". However, Global Titles come in many different numbering plans, and only one of them (E.164) resembles actual phone numbers. |
| RI | Routing Indicator | Determines if message shall be routed on PC+SSN or on GT basis |

### 8.4.2 Global Titles

A Global Title is a (typically) globally unique address in the global telephony network. The body of the Global Title consists of a series of BCD-encoded digits similar to what everyone knows as phone numbers.

A GT is however not only the digits of the "phone number", but also some other equally important information, such as the *Numbering Plan* as well as the *Nature of Address Indication*.

Table 3: Global Title Parts

| Acronym | Name | Description |
|---|---|---|
| GTI | Global Title Indicator | Determines the GT Format. Ranges from no GT (0) to GT+TT+NP+ES+NAI (4) |
| NAI | Nature of Address Indicator | Exists in GTI=1 and is sort of a mixture of TON + NPI |
| TT | Translation Type | Used as a look-up key in Global Title Translation Tables |
| NP | Numbering Plan | Indicates ITU Numbering Plan, such as E.164, E.212, E.214 |
| ES | Encoding Scheme | Just a peculiar way to indicate the length of the digits |
| - | Signals | The actual "phone number digits" |

For more information about SCCP Addresses and Global Titles, please refer to [itu-t-q713]

### 8.4.3 Global Title Translation (GTT)

Global Title Translation is a process of re-writing the Global Title on-the-fly while a signaling message passes a STP.

Basically, a SCCP message is first transported by MTP3 on the MTP level to the Destination Point Code indicated in the MTP Routing Label. This process uses MTP routing and is transparent to SCCP.

Once the SCCP message arrives at the MTP End-Node identified by the Destination Point Code, the message is handed up to the local SCCP stack, which then may implement Global Title Translation.

The input to the GTT process is

Copyright © 2019 sysmocom - s.f.m.c. GmbH 

- the destination address of the SCCP message
- a local list/database of Global Title Translation Rules

The successful output of he GTT includes

- A new Routing Indicator
- The Destination Point Code to which the message is forwarded on MTP level
- a Sub-system Number (if RI is set to "Route on SSN")
- a new Global Title (if RI is set to "Route on GT"), e.g. with translated digits.

Between sender and recipient of a signaling message, there can be many instances of Global Title Translation (up to 15 as per the hop counter).

For more information on Global Title Translation, please refer to [itu-t-q714].

#### 8.4.4 Peculiarities of Connection Oriented SCCP

Interestingly, Connection-Oriented SCCP messages carry SCCP Addresses **only during connection establishment**. All data messages during an ongoing connection do not contain a Called or Calling Party Address. Instead, they are routed only by the MTP label, which is constructed from point code information saved at the time the connection is established.

This means that connection-oriented SCCP can not be routed across MTP network boundaries the same way as connectionless SCCP messages. Instead, an STP would have to perform *connection coupling*, which is basically the equivalent of an application-level proxy between two SCCP connections, each over one of the two MTP networks.

This is probably mostly of theoretical relevance, as connection-oriented SCCP is primarily used between RAN and CN of cellular network inside one operator, i.e. not across multiple MTP networks.

### 8.5 SIGTRAN - SS7 over IP Networks

At some point, IP based networks became more dominant than classic ISDN networks, and 3GPP as well as IETF were working out methods in which telecom signaling traffic can be adapted over IP based networks.

Initially, only the edge of the network (i.e. the applications talking to the network, such as HLR or MSC) were attached to the existing old SS7 backbone by means as SUA and M3UA. Over time, even the links of the actual network backbone networks became more and more IP based.

In order to replace existing TDM-based SS7 links/linksets with SIGTRAN, the M2UA or M2PA variants are used as a kind of drop-in replacement for physical links.

All SIGTRAN share that while they use IP, they don't use TCP or UDP but operate over a (then) newly-introduced Layer 4 transport protocol on top of IP: SCTP (Stream Control Transmission Protocol).

Despite first being specified in October 2000 as IETF RFC 2960, it took a long time until solid implementations of SCTP ended up in general-purpose operating systems. SCTP is not used much outside the context of SIGTRAN, which means implementations often suffer from bugs, and many parts of the public Internet do not carry SCTP traffic due to restrictive firewalls and/or ignorant network administrators.

#### 8.5.1 SIGTRAN Concepts / Terminology

Like every protocol or technology, SIGTRAN brings with it its own terminology and concepts. This section tries to briefly introduce them. For more information, please see the related IETF RFCs.

Copyright © 2019 sysmocom - s.f.m.c. GmbH 

#### 8.5.1.1 Signaling Gateway (SG)

The Signaling Gateway (SG) interconnects the SS7 network with external applications. It translates (parts of) the SS7 protocol stack into an IP based SIGTRAN protocol stack. Which parts at which level of the protocol stack are translated to what depends on the specific SIGTRAN dialect.

A SG is traditionally attached to the TDM-Based SS7 network and offers SIGTRAN/IP based applications a way to remotely attach to the SS7 network.

A SG typically has STP functionality built-in, but it is not mandatory.

#### 8.5.1.2 Application Server (AS)

An Application Server is basically a logical entity representing one particular external application (from the SS7 point of view) which is interfaced with the SS7 network by means of one of the SIGTRAN protocols.

An Application Server can have one or more Application Server Processes associated with it. This functionality can be used for load-balancing or fail-over scenarios.

#### 8.5.1.3 Application Server Process (ASP)

An Application Server Process represents one particular SCTP connection used for SIGTRAN signaling between an external application (e.g. a BSC) and the Signaling Gateway (SG).

One Application Server Process can route traffic for multiple Application Servers. In order to differentiate traffic for different Application Servers, the Routing Context header is used.

### 8.5.2 SIGTRAN variants / stackings

SIGTRAN is the name of an IETF working group, which has released an entire group of different protocol specifications. So rather than one way of transporting classic telecom signaling over IP, there are now half a dozen different ones, and all can claim to be an official IETF standard.

FIXME: Overview picture comparing the different stackings

#### 8.5.2.1 MTP3 User Adaptation (M3UA)

M3UA basically "chops off" everything up to and including the MTP3 protocol layer of the SS7 protocol stack and replaces it with a stack comprised of M3UA over SCTP over IP.

M3UA is specified in [ietf-rfc4666].

M3UA is the SIGTRAN variant chosen by 3GPP for A, IuCs and IuPS interfaces over IP.

#### 8.5.2.2 SCCP User Adaptation (SUA)

SUA basically "chops off" everything up to and including the SCCP protocol layer of the SS7 protocol stack and replaces it with a stack comprised of SUA over SCTP over IP.

This means that SUA can only be used for SCCP based signaling, but not for other SS7 protocols like e.g. TUP and ISUP.

SUA is specified in [ietf-rfc3868].

#### 8.5.2.3 MTP2 User Adaptation (M2UA)

M2UA is specified in [ietf-rfc3331].

**Note**
M2UA is not supported in Osmocom SIGTRAN up to this point. Let us know if we can implement it for you!

Copyright © 2019 sysmocom - s.f.m.c. GmbH 

#### 8.5.2.4 MTP2-User Peer-to-Peer Adaptation (M2PA)

M2PA is specified in [ietf-rfc4165].

**Note**
M2PA is not supported in Osmocom SIGTRAN up to this point. Let us know if we can implement it for you!

### 8.5.3 SIGTRAN security

There simply is none. There are some hints that TLS shall be used over SCTP in order to provide authenticity and/or confidentiality for SIGTRAN, but this is not widely used.

As telecom signaling is not generally carried over public networks, private networks/links by means of MPLS, VLANs or VPNs such as IPsec are often used to isolate and/or secure SIGTRAN.

Under no circumstances should you use unsecured SIGTRAN with production data over the public internet!

### 8.5.4 IPv6 support

SCTP (and thus all the higher layer protocols of the various SIGTRAN stackings) operates on top of both IPv4 and IPv6. As the entire underlying IP transport is transparent to the SS7/SCCP applications, there is no restriction on whether to use SIGTRAN over IPv4 or IPv6.

### 8.5.5 SCTP multi-homing in SIGTRAN

SCTP, unlike more traditional IP L4 protocols (TCP, UDP) doesn't work based on a *connection* between source IP:port and Destination IP:port.

Instead, SCTP creates *associations* between two endpoints, both of which can have any number of IP addresses. This means that in case of network outage, traffic can continue to flow through any of the IP addresses of that association.

The Linux kernel by default advertises all IP addresses of the local system to the peer. This can be seen when inspecting the SCTP INIT chunk e.g. in wireshark. While this may be a reasonable default in some use cases, it is not always the best idea. Imagine addresses of internal/private IP networks, for example local bridge devices between lxc or docker containers, or local VMs. Such addresses have no significance beyond the local machine.

Subsequently, libosmo-sigtran allows the user to explicitly select which local IP addresses shall be used in SCTP multi-homing for the SIGTRAN associations it manages. The user can achieve this by specifying multiple `local-ip` VTY commands within one `asp` (SCTP client role) or within one `listen m3ua 2905` (SCTP server role).

### 8.5.6 SCTP Primary Address

SCTP has the concept of "primary address" in an association. The primary address is a remote address selected from those announced by the peer, and it is the "active" one chosen to transmit user data. The other remote addresses, that are not used, are kept as backups. They are in general only used to transmit user data whenever the SCTP implementation decides to change the primary address, be it due to user policy configuration change or due to the previous primary link becoming unusable. Only confirmed remote addresses (through HEARTBEAT mechanism) are electable to be used as primary address.

By default, the Linux kernel SCTP stack implementation will probably take the first remote address provided at connect() time in order to start the initial handshake, and continue with next provided remote addresses if the first one fails to confirm the handshake. The remote address which successfully confirmed the handshake is then used as a primary address (since it's likely the only confirmed so far), and will be kept until the link is considered down.

Some deployment setups may have requirements on preferred links to be used when transmitting data (eg. network setups with primary and secondary paths). This can be accomplished by explicitly notifying the kernel to use one of the remote addresses through the SCTP_PRIMARY_ADDR sockopt, plus monitoring the address availability changes on the socket and re-enforcing the primary address when it becomes available again. This is supported in the Osmocom SIGTRAN stack by using the `primary` parameter in one of the `remote-ip` commands under the `asp` node:

Copyright © 2019 sysmocom - s.f.m.c. GmbH 

```
cs7 instance 0
 asp my-asp 2905 0 m3ua
  remote-ip 10.11.12.13
  remote-ip 16.17.18.19 primary ❶
  ...
```

❶ Use 16.17.18.19 as primary address for the SCTP association. User data will be in general transmitted over this path.

### 8.5.7 SCTP Peer Primary Address

The SCTP extension ASCONF (RFC5061) allows, when negotiated and supported by both peers, to dynamically announce to the peer the addition or deletion of IP addresses to the association. It also allows one peer announcing to the other peer the desired IP address it should be using as a primary address when sending data to it.

In the Linux kernel SCTP stack, this is accomplished by setting the sockopt SCTP_SET_PEER_PRIMARY_ADDR, which will trigger an ASCONF SCTP message to the peer with the provided local IP address. This is supported in the Osmocom SIGTRAN stack by using the `primary` parameter in one of the `local-ip` commands under the `asp` node:

```
cs7 instance 0
 asp my-asp 2905 0 m3ua
  local-ip 10.11.12.13
  local-ip 16.17.18.19 primary ❶
  ...
```

❶ Announce 16.17.18.19 to the peer as the primary address to be used when transmitting user data to us.

In order to be able to use this feature, the SCTP association peer must support the ASCONF extension. The extension support is negotiated during the INIT handshake of the association. Furthermore, for ASCONF features to work properly, the assoc also needs to announce/use the AUTH extension, as per RFC5061 section 4.2.7. Otherwise, the peer receiving an SCTP INIT with `ExtensionFeatures=ASCONF,ASCONF_ACK` but without AUTH, will reject the association with an ABORT since it's not complying with specifications (this behavior can be tweaked through sysctl "net.sctp.addip_noauth_enable").

As of the time of writing this documentation (linux 6.4.12) and since basically ever, those extensions are runtime-disabled by default. They can be enabled per socket using the kernel sockopts SCTP_ASCONF_SUPPORTED and SCTP_AUTH_SUPPORTED, and that's what the Osmocom stack is currently doing for all its SCTP sockets.

However, those sockopts are farily new (linux v5.4), which means user running older kernels will see in the logs setting those sockopts fail, but connection will keep ongoing, simply without those features available (so setting `primary` in the configuration won't have any effect here). On those older kernels, if this feature is still desired, it can be used by means of enabling the SCTP extensions in all socket system-wide through sysctl:

```
net.sctp.auth_enable=1
net.sctp.addip_enable=1
```

### 8.5.8 SCTP INIT Parameters

Several SCTP INIT parameters can be configured through VTY, which will be passed to the Linux Kernel SCTP stack and used whenever an association is being established.

On the client side (see Section 8.5.9), the parameters are configured in the `asp` node:

```
cs7 instance 0
 asp my-asp 2905 0 m3ua
  sctp-role client
  sctp-param init num-ostreams 250 ❶
  sctp-param init max-instreams 300 ❷
  sctp-param init max-attempts 3 ❸
  sctp-param init timeout 10000 ❹
  ...
```

Copyright © 2019 sysmocom - s.f.m.c. GmbH 

❶ The number of streams from the server to the client. This value is transmitted during SCTP INIT ACK packet.

❷ Announce to the server that a maximum of up to 300 inbound SCTP streams are supported. This value is transmitted during SCTP INIT packet.

❸ Initial SCTP handshake will be attempted 3 times before considering the connection failed.

❹ Retransmit an SCTP INIT message after 10000 ms if no answer is received.

On the server side (see Section 8.5.9), the parameters are configured in the `listen` node:

```
cs7 instance 0
 asp my-asp 2905 0 m3ua
  sctp-role server
 listen m3ua 2905
  sctp-param init num-ostreams 250 ❶
  sctp-param init max-instreams 300 ❷
  ...
```

❶ Announce to the server that up to 250 outbound SCTP streams (server to client) may be requested. This value is transmitted during SCTP INIT packet, and should be equal or lower to the `max-instreams` value received from the client during SCTP INIT packet.

❷ Announce to the server that a maximum of up to 300 inbound SCTP streams are supported. This value is transmitted during SCTP INIT ACK packet.

### 8.5.9 SCTP role

The *SCTP role* defines which of the two L4 protocol roles SCTP assumes:

- The *SCTP server* role binds to a local port and handles incoming connections from clients
- The *SCTP client* role connects to a remote SCTP sever.

### 8.5.10 M3UA/SUA role

The *M3UA role* (or *SUA role*) determines which role a given peer of a M3UA connection implements. 3GPP specifies the following role:

- *SGP* (Signaling Gateway): The entity connected to the larger SS7 network
- *ASP* (Application Server Process): A client application that connects to the SGW to talk to the SS7 network
- *IPSP* (IP Server Process): M3UA in point-to-point mode

Osmocom (libosmo-sigtran) implements both the SGP and ASP roles, but not the IPSP role.

### 8.5.11 Traffic Modes in SIGTRAN

Whenever an AS consists of multiple ASPs, the traffic mode expresses how messages are distributed between those ASPs.

- *Override*: There is always one active ASP and multiple hot standby ASPs. If the active ASP fails, one of the remaining ASPs will become the new active ASP.
- *Loadshare*: The messages will be distributed between the different ASPs in a way to distribute the load among them. Details are implementation specific.
- *Broadcast*: A copy of every incoming signaling message is sent to *all* the ASPs in broadcast traffic mode.

Osmocom (libosmo-sigtran) implements all above-mentioned traffic modes.

Copyright © 2019 sysmocom - s.f.m.c. GmbH 

# 9 Configure SCCP/M3UA

All CNI programs using SCCP/M3UA act as M3UA ASP role and SCTP client, expecting to connect to a Signalling Gateway (STP/SG) implementing the M3UA SG role as SCTP server. The STP/SG then routes M3UA messages between its ASPs, typically by point-codes.

For an introduction about SCCP/M3UA/SS7/SIGTRAN technology, please see the chapter *Signaling Networks: SS7 and SIGTRAN* in the OsmoSTP user manual.

In an all-Osmocom CNI, the typical simple/minimal usage is:

- OsmoSTP acts as the STP/SG (server role) and routes between the ASP,
- All other Osmocom CNI programs act as SCTP client and provide ASP implementations.

For example, in an all-Osmocom minimal setup,

- OsmoMSC contacts an OsmoSTP and subscribes its point-code 0.23.1;
- then OsmoBSC also contacts the same OsmoSTP, subscribes with its own point-code 1.23.3.
- Using these established links, OsmoBSC initiates an A-interface link by directing a BSSAP RESET message to the MSC's point-code 0.23.1,
- and the RESET ACK response from the MSC is routed back to the BSC's point-code 1.23.3.

The details of SCCP/M3UA are configured in the *cs7* section of the VTY configuration.

Osmocom programs automatically configure missing SCCP/M3UA configuration, by assuming sane defaults for small/minimal all-Osmocom installations, which may not be what you want in larger networks integrating with non-Osmocom core network elements.

If no explicit `routing-key` is set, it may be determined at runtime by negotiation with OsmoSTP — see OsmoSTP manual chapter "Osmocom M3UA Routing Key Management Extensions", regarding config option `accept-asp-connections dynamic-permitted`.

The complete active configuration of an Osmocom program can be obtained by the VTY command `show cs7 config` (the usual `show running-config` omits automatically configured items). Here is an example of OsmoMSC's default configuration:

```
OsmoMSC> show cs7 config
cs7 instance 0
 point-code 0.23.1
 asp asp-clnt-OsmoMSC-A-Iu 2905 0 m3ua
  remote-ip 127.0.0.1
  role asp
  sctp-role client
 as as-clnt-OsmoMSC-A-Iu m3ua
  asp asp-clnt-OsmoMSC-A-Iu
  routing-key 2 0.23.1
```

At the time of writing, SCCP/M3UA links involving Osmocom program are:

- A-interface: OsmoBSC to OsmoMSC
- IuCS-interface: OsmoHNBGW to OsmoMSC
- IuPS-interface: OsmoHNBGW to OsmoSGSN
- Lb-interface: OsmoSMLC to OsmoBSC

Copyright © 2019 sysmocom - s.f.m.c. GmbH 

On the SCTP/IP level, those connections are actually all established from the respective program (BSC, MSC, HNBGW, SGSN, SMLC) to OsmoSTP. Hence, if you look at the traffic in a protocol analyzer like wireshark, at IP level, you will see each of those programs establishing an SCTP association from a random local IP to the well-known SCTP port for M3UA (2905) at the OsmoSTP.

Those star-connections for M3UA/SCTP then are the transport network for higher level protocols like SCCP. OsmoSTP then acts as central router for SCCP-level message exchange between all the connected programs.

## 9.1 Connect to STP Instance

Establishing an SCCP/M3UA link towards a remote STP instance can be configured as:

```
cs7 instance 0
 asp my-asp 2905 0 m3ua
  # IP address of the remote STP:
  remote-ip 10.23.24.1
  # optional: local bind to a specific IP
  local-ip 10.9.8.7
  role asp
  sctp-role client
```

Be aware that such an `asp` needs to be linked to an `as`, see Section 9.5.

By default, an STP instance is assumed to listen on the default M3UA port (2905) on the local host. That means in general `127.0.0.1` will be used as default remote SCTP address, and `::1` will be added to the SCTP association if IPv6 support is available on the system.

**Note**
OsmoSTP listens by default on `::` if IPv6 is enabled on the system, and on `0.0.0.0` otherwise. Address `::` actually superseeds `0.0.0.0`, meaning it will listen on all IPv4 and IPv6 addresses available on the system.



**Caution**
Some applications overwrite the default target remote address to be `localhost`. If IPv6 support is available on the system, `localhost` will usually resolve to `::1`, otherwise it will usually resolve to `127.0.0.1`.

## 9.2 Local Point-Code

Each CNI program on an SCCP/M3UA link typically has a local point-code, configurable by:

```
cs7 instance 0
 point-code 7.65.4
```

If an explicit routing context is configured, this point-code is repeated in the `routing-key` configuration:

```
cs7 instance 0
 point-code 0.23.1
 as my-as m3ua
  routing-key 2 0.23.1
```

See also Section 9.4.

Copyright © 2019 sysmocom - s.f.m.c. GmbH 

## 9.3 Remote Point-Code

Programs establishing communication across SCCP links need a remote SCCP address, typically by point-code, to contact.

For example,

- OsmoBSC needs to know the MSC's point-code, to be able to establish the A-interface.
- OsmoHNBGW needs to know the MSC's point-code, to be able to establish the IuCS-interface.
- OsmoHNBGW needs to know the SGSN's point-code, to be able to establish the IuPS-interface.

To maintain remote SCCP addresses, each `cs7` instance maintains an SCCP address book:

```
cs7 instance 0
 sccp-address remote-pc-example
  point-code 1.23.1
```

This address book entry on its own has no effect. It is typically referenced by specific configuration items depending on the individual programs.

Examples:

- An OsmoBSC configures the MSC's remote SCCP address:

```
cs7 instance 0
 sccp-address my-remote-msc
  point-code 1.23.1
msc 0
 msc-addr my-remote-msc
```

- An HNBGW configures both the remote MSC's and SGSN's SCCP addresses:

```
cs7 instance 0
 sccp-address my-msc
  point-code 0.23.1
 sccp-address my-sgsn
  point-code 0.23.2
hnbgw
 iucs
  remote-addr my-msc
 iups
  remote-addr my-sgsn
```

Besides a point-code, an SCCP address can have several routing indicators:

- PC: routing by point-code is the default for Osmocom.
- GT: routing by Global Title is configurable by `routing-indicator GT`.
- IP: routing by IP address is configurable by `routing-indicator IP`.

In OsmoSTP, only routing by point-code is currently implemented.

Copyright © 2019 sysmocom - s.f.m.c. GmbH 

## 9.4 Point-Code Format

Point-codes can be represented in various formats. For details, see OsmoSTP manual, chapter "Point Codes".

By default, Osmocom uses a point-code representation of 3.8.3, i.e. first digit of 3 bit, second digit of 8 bit, and third digit of 3 bit.

```
cs7 instance 0
 point-code format 3 8 3
 point-code 0.23.1
```

Often, point-codes are also represented as a single decimal number:

```
cs7 instance 0
 point-code format 24
 point-code 185
```

It is also possible to use a dash as delimiter.

```
cs7 instance 0
 point-code delimiter dash
 point-code 0-23-1
```

## 9.5 AS and ASP

Each CNI program needs at least one Application Server `as` and one Application Server Process `asp` configured on its `cs7` to be able to communicate on SCCP/M3UA. An `asp` needs to be part of at least one `as`. For details, see the OsmoSTP manual, chapters "Application Server" and "Application Server Process".

In Osmocom's `cs7`, any amount of `as` and `asp` can be configured by name, and an `as` references the `asp` entries belonging to it by their names.

In a simple/minimal Osmocom setup, an Osmocom CNI program would have exactly one `as` with one `asp`.

For example:

```
cs7 instance 0
 asp my-asp 2905 0 m3ua
  # where to reach the STP:
  remote-ip 127.0.0.1
  role asp
  sctp-role client
 as my-as m3ua
  asp my-asp
```

In Osmocom CNI programs, it is possible to omit the `as` and/or `asp` entries, which the program will then attempt to configure automatically.

When configuring both `as` and `asp` manually, make sure to link them by name. For example, the following configuration will **fail**, because `as` and `asp` are not linked:

```
cs7 instance 0
 asp my-asp 2905 0 m3ua
  remote-ip 127.0.0.1
  role asp
  sctp-role client
 as my-as m3ua
  routing-key 2 0.23.1
```

To **fix** above config, link the `asp` to an `as` by adding `asp my-asp`:

Copyright © 2019 sysmocom - s.f.m.c. GmbH 

```
cs7 instance 0
 asp my-asp 2905 0 m3ua
  remote-ip 127.0.0.1
  role asp
  sctp-role client
 as my-as m3ua
  asp my-asp
  routing-key 2 0.23.1
```

## 9.6 Subsystem Number (SSN)

Osmocom CNI programs typically route SCCP/M3UA messages by PC+SSN: each ASP, having a given SCCP address, receives messages for one or more specific subsystems, identified by a Subsystem Number (SSN).

For example, the A-interface between BSC and MSC uses SSN = BSSAP (254). In Osmocom programs, SSNs do not need to be configured; they implicitly, naturally relate to the interfaces that a program implements.

For example, OsmoBSC takes the configured remote MSC's SCCP address and adds the SSN = BSSAP to it in order to contact the MSC's A-interface. To receive A-interface messages from the MSC, OsmoBSC subscribes a local user for this SSN on the ASP.

## 9.7 Routing Context / Routing Key

In SCCP/M3UA, messages can be routed by various Routing Indicators (PC+SSN, PC, GT, ...). Osmocom CNI programs typically use PC+SSN as Routing Indicator.

On the SG (for example OsmoSTP), each ASP's distinct Routing Indicator needs to be indexed by a distinct Routing Context (a simple index number scoped per SG), to forward M3UA to the correct peer.

The Osmocom SG implementation employs Routing Key Management (RKM, see OsmoSTP manual) to automatically determine a distinct Routing Context index for each connected ASP. Routing Contexts can also be configured manually — some non-Osmocom SG implementations require this.

Each Routing Context is associated with a Routing Indicator and address; this association is called a Routing Key.

For example, to configure an OsmoBSC with a local point-code of 1.23.3 to receive M3UA with Routing Context of 2 and RI=PC:

```
cs7 instance 0
 point-code 1.23.3
 as my-as m3ua
  routing-key 2 1.23.3
```

Osmocom programs so far implement Routing Keys by Destination Point Code (DPC), plus optional Subsystem Number (SSN) and/or Service Indicator (SI):

```
routing-key RCONTEXT DPC
routing-key RCONTEXT DPC si (aal2|bicc|b-isup|h248|isup|sat-isup|sccp|tup)
routing-key RCONTEXT DPC ssn SSN
routing-key RCONTEXT DPC si (aal2|bicc|b-isup|h248|isup|sat-isup|sccp|tup) ssn SSN
```

### 9.7.1 M3UA without Routing Context IE / Routing Context *0*

As per the M3UA specification, the use of the routing context IE is optional as long as there is only one AS within an ASP. As soon as there are multiple different AS within one ASP, the routing context IE is mandatory, as it is the only clue to differentiate which of the ASs a given message belongs to.

In the Osmocom M3UA implementation, it is generally assumed that a routing context IE is always used, for the sake of clarity.

Copyright © 2019 sysmocom - s.f.m.c. GmbH 

However, the routing context ID of *0* has the special meaning of *do not encode a routing context IE on transmit*.

So if you configure an application like OsmoBSC to use routing context 0, then no routing context IE will be included in outbound M3UA messages.

This special interpretation of *0* within the Osmocom M3UA implementation however means that we can not represent M3UA with a routing context IE that actually contains *0* as a numeric identifier.

So you only have the following options: * Using M3UA with routing context (1..N) * Using M3UA without routing context (0)

### 9.7.2 Example: Static Routing

Osmocom SS7 supports dynamic routing key registration via M3UA Routing Key Management (RKM), allowing minimal SS7 configuration. If all of your components support dynamic RKM, you should probably use it: see `accept-asp-connections dynamic-permitted` in osmo-stp.cfg.

This chapter explains how to configure osmo-stp if dynamic RKM is not an option.

In this example, let's connect osmo-bsc via osmo-stp to osmo-msc using only static SS7 routing.

```
BSC        <--RK-1-->  STP         <--RK-3--> MSC
IP 1.1.1.1             IP 2.2.2.2             IP 3.3.3.3
M3UA 2905              M3UA 2905              M3UA 2905
PC 1.1.1                                      PC 3.3.3
```

Every one static route fanning out from STP gets assigned a distinct Routing Key — a simple integer number. Above, the BSC's link has RK 1, the MSC's link has RK 3.

For static routing, the M3UA port numbers must be fixed, i.e. there must be no `0` for a client's local port as in `asp foo 2905 0 m3ua`. Instead, you may use `asp foo 2905 2905 m3ua`.

The BSC needs to configure:

- its own point-code — has to match the PC configured for the BSC in osmo-stp.cfg
- the routing key — has to match the RK assigned to BSC's PC in osmo-stp.cfg
- the MSC's point-code — has to match the PC in osmo-stp.cfg and osmo-msc.cfg
- local and remote IP:ports for M3UA — have to match the IP:ports in osmo-stp.cfg

The MSC needs to configure:

- its own point-code — has to match the PC configured for the MSC in osmo-stp.cfg
- the routing key — has to match the RK assigned to MSC's PC in osmo-stp.cfg
- local and remote IP:ports for M3UA — have to match the IP:ports in osmo-stp.cfg

The STP needs to configure:

- all point-codes — they have to match the PCs in osmo-bsc.cfg and osmo-msc.cfg
- all routing keys — they have to match the RKs used in osmo-bsc.cfg and osmo-msc.cfg
- local and remote IP:ports for M3UA — have to match the IP:ports in osmo-bsc.cfg and osmo-msc.cfg

**osmo-bsc.cfg**

Copyright © 2019 sysmocom - s.f.m.c. GmbH 

```
cs7 instance 0
 point-code 1.1.1

 asp mybsc-0 2905 2905 m3ua
  remote-ip 2.2.2.2
  local-ip 1.1.1.1
  sctp-role client
 as mybsc0 m3ua
  asp mybsc0-0
  routing-key 1 1.1.1

 sccp-address mymsc
  routing-indicator PC
  point-code 3.3.3

msc 0
 msc-addr mymsc
```

**osmo-stp.cfg**

```
cs7 instance 0
 xua rkm routing-key-allocation static-only
 listen m3ua 2905
  accept-asp-connections pre-configured
  local-ip 2.2.2.2

 # asp <name> <remote-port> <local-port|0> m3ua
 asp mybsc-0 2905 2905 m3ua
  remote-ip 1.1.1.1
  local-ip 2.2.2.2
 as mybsc m3ua
  asp bsc-0
  routing-key 1 1.1.1

 asp mymsc-0 2905 2905 m3ua
  remote-ip 3.3.3.3
  local-ip 2.2.2.2
 as mymsc m3ua
  asp mymsc-0
  routing-key 3 3.3.3

 route-table system
  update route 1.1.1 7.255.7 linkset mybsc
  update route 3.3.3 7.255.7 linkset mymsc
```

**osmo-msc.cfg**

```
cs7 instance 0
 point-code 3.3.3

 asp mymsc-0 2905 2905 m3ua
  remote-ip 2.2.2.2
  local-ip 3.3.3.3
  sctp-role client
 as mymsc0 m3ua
  asp mymsc0-0
  routing-key 3 3.3.3
```

For comparison, the same setup with dynamic routing key management is a lot shorter, especially at osmo-stp.cfg, and there is no need to manually configure point-codes and routing keys between STP and {BSC,MSC}:

**osmo-bsc.cfg**

Copyright © 2019 sysmocom - s.f.m.c. GmbH 

```
cs7 instance 0
 point-code 1.1.1

 asp mybsc-0 2905 0 m3ua
  remote-ip 2.2.2.2
  local-ip 1.1.1.1
  sctp-role client
 as mybsc0 m3ua
  asp mybsc0-0

 sccp-address mymsc
  routing-indicator PC
  point-code 3.3.3

msc 0
 msc-addr mymsc
```

**osmo-stp.cfg**

```
cs7 instance 0
 xua rkm routing-key-allocation dynamic-permitted
 listen m3ua 2905
  accept-asp-connections dynamic-permitted
  local-ip 2.2.2.2
```

**osmo-msc.cfg**

```
cs7 instance 0
 point-code 3.3.3

 asp mymsc-0 2905 0 m3ua
  remote-ip 2.2.2.2
  local-ip 3.3.3.3
  sctp-role client
 as mymsc0 m3ua
  asp mymsc0-0
```

# 10 MGW Pooling

OsmoHNBGW is able to use a pool of media gateway (MGW) instances. The aim of MGW pooling is to evenly distribute the RTP voice stream load across multiple MGW instances. This can help to scale out over multiple VMs or physical machines. Until osmo-mgw includes multithreading support, it may also be used to scale-out to multiple cores on a single host.

The load distribution is managed in such a way that when a new call is placed, the pool will automatically assign the call to the available MGW with the lowest load.

MGW pooling is recommended for larger RAN or CN installations. For small networks and lab installations the classic method with one MGW per OsmoHNBGW offers sufficient performance.

## 10.1 MGW pool VTY configuration

In OsmoHNBGW the MGW is controlled via an MGCP-Client. The VTY commands to configure the MGCP-Client are part of the several *mgw* nodes, one per MGCP-Client to set up.

To setup an MGW pool, the user must first install multiple OsmoMGW instances, so that they won't interfere with each other. This can be done using different local host IP addresses or different ports. When OsmoMGW is installed from packages, the systemd configuration may also require adjustment.

Copyright © 2019 sysmocom - s.f.m.c. GmbH 

By default, MGCP-Client will use whatever source IP address is resolved by the kernel routing subsystem, and will also ask the kernel to pick a free UDP port.

Example configuration with two MGCP-Client instances in a pool:

```
mgw 0
 description media-gw-0 ❶
 remote-ip 127.0.0.1
 remote-port 2432
 local-ip 127.0.0.1
 local-port 2431
 endpoint-domain mgw0 ❷
mgw 1
 description media-gw-1 ❸
 remote-ip 127.0.0.1
 remote-port 2430
 local-ip 127.0.0.1
 local-port 2429
 endpoint-domain mgw1 ❹
```

❷, ❹ When working with multiple MGW / MGCP-Client instances, the domain name for each MGW should be different. Otherwise it won't be possible to distinguish the endpoint names in the log. It should also be noted that the domain name must match the configuration of the related OsmoMGW instance.

❶, ❸ It is also possible to assign a descriptive name to each MGW instance. The MGCP client specific log lines will then use this name as logging context. If no description is set, the domain name will be used.

## 10.2 MGW pool management

The MGW pool is fully runtime-manageable. The only limitation is that an MGCP-Client can not be restarted or removed as long as it is serving calls (see also: Section 10.2.5).

### 10.2.1 MGW pool status

The VTY implements a *show mgw-pool* command that lists the currently configured MGW pool members, their status and call utilization. The following snippet shows an output example run on OsmoBSC, but it should be also available on other applications supporting the MGW pooling VTY features:

```
OsmoBSC> show mgw-pool
% MGW-Pool:
%  MGW 0:media-gw-0
%   MGCP link:     connected,UP
%   service:       unblocked
%   ongoing calls: 1
%  MGW 1:media-gw-1
%   MGCP link:     connected,UP
%   service:       unblocked
%   ongoing calls: 0
```

### 10.2.2 Adding an MGW / MGCP-Client to the MGW pool

To add a new MGCP-Client to the pool, the *mgw* node is used. Like with the *bts* or the *msc* node a reference number is used that usually starts at 0. However it is still possible to assign any number from 0-255. The enumeration also may contain gaps. The following snippet shows an output example run on OsmoBSC, but it should be also available on other applications supporting the MGW pooling VTY features:

Copyright © 2019 sysmocom - s.f.m.c. GmbH 

```
OsmoBSC> enable
OsmoBSC# configure terminal
OsmoBSC(config)# network
OsmoBSC(config-net)# mgw 2
OsmoBSC(config-mgw)# ?
  keepalive        Monitor if the MGCP link against MGW is still usable
  local-ip         local bind to connect to MGW from
  local-port       local port to connect to MGW from
  remote-ip        remote IP address to reach the MGW at
  remote-port      remote port to reach the MGW at
  endpoint-domain  Set the domain name to send in MGCP messages, e.g. the part 'foo' in ' ↩
      rtpbridge/*@foo'.
  reset-endpoint   Add an endpoint name that should be reset (DLCX) on connect to the reset ↩
      -endpoint list,e.g. 'rtpbridge/*'
```

The newly added MGW will immediately appear in the mgw-pool list but it won't be used until its configuration finished by reconnecting it.

```
% MGW-Pool:
%  MGW 0:media-gw-0
%   MGCP link:     connected,UP
%   service:       unblocked
%   ongoing calls: 2
%  MGW 1:media-gw-1
%   MGCP link:     connected,UP
%   service:       unblocked
%   ongoing calls: 3
%  MGW 2:mgw ❶
%   MGCP link:     disconnected,DOWN
%   service:       unblocked
%   ongoing calls: 0
```

❶ In this example a description is not set yet, so the domain name ("mgw") is displayed.

#### 10.2.3 Reconnecting an MGW / MGCP-Client

It may become necessary to reconnect an MGCP-Client. This is the case when the VTY configuration was changed at runtime. In order to make the changes effective the MGW configuration must be reloaded by reconnecting the MGW connection. Also newly created MGW instances require a reconnect once their configuration is done.

To reconnect an MGCP-Client use the *reconnect* VTY command:

```
OsmoBSC# mgw 2 reconnect
```

The mgcp-client status should immediately change to *connected*. The MGW is now ready to be used for new calls.

```
OsmoBSC# show mgw-pool
% MGW-Pool:
%  MGW 0:media-gw-0
%   MGCP link:     connected,UP
%   service:       unblocked
%   ongoing calls: 2
%  MGW 1:media-gw-1
%   MGCP link:     connected,UP
%   service:       unblocked
%   ongoing calls: 3
%  MGW 2:mgw
%   MGCP link:     connected,UP
%   service:       unblocked
%   ongoing calls: 0
```

Copyright © 2019 sysmocom - s.f.m.c. GmbH 

It should be noted that MGCP a protocol is used via UDP, the connect only happens locally to forward the UDP datagrams properly (state printed in `mgcp-client: (dis)connected` above). Also (unless a reset endpoint is configured like in the example config above) there will be no immediate interaction with the MGW. However, the log should at least confirm the connect worked and the MGCP client has been created successfully.

```
Mon Aug  2 17:15:00 2021 DLMGCP mgcp_client.c:788 MGCP client: using endpoint domain '@mgw'
Mon Aug  2 17:15:00 2021 DLMGCP mgcp_client.c:908 MGCP GW connection: r=127.0.0.1:2427<->l ↩
    =127.0.0.1:2727
```

For that reason, it is strongly advised to enable the `keepalive` feature in OsmoHNBGW to schedule periodical MGCP queries against the MGW, in order to make sure it is reachable (the state `MGCP link: UP|DOWN` printed above). See section Section 10.2.4 below for more information.

#### 10.2.4 Monitor MGCP link (keepalive) / MGCP-Client

The `keepalive` feature in OsmoHNBGW allows scheduling periodical queries on the MGCP layer in order to make sure it is reachable and hence obtain information on the state of the MGCP link. This is in turn used by the MGW Pool when picking an MGW from the pool: MGWs whose link is considered to be DOWN are skipped.

The feature consists of: - A `keepalive request-interval` which will trigger a transmission of an MGCP AuditEndpoint command targeting endpoint with name `keepalive request-endpoint`. This interval is updated every time any message is transmitted in the MGCP link, meaning the MGCP AuditEndpoint message is only triggered if no message has been transmitted since `keepalive request-interval` seconds ago. - A `keepalive timeout` which upon triggering (because no message was received over that amount of time) will then consider the MGW to be non-reachable (link DOWN).

The `keepalive` parameters are to be preferrably configured so that `"keepalive request-interval" * 2 < "keepalive timeout"`.

Example VTY configuration of `keepalive` feature in OsmoHNBGW:

```
mgw 0
 ...
 keepalive request-interval 20 ❶
 keepalive request-endpoint null ❷
 keepalive timeout 50 ❸
```

❶ Transmit an MGCP AuditEndpoint message to the MGW if no message has been sent to it over last 20 seconds

❷ The MGCP AuditEndpoint targets the `null` endpoint. This is a special endpoint available at OsmoMGW for those purposes, but any available endpoint can be configured and used instead.

❸ Consider the MGCP link to be DOWN if no message is received from the MGW over the last 50 seconds

**Note**
The `keepalive` feature is disabled by default, and must be explicitly configured in order to enable it.

#### 10.2.5 Blocking an MGW / MGCP-Client

If it becomes apparent that an MGCP-Client must be restarted or removed from the config (maintenance) the operator can put that MGCP-Client into a blocked mode. A blocked MGCP-Client will still serve the ongoing calls but it will not be picked for the assignment of new calls.

To block an MGCP-Client use the *block* VTY command:

```
OsmoBSC# mgw 2 block
OsmoBSC# show mgw-pool
% MGW-Pool:
%  MGW 0:media-gw-0
```

Copyright © 2019 sysmocom - s.f.m.c. GmbH 

```
%   MGCP link:     connected,UP
%   service:       unblocked
%   ongoing calls: 11
%  MGW 1:media-gw-1
%   MGCP link:     connected,UP
%   service:       unblocked
%   ongoing calls: 12
%  MGW 2:mgw
%   MGCP link:     connected,UP
%   service:       blocked
%   ongoing calls: 10
```

When the number of ongoing calls has tapered off, the MGW / MGCP-Client can be restarted or removed if necessary.

```
OsmoBSC# show mgw-pool
% MGW-Pool:
%  MGW 0:media-gw-0
%   MGCP link:     connected,UP
%   service:       unblocked
%   ongoing calls: 15
%  MGW 1:media-gw-1
%   MGCP link:     connected,UP
%   service:       unblocked
%   ongoing calls: 14
%  MGW 2:mgw
%   MGCP link:     connected,UP
%   service:       blocked
%   ongoing calls: 0
```

If the blockade should be reverted, the *unblock* VTY command can be used in the same way to remove the blockade. (Reconnecting will not remove the blockade.)

### 10.2.6 Removing an MGW / MGCP-Client

An MGCP-Client is removed from the pool using the *no mgw* command from the configure terminal. The MGCP-Client instance will automatically be terminated and the related resources are freed. The only requirement is that there are no ongoing calls on the selected instance.

```
OsmoBSC# configure terminal
OsmoBSC(config)# network
OsmoBSC(config-net)# no mgw 2
```

# 11 Glossary

**2FF**
: 2nd Generation Form Factor; the so-called plug-in SIM form factor

**3FF**
: 3rd Generation Form Factor; the so-called microSIM form factor

**3GPP**
: 3rd Generation Partnership Project

**4FF**
: 4th Generation Form Factor; the so-called nanoSIM form factor

**A Interface**
: Interface between BTS and BSC, traditionally over E1 (*3GPP TS 48.008* [3gpp-ts-48-008])

Copyright © 2019 sysmocom - s.f.m.c. GmbH 

**A3/A8**
Algorithm 3 and 8; Authentication and key generation algorithm in GSM and GPRS, typically COMP128v1/v2/v3 or MILENAGE are typically used

**A5**
Algorithm 5; Air-interface encryption of GSM; currently only A5/0 (no encryption), A5/1 and A5/3 are in use

**Abis Interface**
Interface between BTS and BSC, traditionally over E1 (*3GPP TS 48.058* [3gpp-ts-48-058] and *3GPP TS 52.021* [3gpp-ts-52-021])

**ACC**
Access Control Class; every BTS broadcasts a bit-mask of permitted ACC, and only subscribers with a SIM of matching ACC are permitted to use that BTS

**AGCH**
Access Grant Channel on Um interface; used to assign a dedicated channel in response to RACH request

**AGPL**
GNU Affero General Public License, a copyleft-style Free Software License

**AQPSK**
Adaptive QPSK, a modulation scheme used by VAMOS channels on Downlink

**ARFCN**
Absolute Radio Frequency Channel Number; specifies a tuple of uplink and downlink frequencies

**AUC**
Authentication Center; central database of authentication key material for each subscriber

**BCCH**
Broadcast Control Channel on Um interface; used to broadcast information about Cell and its neighbors

**BCC**
Base Station Color Code; short identifier of BTS, lower part of BSIC

**BTS**
Base Transceiver Station

**BSC**
Base Station Controller

**BSIC**
Base Station Identity Code; 16bit identifier of BTS within location area

**BSSGP**
Base Station Subsystem Gateway Protocol (*3GPP TS 48.018* [3gpp-ts-48-018])

**BVCI**
BSSGP Virtual Circuit Identifier

**CBC**
Cell Broadcast Centre; central entity of Cell Broadcast service

**CBCH**
Cell Broadcast Channel; used to transmit Cell Broadcast SMS (SMS-CB)

**CBS**
Cell Broadcast Service

**CBSP**
Cell Broadcast Service Protocol (*3GPP TS 48.049* [3gpp-ts-48-049])

Copyright © 2019 sysmocom - s.f.m.c. GmbH 

**CC**
Call Control; Part of the GSM Layer 3 Protocol

**CCCH**
Common Control Channel on Um interface; consists of RACH (uplink), BCCH, PCH, AGCH (all downlink)

**Cell**
A cell in a cellular network, served by a BTS

**CEPT**
Conférence européenne des administrations des postes et des télécommunications; European Conference of Postal and Telecommunications Administrations.

**CGI**
Cell Global Identifier comprised of MCC, MNC, LAC and BSIC

**CSFB**
Circiut-Switched Fall Back; Mechanism for switching from LTE/EUTRAN to UTRAN/GERAN when circuit-switched services such as voice telephony are required.

**dB**
deci-Bel; relative logarithmic unit

**dBm**
deci-Bel (milliwatt); unit of measurement for signal strength of radio signals

**DHCP**
Dynamic Host Configuration Protocol (*IETF RFC 2131* [ietf-rfc2131])

**downlink**
Direction of messages / signals from the network core towards the mobile phone

**DSCP**
Differentiated Services Code Point (*IETF RFC 2474* [ietf-rfc2474])

**DSP**
Digital Signal Processor

**dvnixload**
Tool to program UBL and the Bootloader on a sysmoBTS

**EDGE**
Enhanced Data rates for GPRS Evolution; Higher-speed improvement of GPRS; introduces 8PSK

**EGPRS**
Enhanced GPRS; the part of EDGE relating to GPRS services

**EIR**
Equipment Identity Register; core network element that stores and manages IMEI numbers

**ESME**
External SMS Entity; an external application interfacing with a SMSC over SMPP

**ETSI**
European Telecommunications Standardization Institute

**FPGA**
Field Programmable Gate Array; programmable digital logic hardware

**Gb**
Interface between PCU and SGSN in GPRS/EDGE network; uses NS, BSSGP, LLC

**GERAN**
GPRS/EDGE Radio Access Network

Copyright © 2019 sysmocom - s.f.m.c. GmbH 

**GFDL**
GNU Free Documentation License; a copyleft-style Documentation License

**GGSN**
GPRS Gateway Support Node; gateway between GPRS and external (IP) network

**GMSK**
Gaussian Minimum Shift Keying; modulation used for GSM and GPRS

**GPL**
GNU General Public License, a copyleft-style Free Software License

**Gp**
Gp interface between SGSN and GGSN; uses GTP protocol

**GPRS**
General Packet Radio Service; the packet switched 2G technology

**GPS**
Global Positioning System; provides a highly accurate clock reference besides the global position

**GSM**
Global System for Mobile Communications. ETSI/3GPP Standard of a 2G digital cellular network

**GSMTAP**
GSM tap; pseudo standard for encapsulating GSM protocol layers over UDP/IP for analysis

**GSUP**
Generic Subscriber Update Protocol. Osmocom-specific alternative to TCAP/MAP

**GT**
Global Title; an address in SCCP

**GTP**
GPRS Tunnel Protocol; used between SGSN and GGSN

**HLR**
Home Location Register; central subscriber database of a GSM network

**HNB-GW**
Home NodeB Gateway. Entity between femtocells (Home NodeB) and CN in 3G/UMTS.

**HPLMN**
Home PLMN; the network that has issued the subscriber SIM and has his record in HLR

**IE**
Information Element

**IMEI**
International Mobile Equipment Identity; unique 14-digit decimal number to globally identify a mobile device, optionally with a 15th checksum digit

**IMEISV**
IMEI software version; unique 14-digit decimal number to globally identify a mobile device (same as IMEI) plus two software version digits (total digits: 16)

**IMSI**
International Mobile Subscriber Identity; 15-digit unique identifier for the subscriber/SIM; starts with MCC/MNC of issuing operator

**IP**
Internet Protocol (*IETF RFC 791* [ietf-rfc791])

Copyright © 2019 sysmocom - s.f.m.c. GmbH 

**IPA**
*ip.access GSM over IP* protocol; used to multiplex a single TCP connection

**Iu**
Interface in 3G/UMTS between RAN and CN

**IuCS**
Iu interface for circuit-switched domain. Used in 3G/UMTS between RAN and MSC

**IuPS**
Iu interface for packet-switched domain. Used in 3G/UMTS between RAN and SGSN

**LAC**
Location Area Code; 16bit identifier of Location Area within network

**LAPD**
Link Access Protocol, D-Channel (*ITU-T Q.921* [itu-t-q921])

**LAPDm**
Link Access Protocol Mobile (*3GPP TS 44.006* [3gpp-ts-44-006])

**LLC**
Logical Link Control; GPRS protocol between MS and SGSN (*3GPP TS 44.064* [3gpp-ts-44-064])

**Location Area**
Location Area; a geographic area containing multiple BTS

**LU**
Location Updating; can be of type IMSI-Attach or Periodic. Procedure that indicates a subscriber's physical presence in a given radio cell.

**M2PA**
MTP2 Peer-to-Peer Adaptation; a SIGTRAN Variant (*RFC 4165* [ietf-rfc4165])

**M2UA**
MTP2 User Adaptation; a SIGTRAN Variant (*RFC 3331* [ietf-rfc3331])

**M3UA**
MTP3 User Adaptation; a SIGTRAN Variant (*RFC 4666* [ietf-rfc4666])

**MCC**
Mobile Country Code; unique identifier of a country, e.g. 262 for Germany

**MFF**
Machine-to-Machine Form Factor; a SIM chip package that is soldered permanently onto M2M device circuit boards.

**MGW**
Media Gateway

**MM**
Mobility Management; part of the GSM Layer 3 Protocol

**MNC**
Mobile Network Code; identifies network within a country; assigned by national regulator

**MNCC**
Mobile Network Call Control; Unix domain socket based Interface between MSC and external call control entity like osmo-sip-connector

**MNO**
Mobile Network Operator; operator with physical radio network under his MCC/MNC

**MO**
Mobile Originated. Direction from Mobile (MS/UE) to Network

Copyright © 2019 sysmocom - s.f.m.c. GmbH 

**MS**
Mobile Station; a mobile phone / GSM Modem

**MSC**
Mobile Switching Center; network element in the circuit-switched core network

**MSC pool**
A number of redundant MSCs serving the same core network, which a BSC / RNC distributes load across; see also the "MSC Pooling" chapter in OsmoBSC's user manual [userman-osmobsc] and *3GPP TS 23.236* [3gpp-ts-23-236]

**MSISDN**
Mobile Subscriber ISDN Number; telephone number of the subscriber

**MT**
Mobile Terminated. Direction from Network to Mobile (MS/UE)

**MTP**
Message Transfer Part; SS7 signaling protocol (*ITU-T Q.701* [itu-t-q701])

**MVNO**
Mobile Virtual Network Operator; Operator without physical radio network

**NCC**
Network Color Code; assigned by national regulator

**NITB**
Network In The Box; combines functionality traditionally provided by BSC, MSC, VLR, HLR, SMSC functions; see OsmoNITB

**NRI**
Network Resource Indicator, typically 10 bits of a TMSI indicating which MSC of an MSC pool attached the subscriber; see also the "MSC Pooling" chapter in OsmoBSC's user manual [userman-osmobsc] and *3GPP TS 23.236* [3gpp-ts-23-236]

**NSEI**
NS Entity Identifier

**NVCI**
NS Virtual Circuit Identifier

**NWL**
Network Listen; ability of some BTS to receive downlink from other BTSs

**NS**
Network Service; protocol on Gb interface (*3GPP TS 48.016* [3gpp-ts-48-016])

**OCXO**
Oven Controlled Crystal Oscillator; very high precision oscillator, superior to a VCTCXO

**OML**
Operation & Maintenance Link (ETSI/*3GPP TS 52.021* [3gpp-ts-52-021])

**OpenBSC**
Open Source implementation of GSM network elements, specifically OsmoBSC, OsmoNITB, OsmoSGSN

**OpenGGSN**
Open Source implementation of a GPRS Packet Control Unit

**OpenVPN**
Open-Source Virtual Private Network; software employed to establish encrypted private networks over untrusted public networks

Copyright © 2019 sysmocom - s.f.m.c. GmbH 

**Osmocom**
Open Source MObile COMmunications; collaborative community for implementing communications protocols and systems, including GSM, GPRS, TETRA, DECT, GMR and others

**OsmoBSC**
Open Source implementation of a GSM Base Station Controller

**OsmoNITB**
Open Source implementation of a GSM Network In The Box, combines functionality traditionally provided by BSC, MSC, VLR, HLR, AUC, SMSC

**OsmoSGSN**
Open Source implementation of a Serving GPRS Support Node

**OsmoPCU**
Open Source implementation of a GPRS Packet Control Unit

**OTA**
Over-The-Air; Capability of operators to remotely reconfigure/reprogram ISM/USIM cards

**PC**
Point Code; an address in MTP

**PCH**
Paging Channel on downlink Um interface; used by network to page an MS

**PCP**
Priority Code Point (*IEEE 802.1Q* [?])

**PCU**
Packet Control Unit; used to manage Layer 2 of the GPRS radio interface

**PDCH**
Packet Data Channel on Um interface; used for GPRS/EDGE signalling + user data

**PIN**
Personal Identification Number; a number by which the user authenticates to a SIM/USIM or other smart card

**PLMN**
Public Land Mobile Network; specification language for a single GSM network

**PUK**
PIN Unblocking Code; used to unblock a blocked PIN (after too many wrong PIN attempts)

**RAC**
Routing Area Code; 16bit identifier for a Routing Area within a Location Area

**RACH**
Random Access Channel on uplink Um interface; used by MS to request establishment of a dedicated channel

**RAM**
Remote Application Management; Ability to remotely manage (install, remove) Java Applications on SIM/USIM Card

**RF**
Radio Frequency

**RFM**
Remote File Management; Ability to remotely manage (write, read) files on a SIM/USIM card

**Roaming**
Procedure in which a subscriber of one network is using the radio network of another network, often in different countries; in some countries national roaming exists

Copyright © 2019 sysmocom - s.f.m.c. GmbH 

**Routing Area**
Routing Area; GPRS specific sub-division of Location Area

**RR**
Radio Resources; Part of the GSM Layer 3 Protocol

**RSL**
Radio Signalling Link (*3GPP TS 48.058* [3gpp-ts-48-058])

**RTP**
Real-Time Transport Protocol (*IETF RFC 3550* [ietf-rfc3550]); Used to transport audio/video streams over UDP/IP

**SACCH**
Slow Associate Control Channel on Um interface; bundled to a TCH or SDCCH, used for signalling in parallel to active dedicated channel

**SCCP**
Signaling Connection Control Part; SS7 signaling protocol (*ITU-T Q.711* [itu-t-q711])

**SDCCH**
Slow Dedicated Control Channel on Um interface; used for signalling and SMS transport in GSM

**SDK**
Software Development Kit

**SGs**
Interface between MSC (GSM/UMTS) and MME (LTE/EPC) to facilitate CSFB and SMS.

**SGSN**
Serving GPRS Support Node; Core network element for packet-switched services in GSM and UMTS.

**SIGTRAN**
Signaling Transport over IP (*IETF RFC 2719* [ietf-rfc2719])

**SIM**
Subscriber Identity Module; small chip card storing subscriber identity

**Site**
A site is a location where one or more BTSs are installed, typically three BTSs for three sectors

**SMPP**
Short Message Peer-to-Peer; TCP based protocol to interface external entities with an SMSC

**SMSC**
Short Message Service Center; store-and-forward relay for short messages

**SS7**
Signaling System No. 7; Classic digital telephony signaling system

**SS**
Supplementary Services; query and set various service parameters between subscriber and core network (e.g. USSD, 3rd-party calls, hold/retrieve, advice-of-charge, call deflection)

**SSH**
Secure Shell; *IETF RFC 4250* [ietf-rfc4251] to 4254

**SSN**
Sub-System Number; identifies a given SCCP Service such as MSC, HLR

**STP**
Signaling Transfer Point; A Router in SS7 Networks

**SUA**
SCCP User Adaptation; a SIGTRAN Variant (*RFC 3868* [ietf-rfc3868])

Copyright © 2019 sysmocom - s.f.m.c. GmbH 

**syslog**
System logging service of UNIX-like operating systems

**System Information**
A set of downlink messages on the BCCH and SACCH of the Um interface describing properties of the cell and network

**TCH**
Traffic Channel; used for circuit-switched user traffic (mostly voice) in GSM

**TCP**
Transmission Control Protocol; (*IETF RFC 793* [ietf-rfc793])

**TFTP**
Trivial File Transfer Protocol; (*IETF RFC 1350* [ietf-rfc1350])

**TOS**
Type Of Service; bit-field in IPv4 header, now re-used as DSCP (*IETF RFC 791* [ietf-rfc791])

**TRX**
Transceiver; element of a BTS serving a single carrier

**TS**
Technical Specification

**u-Boot**
Boot loader used in various embedded systems

**UBI**
An MTD wear leveling system to deal with NAND flash in Linux

**UBL**
Initial bootloader loaded by the TI Davinci SoC

**UDP**
User Datagram Protocol (*IETF RFC 768* [ietf-rfc768])

**UICC**
Universal Integrated Chip Card; A smart card according to *ETSI TR 102 216* [etsi-tr102216]

**Um interface**
U mobile; Radio interface between MS and BTS

**uplink**
Direction of messages: Signals from the mobile phone towards the network

**USIM**
Universal Subscriber Identity Module; application running on a UICC to provide subscriber identity for UMTS and GSM networks

**USSD**
Unstructured Supplementary Service Data; textual dialog between subscriber and core network, e.g. **100 → Your extension is 1234*

**VAMOS**
Voice services over Adaptive Multi-user channels on One Slot; an optional extension for GSM specified in Release 9 of 3GPP GERAN specifications (*3GPP TS 48.018* [3gpp-ts-48-018]) allowing two independent UEs to transmit and receive simultaneously on traffic channels

**VCTCXO**
Voltage Controlled, Temperature Compensated Crystal Oscillator; a precision oscillator, superior to a classic crystal oscillator, but inferior to an OCXO

**VLAN**
Virtual LAN in the context of Ethernet (*IEEE 802.1Q* [ieee-802.1q])

Copyright © 2019 sysmocom - s.f.m.c. GmbH 

**VLR**
Visitor Location Register; volatile storage of attached subscribers in the MSC

**VPLMN**
Visited PLMN; the network in which the subscriber is currently registered; may differ from HPLMN when on roaming

**VTY**
Virtual TeletYpe; a textual command-line interface for configuration and introspection, e.g. the OsmoBSC configuration file as well as its telnet link on port 4242

# A Osmocom TCP/UDP Port Numbers

The Osmocom GSM system utilizes a variety of TCP/IP based protocols. The table below provides a reference as to which port numbers are used by which protocol / interface.

Table 4: TCP/UDP port numbers

| L4 Protocol | Port Number | Purpose | Software |
|---|---|---|---|
| UDP | 1984 | Osmux | osmo-mgw, osmo-bts |
| UDP | 2427 | MGCP GW | osmo-bsc_mgcp, osmo-mgw |
| TCP | 2775 | SMPP (SMS interface for external programs) | osmo-nitb |
| TCP | 3002 | A-bis/IP OML | osmo-bts, osmo-bsc, osmo-nitb |
| TCP | 3003 | A-bis/IP RSL | osmo-bts, osmo-bsc, osmo-nitb |
| TCP | 4227 | telnet (VTY) | osmo-pcap-client |
| TCP | 4228 | telnet (VTY) | osmo-pcap-server |
| TCP | 4236 | Control Interface | osmo-trx |
| TCP | 4237 | telnet (VTY) | osmo-trx |
| TCP | 4238 | Control Interface | osmo-bts |
| TCP | 4239 | telnet (VTY) | osmo-stp |
| TCP | 4240 | telnet (VTY) | osmo-pcu |
| TCP | 4241 | telnet (VTY) | osmo-bts |
| TCP | 4242 | telnet (VTY) | osmo-nitb, osmo-bsc, cellmgr-ng |
| TCP | 4243 | telnet (VTY) | osmo-bsc_mgcp, osmo-mgw |
| TCP | 4244 | telnet (VTY) | osmo-bsc_nat |
| TCP | 4245 | telnet (VTY) | osmo-sgsn |
| TCP | 4246 | telnet (VTY) | osmo-gbproxy |
| TCP | 4247 | telnet (VTY) | OsmocomBB |
| TCP | 4249 | Control Interface | osmo-nitb, osmo-bsc |
| TCP | 4250 | Control Interface | osmo-bsc_nat |
| TCP | 4251 | Control Interface | osmo-sgsn |
| TCP | 4252 | telnet (VTY) | sysmobts-mgr |
| TCP | 4253 | telnet (VTY) | osmo-gtphub |
| TCP | 4254 | telnet (VTY) | osmo-msc |
| TCP | 4255 | Control Interface | osmo-msc |
| TCP | 4256 | telnet (VTY) | osmo-sip-connector |
| TCP | 4257 | Control Interface | osmo-ggsn, ggsn (OpenGGSN) |
| TCP | 4258 | telnet (VTY) | osmo-hlr |
| TCP | 4259 | Control Interface | osmo-hlr |
| TCP | 4260 | telnet (VTY) | osmo-ggsn |
| TCP | 4261 | telnet (VTY) | osmo-hnbgw |
| TCP | 4262 | Control Interface | osmo-hnbgw |
| TCP | 4263 | Control Interface | osmo-gbproxy |
| TCP | 4264 | telnet (VTY) | osmo-cbc |
| TCP | 4265 | Control Interface | osmo-cbc |

Copyright © 2019 sysmocom - s.f.m.c. GmbH 

Table 4: (continued)

| L4 Protocol | Port Number | Purpose | Software |
|---|---|---|---|
| TCP | 4266 | D-GSM MS Lookup: mDNS serve | osmo-hlr |
| TCP | 4267 | Control Interface | osmo-mgw |
| TCP | 4268 | telnet (VTY) | osmo-uecups |
| SCTP | 4268 | UECUPS | osmo-uecups |
| TCP | 4269 | telnet (VTY) | osmo-e1d |
| TCP | 4270 | telnet (VTY) | osmo-isdntap |
| TCP | 4271 | telnet (VTY) | osmo-smlc |
| TCP | 4272 | Control Interface | osmo-smlc |
| TCP | 4273 | telnet (VTY) | osmo-hnodeb |
| TCP | 4274 | Control Interface | osmo-hnodeb |
| TCP | 4275 | telnet (VTY) | osmo-upf |
| TCP | 4276 | Control Interface | osmo-upf |
| TCP | 4277 | telnet (VTY) | osmo-pfcp-tool |
| TCP | 4278 | Control Interface | osmo-pfcp-tool |
| UDP | 4729 | GSMTAP | Almost every osmocom project |
| TCP | 5000 | A/IP | osmo-bsc, osmo-bsc_nat |
| UDP | 23000 | GPRS-NS over IP default port | osmo-pcu, osmo-sgsn, osmo-gbproxy |
| TCP | 48049 | BSC-CBC (CBSP) default port | osmo-bsc, osmo-cbc |

# B Bibliography / References

#### B.0.0.0.1 References


[1] [userman-ice1usb] Osmocom Project: icE1usb User Manual.

[2] [userman-ogt] Pau Espin: osmo-gsm-tester User Manual.

[3] [userman-remsim] Harald Welte: osmo-remsim User Manual.

[4] [osmobts-abis-spec] Neels Hofmeyr & Harald Welte. OsmoBTS Abis Protocol Specification. https://ftp.osmocom.org/docs/latest/osmobts-abis.pdf

[5] [userman-osmobsc] Osmocom Project: OsmoBSC User Manual. https://ftp.osmocom.org/docs/latest/osmobsc-usermanual.pdf

[6] [vty-ref-osmobsc] Osmocom Project: OsmoBSC VTY Reference Manual. https://ftp.osmocom.org/docs/-latest/osmobsc-vty-reference.pdf

[7] [userman-osmobts] Osmocom Project: OsmoBTS User Manual. https://ftp.osmocom.org/docs/latest/osmobts-usermanual.pdf

[8] [vty-ref-osmobts] Osmocom Project: OsmoBTS VTY Reference Manual. https://ftp.osmocom.org/docs/-latest/osmobts-trx-vty-reference.pdf https://ftp.osmocom.org/docs/latest/osmobts-sysmo-vty-reference.pdf https://ftp.osmocom.org/docs/latest/osmobts-lc15-vty-reference.pdf https://ftp.osmocom.org/docs/latest/-osmobts-oc2g-vty-reference.pdf https://ftp.osmocom.org/docs/latest/osmobts-octphy-vty-reference.pdf https://ftp.osmocom.org/docs/latest/osmobts-virtual-vty-reference.pdf

[9] [userman-osmocbc] Osmocom Project: OsmoCBC User Manual. https://ftp.osmocom.org/docs/latest/-osmocbc-usermanual.pdf

[10] [vty-ref-osmocbc] Osmocom Project: OsmoCBC VTY Reference Manual. https://ftp.osmocom.org/docs/-latest/osmocbc-vty-reference.pdf


Copyright © 2019 sysmocom - s.f.m.c. GmbH 

[11] [userman-osmogbproxy] Osmocom Project: OsmoGBProxy User Manual. https://ftp.osmocom.org/docs/-latest/osmogbproxy-usermanual.pdf

[12] [vty-ref-osmogbproxy] Osmocom Project: OsmoGBPRoxy VTY Reference Manual. https://ftp.osmocom.org/-docs/latest/osmogbproxy-vty-reference.pdf

[13] [userman-osmoggsn] Osmocom Project: OpenGGSN User Manual. https://ftp.osmocom.org/docs/latest/-osmoggsn-usermanual.pdf

[14] [vty-ref-osmoggsn] Osmocom Project: OsmoGGSN VTY Reference Manual. https://ftp.osmocom.org/docs/-latest/osmoggsn-vty-reference.pdf

[15] [userman-osmohlr] Osmocom Project: OsmoHLR User Manual. https://ftp.osmocom.org/docs/latest/osmohlr-usermanual.pdf

[16] [vty-ref-osmohlr] Osmocom Project: OsmoHLR VTY Reference Manual. https://ftp.osmocom.org/docs/latest/-osmohlr-vty-reference.pdf

[17] [userman-osmohnbgw] Osmocom Project: OsmoHNBGW User Manual. https://ftp.osmocom.org/docs/latest/-osmohnbgw-usermanual.pdf

[18] [vty-ref-osmohnbgw] Osmocom Project: OsmoHNBGW VTY Reference Manual. https://ftp.osmocom.org/-docs/latest/osmohnbgw-vty-reference.pdf

[19] [userman-osmomgw] Osmocom Project: OsmoMGW User Manual. https://ftp.osmocom.org/docs/latest/-osmomgw-usermanual.pdf

[20] [vty-ref-osmomgw] Osmocom Project: OsmoMGW VTY Reference Manual. https://ftp.osmocom.org/docs/-latest/osmomgw-vty-reference.pdf

[21] [userman-osmomsc] Osmocom Project: OsmoMSC User Manual. https://ftp.osmocom.org/docs/latest/-osmomsc-usermanual.pdf

[22] [vty-ref-osmomsc] Osmocom Project: OsmoMSC VTY Reference Manual. https://ftp.osmocom.org/docs/-latest/osmomsc-vty-reference.pdf

[23] [userman-osmonitb] Osmocom Project: OsmoNITB User Manual. https://ftp.osmocom.org/docs/latest/-osmonitb-usermanual.pdf

[24] [vty-ref-osmonitb] Osmocom Project: OsmoNITB VTY Reference Manual. https://ftp.osmocom.org/docs/-latest/osmonitb-vty-reference.pdf

[25] [userman-osmopcu] Osmocom Project: OsmoPCU User Manual. https://ftp.osmocom.org/docs/latest/-osmopcu-usermanual.pdf

[26] [vty-ref-osmopcu] Osmocom Project: OsmoPCU VTY Reference Manual. https://ftp.osmocom.org/docs/-latest/osmopcu-vty-reference.pdf

[27] [userman-osmosgsn] Osmocom Project: OsmoSGSN User Manual. https://ftp.osmocom.org/docs/latest/-osmosgsn-usermanual.pdf

[28] [vty-ref-osmosgsn] Osmocom Project: OsmoSGSN VTY Reference Manual. https://ftp.osmocom.org/docs/-latest/osmosgsn-vty-reference.pdf

[29] [userman-osmosipconnector] Osmocom Project: OsmoSIPconnector User Manual. https://ftp.osmocom.org/-docs/latest/osmosipconnector-usermanual.pdf

[30] [vty-ref-osmosipconnector] Osmocom Project: OsmoSIPconnector VTY Reference Manual. https://ftp.osmocom.org/docs/latest/osmosipconnector-vty-reference.pdf

[31] [userman-osmosmlc] Osmocom Project: OsmoSMLC User Manual. https://ftp.osmocom.org/docs/latest/-osmosmlc-usermanual.pdf

[32] [vty-ref-osmosmlc] Osmocom Project: OsmoSMLC VTY Reference Manual. https://ftp.osmocom.org/docs/-latest/osmosmlc-vty-reference.pdf

Copyright © 2019 sysmocom - s.f.m.c. GmbH 

[33] [userman-osmostp] Osmocom Project: OsmoSTP User Manual. https://ftp.osmocom.org/docs/latest/osmostp-usermanual.pdf

[34] [vty-ref-osmostp] Osmocom Project: OsmoSTP VTY Reference Manual. https://ftp.osmocom.org/docs/latest/-osmostp-vty-reference.pdf

[35] [userman-osmotrx] Osmocom Project: OsmoTRX User Manual. https://ftp.osmocom.org/docs/latest/osmotrx-usermanual.pdf

[36] [vty-ref-osmotrx] Osmocom Project: OsmoTRX VTY Reference Manual. https://ftp.osmocom.org/docs/-latest/osmotrx-uhd-vty-reference.pdf https://ftp.osmocom.org/docs/latest/osmotrx-lms-vty-reference.pdf https://ftp.osmocom.org/docs/latest/osmotrx-ipc-vty-reference.pdf https://ftp.osmocom.org/docs/latest/-osmotrx-usrp1-vty-reference.pdf

[37] [3gpp-ts-23-041] 3GPP TS 23.041: Technical realization of Cell Broadcast Service (CBS)

[38] [3gpp-ts-23-048] 3GPP TS 23.048: Security mechanisms for the (U)SIM application toolkit; Stage 2 https://www.3gpp.org/DynaReport/23048.htm

[39] [3gpp-ts-23-236] 3GPP TS 23.236: Intra-domain connection of Radio Access Network (RAN) nodes to multiple Core Network (CN) nodes https://www.3gpp.org/DynaReport/23236.htm

[40] [3gpp-ts-24-007] 3GPP TS 24.007: Mobile radio interface signalling layer 3; General Aspects https://www.3gpp.org/DynaReport/24007.htm

[41] [3gpp-ts-24-008] 3GPP TS 24.008: Mobile radio interface Layer 3 specification; Core network protocols; Stage 3. https://www.3gpp.org/dynareport/24008.htm

[42] [3gpp-ts-31-101] 3GPP TS 31.101: UICC-terminal interface; Physical and logical characteristics https://www.3gpp.org/DynaReport/31101.htm

[43] [3gpp-ts-31-102] 3GPP TS 31.102: Characteristics of the Universal Subscriber Identity Module (USIM) application https://www.3gpp.org/DynaReport/31102.htm

[44] [3gpp-ts-31-103] 3GPP TS 31.103: Characteristics of the IMS Subscriber Identity Module (ISIM) application https://www.3gpp.org/DynaReport/31103.htm

[45] [3gpp-ts-31-111] 3GPP TS 31.111: Universal Subscriber Identity Module (USIM) Application Toolkit (USAT) https://www.3gpp.org/DynaReport/31111.htm

[46] [3gpp-ts-31-115] 3GPP TS 31.115: Secured packet structure for (Universal) Subscriber Identity Module (U)SIM Toolkit applications https://www.3gpp.org/DynaReport/31115.htm

[47] [3gpp-ts-31-116] 3GPP TS 31.116: Remote APDU Structure for (U)SIM Toolkit applications https://www.3gpp.org/DynaReport/31116.htm

[48] [3gpp-ts-35-205] 3GPP TS 35.205: 3G Security; Specification of the MILENAGE algorithm set: General

[49] [3gpp-ts-35-206] 3GPP TS 35.206: 3G Security; Specification of the MILENAGE algorithm set: Algorithm specification https://www.3gpp.org/DynaReport/35206.htm

[50] [3gpp-ts-44-006] 3GPP TS 44.006: Mobile Station - Base Station System (MS - BSS) interface; Data Link (DL) layer specification https://www.3gpp.org/DynaReport/44006.htm

[51] [3gpp-ts-44-018] 3GPP TS 44.018: Mobile radio interface layer 3 specification; Radio Resource Control (RRC) protocol https://www.3gpp.org/DynaReport/44018.htm

[52] [3gpp-ts-44-064] 3GPP TS 44.064: Mobile Station - Serving GPRS Support Node (MS-SGSN); Logical Link Control (LLC) Layer Specification https://www.3gpp.org/DynaReport/44064.htm

[53] [3gpp-ts-45-002] 3GPP TS 45.002: Digital cellular telecommunications system (Phase 2+) (GSM); GSM/EDGE Multiplexing and multiple access on the radio path https://www.3gpp.org/DynaReport/45002.htm

[54] [3gpp-ts-48-008] 3GPP TS 48.008: Mobile Switching Centre - Base Station system (MSC-BSS) interface; Layer 3 specification https://www.3gpp.org/DynaReport/48008.htm

Copyright © 2019 sysmocom - s.f.m.c. GmbH 

[55] [3gpp-ts-48-016] 3GPP TS 48.016: General Packet Radio Service (GPRS); Base Station System (BSS) - Serving GPRS Support Node (SGSN) interface; Network service https://www.3gpp.org/DynaReport/48016.htm

[56] [3gpp-ts-48-018] 3GPP TS 48.018: General Packet Radio Service (GPRS); Base Station System (BSS) - Serving GPRS Support Node (SGSN); BSS GPRS protocol (BSSGP) https://www.3gpp.org/DynaReport/48018.htm

[57] [3gpp-ts-48-049] 3GPP TS 48.049: Digital cellular communications system; Base Station Controller - Cell Broadcast Centre (BSC-CBC) interface specification; Cell Broadcast Service Protocol (CBSP) https://www.3gpp.org/DynaReport/48049.htm

[58] [3gpp-ts-48-056] 3GPP TS 48.056: Base Station Controller - Base Transceiver Station (BSC - BTS) interface; Layer 2 specification https://www.3gpp.org/DynaReport/48056.htm

[59] [3gpp-ts-48-058] 3GPP TS 48.058: Base Station Controller - Base Transceiver Station (BSC - BTS) Interface; Layer 3 specification https://www.3gpp.org/DynaReport/48058.htm

[60] [3gpp-ts-51-011] 3GPP TS 51.011: Specification of the Subscriber Identity Module - Mobile Equipment (SIM-ME) interface

[61] [3gpp-ts-51-014] 3GPP TS 51.014: Specification of the SIM Application Toolkit for the Subscriber Identity Module - Mobile Equipment (SIM - ME) interface https://www.3gpp.org/DynaReport/51014.htm

[62] [3gpp-ts-52-021] 3GPP TS 52.021: Network Management (NM) procedures and messages on the A-bis interface https://www.3gpp.org/DynaReport/52021.htm

[63] [etsi-tr102216] ETSI TR 102 216: Smart cards https://www.etsi.org/deliver/etsi_tr/102200_102299/102216/03.00.00_60/tr_102216v030000p.pdf

[64] [etsi-ts102221] ETSI TS 102 221: Smart Cards; UICC-Terminal interface; Physical and logical characteristics https://www.etsi.org/deliver/etsi_ts/102200_102299/102221/13.01.00_60/ts_102221v130100p.pdf

[65] [etsi-ts101220] ETSI TS 101 220: Smart Cards; ETSI numbering system for telecommunication application providers https://www.etsi.org/deliver/etsi_ts/101200_101299/101220/12.00.00_60/ts_101220v120000p.pdf

[66] [etsi-ts102671] ETSI TS 102 671: Smart Cards; Machine to Machine UICC; Physical and logical characteristics https://www.etsi.org/deliver/etsi_ts/102600_102699/102671/18.01.00_60/ts_102671v180100p.pdf

[67] [ieee-802.1q] IEEE 802.1Q: Bridges and Bridged Networks https://ieeexplore.ieee.org/document/6991462

[68] [ietf-rfc768] IETF RFC 768: User Datagram Protocol https://tools.ietf.org/html/rfc768

[69] [ietf-rfc791] IETF RFC 791: Internet Protocol https://tools.ietf.org/html/rfc791

[70] [ietf-rfc793] IETF RFC 793: Transmission Control Protocol https://tools.ietf.org/html/rfc793

[71] [ietf-rfc1035] IETF RFC 1035: Domain Names - Implementation and Specification https://tools.ietf.org/html/rfc1035

[72] [ietf-rfc1350] IETF RFC 1350: Trivial File Transfer Protocol https://tools.ietf.org/html/rfc1350

[73] [ietf-rfc2131] IETF RFC 2131: Dynamic Host Configuration Protocol https://tools.ietf.org/html/rfc2131

[74] [ietf-rfc2474] IETF RFC 2474: Definition of the Differentiated Services Field (DS Field) in the IPv44 and IPv6 Headers https://tools.ietf.org/html/rfc2474

[75] [ietf-rfc2719] IETF RFC 2719: Signal Transport over IP https://tools.ietf.org/html/rfc2719

[76] [ietf-rfc3331] IETF RFC 3331: Message Transfer Part 2 User Adaptation Layer https://tools.ietf.org/html/rfc3331

[77] [ietf-rfc3550] IETF RFC 3550: RTP: A Transport protocol for Real-Time Applications https://tools.ietf.org/html/rfc3550

[78] [ietf-rfc3596] IETF RFC 3596: DNS Extensions to Support IP Version 6 https://tools.ietf.org/html/rfc3596

Copyright © 2019 sysmocom - s.f.m.c. GmbH 

[79] [ietf-rfc3868] IETF RFC 3868: SCCP User Adaptation Layer https://tools.ietf.org/html/rfc3868

[80] [ietf-rfc4165] IETF RFC 4165: Message Transfer Part 2 Peer-to-Peer Adaptation Layer https://tools.ietf.org/-html/rfc4165

[81] [ietf-rfc4251] IETF RFC 4251: The Secure Shell (SSH) Protocol Architecture https://tools.ietf.org/html/-rfc4251

[82] [ietf-rfc4666] IETF RFC 4666: Message Transfer Part 3 User Adaptation Layer https://tools.ietf.org/html/-rfc4666

[83] [ietf-rfc5771] IETF RFC 5771: IANA Guidelines for IPv4 Multicast Address Assignments https://tools.ietf.org/html/rfc5771

[84] [itu-t-q701] ITU-T Q.701: Functional Description of the Message Transfer Part (MTP) https://www.itu.int/rec/-T-REC-Q.701/en/

[85] [itu-t-q711] ITU-T Q.711: Functional Description of the Signalling Connection Control Part https://www.itu.int/rec/T-REC-Q.711/en/

[86] [itu-t-q713] ITU-T Q.713: Signalling connection control part formats and codes https://www.itu.int/rec/T-REC-Q.713/en/

[87] [itu-t-q714] ITU-T Q.714: Signalling connection control part procedures https://www.itu.int/rec/T-REC-Q.714/en/

[88] [itu-t-q921] ITU-T Q.921: ISDN user-network interface - Data link layer specification https://www.itu.int/rec/-T-REC-Q.921/en

[89] [smpp-34] SMPP Develoepers Forum. Short Message Peer-to-Peer Protocol Specification v3.4 https://docs.nimta.com/SMPP_v3_4_Issue1_2.pdf

[90] [gnu-agplv3] Free Software Foundation. GNU Affero General Public License. https://www.gnu.org/licenses/-agpl-3.0.en.html

[91] [freeswitch_pbx] FreeSWITCH SIP PBX https://freeswitch.org

[92] [tw-ts-001] TW-TS-001: Enhanced RTP transport of FR and EFR codec frames in an IP-based GSM RAN https://www.freecalypso.org/specs/tw-ts-001-v010100.txt

[93] [tw-ts-002] TW-TS-002: Enhanced RTP transport of HRv1 codec frames in an IP-based GSM RAN https://www.freecalypso.org/specs/tw-ts-002-v010100.txt

[94] [tw-ts-003] TW-TS-003: BSSMAP extension for selection of enhanced RTP transport formats https://www.freecalypso.org/specs/tw-ts-003-v010002.txt

# C GNU Free Documentation License

Version 1.3, 3 November 2008

Copyright © 2000, 2001, 2002, 2007, 2008 Free Software Foundation, Inc. http://fsf.org/

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

## C.1 PREAMBLE

The purpose of this License is to make a manual, textbook, or other functional and useful document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondarily, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

## C.2 APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you". You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Section whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not Transparent is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, Title Page means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

The "publisher" means any person or entity that distributes copies of the Document to the public.

Copyright © 2019 sysmocom - s.f.m.c. GmbH 

A section "Entitled XYZ" means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as "Acknowledgements", "Dedications", "Endorsements", or "History".) To "Preserve the Title" of such a section when you modify the Document means that it remains a section "Entitled XYZ" according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

## C.3 VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section Section C.4.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

## C.4 COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

## C.5 MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

a. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.

b. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.

Copyright © 2019 sysmocom - s.f.m.c. GmbH 

c. State on the Title Page the name of the publisher of the Modified Version, as the publisher.

d. Preserve all the copyright notices of the Document.

e. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.

f. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.

g. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.

h. Include an unaltered copy of this License.

i. Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.

j. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.

k. For any section Entitled "Acknowledgements" or "Dedications", Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.

l. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.

m. Delete any section Entitled "Endorsements". Such a section may not be included in the Modified Version.

n. Do not retitle any existing section to be Entitled "Endorsements" or to conflict in title with any Invariant Sections.

o. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Section and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties—for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

## C.6 COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such

Copyright © 2019 sysmocom - s.f.m.c. GmbH 

section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled "History" in the various original documents, forming one section Entitled "History"; likewise combine any sections Entitled "Acknowledgements", and any sections Entitled "Dedications". You must delete all sections Entitled "Endorsements".

## C.7 COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

## C.8 AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an "aggregate" if the copyright resulting from the compilation is not used to limit the legal rights of the compilation's users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

## C.9 TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled "Acknowledgements", "Dedications", or "History", the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

## C.10 TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, or distribute it is void, and will automatically terminate your rights under this License.

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, receipt of a copy of some or all of the same material does not give you any rights to use it.

Copyright © 2019 sysmocom - s.f.m.c. GmbH 

## C.11 FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See http://www.gnu.org/copyleft/.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation. If the Document specifies that a proxy can decide which future versions of this License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Document.

## C.12 RELICENSING

"Massive Multiauthor Collaboration Site" (or "MMC Site") means any World Wide Web server that publishes copyrightable works and also provides prominent facilities for anybody to edit those works. A public wiki that anybody can edit is an example of such a server. A "Massive Multiauthor Collaboration" (or "MMC") contained in the site means any set of copyrightable works thus published on the MMC site.

"CC-BY-SA" means the Creative Commons Attribution-Share Alike 3.0 license published by Creative Commons Corporation, a not-for-profit corporation with a principal place of business in San Francisco, California, as well as future copyleft versions of that license published by that same organization.

"Incorporate" means to publish or republish a Document, in whole or in part, as part of another Document.

An MMC is "eligible for relicensing" if it is licensed under this License, and if all works that were first published under this License somewhere other than this MMC, and subsequently incorporated in whole or in part into the MMC, (1) had no cover texts or invariant sections, and (2) were thus incorporated prior to November 1, 2008.

The operator of an MMC Site may republish an MMC contained in the site under CC-BY-SA on the same site at any time before August 1, 2009, provided the MMC is eligible for relicensing.

## C.13 ADDENDUM: How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

```
Copyright (c)  YEAR  YOUR NAME.
Permission is granted to copy, distribute and/or modify this document
under the terms of the GNU Free Documentation License, Version 1.3
or any later version published by the Free Software Foundation;
with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts.
A copy of the license is included in the section entitled ``GNU
Free Documentation License''.
```

If you have Invariant Sections, Front-Cover Texts and Back-Cover Texts, replace the "with…Texts." line with this:

```
with the Invariant Sections being LIST THEIR TITLES, with the
Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST.
```

If you have Invariant Sections without Cover Texts, or some other combination of the three, merge those two alternatives to suit the situation.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.

Copyright © 2019 sysmocom - s.f.m.c. GmbH 