

# QMI QCMAP 1.3 for MPSS.NI.5.0.x

### QMI Qualcomm Mobile Access Point Svc Spec

80-ND591-34 A

February 1, 2013

Submit technical questions at:

https://support.cdmatech.com

#### Confidential and Proprietary - Qualcomm Technologies, Inc.

**NO PUBLIC DISCLOSURE PERMITTED:** Please report postings of this document on public servers or websites to: DocCtrlAgent@qualcomm.com.

**Restricted Distribution.** Not to be distributed to anyone who is not an employee of either Qualcomm or its subsidiaries without the express approval of Qualcomm's Configuration Management.

Not to be used, copied, reproduced, or modified in whole or in part, nor its contents revealed in any manner to others without the express written permission of Qualcomm Technologies, Inc.

Qualcomm reserves the right to make changes to the product(s) or information contained herein without notice. No liability is assumed for any damages arising directly or indirectly by their use or application. The information provided in this document is provided on an "as is" basis.

This document contains confidential and proprietary information and must be shredded when discarded.

Qualcomm and MSM are trademarks of Qualcomm Technologies, Inc. registered in the United States and other countries. All Qualcomm Technologies, Inc.trademarks are used with permission. Other product and brand names may be trademarks or registered trademarks of their respective owners.

This technical data may be subject to U.S. and international export, re-export, or transfer ("export") laws. Diversion contrary to U.S. and international law is strictly prohibited.

Qualcomm Technologies, Inc.
5775 Morehouse Drive
San Diego, CA 92121
U.S.A.
© 2013 Qualcomm Technologies, Inc.
All rights reserved.

# **Contents**

1	Intro	oduction	8
	1.1	Purpose	8
	1.2	Scope	8
	1.3	Conventions	8
	1.4	References	9
	1.5	Technical Assistance	ç
	1.6	Acronyms	
2	The	Generalized QMI Service Compliance	11
	2.1	Generalized QMI Service Compliance	11
	2.2	QCMAP Service Type	11
	2.3		11
		2.3.1 Response Message Result TLV	11
	2.4	2.3.1 Response Message Result TLV	12
	2.5	Service State Variables	12
		2.5.1 Shared State Variables	12
		2.5.2 State Variables Per Control Point	12
		(0) (1) (1)	4.0
3		QCMAP Messages	13
	3.1	QMI_QCMAP_MOBILE_AP_ENABLE	
		3.1.1 Request - QMI_QCMAP_MOBILE_AP_ENABLE_REQ	
		3.1.2 Response - QMI_QCMAP_MOBILE_AP_ENABLE_RESP	
		3.1.3 Description of QMI_QCMAP_MOBILE_AP_ENABLE REQ/RESP	
	3.2	QMI_QCMAP_MOBILE_AP_DISABLE	
		3.2.1 Request - QMI_QCMAP_MOBILE_AP_DISABLE_REQ	
		3.2.2 Response - QMI_QCMAP_DISABLE_RESP	
		3.2.3 Description of QMI_QCMAP_MOBILE_AP_DISABLE REQ/RESP	
	3.3	QMI_QCMAP_BRING_UP_WWAN	
		3.3.1 Request - QMI_QCMAP_BRING_UP_WWAN_REQ	
		3.3.2 Response - QMI_QCMAP_BRING_UP_WWAN_RESP	
		3.3.3 Description of QMI_QCMAP_BRING_UP_WWAN REQ/RESP	
		3.3.4 Indication - QMI_QCMAP_BRING_UP_WWAN_IND	
		3.3.5 Description of QMI_QCMAP_BRING_UP_WWAN_IND	
	3.4	QMI_QCMAP_TEAR_DOWN_WWAN	
		3.4.1 Request - QMI_QCMAP_TEAR_DOWN_WWAN_REQ	
		3.4.2 Response - QMI_QCMAP_TEAR_DOWN_WWAN_RESP	
		3.4.3 Description of QMI_QCMAP_TEAR_DOWN_WWAN REQ/RESP	
		3.4.4 Indication - QMI_QCMAP_TEAR_DOWN_WWAN_IND	
		3.4.5 Description of QMI_QCMAP_TEAR_DOWN_WWAN_IND	26
	3.5	QMI_QCMAP_GET_WWAN_STATUS	27

		Request - QMI_QCMAP_GET_WWAN_STATUS_REQ	
		Response - QMI_QCMAP_GET_WWAN_STATUS_RESP	
	3.5.3	Description of QMI_QCMAP_GET_WWAN_STATUS REQ/RESP	29
3.6		CMAP_WWAN_STATUS_IND_REG	
		Request - QMI_QCMAP_WWAN_STATUS_IND_REG_REQ	
		Response - QMI_QCMAP_WWAN_STATUS_IND_REG_RESP	31
	3.6.3	Description of QMI_QCMAP_WWAN_STATUS_IND_REG_REQ/RESP	31
3.7		CMAP_WWAN_STATUS_IND	
		Indication - QMI_QCMAP_WWAN_STATUS_IND	
		Description of QMI_QCMAP_WWAN_STATUS_IND	33
3.8		CMAP_SET_IPSEC_VPN_PASS_THROUGH	
		Request - QMI_QCMAP_SET_IPSEC_VPN_PASS_THROUGH_REQ	34
		Response - QMI_QCMAP_SET_IPSEC_VPN_PASS_THROUGH_RESP	35
		Description of QMI_QCMAP_SET_IPSEC_VPN_PASS_THROUGH REQ/RESP	35
3.9		CMAP_GET_IPSEC_VPN_PASS_THROUGH	
		Request - QMI QCMAP GET IPSEC VPN PASS THROUGH REQ	36
		Response - QMI_QCMAP_GET_IPSEC_VPN_PASS_THROUGH_RESP	37
		Description of QMI_QCMAP_GET_IPSEC_VPN_PASS_THROUGH REQ/RESP	37
3 10		CMAP_SET_PPTP_VPN_PASS_THROUGH	
00		Request - QMI QCMAP SET PPTP VPN PASS THROUGH REQ	
		Response - QMI QCMAP SET PPTP VPN PASS THROUGH RESP	39
		Description of QMI_QCMAP_SET_PPTP_VPN_PASS_THROUGH REQ/RESP	39
3 11		CMAP_GET_PPTP_VPN_PASS_THROUGH	40
0.11		Request - QMI QCMAP GET PPTP VPN PASS THROUGH REQ	40
		Response - QMI_QCMAP_GET_PPTP_VPN_PASS_THROUGH_RESP	41
		Description of QMI_QCMAP_GET_PPTP_VPN_PASS_THROUGH REQ/RESP	41
3 12		CMAP_SET_L2TP_VPN_PASS_THROUGH	42
0.12		Request - QMI_QCMAP_SET_L2TP_VPN_PASS_THROUGH_REQ	42
		Response - QMI_QCMAP_SET_L2TP_VPN_PASS_THROUGH_RESP	43
		Description of QMI_QCMAP_SET_L2TP_VPN_PASS_THROUGH REQ/RESP	43
3 13		CMAP_GET_L2TP_VPN_PASS_THROUGH	
0.10		Request - QMI_QCMAP_GET_L2TP_VPN_PASS_THROUGH_REQ	44
		Response - QMI QCMAP GET L2TP VPN PASS THROUGH RESP	
		Description of QMI_QCMAP_GET_L2TP_VPN_PASS_THROUGH REQ/RESP	
3 1/1		CMAP_SET_DYNAMIC_NAT_ENTRY_TIMEOUT	
0.14		Request - QMI_QCMAP_SET_DYNAMIC_NAT_ENTRY_TIMEOUT_REQ	
		Response - QMI_QCMAP_SET_DYNAMIC_NAT_ENTRY_TIMEOUT_RESP	
		Description of QMI_QCMAP_SET_DYNAMIC_NAT_ENTRY	77
		TIMEOUT REQ/RESP	47
3 15		CMAP_GET_DYNAMIC_NAT_ENTRY_TIMEOUT	48
0.10		Request - QMI_QCMAP_GET_DYNAMIC_NAT_ENTRY_TIMEOUT_REQ	48
		Response - QMI QCMAP GET DYNAMIC NAT ENTRY TIMEOUT RESP	49
		Description of QMI_QCMAP_GET_DYNAMIC_NAT_ENTRY	
		TIMEOUT REQ/RESP	10
3 16		CMAP_ADD_STATIC_NAT_ENTRY	
5.10		Request - QMI_QCMAP_ADD_STATIC_NAT_ENTRY_REQ	
		Response - QMI QCMAP ADD STATIC NAT ENTRY RESP	
		Description of QMI_QCMAP_ADD_STATIC_NAT_ENTRY_RESP	
2 17		CMAP_DELETE_STATIC_NAT_ENTRY	
3.17		Request - QMI QCMAP DELETE STATIC NAT ENTRY REQ	
	J. I / . I	HEGUEST WIN WONAF DELETE STATIO NAT ENTAT NEW	- :02

	3.17.2 Response - QMI_QCMAP_DELETE_STATIC_NAT_ENTRY_RESP	
	3.17.3 Description of QMI_QCMAP_DELETE_STATIC_NAT_ENTRY REQ/RESP	53
3.18	QMI_QCMAP_GET_STATIC_NAT_ENTRIES	54
	3.18.1 Request - QMI_QCMAP_GET_STATIC_NAT_ENTRIES_REQ	54
	3.18.2 Response - QMI_QCMAP_GET_STATIC_NAT_ENTRIES_RESP	
	3.18.3 Description of QMI_QCMAP_GET_STATIC_NAT_ENTRIES REQ/RESP	56
3.19	QMI_QCMAP_SET_DMZ	
	3.19.1 Request - QMI_QCMAP_SET_DMZ_REQ	
	3.19.2 Response - QMI_QCMAP_SET_DMZ_RESP	
	3.19.3 Description of QMI_QCMAP_SET_DMZ REQ/RESP	
3 20	QMI_QCMAP_GET_DMZ	
0.20	3.20.1 Request - QMI_QCMAP_GET_DMZ_REQ	
	3.20.2 Response - QMI_QCMAP_GET_DMZ_RESP	
	3.20.3 Description of QMI_QCMAP_GET_DMZ_REQ/RESP	
2 21	QMI_QCMAP_DELETE_DMZ	
5.21	3.21.1 Request - QMI_QCMAP_DELETE_DMZ_REQ	
	3.21.2 Response - QMI_QCMAP_DELETE_DMZ_RESP	
2.00	3.21.3 Description of QMI_QCMAP_DELETE_DMZ REQ/RESP	
3.22	QMI_QCMAP_GET_WWAN_CONFIG	
	3.22.1 Request - QMI_QCMAP_GET_WWAN_CONFIG_REQ	
	3.22.2 Response - QMI_QCMAP_GET_WWAN_CONFIG_RESP	
	3.22.3 Description of QMI_QCMAP_GET_WWAN_CONFIG REQ/RESP	
3.23	QMI_QCMAP_ENABLE_FIREWALL_SETTING	
	3.23.1 Request - QMI_QCMAP_ENABLE_FIREWALL_SETTING_REQ	
	3.23.2 Response - QMI_QCMAP_ENABLE_FIREWALL_SETTING_RESP	
	3.23.3 Description of QMI_QCMAP_ENABLE_FIREWALL_SETTING REQ/RESP	
3.24	QMI_QCMAP_GET_FIREWALL_SETTING	
	3.24.1 Request - QMI_QCMAP_GET_FIREWALL_SETTING_REQ	
	3.24.2 Response - QMI_QCMAP_GET_FIREWALL_SETTING_RESP	
	3.24.3 Description of QMI_QCMAP_GET_FIREWALL_SETTING REQ/RESP	
3.25	QMI_QCMAP_DISABLE_FIREWALL_SETTING	
	3.25.1 Request - QMI_QCMAP_DISABLE_FIREWALL_SETTING_REQ	
	3.25.2 Response - QMI_QCMAP_DISABLE_FIREWALL_SETTING_RESP	72
	3.25.3 Description of QMI_QCMAP_DISABLE_FIREWALL_SETTING REQ/RESP	72
3.26	QMI_QCMAP_ADD_FIREWALL_CONFIG	
	3.26.1 Request - QMI_QCMAP_ADD_FIREWALL_CONFIG_REQ	
	3.26.2 Response - QMI_QCMAP_ADD_FIREWALL_CONFIG_RESP	
	3.26.3 Description of QMI_QCMAP_ADD_FIREWALL_CONFIG REQ/RESP	75
3.27	QMI_QCMAP_DELETE_FIREWALL_CONFIG	76
	3.27.1 Request - QMI_QCMAP_DELETE_FIREWALL_CONFIG_REQ	76
	3.27.2 Response - QMI_QCMAP_DELETE_FIREWALL_CONFIG_RESP	77
	3.27.3 Description of QMI_QCMAP_DELETE_FIREWALL_CONFIG REQ/RESP	77
3.28	QMI_QCMAP_GET_FIREWALL_CONFIG	78
		78
	3.28.2 Response - QMI_QCMAP_GET_FIREWALL_CONFIG_RESP	79
	3.28.3 Description of QMI_QCMAP_GET_FIREWALL_CONFIG REQ/RESP	
3.29	QMI QCMAP STATION MODE ENABLE	
	3.29.1 Request - QMI_QCMAP_STATION_MODE_ENABLE_REQ	
	3.29.2 Response - QMI_QCMAP_STATION_MODE_ENABLE_RESP	
	3.29.3 Description of QMI QCMAP STATION MODE ENABLE REQ/RESP	

	3.30	QMI_QCMAP_STATION_MODE_DISABLE	83
		3.30.1 Request - QMI_QCMAP_STATION_MODE_DISABLE_REQ	83
		3.30.2 Response - QMI_QCMAP_STATION_MODE_DISABLE_RESP	84
		3.30.3 Description of QMI_QCMAP_STATION_MODE_DISABLE REQ/RESP	84
	3.31	QMI_QCMAP_GET_STATION_MODE	85
		3.31.1 Request - QMI_QCMAP_GET_STATION_MODE_REQ	85
		3.31.2 Response - QMI_QCMAP_GET_STATION_MODE_RESP	86
		3.31.3 Description of QMI_QCMAP_GET_STATION_MODE REQ/RESP	86
	3.32	QMI_QCMAP_ADD_EXTD_FIREWALL_CONFIG	
		3.32.1 Request - QMI_QCMAP_ADD_EXTD_FIREWALL_CONFIG_REQ	87
		3.32.2 Response - QMI_QCMAP_ADD_EXTD_FIREWALL_CONFIG_RESP	90
		3.32.3 Description of QMI_QCMAP_ADD_EXTD_FIREWALL_CONFIG REQ/RESP	91
	3.33	QMI_QCMAP_GET_EXTD_FIREWALL_CONFIG	
		3.33.1 Request - QMI_QCMAP_GET_EXTD_FIREWALL_CONFIG_REQ	
		3.33.2 Response - QMI_QCMAP_GET_EXTD_FIREWALL_CONFIG_RESP	93
		3.33.3 Description of QMI_QCMAP_GET_EXTD_FIREWALL_CONFIG REQ/RESP	
	3.34	QMI_QCMAP_GET_FIREWALL_CONFIG_HANDLE_LIST	
		3.34.1 Request - QMI_QCMAP_GET_FIREWALL_CONFIG_HANDLE_LIST_REQ	97
		3.34.2 Response - QMI_QCMAP_GET_FIREWALL_CONFIG_HANDLE_LIST_RESP .	98
		3.34.3 Description of QMI_QCMAP_GET_FIREWALL_CONFIG	
		HANDLE_LIST REQ/RESP	99
	3.35	QMI_QCMAP_CHANGE_NAT_TYPE	100
		3.35.1 Request - QMI_QCMAP_CHANGE_NAT_TYPE_REQ	
		3.35.2 Response - QMI_QCMAP_CHANGE_NAT_TYPE_RESP	
		3.35.3 Description of QMI_QCMAP_CHANGE_NAT_TYPE REQ/RESP	
	3.36	QMI_QCMAP_GET_NAT_TYPE	
		3.36.1 Request - QMI_QCMAP_GET_NAT_TYPE_REQ	
		3.36.2 Response - QMI_QCMAP_GET_NAT_TYPE_RESP	
		3.36.3 Description of QMI_QCMAP_GET_NAT_TYPE REQ/RESP	104
	Oall	End Bosses	105
A			105
		Call End Reasons	
	H.Z	Verbose Call End Reasons	10/

# **List of Tables**

1-1	Reference documents and standards
1-2	Acronyms
3-1	QMI_QCMAP messages
A-1	Call end reasons
A-2	Verbose call end reasons

# **Revision History**

Revision	Date	Description		
A Feb 2013		Initial release. Created from 80-VB816-34 B.		
		Updates for this revision include minor version 2 and minor version 3.		
		Updated sections 2.3.1 and 3.1.3.		
		Added new TLVs:  • SSID2 IP address info  • NAT type info		
		Added new messages:  • QMI_QCMAP_CHANGE_NAT_TYPE (Section 3.35)  • QMI_QCMAP_GET_NAT_TYPE (Section 3.36)		

# 1 Introduction

### 1.1 Purpose

This specification documents Major Version 1 of the Qualcomm Messaging Interface (QMI) for Qualcomm Mobile Access Point Service (QMI\_QCMAP).

QMI\_QCMAP provides a command set to interface with a wireless mobile station to access mobile AP services.

### 1.2 Scope

This document is intended for software developers using QMI\_QCMAP on a host processor and interacting with a Qualcomm MSM<sup>TM</sup> device for controlling Qualcomm mobile access point functionality.

This document provides the following details about QMI\_QCMAP:

- Theory of operation Chapter 2 provides the theory of operation of QMI\_QCMAP. The chapter includes messaging conventions, assigned QMI service type, fundamental service concepts, and state variables related to the service.
- Message formats, syntax, and semantics Chapter 3 provides the specific syntax and semantics of messages included in this version of the QMI\_QCMAP specification.
- Additional information Appendix A provides tables for call end reasons and verbose call end reasons.

### 1.3 Conventions

Function declarations, function names, type declarations, and code samples appear in a different font. For example, #include.

Parameter types are indicated by arrows:

- → Designates an input parameter
- ← Designates an output parameter
- → Designates a parameter used for both input and output

### 1.4 References

Reference documents are listed in Table 1-1. Reference documents that are no longer applicable are deleted from this table; therefore, reference numbers might not be sequential.

Table 1-1 Reference documents and standards

Ref.	Document						
Qual	Qualcomm Technologies						
Q1	Application Note: Software Glossary for Customers CL93-V3077-1						
Q2	Qualcomm MSM Interface (QMI) Architecture	80-VB816-1					
Q3	QMI WDS for MPSS.NI.5.0.x, QMI Wireless Data Svc Spec	80-ND591-5					
Stand	dards						
S1	User Datagram Protocol	RFC 768 (Aug 1980)					
S2	Internet Protocol DARPA Internet Program Protocol	RFC 791 (Sep 1981)					
	Specification						
S3	Internet Control Message Protocol DARPA Internet Program	RFC 792 (Sep 1981)					
	Protocol Specification						
S4	Transmission Control Protocol DARPA Internet Program	RFC 793 (Sep 1981)					
	Protocol Specification						
S5	Internet Protocol Version 6 (IPv6) Specification	RFC 2460 (Dec 1998)					
S6	Internet Protocol Version 6 (IPv6) Addressing Architecture	RFC 3513 (Apr 2003)					
S7	IP Encapsulating Security Payload (ESP) RFC 4303 (Dec 2005)						

### 1.5 Technical Assistance

For assistance or clarification on information in this document, submit a case to Qualcomm Technologies at <a href="https://support.cdmatech.com">https://support.cdmatech.com</a>.

If you do not have access to the CDMATech Support website, register for access or send email to support.cdmatech@qti.qualcomm.com.

### 1.6 Acronyms

For definitions of terms and abbreviations, refer to [Q1]. Table 1-2 lists terms that are specific to this document.

Table 1-2 Acronyms

Acronym	Definition
AP	access point
DMZ	DMZ (sometimes referred to as a perimeter network) is a physical or logical
	subnetwork that contains and exposes an organization's external services to a larger
	untrusted network, usually the Internet. The purpose of a DMZ is to add an additional
	layer of security to an organization's LAN.
DNS	domain name service
ESP	Encapsulating Security Payload Protocol
ICMP	Internet Control Message Protocol
IPSec	Internet Protocol security

Table 1-2 Acronyms (cont.)

Acronym Definition					
L2TP	Layer 2 Tunneling Protocol				
MIP	Mobile Internet Protocol				
NAT	network address translation				
PPTP	Point-to-Point Tunneling Protocol				
QCMAP	Qualcomm Mobile Access Point Service				
QMI	Qualcomm messaging interface				
SNAT	static NAT				
SSID	service set identifier				
STA	station				
TCP	Transmission Control Protocol				
TE	terminal equipment				
TLV	type-length-value				
TOS	type of service				
UDP	User Datagram Protocol				
VPN	virtual private network				

# 2 Theory of Operation

### 2.1 Generalized QMI Service Compliance

The QMI\_QCMAP service complies with the generalized QMI service specification, including the rules for messages, indications and responses, byte ordering, arbitration, constants, result, and error code values described in [Q2]. Extensions to the generalized QMI service theory of operation are noted in subsequent sections of this chapter.

### 2.2 QCMAP Service Type

QCMAP is assigned QMI service type 0x1E.

### 2.3 Message Definition Template

### 2.3.1 Response Message Result TLV

This Type-Length-Value (TLV) is present in all Response messages defined in this document. It is not present in the Indication messages.

	Name	Version introduced	Version last modified
Result Code		Corresponding	N/A
		command's Version	
		introduced	

Field	Field	Field	Parameter	Size	Description	
	value	type		(byte)		
Туре	0x02			1	Result Code	
Length	4			2		
Value	$\rightarrow$	uint16	qmi_result	2	Result code	
					• QMI_RESULT_SUCCESS	
					• QMI_RESULT_FAILURE	
		uint16	qmi_error	2	Error code – Possible error code values	
					are described in the error codes section	
					of each message definition	

### 2.4 QMI\_QCMAP Fundamental Concepts

QMI\_QCMAP provides a command set to interface with a wireless mobile station to access mobile AP services. The QMI\_QCMAP service supports only one client per QMI control channel.

### 2.5 Service State Variables

#### 2.5.1 Shared State Variables

No QMI\_QCMAP state variables are shared across control points.

#### 2.5.2 State Variables Per Control Point

Name	Description		Default
	, ax	values	value
register_indication	WWAN status indication registration per	• FALSE	FALSE
	mobile AP handle	• TRUE	

Table 3-1 QMI\_QCMAP messages

Command	ID	Description
QMI_QCMAP_MOBILE_AP_ENABLE	0x0020	Enables the mobile AP functionality via a single mobile AP instance on the modem.
QMI_QCMAP_MOBILE_AP_DISABLE	0x0021	Disables the mobile AP functionality for a mobile AP instance on the modem.
QMI_QCMAP_BRING_UP_WWAN	0x0022	Invokes bringing up the WWAN from the mobile AP.
QMI_QCMAP_BRING_UP_WWAN_IND	0x0022 indication	Indicates the completion of processing a QMI_QCMAP_BRING_UP_WWAN_REQ.
QMI_QCMAP_TEAR_DOWN_WWAN	0x0023	Tears down the WWAN.
QMI_QCMAP_TEAR_DOWN_WWAN_IND	0x0023 indication	Indicates the completion of processing a QMI_QCMAP_TEAR_DOWN_WWAN_REQ.
QMI_QCMAP_GET_WWAN_STATUS	0x0024	Queries the current WWAN status.
QMI_QCMAP_WWAN_STATUS_IND_REG	0x003A	Registers/deregisters the control point to receive QMI_QCMAP_WWAN_STATUS_IND.
QMI_QCMAP_WWAN_STATUS_IND	0x003E	Indicates a change in the current mobile AP WWAN connection status.
QMI_QCMAP_SET_IPSEC_VPN_PASS_ THROUGH	0x0026	Configures the Internet Protocol security (IPSec) Virtual Private Network (VPN) passthrough setting.
QMI_QCMAP_GET_IPSEC_VPN_PASS_ THROUGH	0x0025	Queries the IPSec VPN passthrough setting.
QMI_QCMAP_SET_PPTP_VPN_PASS_ THROUGH	0x0028	Configures the Point-to-Point Tunneling Protocol (PPTP) VPN passthrough setting.
QMI_QCMAP_GET_PPTP_VPN_PASS_ THROUGH	0x0027	Queries the PPTP VPN passthrough setting.
QMI_QCMAP_SET_L2TP_VPN_PASS_ THROUGH	0x002A	Configures the Layer 2 Tunneling Protocol (L2TP) VPN passthrough setting.
QMI_QCMAP_GET_L2TP_VPN_PASS_ THROUGH	0x0029	Queries the L2TP VPN passthrough setting.

Table 3-1 QMI\_QCMAP messages (cont.)

QMI_QCMAP_SET_DYNAMIC_NAT_ ENTRY_TIMEOUT         0x002C         Sets the Network Address Translation (NAT) entry timeout.           QMI_QCMAP_GET_DYNAMIC_NAT_ ENTRY_TIMEOUT         0x002B         Queries the NAT entry timeout.           QMI_QCMAP_ADD_STATIC_NAT_ENTRY         0x002D         Adds a static NAT entry.           QMI_QCMAP_DELETE_STATIC_NAT_ENTRY         0x002E         Deletes a static NAT entry.           QMI_QCMAP_GET_STATIC_NAT_ENTRIES         0x002F         Queries all static NAT entry.           QMI_QCMAP_GET_DMZ         0x0030         Sets the DMZ (perimeter network) IP address for the mobile AP.           QMI_QCMAP_GET_DMZ         0x0031         Deletes the DMZ entry or DMZ IP address.           QMI_QCMAP_DELETE_DMZ         0x0031         Deletes the DMZ entry or DMZ IP address.           QMI_QCMAP_GET_WWAN_CONFIG         0x0033         Queries the WWAN IP configuration.           QMI_QCMAP_ENABLE_FIREWALL_SETTING         0x0034         Enables the firewall setting.           QMI_QCMAP_DISABLE_FIREWALL_SETTING         0x0035         Queries the firewall setting.           QMI_QCMAP_ADD_FIREWALL_CONFIG         0x0037         Adds a firewall configuration rule.           QMI_QCMAP_GET_FIREWALL_CONFIG         0x0038         Queries the firewall configuration rule.           QMI_QCMAP_STATION_MODE_ENABLE         0x0038         Enables Station (STA) mode functionality for a mobile AP instance on t	Command	ID	Description
ENTRY_TIMEOUT  QMI_QCMAP_GET_DYNAMIC_NAT_ ENTRY_TIMEOUT  QMI_QCMAP_ADD_STATIC_NAT_ENTRY  QMI_QCMAP_ADD_STATIC_NAT_ENTRY  QMI_QCMAP_DELETE_STATIC_NAT_ENTRIES  QMI_QCMAP_GET_STATIC_NAT_ENTRIES  QMI_QCMAP_GET_DMZ  QMI_QCMAP_GET_DMZ  QMI_QCMAP_GET_DMZ  QMI_QCMAP_GET_DMZ  QMI_QCMAP_GET_DMZ  QMI_QCMAP_GET_DMZ  QMI_QCMAP_GET_WWAN_CONFIG  QMI_QCMAP_GET_WWAN_CONFIG  QMI_QCMAP_GET_FIREWALL_ SETTING  QMI_QCMAP_GET_FIREWALL_ QMI_QCMAP_DISABLE_FIREWALL_ QMI_QCMAP_DISABLE_ QMI_QCMAP_DISABLE_FIREWALL_ QMI_QCMAP_DISABLE_ QMI_QCMAP_DISABLE_ QMI_QCMAP_DISABLE_ QMI_QCMAP_DISABLE_ QMI_QCMAP_DISABLE_ QMI_QCMAP_DISABLE_ QMI_QCMAP_DISABLE_ QMI_QCMAP_DISABLE_ QMI_QCMAP_STATION_MODE_DISABLE  QMI_QCMAP_STATION_MODE_DISABLE  QMI_QCMAP_STATION_MODE_DISABLE  QMI_QCMAP_STATION_MODE_DISABLE  QMI_QCMAP_DISABLE_FIREWALL_ QMI_QCMAP_DISABLE_ QMI_QCMAP_			-
QMI_QCMAP_GET_DYNAMIC_NAT_ ENTRY_TIMEOUT         0x002D         Queries the NAT entry timeout.           QMI_QCMAP_ADD_STATIC_NAT_ENTRY         0x002D         Adds a static NAT entry.           QMI_QCMAP_DELETE_STATIC_NAT_ENTRIES         0x002E         Deletes a static NAT entry.           QMI_QCMAP_GET_STATIC_NAT_ENTRIES         0x003F         Queries all static NAT entries.           QMI_QCMAP_GET_DMZ         0x0030         Sets the DMZ (perimeter network) IP address for the mobile AP.           QMI_QCMAP_GET_DMZ         0x0031         Queries the DMZ IP address on the mobile AP.           QMI_QCMAP_GET_DMZ         0x0031         Deletes the DMZ entry or DMZ IP address.           QMI_QCMAP_GET_WWAN_CONFIG         0x0033         Queries the WWAN IP configuration.           QMI_QCMAP_ENABLE_FIREWALL_SETTING         0x0034         Enables the firewall setting.           QMI_QCMAP_GET_FIREWALL_SETTING         0x0035         Queries the firewall setting.           QMI_QCMAP_DISABLE_FIREWALL_CONFIG         0x0036         Disables the firewall setting.           QMI_QCMAP_ADD_FIREWALL_CONFIG         0x0037         Adds a firewall configuration rule.           QMI_QCMAP_GET_FIREWALL_CONFIG         0x0038         Queries the firewall configuration rule.           QMI_QCMAP_STATION_MODE_ENABLE         0x003B         Enables Station (STA) mode functionality for a mobile AP instance on the modem.		0.0020	
ENTRY_TIMEOUT  QMI_QCMAP_ADD_STATIC_NAT_ENTRY  QMI_QCMAP_DELETE_STATIC_NAT_ENTRIES  QMI_QCMAP_DELETE_STATIC_NAT_ENTRIES  QMI_QCMAP_GET_STATIC_NAT_ENTRIES  QMI_QCMAP_SET_DMZ  QMI_QCMAP_GET_DMZ  QMI_QCMAP_GET_DMZ  QMI_QCMAP_DELETE_DMZ  QMI_QCMAP_DELETE_DMZ  QMI_QCMAP_DELETE_DMZ  QMI_QCMAP_GET_WWAN_CONFIG  QMI_QCMAP_GET_WWAN_CONFIG  QMI_QCMAP_ENABLE_FIREWALL_ SETTING  QMI_QCMAP_GET_FIREWALL_ SETTING  QMI_QCMAP_DELETE_DMZ  QMI_QCMAP_DELETE_DMZ  QMI_QCMAP_DELETE_DMZ  QMI_QCMAP_DELETE_DMZ  QMI_QCMAP_DELETE_FIREWALL_ SETTING  QMI_QCMAP_DELETE_FIREWALL_ QMI_QCMAP_DELETE_FIREWALL_ SETTING  QMI_QCMAP_DELETE_FIREWALL_ QMI_QCMAP_DELETE_FIREWALL_ QMI_QCMAP_DELETE_FIREWALL_ QMI_QCMAP_DELETE_FIREWALL_ QMI_QCMAP_DELETE_FIREWALL_CONFIG  QMI_QCMAP_DELETE_FIREWALL_CONFIG  QMI_QCMAP_DELETE_FIREWALL_CONFIG  QMI_QCMAP_STATION_MODE_ENABLE  QMI_QCMAP_STATION_MODE_DISABLE  QMI_QCMAP_STATION_MODE_DISABLE  QMI_QCMAP_GET_STATION_MODE_DISABLE  QMI_QCMAP_GET_STATION_MODE_DISABLE  QMI_QCMAP_GET_STATION_MODE  QMI_QCMAP_GET_EXTD_FIREWALL_ QX003C  Queries the STA mode functionality for a mobile AP instance on the modem.  QMI_QCMAP_GET_EXTD_FIREWALL_ QX003C  QUERIEN THE MODE  QX003D  QUERIEN THE MODE	_	0x002B	
QMI_QCMAP_ADD_STATIC_NAT_ENTRY         0x002D         Adds a static NAT entry.           QMI_QCMAP_DELETE_STATIC_NAT_ENTRIES         0x002E         Deletes a static NAT entry.           QMI_QCMAP_GET_STATIC_NAT_ENTRIES         0x002F         Queries all static NAT entry.           QMI_QCMAP_SET_DMZ         0x0030         Sets the DMZ (perimeter network) IP address for the mobile AP.           QMI_QCMAP_GET_DMZ         0x0032         Queries the DMZ (perimeter network) IP address for the mobile AP.           QMI_QCMAP_GET_DMZ         0x0031         Deletes the DMZ entry or DMZ IP address.           QMI_QCMAP_DELETE_DMZ         0x0031         Deletes the DMZ entry or DMZ IP address.           QMI_QCMAP_ENABLE_FIREWALL_SETTING         0x0034         Enables the firewall setting.           QMI_QCMAP_GET_FIREWALL_SETTING         0x0035         Queries the firewall setting.           QMI_QCMAP_DISABLE_FIREWALL_SETTING         0x0036         Disables the firewall setting.           QMI_QCMAP_ADD_FIREWALL_CONFIG         0x0037         Adds a firewall configuration rule.           QMI_QCMAP_DELETE_FIREWALL_CONFIG         0x0039         Deletes a firewall configuration rule.           QMI_QCMAP_GET_FIREWALL_CONFIG         0x0038         Queries the firewall configuration rules.           QMI_QCMAP_STATION_MODE_DISABLE         0x0036         Disables STA mode functionality for a mobile AP instance on the modem.		0X002 <b>D</b>	Queries the 1411 entry timeout.
QMI_QCMAP_DELETE_STATIC_NAT_ENTRIES  QMI_QCMAP_SET_DMZ  QMI_QCMAP_SET_DMZ  QMI_QCMAP_SET_DMZ  QMI_QCMAP_GET_DMZ  QMI_QCMAP_GET_DMZ  QMI_QCMAP_GET_DMZ  QMI_QCMAP_DELETE_DMZ  QMI_QCMAP_DELETE_DMZ  QMI_QCMAP_DELETE_DMZ  QMI_QCMAP_GET_WWAN_CONFIG  QMI_QCMAP_ENABLE_FIREWALL_ SETTING  QMI_QCMAP_GET_FIREWALL_SETTING  QMI_QCMAP_DISABLE_FIREWALL_ QMI_QCMAP_DISABLE_FIREWALL_CONFIG  QMI_QCMAP_DISABLE_FIREWALL_CONFIG  QMI_QCMAP_DISABLE_FIREWALL_CONFIG  QMI_QCMAP_STATION_MODE_ENABLE  QMI_QCMAP_STATION_MODE_DISABLE  QMI_QCMAP_STATION_MODE_DISABLE  QMI_QCMAP_GET_STATION_MODE  QMI_QCMAP_GET_STATION_FIREWALL_ QX003F  QMI_QCMAP_GET_STATION_FIREWALL_ QX004F  QMI_QCMAP_GET_STATION_FIREWALL_ QX004F  QMI_QCMAP_GET_STATION_FIREWALL_ QX004F  QMI_QCMAP_GET_STATION_FIREWALL_ QX004G  QMI_QCMAP_GET_STATION_FIREWALL_ QX004G  QMI_QCMAP_GET_STATION_FIREWALL_ QX004G  QMI_QCMAP_GET_FIREWALL_ QX004G  QMI_QCMAP_GET_FIREWALL_ QX004G  Gets the handles of all the firewall rules.	_	0x002D	Adds a static NAT entry
ENTRY  QMI_QCMAP_GET_STATIC_NAT_ENTRIES  QMI_QCMAP_SET_DMZ  QMI_QCMAP_GET_DMZ  QMI_QCMAP_GET_DMZ  QMI_QCMAP_GET_DMZ  QMI_QCMAP_GET_DMZ  QMI_QCMAP_DELETE_DMZ  QMI_QCMAP_DELETE_DMZ  QMI_QCMAP_GET_WWAN_CONFIG  QMI_QCMAP_GET_WWAN_CONFIG  QMI_QCMAP_ENABLE_FIREWALL_ SETTING  QMI_QCMAP_GET_FIREWALL_SETTING  QMI_QCMAP_GET_FIREWALL_SETTING  QMI_QCMAP_DISABLE_FIREWALL_ SETTING  QMI_QCMAP_DISABLE_FIREWALL_ QMI_QCMAP_DISABLE_FIREWALL_CONFIG  QMI_QCMAP_DISABLE_FIREWALL_CONFIG  QMI_QCMAP_GET_FIREWALL_CONFIG  QMI_QCMAP_GET_FIREWALL_CONFIG  QMI_QCMAP_GET_FIREWALL_CONFIG  QMI_QCMAP_GET_FIREWALL_CONFIG  QMI_QCMAP_STATION_MODE_ENABLE  QMI_QCMAP_STATION_MODE_DISABLE  QMI_QCMAP_GET_STATION_MODE  QMI_QCMAP_GET_STATION_MODE  QMI_QCMAP_ADD_EXTD_FIREWALL_ QMI_QCMAP_ADD_EXTD_FIREWALL_ QMI_QCMAP_GET_EXTD_FIREWALL_ QMI_QCMAP_GET_EXTD_FIREWALL_ QMI_QCMAP_GET_EXTD_FIREWALL_ QMI_QCMAP_GET_FIREWALL_ QMI_QCMAP_GET_EXTD_FIREWALL_ QMI_QCMAP_GET_FIREWALL_ QMI_QCMA	QMI_QCMAI_ADD_STATIC_NAI_ENTRI	0X002D	Adds a static IVAI chiry.
ENTRY  QMI_QCMAP_GET_STATIC_NAT_ENTRIES  QMI_QCMAP_SET_DMZ  QMI_QCMAP_GET_DMZ  QMI_QCMAP_GET_DMZ  QMI_QCMAP_GET_DMZ  QMI_QCMAP_GET_DMZ  QMI_QCMAP_DELETE_DMZ  QMI_QCMAP_DELETE_DMZ  QMI_QCMAP_GET_WWAN_CONFIG  QMI_QCMAP_GET_WWAN_CONFIG  QMI_QCMAP_ENABLE_FIREWALL_ SETTING  QMI_QCMAP_GET_FIREWALL_SETTING  QMI_QCMAP_GET_FIREWALL_SETTING  QMI_QCMAP_DISABLE_FIREWALL_ SETTING  QMI_QCMAP_DISABLE_FIREWALL_ QMI_QCMAP_DISABLE_FIREWALL_CONFIG  QMI_QCMAP_DISABLE_FIREWALL_CONFIG  QMI_QCMAP_GET_FIREWALL_CONFIG  QMI_QCMAP_GET_FIREWALL_CONFIG  QMI_QCMAP_GET_FIREWALL_CONFIG  QMI_QCMAP_GET_FIREWALL_CONFIG  QMI_QCMAP_STATION_MODE_ENABLE  QMI_QCMAP_STATION_MODE_DISABLE  QMI_QCMAP_GET_STATION_MODE  QMI_QCMAP_GET_STATION_MODE  QMI_QCMAP_ADD_EXTD_FIREWALL_ QMI_QCMAP_ADD_EXTD_FIREWALL_ QMI_QCMAP_GET_EXTD_FIREWALL_ QMI_QCMAP_GET_EXTD_FIREWALL_ QMI_QCMAP_GET_EXTD_FIREWALL_ QMI_QCMAP_GET_FIREWALL_ QMI_QCMAP_GET_EXTD_FIREWALL_ QMI_QCMAP_GET_FIREWALL_ QMI_QCMA	OMI OCMAP DELETE STATIC NAT	0x002E	Deletes a static NAT entry.
QMI_QCMAP_SET_DMZ  QMI_QCMAP_GET_DMZ  QMI_QCMAP_GET_DMZ  QMI_QCMAP_DELETE_DMZ  QMI_QCMAP_DELETE_DMZ  QMI_QCMAP_GET_WWAN_CONFIG  QMI_QCMAP_ENABLE_FIREWALL_ SETTING  QMI_QCMAP_GET_FIREWALL_SETTING  QMI_QCMAP_DELETE_DMZ  QMI_QCMAP_DISABLE_FIREWALL_ SETTING  QMI_QCMAP_DISABLE_ QMI_QCMAP_DISABLE  QMI_QCMAP_DISABLE  QMI_QCMAP_STATION_MODE_ENABLE  QMI_QCMAP_STATION_MODE_DISABLE  QMI_QCMAP_STATION_MODE_DISABLE  QMI_QCMAP_GET_STATION_MODE  QMI_QCMAP_GET_STATION_MODE  QMI_QCMAP_GET_STATION_FIREWALL_ QMI_QCMAP_ADD_EXTD_FIREWALL_ QMI_QCMAP_ADD_EXTD_FIREWALL_ QMI_QCMAP_GET_EXTD_FIREWALL_ QMI_QCMAP_GET_EXTD_FIREWALL_ QMI_QCMAP_GET_EXTD_FIREWALL_ QMI_QCMAP_GET_EXTD_FIREWALL_ QMI_QCMAP_GET_EXTD_FIREWALL_ QMI_QCMAP_GET_EXTD_FIREWALL_ QMI_QCMAP_GET_FIREWALL_ QMI_QCMAP_GET_FIREWALL			
QMI_QCMAP_SET_DMZ  QMI_QCMAP_GET_DMZ  QMI_QCMAP_GET_DMZ  QMI_QCMAP_DELETE_DMZ  QMI_QCMAP_DELETE_DMZ  QMI_QCMAP_GET_WWAN_CONFIG  QMI_QCMAP_ENABLE_FIREWALL_ SETTING  QMI_QCMAP_GET_FIREWALL_SETTING  QMI_QCMAP_DELETE_FIREWALL_ SETTING  QMI_QCMAP_DISABLE_FIREWALL_ SETTING  QMI_QCMAP_DISABLE_FIREWALL_CONFIG  QMI_QCMAP_DISABLE_FIREWALL_CONFIG  QMI_QCMAP_DISABLE_FIREWALL_CONFIG  QMI_QCMAP_DISABLE_FIREWALL_CONFIG  QMI_QCMAP_DISABLE_ QMI_QCMAP_STATION_MODE_ENABLE  QMI_QCMAP_STATION_MODE_ENABLE  QMI_QCMAP_STATION_MODE_DISABLE  QMI_QCMAP_GET_STATION_MODE  QMI_QCMAP_GET_STATION_MODE  QMI_QCMAP_GET_STATION_MODE  QMI_QCMAP_ADD_EXTD_FIREWALL_ QMI_QCMAP_ADD_EXTD_FIREWALL_ CONFIG  QMI_QCMAP_GET_EXTD_FIREWALL_ QMI_QCMAP_GET_FIREWALL_ CONFIG  QMI_QCMAP_GET_FIREWALL_ QMI_QCMAP_GET_FIREWALL	QMI QCMAP GET STATIC NAT ENTRIES	0x002F	Queries all static NAT entries.
Address for the mobile AP.  QMI_QCMAP_GET_DMZ  QMI_QCMAP_DELETE_DMZ  QMI_QCMAP_DELETE_DMZ  QMI_QCMAP_GET_WWAN_CONFIG  QMI_QCMAP_ENABLE_FIREWALL_ SETTING  QMI_QCMAP_GET_FIREWALL_SETTING  QMI_QCMAP_GET_FIREWALL_SETTING  QMI_QCMAP_DISABLE_FIREWALL_ SETTING  QMI_QCMAP_ADD_FIREWALL_CONFIG  QMI_QCMAP_DELETE_FIREWALL_CONFIG  QMI_QCMAP_DELETE_FIREWALL_CONFIG  QMI_QCMAP_DELETE_FIREWALL_CONFIG  QMI_QCMAP_DELETE_FIREWALL_CONFIG  QMI_QCMAP_STATION_MODE_ENABLE  QMI_QCMAP_STATION_MODE_DISABLE  QMI_QCMAP_STATION_MODE_DISABLE  QMI_QCMAP_GET_STATION_MODE  QMI_QCMAP_GET_STATION_MODE  QMI_QCMAP_GET_STATION_MODE  QMI_QCMAP_ADD_EXTD_FIREWALL_ QMI_QCMAP_ADD_EXTD_FIREWALL_ CONFIG  QMI_QCMAP_GET_EXTD_FIREWALL_ CONFIG  QMI_QCMAP_GET_EXTD_FIREWALL_ CONFIG  QMI_QCMAP_GET_EXTD_FIREWALL_ CONFIG  QMI_QCMAP_GET_EXTD_FIREWALL_ CONFIG  QMI_QCMAP_GET_FIREWALL_ CONFIG  QMI_QCMAP_GET_FIREWALL_CONFIG_ QMI_QCMAP_GET_FIREWALL_ CONFIG  QMI_QCMAP_GET_FIREWALL_CONFIG_ QMI_QCMAP_GE			
address for the mobile AP.  QMI_QCMAP_GET_DMZ  QMI_QCMAP_DELETE_DMZ  QMI_QCMAP_DELETE_DMZ  QMI_QCMAP_GET_WWAN_CONFIG  QMI_QCMAP_ENABLE_FIREWALL_ SETTING  QMI_QCMAP_GET_FIREWALL_SETTING  QMI_QCMAP_GET_FIREWALL_SETTING  QMI_QCMAP_BISABLE_FIREWALL_ SETTING  QMI_QCMAP_ADD_FIREWALL_CONFIG  QMI_QCMAP_DELETE_FIREWALL_CONFIG  QMI_QCMAP_BET_FIREWALL_CONFIG  QMI_QCMAP_BET_FIREWALL_CONFIG  QMI_QCMAP_BET_FIREWALL_CONFIG  QMI_QCMAP_BET_FIREWALL_CONFIG  QMI_QCMAP_STATION_MODE_ENABLE  QMI_QCMAP_STATION_MODE_DISABLE  QMI_QCMAP_GET_STATION_MODE  QMI_QCMAP_GET_STATION_MODE  QMI_QCMAP_GET_STATION_MODE  QMI_QCMAP_GET_STATION_MODE  QMI_QCMAP_ADD_EXTD_FIREWALL_ QMI_QCMAP_ADD_EXTD_FIREWALL_ CONFIG  QMI_QCMAP_GET_EXTD_FIREWALL_ CONFIG  QMI_QCMAP_GET_EXTD_FIREWALL_ CONFIG  QMI_QCMAP_GET_EXTD_FIREWALL_ CONFIG  QMI_QCMAP_GET_FIREWALL_ CONFIG  QMI_QCMAP_GET_FIREWALL_CONFIG_ QMI_QCMAP_GET_FIREWALL_ CONFIG  QMI_QCMAP_GET_FIREWALL_CONFIG_ QM	QMI_QCMAP_SET_DMZ	0x0030	Sets the DMZ (perimeter network) IP
QMI_QCMAP_GET_WWAN_CONFIG  QMI_QCMAP_ENABLE_FIREWALL_ SETTING  QMI_QCMAP_DISABLE_FIREWALL_ QMI_QCMAP_DISABLE_FIREWALL_ QMI_QCMAP_DISABLE_FIREWALL_ QMI_QCMAP_DISABLE_FIREWALL_ QMI_QCMAP_DISABLE_FIREWALL_CONFIG  QMI_QCMAP_DELETE_FIREWALL_CONFIG  QMI_QCMAP_DELETE_FIREWALL_CONFIG  QMI_QCMAP_GET_FIREWALL_CONFIG  QMI_QCMAP_GET_FIREWALL_CONFIG  QMI_QCMAP_GET_FIREWALL_CONFIG  QMI_QCMAP_GET_FIREWALL_CONFIG  QMI_QCMAP_GET_FIREWALL_CONFIG  QMI_QCMAP_STATION_MODE_ENABLE  QMI_QCMAP_STATION_MODE_DISABLE  QMI_QCMAP_STATION_MODE_DISABLE  QMI_QCMAP_GET_STATION_MODE  QMI_QCMAP_GET_STATION_MODE  QMI_QCMAP_GET_STATION_MODE  QMI_QCMAP_GET_STATION_MODE  QMI_QCMAP_GET_STATION_MODE  QMI_QCMAP_GET_STATION_MODE  QMI_QCMAP_GET_STATION_MODE  QMI_QCMAP_GET_STATION_MODE  QMI_QCMAP_GET_STATION_MODE  QMI_QCMAP_GET_EXTD_FIREWALL_ CONFIG  QMI_QCMAP_GET_EXTD_FIREWALL_ CONFIG  QMI_QCMAP_GET_EXTD_FIREWALL_ CONFIG  QMI_QCMAP_GET_FIREWALL_ CONFIG  QMI_QCMAP_GET_FIREWALL_CONFIG  QMI_GCMAP_GET_FIREWALL_CONFIG  QMI_QCMAP_GET_FIREWALL_CONFIG  QMI_QCM			•
QMI_QCMAP_GET_WWAN_CONFIG  QMI_QCMAP_ENABLE_FIREWALL_ SETTING  QMI_QCMAP_DISABLE_FIREWALL_ QMI_QCMAP_DISABLE_FIREWALL_ QMI_QCMAP_DISABLE_FIREWALL_ QMI_QCMAP_DISABLE_FIREWALL_ QMI_QCMAP_DISABLE_FIREWALL_CONFIG  QMI_QCMAP_DELETE_FIREWALL_CONFIG  QMI_QCMAP_DELETE_FIREWALL_CONFIG  QMI_QCMAP_GET_FIREWALL_CONFIG  QMI_QCMAP_GET_FIREWALL_CONFIG  QMI_QCMAP_GET_FIREWALL_CONFIG  QMI_QCMAP_GET_FIREWALL_CONFIG  QMI_QCMAP_GET_FIREWALL_CONFIG  QMI_QCMAP_STATION_MODE_ENABLE  QMI_QCMAP_STATION_MODE_DISABLE  QMI_QCMAP_STATION_MODE_DISABLE  QMI_QCMAP_GET_STATION_MODE  QMI_QCMAP_GET_STATION_MODE  QMI_QCMAP_GET_STATION_MODE  QMI_QCMAP_GET_STATION_MODE  QMI_QCMAP_GET_STATION_MODE  QMI_QCMAP_GET_STATION_MODE  QMI_QCMAP_GET_STATION_MODE  QMI_QCMAP_GET_STATION_MODE  QMI_QCMAP_GET_STATION_MODE  QMI_QCMAP_GET_EXTD_FIREWALL_ CONFIG  QMI_QCMAP_GET_EXTD_FIREWALL_ CONFIG  QMI_QCMAP_GET_EXTD_FIREWALL_ CONFIG  QMI_QCMAP_GET_FIREWALL_ CONFIG  QMI_QCMAP_GET_FIREWALL_CONFIG  QMI_GCMAP_GET_FIREWALL_CONFIG  QMI_QCMAP_GET_FIREWALL_CONFIG  QMI_QCM	QMI QCMAP GET DMZ	0x0032	Queries the DMZ IP address on the
QMI_QCMAP_ENABLE_FIREWALL_ SETTING QMI_QCMAP_GET_FIREWALL_SETTING QMI_QCMAP_GET_FIREWALL_SETTING QMI_QCMAP_DISABLE_FIREWALL_ SETTING QMI_QCMAP_DISABLE_FIREWALL_ SETTING QMI_QCMAP_ADD_FIREWALL_CONFIG QMI_QCMAP_DELETE_FIREWALL_CONFIG QMI_QCMAP_DELETE_FIREWALL_CONFIG QMI_QCMAP_DELETE_FIREWALL_CONFIG QMI_QCMAP_STATION_MODE_ENABLE QMI_QCMAP_STATION_MODE_DISABLE QMI_QCMAP_STATION_MODE_DISABLE QMI_QCMAP_GET_STATION_MODE QMI_QCMAP_GET_EXTD_FIREWALL_ CONFIG QMI_QCMAP_GET_EXTD_FIREWALL_ CONFIG QMI_QCMAP_GET_FIREWALL_CONFIG_ QMI_QCMAP_GET_FIREWALL_C			- UV9
QMI_QCMAP_ENABLE_FIREWALL_ SETTING QMI_QCMAP_GET_FIREWALL_SETTING QMI_QCMAP_GET_FIREWALL_SETTING QMI_QCMAP_DISABLE_FIREWALL_ SETTING QMI_QCMAP_DISABLE_FIREWALL_ SETTING QMI_QCMAP_ADD_FIREWALL_CONFIG QMI_QCMAP_DELETE_FIREWALL_CONFIG QMI_QCMAP_DELETE_FIREWALL_CONFIG QMI_QCMAP_DELETE_FIREWALL_CONFIG QMI_QCMAP_STATION_MODE_ENABLE QMI_QCMAP_STATION_MODE_DISABLE QMI_QCMAP_STATION_MODE_DISABLE QMI_QCMAP_GET_STATION_MODE QMI_QCMAP_GET_EXTD_FIREWALL_ CONFIG QMI_QCMAP_GET_EXTD_FIREWALL_ CONFIG QMI_QCMAP_GET_FIREWALL_CONFIG_ QMI_QCMAP_GET_FIREWALL_CONFI	QMI_QCMAP_DELETE_DMZ	0x0031	
QMI_QCMAP_GET_WWAN_CONFIG       0x0033       Queries the WWAN IP configuration.         QMI_QCMAP_ENABLE_FIREWALL_ SETTING       0x0034       Enables the firewall setting.         QMI_QCMAP_GET_FIREWALL_SETTING       0x0035       Queries the firewall setting.         QMI_QCMAP_DISABLE_FIREWALL_ SETTING       0x0036       Disables the firewall setting.         QMI_QCMAP_ADD_FIREWALL_CONFIG       0x0037       Adds a firewall configuration rule.         QMI_QCMAP_DELETE_FIREWALL_CONFIG       0x0039       Deletes a firewall configuration rule.         QMI_QCMAP_GET_FIREWALL_CONFIG       0x0038       Queries the firewall configuration rules.         QMI_QCMAP_STATION_MODE_ENABLE       0x003B       Enables Station (STA) mode functionality for a mobile AP instance on the modem.         QMI_QCMAP_STATION_MODE_DISABLE       0x003C       Disables STA mode functionality for a mobile AP instance on the modem.         QMI_QCMAP_GET_STATION_MODE       0x003D       Queries the STA mode functionality for a mobile AP instance on the modem.         QMI_QCMAP_ADD_EXTD_FIREWALL_ CONFIG       0x003F       Adds IP filter-based firewall rules (extended firewall).         QMI_QCMAP_GET_EXTD_FIREWALL_ CONFIG       0x0040       Gets the firewall rules.         QMI_QCMAP_GET_FIREWALL_CONFIG_ ANDLE_LIST       0x0041       Gets the handles of all the firewall rules.			
QMI_QCMAP_ENABLE_FIREWALL_ SETTING QMI_QCMAP_GET_FIREWALL_SETTING QMI_QCMAP_DISABLE_FIREWALL_ SETTING QMI_QCMAP_DISABLE_FIREWALL_ SETTING QMI_QCMAP_ADD_FIREWALL_CONFIG QMI_QCMAP_ADD_FIREWALL_CONFIG QMI_QCMAP_DELETE_FIREWALL_CONFIG QMI_QCMAP_DELETE_FIREWALL_CONFIG QMI_QCMAP_GET_FIREWALL_CONFIG QMI_QCMAP_STATION_MODE_ENABLE QMI_QCMAP_STATION_MODE_DISABLE QMI_QCMAP_STATION_MODE_DISABLE QMI_QCMAP_GET_STATION_MODE QMI_QCMAP_GET_STATION_MODE QMI_QCMAP_GET_STATION_MODE QMI_QCMAP_GET_STATION_MODE QMI_QCMAP_GET_STATION_MODE QMI_QCMAP_GET_STATION_MODE QMI_QCMAP_GET_STATION_MODE QMI_QCMAP_GET_STATION_MODE QMI_QCMAP_ADD_EXTD_FIREWALL_ CONFIG QMI_QCMAP_GET_EXTD_FIREWALL_ CONFIG QMI_QCMAP_GET_EXTD_FIREWALL_ CONFIG QMI_QCMAP_GET_FIREWALL_CONFIG_ QMI	QMI QCMAP GET WWAN CONFIG	0x0033	Queries the WWAN IP configuration.
QMI_QCMAP_DISABLE_FIREWALL_ QMI_QCMAP_ADD_FIREWALL_CONFIG  QMI_QCMAP_DELETE_FIREWALL_CONFIG  QMI_QCMAP_DELETE_FIREWALL_CONFIG  QMI_QCMAP_GET_FIREWALL_CONFIG  QMI_QCMAP_STATION_MODE_ENABLE  QMI_QCMAP_STATION_MODE_DISABLE  QMI_QCMAP_STATION_MODE_DISABLE  QMI_QCMAP_GET_STATION_MODE  QMI_QCMAP_ADD_EXTD_FIREWALL_ CONFIG  QMI_QCMAP_GET_EXTD_FIREWALL_ CONFIG  QMI_QCMAP_GET_EXTD_FIREWALL_ CONFIG  QMI_QCMAP_GET_FIREWALL_CONFIG_ QMI_QCMAP_			
QMI_QCMAP_DISABLE_FIREWALL_ QMI_QCMAP_ADD_FIREWALL_CONFIG  QMI_QCMAP_DELETE_FIREWALL_CONFIG  QMI_QCMAP_DELETE_FIREWALL_CONFIG  QMI_QCMAP_GET_FIREWALL_CONFIG  QMI_QCMAP_STATION_MODE_ENABLE  QMI_QCMAP_STATION_MODE_DISABLE  QMI_QCMAP_STATION_MODE_DISABLE  QMI_QCMAP_GET_STATION_MODE  QMI_QCMAP_ADD_EXTD_FIREWALL_ CONFIG  QMI_QCMAP_GET_EXTD_FIREWALL_ CONFIG  QMI_QCMAP_GET_EXTD_FIREWALL_ CONFIG  QMI_QCMAP_GET_FIREWALL_CONFIG_ QMI_QCMAP_	QMI_QCMAP_ENABLE_FIREWALL_	0x0034	Enables the firewall setting.
QMI_QCMAP_DISABLE_FIREWALL_ QMI_QCMAP_ADD_FIREWALL_CONFIG QMI_QCMAP_DELETE_FIREWALL_CONFIG QMI_QCMAP_DELETE_FIREWALL_CONFIG QMI_QCMAP_GET_FIREWALL_CONFIG QMI_QCMAP_STATION_MODE_ENABLE QMI_QCMAP_STATION_MODE_DISABLE QMI_QCMAP_STATION_MODE_DISABLE QMI_QCMAP_GET_STATION_MODE QMI_QCMAP_GET_STATION_MODE QMI_QCMAP_GET_STATION_MODE QMI_QCMAP_GET_STATION_MODE QMI_QCMAP_GET_STATION_MODE QMI_QCMAP_GET_STATION_MODE QMI_QCMAP_GET_STATION_MODE QMI_QCMAP_GET_STATION_MODE QMI_QCMAP_GET_STATION_MODE QMI_QCMAP_ADD_EXTD_FIREWALL_ CONFIG QMI_QCMAP_GET_EXTD_FIREWALL_ CONFIG QMI_QCMAP_GET_EXTD_FIREWALL_ CONFIG QMI_QCMAP_GET_FIREWALL_CONFIG_ QMI_QCMAP_GET_FIREWALL_		CO/ 8/	,
QMI_QCMAP_DISABLE_FIREWALL_ QMI_QCMAP_ADD_FIREWALL_CONFIG QMI_QCMAP_DELETE_FIREWALL_CONFIG QMI_QCMAP_DELETE_FIREWALL_CONFIG QMI_QCMAP_GET_FIREWALL_CONFIG QMI_QCMAP_STATION_MODE_ENABLE QMI_QCMAP_STATION_MODE_DISABLE QMI_QCMAP_STATION_MODE_DISABLE QMI_QCMAP_GET_STATION_MODE QMI_QCMAP_GET_STATION_MODE QMI_QCMAP_GET_STATION_MODE QMI_QCMAP_GET_STATION_MODE QMI_QCMAP_GET_STATION_MODE QMI_QCMAP_GET_STATION_MODE QMI_QCMAP_GET_STATION_MODE QMI_QCMAP_GET_STATION_MODE QMI_QCMAP_GET_STATION_MODE QMI_QCMAP_ADD_EXTD_FIREWALL_ CONFIG QMI_QCMAP_GET_EXTD_FIREWALL_ CONFIG QMI_QCMAP_GET_EXTD_FIREWALL_ CONFIG QMI_QCMAP_GET_FIREWALL_CONFIG_ QMI_QCMAP_GET_FIREWALL_	QMI_QCMAP_GET_FIREWALL_SETTING	0x0035	Queries the firewall setting.
QMI_QCMAP_ADD_FIREWALL_CONFIG		( ) ( )	
QMI_QCMAP_ADD_FIREWALL_CONFIG	QMI_QCMAP_DISABLE_FIREWALL_	0x0036	Disables the firewall setting.
QMI_QCMAP_DELETE_FIREWALL_CONFIG 0x0039 Deletes a firewall configuration rule.  QMI_QCMAP_GET_FIREWALL_CONFIG 0x0038 Queries the firewall configuration rules.  QMI_QCMAP_STATION_MODE_ENABLE 0x003B Enables Station (STA) mode functionality for a mobile AP instance on the modem.  QMI_QCMAP_STATION_MODE_DISABLE 0x003C Disables STA mode functionality for a mobile AP instance on the modem.  QMI_QCMAP_GET_STATION_MODE 0x003D Queries the STA mode functionality for a mobile AP instance on the modem.  QMI_QCMAP_ADD_EXTD_FIREWALL_ 0x003F Adds IP filter-based firewall rules (extended firewall).  QMI_QCMAP_GET_EXTD_FIREWALL_ 0x0040 Gets the firewall rules.  CONFIG QMI_QCMAP_GET_FIREWALL_CONFIG_ 0x0041 Gets the handles of all the firewall rules.		- 15°	
QMI_QCMAP_DELETE_FIREWALL_CONFIG 0x0039 Deletes a firewall configuration rule.  QMI_QCMAP_GET_FIREWALL_CONFIG 0x0038 Queries the firewall configuration rules.  QMI_QCMAP_STATION_MODE_ENABLE 0x003B Enables Station (STA) mode functionality for a mobile AP instance on the modem.  QMI_QCMAP_STATION_MODE_DISABLE 0x003C Disables STA mode functionality for a mobile AP instance on the modem.  QMI_QCMAP_GET_STATION_MODE 0x003D Queries the STA mode functionality for a mobile AP instance on the modem.  QMI_QCMAP_ADD_EXTD_FIREWALL_ 0x003F Adds IP filter-based firewall rules (extended firewall).  QMI_QCMAP_GET_EXTD_FIREWALL_ 0x0040 Gets the firewall rules.  CONFIG QMI_QCMAP_GET_FIREWALL_CONFIG_ 0x0041 Gets the handles of all the firewall rules.	QMI_QCMAP_ADD_FIREWALL_CONFIG	0x0037	Adds a firewall configuration rule.
QMI_QCMAP_STATION_MODE_ENABLE  QMI_QCMAP_STATION_MODE_ENABLE  QMI_QCMAP_STATION_MODE_ENABLE  QMI_QCMAP_STATION_MODE_DISABLE  QMI_QCMAP_STATION_MODE_DISABLE  QMI_QCMAP_GET_STATION_MODE  QMI_QCMAP_ADD_EXTD_FIREWALL_ CONFIG  QMI_QCMAP_GET_EXTD_FIREWALL_ CONFIG  QMI_QCMAP_GET_FIREWALL_ CONFIG  QMI_QCMAP_GET_FIREWALL_CONFIG_ QMI_QCMAP_GET_FIREWALL_CONFIG_ CONFIG  QMI_QCMAP_GET_FIREWALL_CONFIG_ QMI_QCMAP_GET_FIREWALL_CONFIG_ CONFIG  QMI_QCMAP_GET_FIREWALL_CONFIG_ QMI_QCMAP_GET_FIREWALL_CONFIG_ CONFIG  CONFIG  CONFIG  CONFIG  CONFIG  CONFIG  CONFIG  CONFIG  CONFIG  CO	CO. 27. 24.)		
QMI_QCMAP_STATION_MODE_ENABLE  QMI_QCMAP_STATION_MODE_DISABLE  QMI_QCMAP_STATION_MODE_DISABLE  QMI_QCMAP_GET_STATION_MODE  QMI_QCMAP_ADD_EXTD_FIREWALL_ CONFIG  QMI_QCMAP_GET_EXTD_FIREWALL_ QMI_QCMAP_GET_EXTD_FIREWALL_ CONFIG  QMI_QCMAP_GET_FIREWALL_ CONFIG	QMI_QCMAP_DELETE_FIREWALL_CONFIG	0x0039	Deletes a firewall configuration rule.
QMI_QCMAP_STATION_MODE_ENABLE  QMI_QCMAP_STATION_MODE_DISABLE  QMI_QCMAP_STATION_MODE_DISABLE  QMI_QCMAP_GET_STATION_MODE  QMI_QCMAP_ADD_EXTD_FIREWALL_ CONFIG  QMI_QCMAP_GET_EXTD_FIREWALL_ QMI_QCMAP_GET_EXTD_FIREWALL_ CONFIG  QMI_QCMAP_GET_FIREWALL_ CONFIG	Al.		_
functionality for a mobile AP instance on the modem.  QMI_QCMAP_STATION_MODE_DISABLE  QMI_QCMAP_GET_STATION_MODE  QMI_QCMAP_ADD_EXTD_FIREWALL_ CONFIG  QMI_QCMAP_GET_EXTD_FIREWALL_ CONFIG  QMI_QCMAP_GET_EXTD_FIREWALL_ CONFIG  QMI_QCMAP_GET_EXTD_FIREWALL_ CONFIG  QMI_QCMAP_GET_EXTD_FIREWALL_ CONFIG  QMI_QCMAP_GET_FIREWALL_CONFIG_ QMI_QCMAP_GET_FIREWALL_CONFIG_ CONFIG  QMI_QCMAP_GET_FIREWALL_CONFIG_ CONFIG  QMI_QCMAP_GET_FIREWALL_CONFIG_ CONFIG  QMI_QCMAP_GET_FIREWALL_CONFIG_ CONFIG  QMI_QCMAP_GET_FIREWALL_CONFIG_ CONFIG  QMI_QCMAP_GET_FIREWALL_CONFIG_ CONFIG  QMI_QCMAP_GET_FIREWALL_CONFIG_ CONFIG_ CON	QMI_QCMAP_GET_FIREWALL_CONFIG	0x0038	Queries the firewall configuration rules.
functionality for a mobile AP instance on the modem.  QMI_QCMAP_STATION_MODE_DISABLE  QMI_QCMAP_GET_STATION_MODE  QMI_QCMAP_ADD_EXTD_FIREWALL_ CONFIG  QMI_QCMAP_GET_EXTD_FIREWALL_ CONFIG  QMI_QCMAP_GET_EXTD_FIREWALL_ CONFIG  QMI_QCMAP_GET_EXTD_FIREWALL_ CONFIG  QMI_QCMAP_GET_EXTD_FIREWALL_ CONFIG  QMI_QCMAP_GET_FIREWALL_CONFIG_ QMI_QCMAP_GET_FIREWALL_CONFIG_ CONFIG  QMI_QCMAP_GET_FIREWALL_CONFIG_ CONFIG  QMI_QCMAP_GET_FIREWALL_CONFIG_ CONFIG  QMI_QCMAP_GET_FIREWALL_CONFIG_ CONFIG  QMI_QCMAP_GET_FIREWALL_CONFIG_ CONFIG  QMI_QCMAP_GET_FIREWALL_CONFIG_ CONFIG  QMI_QCMAP_GET_FIREWALL_CONFIG_ CONFIG_ CON			
On the modem.  QMI_QCMAP_STATION_MODE_DISABLE  QMI_QCMAP_GET_STATION_MODE  QMI_QCMAP_ADD_EXTD_FIREWALL_ CONFIG  QMI_QCMAP_GET_EXTD_FIREWALL_ CONFIG  QMI_QCMAP_GET_EXTD_FIREWALL_ CONFIG  QMI_QCMAP_GET_EXTD_FIREWALL_ CONFIG  QMI_QCMAP_GET_FIREWALL_ CONFIG  QMI_QCMAP_GET_FIREWALL_CONFIG_ QMI_QCMAP_GET_FIREWALL_CONFIG_ QMI_QCMAP_GET_FIREWALL_CONFIG_ CONFIG  QMI_QCMAP_GET_FIREWALL_CONFIG_ QMI_QCMAP_GET_FIREWALL_CONFIG_ CONFIG  QMI_QCMAP_GET_FIREWALL_CONFIG_ QMI_QCMAP_GET_FIREWALL_CONFIG_ CONFIG  QMI_QCMAP_GET_FIREWALL_CONFIG_ CONFIG_ QMI_QCMAP_GET_FIREWALL_CONFIG_ CONFIG_ QMI_QCMAP_GET_FIREWALL_CONFIG_ CONFIG_ QMI_QCMAP_GET_FIREWALL_CONFIG_ CONFIG_ CO	QMI_QCMAP_STATION_MODE_ENABLE	0x003B	Enables Station (STA) mode
QMI_QCMAP_STATION_MODE_DISABLE0x003CDisables STA mode functionality for a mobile AP instance on the modem.QMI_QCMAP_GET_STATION_MODE0x003DQueries the STA mode functionality for a mobile AP instance on the modem.QMI_QCMAP_ADD_EXTD_FIREWALL_ CONFIG0x003FAdds IP filter-based firewall rules (extended firewall).QMI_QCMAP_GET_EXTD_FIREWALL_ CONFIG0x0040Gets the firewall rules.QMI_QCMAP_GET_FIREWALL_CONFIG_ HANDLE_LIST0x0041Gets the handles of all the firewall rules.			functionality for a mobile AP instance
QMI_QCMAP_GET_STATION_MODE0x003DQueries the STA mode functionality for a mobile AP instance on the modem.QMI_QCMAP_ADD_EXTD_FIREWALL_ CONFIG0x003FAdds IP filter-based firewall rules (extended firewall).QMI_QCMAP_GET_EXTD_FIREWALL_ CONFIG0x0040Gets the firewall rules.QMI_QCMAP_GET_FIREWALL_CONFIG_ HANDLE_LIST0x0041Gets the handles of all the firewall rules.			on the modem.
QMI_QCMAP_GET_STATION_MODE  QMI_QCMAP_ADD_EXTD_FIREWALL_ CONFIG  QMI_QCMAP_GET_EXTD_FIREWALL_ CONFIG  QMI_QCMAP_GET_EXTD_FIREWALL_ CONFIG  QMI_QCMAP_GET_FIREWALL_ CONFIG  QMI_QCMAP_GET_FIREWALL_CONFIG_ HANDLE_LIST  Qx003D  Queries the STA mode functionality for a mobile AP instance on the modem.  0x003F  Adds IP filter-based firewall rules  (extended firewall).  Gets the firewall rules.  Ox0040  Gets the handles of all the firewall rules.	QMI_QCMAP_STATION_MODE_DISABLE	0x003C	Disables STA mode functionality for a
QMI_QCMAP_ADD_EXTD_FIREWALL_ CONFIG  QMI_QCMAP_GET_EXTD_FIREWALL_ CONFIG  QMI_QCMAP_GET_FIREWALL_ CONFIG  QMI_QCMAP_GET_FIREWALL_CONFIG_ HANDLE_LIST  a mobile AP instance on the modem.  Adds IP filter-based firewall rules (extended firewall).  Gets the firewall rules.  Gets the handles of all the firewall rules.			mobile AP instance on the modem.
QMI_QCMAP_ADD_EXTD_FIREWALL_ CONFIG  QMI_QCMAP_GET_EXTD_FIREWALL_ CONFIG  QMI_QCMAP_GET_FIREWALL_CONFIG_ QMI_QCMAP_GET_FIREWALL_CONFIG_ HANDLE_LIST  Qx003F  Adds IP filter-based firewall rules (extended firewall).  Gets the firewall rules.	QMI_QCMAP_GET_STATION_MODE	0x003D	Queries the STA mode functionality for
CONFIG (extended firewall).  QMI_QCMAP_GET_EXTD_FIREWALL_ CONFIG  QMI_QCMAP_GET_FIREWALL_CONFIG_ HANDLE_LIST  (extended firewall).  Gets the firewall rules.  Gets the handles of all the firewall rules.			a mobile AP instance on the modem.
QMI_QCMAP_GET_EXTD_FIREWALL_ CONFIG  QMI_QCMAP_GET_FIREWALL_CONFIG_ HANDLE_LIST  Ox0040  Gets the firewall rules.  Gets the handles of all the firewall rules.	QMI_QCMAP_ADD_EXTD_FIREWALL_	0x003F	Adds IP filter-based firewall rules
CONFIG  QMI_QCMAP_GET_FIREWALL_CONFIG_ 0x0041 Gets the handles of all the firewall rules.  HANDLE_LIST	CONFIG		(extended firewall).
QMI_QCMAP_GET_FIREWALL_CONFIG_ 0x0041 Gets the handles of all the firewall rules. HANDLE_LIST	QMI_QCMAP_GET_EXTD_FIREWALL_	0x0040	Gets the firewall rules.
HANDLE_LIST	CONFIG		
	QMI_QCMAP_GET_FIREWALL_CONFIG_	0x0041	Gets the handles of all the firewall rules.
QMI_QCMAP_CHANGE_NAT_TYPE 0x0042 Changes the currently existing NAT	HANDLE_LIST		
	QMI_QCMAP_CHANGE_NAT_TYPE	0x0042	Changes the currently existing NAT
type.			type.
QMI_QCMAP_GET_NAT_TYPE 0x0043 Gets the currently enabled NAT type.	QMI_QCMAP_GET_NAT_TYPE	0x0043	Gets the currently enabled NAT type.

# 3.1 QMI\_QCMAP\_MOBILE\_AP\_ENABLE

Enables the mobile AP functionality via a single mobile AP instance on the modem.

**QCMAP** message ID

0x0020

**Version introduced** 

Major - 1, Minor - 0

### 3.1.1 Request - QMI\_QCMAP\_MOBILE\_AP\_ENABLE\_REQ

Message type

Request

Sender

Control point

#### **Mandatory TLVs**

Name	Version introduced	Version last modified
IP Family	1.0	1.0

Field	Field	Field	Parameter	Size	Description
	value	type	C 60 1891	(byte)	
Туре	0x01		×10,	1	IP Family
Length	4			2	
Value	$\rightarrow$	enum	ip_family	4	Determines whether mobile AP IPv4 or
					IPv6 must be enabled. Values:
					• 4 – IPv4
					• 6 – IPv6

#### **Optional TLVs**

Name	Version introduced	Version last modified
IP Address	1.0	1.0
Network Policy	1.0	1.0
SSID2 IP Address Info	1.2	1.2
NAT Type Info	1.3	1.3

Field	Field	Field	Parameter	Size	Description
	value	type		(byte)	
Туре	0x10			1	IP Address
Length	28			2	

Field	Field	Field	Parameter	Size	Description
	value	type		(byte)	
Value	$\rightarrow$	uint32	subnet_mask	4	Subnet mask.
		uint32	nat_ip_addr	4	NAT IP address.
		uint32	nat_dns_addr	4	NAT Domain Name Service (DNS)
					address.
		uint32	usb_rmnet_ip_addr	4	RmNet USB Terminal Equipment (TE)
					address.
		uint32	usb_rmnet_gateway_addr	4	RmNet USB gateway address.
		uint32	apps_rmnet_ip_addr	4	RmNet applications IP address.
		uint32	apps_rmnet_gateway_addr	4	RmNet applications gateway address.
Туре	0x11			1	Network Policy
Length	10			2	
Value	$\rightarrow$	mask	tech_pref	8	Bitmap indicating the technology
					preference. A single connection is
					attempted using the following specified
					technology preferences:
					• Bit 0 – 3GPP
					• Bit 1 – 3GPP2
				,	All other bits are reserved and ignored
				Jilo.	even if they are set in the request. If a
				. of 170.	single value of the technology preference
			4	0,1	bitmask is set, the device attempts to use
			Ma	J: , o	that technology. If two or more bits in
			200	and in	the technology preference bitmask are
			The state of the s	Star	set, the device determines the technology
		1	. S. C. C.		to be used from those specified.
		uint8	profile_id_3gpp2	1	CDMA profile ID.
		uint8	profile_id_3gpp	1	UMTS profile ID.
Туре	0x12		×100	1	SSID2 IP Address Info
Length	8		ì	2	
Value	$\rightarrow$	uint32	addr	4	IPv4 address as specified in the IPv4
					protocol specification (RFC 791 [S2]).
		uint32	subnet_mask	4	IPv4 subnet mask as specified in the IPv4
					protocol specification (RFC 791 [S2]).
Туре	0x13			1	NAT Type Info
Length	4			2	
Value	$\rightarrow$	enum	qcmap_nat_type_info	4	NAT type specified during mobile AP
					enable. Values:
					• 0x00 – QCMAP_NAT_TYPE_
					SYMMETRIC – Symmetric NAT
					• 0x01 – QCMAP_NAT_TYPE_PORT_
					RESTRICTED_CONE – Port restricted
					cone NAT

### 3.1.2 Response - QMI\_QCMAP\_MOBILE\_AP\_ENABLE\_RESP

Message	type
---------	------

Response

#### Sender

Service

#### **Mandatory TLVs**

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

#### **Optional TLVs**

Name	Version introduced	Version last modified
Mobile AP Handle	1.0	1.0

Field	Field	Field	Parameter	Size	Description
	value	type		(byte)	_
Туре	0x10			oT of	Mobile AP Handle
Length	4		34	2	
Value	$\rightarrow$	uint32	mobile_ap_handle	(4.0	Handle identifying the mobile AP call
			3, 50,	(0)	instance.
			and the second	Star	The mobile AP handle must be retained
			4,0° 6,0°		by the control point and specified in all
					mobile AP-specific QCMAP messages.
			20, 993		For example, QMI_QCMAP_DISABLE,
			4100		QMI_QCMAP_BRING_UP_WWAN,
					etc.

#### **Error codes**

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_MALFORMED_MSG	Message was not formulated correctly by the control point
	or the message was corrupted during transmission
QMI_ERR_MISSING_ARG	Some TLV was missing
QMI_ERR_NOT_SUPPORTED	Operation is not supported
QMI_ERR_NO_EFFECT	Mobile AP instance is already enabled

### 3.1.3 Description of QMI\_QCMAP\_MOBILE\_AP\_ENABLE REQ/RESP

This command enables the mobile AP functionality at the modem. The control point passes the network policy that is used to bring up the WWAN when QMI\_QCMAP\_BRING\_UP\_WWAN is called. After QMI\_QCMAP\_MOBILE\_AP\_ENABLE is successfully processed, any subsequent RmNet call using the same network policy is brought up in the Mobile AP mode. If the IP family is QCMAP\_IP\_V4, the control point must fill in the optional IP Address TLV.

The control point is expected to store the mobile AP handle that is returned and to pass it in all mobile AP-specific messages.

The Network Policy TLV provides the network policy that is used by the mobile AP to select the WWAN network. If this value is not specified, the default WWAN network is selected.

The IP Address TLV is required when the mobile AP IPv4 is enabled. The value is ignored when the mobile AP IPv6 is enabled. If the TLV is not specified when enabling the mobile AP IPv4, a QMI\_ERR\_MISSING\_ARG error is returned.

The SSID2 IP Address Info TLV is required when the mobile AP IPv4 Service Set Identifier 2 (SSID2) is enabled. The value is ignored when the mobile AP IPv6 is enabled. If this TLV is not specified when enabling the mobile AP IPv4, it is assumed that SSID2 is not enabled.

The mobile AP instance enabled by this command remains enabled until the control point or client issues a QMI\_QCMAP\_MOBILE\_AP\_DISABLE\_REQ request or until the control point disassociates from the service.

#### QMI\_QCMAP\_MOBILE\_AP\_DISABLE 3.2

Disables the mobile AP functionality for a mobile AP instance on the modem.

**QCMAP** message ID

0x0021

**Version introduced** 

Major - 1, Minor - 0

#### Request - QMI\_QCMAP\_MOBILE\_AP\_DISABLE\_REQ 3.2.1

Message type

Request

Sender

Control point

#### **Mandatory TLVs**

Name	Version introduced	Version last modified
Mobile AP Handle	1.0	1.0

Field	Field	Field	Parameter	Size	Description
	value	type	600,00	(byte)	
Туре	0x01		440	1	Mobile AP Handle
Length	4			2	
Value	$\rightarrow$	uint32	mobile_ap_handle	4	Handle identifying the mobile AP call
				instance.	
				The value must be the handle previous	
					returned by QMI_QCMAP_MOBILE_
					AP_ENABLE_REQ.

#### **Optional TLVs**

None

#### 3.2.2 Response - QMI QCMAP DISABLE RESP

Message type

Response

Sender

Service

#### **Mandatory TLVs**

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

#### **Optional TLVs**

#### **Error codes**

Optional TLVs	
None	* dX <sup>c</sup>
Error codes	de Sec.
QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_MALFORMED_MSG	Message was not formulated correctly by the control point
	or the message was corrupted during transmission
QMI_ERR_MISSING_ARG	Some TLV was missing
QMI_ERR_INVALID_HANDLE	Mobile AP handle provided in the request is not valid, i.e., it
	is not assigned to the control point
QMI_ERR_NO_EFFECT	WWAN is connected or in a transient state; the control point
Office	must terminate the WWAN connection using
	QMI_QCMAP_TEAR_DOWN_WWAN_REQ and wait for
54	the final WWAN status before disabling the mobile AP

#### Description of QMI\_QCMAP\_MOBILE\_AP\_DISABLE REQ/RESP 3.2.3

This command disables the mobile AP functionality at the modem for a single mobile AP instance. After the request is successfully processed, the ongoing RmNet and WWAN calls (if any) are torn down and subsequent RmNet calls are brought up in the non-Mobile AP mode. If the RmNet call is up in the Mobile AP mode at the time this command is sent, the control point considers that the packet data connection state is unchanged until notified of a state change via QMI\_WDS\_PKT\_SRVC\_STATUS\_IND (refer to Q3) for the RmNet session. If the WWAN call is active, the mobile AP is not disabled and a QMI ERR NO EFFECT error is returned.

The mobile AP instance associated with the control point can be disabled using either this command or when the control point disconnects from the QMI\_QCMAP service. Qualcomm recommends that the client disable the mobile AP instances specifically using this command and then proceed by disconnecting from the service.

All NAT-specific functionalities associated with this mobile AP instance are disabled when the command is used or when the control point disassociates from the QMI QCMAP service. The control point must reactivate or set functionalities such as the DMZ, VPN passthrough, static NAT, and the firewall after enabling the mobile AP again.

### 3.3 QMI\_QCMAP\_BRING\_UP\_WWAN

Invokes bringing up the WWAN from the mobile AP.

**QCMAP** message ID

0x0022

**Version introduced** 

Major - 1, Minor - 0

### 3.3.1 Request - QMI\_QCMAP\_BRING\_UP\_WWAN\_REQ

Message type

Request

Sender

Control point

#### **Mandatory TLVs**

Name	Version introduced	Version last modified
Mobile AP Handle	1.0	1.0

Field	Field	Field	Parameter	Size	Description
	value	type	600,00	(byte)	
Туре	0x01		4400	1	Mobile AP Handle
Length	4			2	
Value	$\rightarrow$	uint32	mobile_ap_handle	4	Handle identifying the mobile AP call
					instance.
				The value must be the handle previous	
					returned by QMI_QCMAP_MOBILE_
					AP_ENABLE_REQ.

#### **Optional TLVs**

None

### Response - QMI QCMAP BRING UP WWAN RESP

Message type

Response

Sender

Service

#### **Mandatory TLVs**

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

#### **Optional TLVs**

#### **Error codes**

Optional TLVs	
None	A Section 1
Error codes	alesect
QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_MALFORMED_MSG	Message was not formulated correctly by the control point
	or the message was corrupted during transmission
QMI_ERR_MISSING_ARG	Some TLV was missing
QMI_ERR_INVALID_HANDLE	Mobile AP handle provided in the request is not valid, i.e., it
	is not assigned to the control point
QMI_ERR_NO_EFFECT	WWAN is already up or a previous request is still in process
COLLIN	(WWAN is connecting)

#### Description of QMI QCMAP BRING UP WWAN REQ/RESP 3.3.3

This command brings up the WWAN connection. The call is established using the stored network policy that enabled the mobile AP via QMI\_QCMAP\_MOBILE\_AP\_ENABLE\_REQ.

If the response returned is SUCCESS, the corresponding QMI QCMAP BRING UP WWAN IND indication determines that the request has been completely processed by the modem.

The WWAN status can be queried using QMI\_QCMAP\_GET\_WWAN\_STATUS or sent as an indication for registered clients. See QMI\_QCMAP\_WWAN\_STATUS\_IND\_REG (Section 3.6) for information on registration.

If the control point issues multiple requests in short intervals, a QMI ERR NO EFFECT error is returned indicating that the previous request is still in process.

### 3.3.4 Indication - QMI QCMAP BRING UP WWAN IND

Message type

Indication

Sender

Service

Indication scope

Unicast

#### **Mandatory TLVs**

Name	Version introduced	Version last modified
Mobile AP Handle	1.0	1.0
IP Family	1.0	1.0

Field	Field	Field	Parameter	Size	Description
	value	type		(byte)	) ·
Туре	0x01		24	~.(1)	Mobile AP Handle
Length	4		Mic	20	
Value	$\rightarrow$	uint32	mobile_ap_handle	4	Handle identifying the mobile AP call
			10 m	Stan	instance.
Туре	0x02		STO ST. CO	1	IP Family
Length	4		(2) (2)	2	
Value	$\rightarrow$	enum	ip_family	4	Determines whether the mobile AP is
			410		IPv4 or IPv6. Values:
					• 4 – IPv4
					• 6 – IPv6

#### **Optional TLVs**

None

### 3.3.5 Description of QMI\_QCMAP\_BRING\_UP\_WWAN\_IND

This indication communicates the completion of processing a QMI\_QCMAP\_BRING\_UP\_WWAN\_REQ received from the control point. If the client registered for the QMI\_QCMAP\_WWAN\_STATUS\_IND indication, it receives the corresponding event indication that reports the WWAN status. Alternatively, the control point can issue QMI\_QCMAP\_GET\_WWAN\_STATUS\_REQ to query the current WWAN status.

### 3.4 QMI\_QCMAP\_TEAR\_DOWN\_WWAN

Tears down the WWAN.

**QCMAP** message ID

0x0023

**Version introduced** 

Major - 1, Minor - 0

# 3.4.1 Request - QMI\_QCMAP\_TEAR\_DOWN\_WWAN\_REQ

Message type

Request

Sender

Control point

#### **Mandatory TLVs**

Name	Version introduced	Version last modified
Mobile AP Handle	1.0	1.0

Field	Field	Field	Parameter	Size	Description	
	value	type	C 05 793	(byte)		
Туре	0x01		4400	1	Mobile AP Handle	
Length	4			2		
Value	$\rightarrow$	uint32	mobile_ap_handle	4	Handle identifying the mobile AP call	
					instance.	
				The value must be the handle previou		
					returned by QMI_QCMAP_MOBILE_	
					AP_ENABLE_REQ.	

#### **Optional TLVs**

None

### Response - QMI QCMAP TEAR DOWN WWAN RESP

Message	type
---------	------

Response

Sender

Service

#### **Mandatory TLVs**

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

#### **Optional TLVs**

#### **Error codes**

Optional TLVs	
None	, a <sup>co</sup>
Error codes	desect
QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_MALFORMED_MSG	Message was not formulated correctly by the control point
	or the message was corrupted during transmission
QMI_ERR_MISSING_ARG	Some TLV was missing
QMI_ERR_INVALID_HANDLE	Mobile AP handle provided in the request is not valid, i.e., it
	is not assigned to the control point
QMI_ERR_NO_EFFECT	WWAN is already down or a previous request is still in
COLLIN	process (WWAN is disconnecting)

#### Description of QMI\_QCMAP\_TEAR\_DOWN\_WWAN REQ/RESP 3.4.3

This command tears down the mobile AP WWAN interface that was brought up via QMI\_QCMAP\_BRING\_UP\_WWAN.

If the response returned is SUCCESS, the corresponding QMI QCMAP TEAR DOWN WWAN IND indication determines that the request has been completely processed by the modem.

The WWAN status can be queried using QMI\_QCMAP\_GET\_WWAN\_STATUS or sent as an indication for registered clients. See QMI\_QCMAP\_WWAN\_STATUS\_IND\_REG (Section 3.6) for information on registration.

If the control point issues multiple requests in short intervals, a QMI ERR NO EFFECT error is returned indicating that the previous request is still in process.

#### 3.4.4 Indication - QMI QCMAP TEAR DOWN WWAN IND

Message type

Indication

Sender

Service

Indication scope

Unicast

#### **Mandatory TLVs**

Name	Version introduced	Version last modified
Mobile AP Handle	1.0	1.0
IP Family	1.0	1.0

Field	Field	Field	Parameter	Size	Description
	value	type		(byte)	D`
Туре	0x01		4	~1\)	Mobile AP Handle
Length	4		Mic	(2.0)	
Value	$\rightarrow$	uint32	mobile_ap_handle	4	Handle identifying the mobile AP call
			1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	Stan	instance.
Туре	0x02		STO ST. CO	1	IP Family
Length	4		(2) 12/12	2	
Value	$\rightarrow$	enum	ip_family	4	Determines whether the mobile AP is
			410		IPv4 or IPv6. Values:
					• 4 – IPv4
					• 6 – IPv6

#### **Optional TLVs**

None

### 3.4.5 Description of QMI\_QCMAP\_TEAR\_DOWN\_WWAN\_IND

This indication communicates the completion of processing a

QMI\_QCMAP\_TEAR\_DOWN\_WWAN\_REQ received from the control point. If the client registered for the QMI\_QCMAP\_WWAN\_STATUS\_IND indication, it receives the corresponding event indication that reports the WWAN status. Alternatively, the control point can issue

QMI\_QCMAP\_GET\_WWAN\_STATUS\_REQ to query the current WWAN status.

### 3.5 QMI\_QCMAP\_GET\_WWAN\_STATUS

Queries the current WWAN status.

**QCMAP** message ID

0x0024

**Version introduced** 

Major - 1, Minor - 0

### 3.5.1 Request - QMI\_QCMAP\_GET\_WWAN\_STATUS\_REQ

Message type

Request

Sender

Control point

#### **Mandatory TLVs**

Name	Version introduced	Version last modified
Mobile AP Handle	1.0	1.0

Field	Field	Field	Parameter	Size	Description
	value	type	C 05 793	(byte)	
Туре	0x01		4400	1	Mobile AP Handle
Length	4			2	
Value	$\rightarrow$	uint32	mobile_ap_handle	4	Handle identifying the mobile AP call
					instance.
					The value must be the handle previously
					returned by QMI_QCMAP_MOBILE_
					AP_ENABLE_REQ.

#### **Optional TLVs**

None

### Response - QMI\_QCMAP\_GET\_WWAN\_STATUS\_RESP

#### Message type

Response

#### Sender

Service

#### **Mandatory TLVs**

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

#### **Optional TLVs**

Name	Version introduced	Version last modified
Call End Reason	1.0	1.0
Verbose Call End Reason	1.0	1.0
Packet Service Status	1.0	1.0

Field	Field	Field	Parameter	Size	Description
	value	type	33	(byte)	
Туре	0x10		Me	\$\frac{1}{2}	Call End Reason
Length	4		18 20	2	
Value	$\rightarrow$	enum	call_end_reason	4	Reason the call ended; see Table A-1 for
			Silo College		the definition of these values.
Type	0x11		(2) 27 4	1	Verbose Call End Reason
Length	4		30,90,	2	
Value	$\rightarrow$	enum	verbose_call_end_reason	4	Reason the call ended (verbose); see
					Table A-2 for the definition of these
					values.
Type	0x12			1	Packet Service Status
Length	4			2	
Value	$\rightarrow$	enum	wwan_status	4	If the response is QMI_ERR_NONE,
					this indicates the WWAN status. Values:
					• 1 – Connecting
					• 2 – Connected
					• 3 – Disconnecting
					• 4 – Disconnected

#### **Error codes**

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_MALFORMED_MSG	Message was not formulated correctly by the control point
	or the message was corrupted during transmission
QMI_ERR_INVALID_HANDLE	Mobile AP handle provided in the request is not valid, i.e., it
	is not assigned to the control point
QMI_ERR_NO_MEMORY	Device could not allocate memory to formulate a response
QMI_ERR_MISSING_ARG	One or more mandatory TLVs are missing

### 3.5.3 Description of QMI\_QCMAP\_GET\_WWAN\_STATUS REQ/RESP

This command queries the state of the WWAN instantaneously corresponding to the mobile AP handle. The WWAN state could have changed for the following reasons:

- The WWAN state was earlier changed via QMI\_QCMAP\_BRING\_UP\_WWAN or QMI\_QCMAP\_TEAR\_DOWN\_WWAN
- If the network-initiated call status changes

### 3.6 QMI\_QCMAP\_WWAN\_STATUS\_IND\_REG

Registers/deregisters the control point to receive QMI\_QCMAP\_WWAN\_STATUS\_IND.

**QCMAP** message ID

0x003A

**Version introduced** 

Major - 1, Minor - 0

### 3.6.1 Request - QMI\_QCMAP\_WWAN\_STATUS\_IND\_REG\_REQ

Message type

Request

Sender

Control point

#### **Mandatory TLVs**

	Name	Pla	Version	nintroduced	Version last modified
Mobile AP Handle		30 05	5:03:	1.0	1.0
Register Indication	1 4	- Ch 65	25/111	1.0	1.0

Field	Field	Field	Parameter	Size	Description
	value	type	8400	(byte)	
Туре	0x01			1	Mobile AP Handle
Length	4			2	
Value	$\rightarrow$	uint32	mobile_ap_handle	4	Handle identifying the mobile AP call
					instance.
					The value must be the handle previously
					returned by QMI_QCMAP_MOBILE_
					AP_ENABLE_REQ.
Туре	0x02			1	Register Indication
Length	1			2	
Value	$\rightarrow$	boolean	register_indication	1	Specifies the registration. Values:
					• 0 – Do not register or deregister if
					already registered
					• 1 – Register for the indication; ignore if
					already registered

#### **Optional TLVs**

None

### 3.6.2 Response - QMI\_QCMAP\_WWAN\_STATUS\_IND\_REG\_RESP

#### Message type

Response

#### Sender

Service

#### **Mandatory TLVs**

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

#### **Optional TLVs**

None

#### **Error codes**

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_MALFORMED_MSG	Message was not formulated correctly by the control point
1613	or the message was corrupted during transmission
QMI_ERR_INVALID_HANDLE	Mobile AP handle provided in the request is not valid, i.e., it
	is not assigned to the control point
QMI_ERR_NO_MEMORY	Device could not allocate memory to formulate a response
QMI_ERR_MISSING_ARG	One or more of mandatory TLVs are missing

ortal Trade

# 3.6.3 Description of QMI\_QCMAP\_WWAN\_STATUS\_IND\_REG REQ/RESP

This command registers/deregisters the control point to receive the QMI\_QCMAP\_WWAN\_STATUS\_IND indication.

### 3.7 QMI\_QCMAP\_WWAN\_STATUS\_IND

Indicates a change in the current mobile AP WWAN connection status.

**QCMAP** message ID

0x003E

**Version introduced** 

Major - 1, Minor - 0

### 3.7.1 Indication - QMI\_QCMAP\_WWAN\_STATUS\_IND

Message type

Indication

Sender

Service

Indication scope

Unicast

**Mandatory TLVs** 

Name	Version introduced	Version last modified
Mobile AP Handle	1.0	1.0
IP Family	1.0	1.0
Packet Service Status	1.0	1.0
Reconfiguration Required	1.0	1.0

Field	Field	Field	Parameter	Size	Description
	value	type		(byte)	
Туре	0x01			1	Mobile AP Handle
Length	4			2	
Value	$\rightarrow$	uint32	mobile_ap_handle	4	Handle identifying the mobile AP call
					instance.
Туре	0x02			1	IP Family
Length	4			2	
Value	$\rightarrow$	enum	ip_family	4	Determines whether the mobile AP is
					IPv4 or IPv6. Value:
					• 4 – IPv4
					• 6 – IPv6
Туре	0x03			1	Packet Service Status
Length	4			2	

Field	Field	Field	Parameter	Size	Description
	value	type		(byte)	
Value	$\rightarrow$	enum	wwan_status	4	Indicates the WWAN status. Values:
					• 1 – Connecting
					• 2 – Connected
					• 3 – Disconnecting
					• 4 – Disconnected
Туре	0x04			1	Reconfiguration Required
Length	1			2	
Value	$\rightarrow$	uint8	reconfig_required	1	Indicates whether the IP reconfiguration
					is required by the control point.

#### **Optional TLVs**

Name	Version introduced	Version last modified
Call End Reason	1.0	1.0
Verbose Call End Reason	1.0	1.0

Field	Field	Field	Parameter	Size	Description
	value	type		(byte)	<u>`</u>
Туре	0x10			COK	Call End Reason
Length	4		Ma	2	
Value	$\rightarrow$	enum	call_end_reason	4	Reason the call ended; see Table A-1 for
			Zia Z	YES?	the definition of these values.
Туре	0x11		180 7 1 0°	1	Verbose Call End Reason
Length	4		20,111	2	
Value	$\rightarrow$	enum	verbose_call_end_reason	4	Reason the call ended (verbose); see
			1,00		Table A-2 for the definition of these
					values.

### 3.7.2 Description of QMI\_QCMAP\_WWAN\_STATUS\_IND

This indication communicates changes in the WWAN state.

The indication is also sent when the WWAN technology changes after a handoff is performed on the modem. The Reconfiguration Required TLV value will be set to indicate that an IP address reconfiguration is required by the control point.

If the indication is sent due to a WWAN Down state, the optional Call End Reason TLV and optional Verbose Call End Reason TLV are included and will contain the reason the call was terminated. These reasons include network and user-generated reasons. See Table A-1 for the call end reasons. See Table A-2 for the verbose call end reasons.

The Call End Reason TLV has been kept for backward compatibility. All new QMI clients must use the Verbose Call End Reason TLV. Any new call end reason will be added to the Verbose Call End Reason TLV.

### 3.8 QMI\_QCMAP\_SET\_IPSEC\_VPN\_PASS\_THROUGH

Configures the Internet Protocol security (IPSec) Virtual Private Network (VPN) passthrough setting.

**QCMAP** message ID

0x0026

**Version introduced** 

Major - 1, Minor - 0

### 3.8.1 Request - QMI\_QCMAP\_SET\_IPSEC\_VPN\_PASS\_THROUGH\_REQ

Message type

Request

Sender

Control point

#### **Mandatory TLVs**

Name	Version introduced	Version last modified
Mobile AP Handle	1.0	1.0
VPN Passthrough Value	1.0	1.0

Field	Field	Field	Parameter	Size	Description
	value	type	*100	(byte)	
Туре	0x01			1	Mobile AP Handle
Length	4			2	
Value	$\rightarrow$	uint32	mobile_ap_handle	4	Handle identifying the mobile AP call
					instance.
					The value must be the handle previously
					returned by QMI_QCMAP_MOBILE_
					AP_ENABLE_REQ.
Туре	0x02			1	VPN Passthrough Value
Length	1			2	
Value	$\rightarrow$	boolean	vpn_pass_through_value	1	Indicates whether an IPSec VPN
					passthrough is allowed; boolean value.

#### **Optional TLVs**

None

# 3.8.2 Response - QMI\_QCMAP\_SET\_IPSEC\_VPN\_PASS\_THROUGH\_RESP

Message type

Response

Sender

Service

**Mandatory TLVs** 

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

**Optional TLVs** 

None

#### **Error codes**

QMI_ERR_NONE	No error in the request	
QMI_ERR_INTERNAL	Unexpected error occurred during processing	
QMI_ERR_MALFORMED_MSG	Message was not formulated correctly by the control point	
CO. V	or the message was corrupted during transmission	
QMI_ERR_MISSING_ARG	Some TLV was missing	
QMI_ERR_INVALID_HANDLE	Mobile AP handle provided in the request is not valid, i.e., it	
	is not assigned to the control point	
QMI_ERR_INVALID_ARG	Argument is not correct	
QMI_ERR_NOT_SUPPORTED	Operation is not supported	

# 3.8.3 Description of QMI\_QCMAP\_SET\_IPSEC\_VPN\_PASS\_THROUGH REQ/RESP

This command sets the IPSec VPN passthrough on the device. The command handler overwrites any previously configured value with the current value.

### 3.9 QMI\_QCMAP\_GET\_IPSEC\_VPN\_PASS\_THROUGH

Queries the IPSec VPN passthrough setting.

**QCMAP** message ID

0x0025

Version introduced

Major - 1, Minor - 0

### 3.9.1 Request - QMI\_QCMAP\_GET\_IPSEC\_VPN\_PASS\_THROUGH\_REQ

Message type

Request

Sender

Control point

#### **Mandatory TLVs**

Name	Version introduced	Version last modified
Mobile AP Handle	1.0	1.0

Field	Field	Field	Parameter	Size	Description
	value	type	C 05 793	(byte)	
Туре	0x01		4400	1	Mobile AP Handle
Length	4			2	
Value	$\rightarrow$	uint32	mobile_ap_handle	4	Handle identifying the mobile AP call
					instance.
					The value must be the handle previously
					returned by QMI_QCMAP_MOBILE_
					AP_ENABLE_REQ.

#### **Optional TLVs**

None

# 3.9.2 Response - QMI\_QCMAP\_GET\_IPSEC\_VPN\_PASS\_THROUGH\_RESP

Message type

Response

Sender

Service

#### **Mandatory TLVs**

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

#### **Optional TLVs**

Name	Version introduced	Version last modified
VPN Passthrough Value	1.0	1.0

Field	Field	Field	Parameter	Size	Description
	value	type		(byte)	
Туре	0x10		100	$\sqrt{3}$	VPN Passthrough Value
Length	1		, ia.	2	
Value	$\rightarrow$	boolean	vpn_pass_through_value	CP	Indicates whether an IPSec VPN
			Val. Val.	0	passthrough is allowed; boolean value.

#### **Error codes**

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_MALFORMED_MSG	Message was not formulated correctly by the control point
	or the message was corrupted during transmission
QMI_ERR_MISSING_ARG	Some TLV was missing
QMI_ERR_INVALID_HANDLE	Mobile AP handle provided in the request is not valid, i.e., it
	is not assigned to the control point
QMI_ERR_NOT_SUPPORTED	Operation is not supported

# 3.9.3 Description of QMI\_QCMAP\_GET\_IPSEC\_VPN\_PASS\_THROUGH REQ/RESP

This command queries the IPSec VPN passthrough value on the device.

## 3.10 QMI\_QCMAP\_SET\_PPTP\_VPN\_PASS\_THROUGH

Configures the Point-to-Point Tunneling Protocol (PPTP) VPN passthrough setting.

**QCMAP** message ID

0x0028

Version introduced

Major - 1, Minor - 0

## 3.10.1 Request - QMI\_QCMAP\_SET\_PPTP\_VPN\_PASS\_THROUGH\_REQ

Message type

Request

Sender

Control point

#### **Mandatory TLVs**

Name	Version introduced	Version last modified
Mobile AP Handle	1.0	1.0
VPN Passthrough Value	1.0	1.0

Field	Field	Field	Parameter	Size	Description
	value	type	4400	(byte)	
Туре	0x01			1	Mobile AP Handle
Length	4			2	
Value	$\rightarrow$	uint32	mobile_ap_handle	4	Handle identifying the mobile AP call
					instance.
					The value must be the handle previously
					returned by QMI_QCMAP_MOBILE_
					AP_ENABLE_REQ.
Туре	0x02			1	VPN Passthrough Value
Length	1			2	
Value	$\rightarrow$	boolean	vpn_pass_through_value	1	Indicates whether an IPSec VPN
					passthrough is allowed; boolean value.

#### **Optional TLVs**

None

## 3.10.2 Response - QMI\_QCMAP\_SET\_PPTP\_VPN\_PASS\_THROUGH\_-RESP

Message type

Response

Sender

Service

#### **Mandatory TLVs**

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

#### **Optional TLVs**

None

#### **Error codes**

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_MALFORMED_MSG	Message was not formulated correctly by the control point
CON 1	or the message was corrupted during transmission
QMI_ERR_MISSING_ARG	Some TLV was missing
QMI_ERR_INVALID_HANDLE	Mobile AP handle provided in the request is not valid, i.e., it
	is not assigned to the control point
QMI_ERR_INVALID_ARG	Argument is not correct
QMI_ERR_NOT_SUPPORTED	Operation is not supported

# 3.10.3 Description of QMI\_QCMAP\_SET\_PPTP\_VPN\_PASS\_THROUGH REQ/RESP

This command sets the PPTP VPN passthrough on the device. The command handler overwrites any previously configured value with the current value.

# 3.11 QMI\_QCMAP\_GET\_PPTP\_VPN\_PASS\_THROUGH

Queries the PPTP VPN passthrough setting.

**QCMAP** message ID

0x0027

Version introduced

Major - 1, Minor - 0

## 3.11.1 Request - QMI\_QCMAP\_GET\_PPTP\_VPN\_PASS\_THROUGH\_REQ

Message type

Request

Sender

Control point

#### **Mandatory TLVs**

Name	Version introduced	Version last modified
Mobile AP Handle	1.0	1.0

Field	Field	Field	Parameter	Size	Description
	value	type	C 05 793	(byte)	
Туре	0x01		4400	1	Mobile AP Handle
Length	4			2	
Value	$\rightarrow$	uint32	mobile_ap_handle	4	Handle identifying the mobile AP call
					instance.
					The value must be the handle previously
					returned by QMI_QCMAP_MOBILE_
					AP_ENABLE_REQ.

#### **Optional TLVs**

## 3.11.2 Response - QMI\_QCMAP\_GET\_PPTP\_VPN\_PASS\_THROUGH\_-RESP

Message type

Response

Sender

Service

#### **Mandatory TLVs**

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

#### **Optional TLVs**

Name	Version introduced	Version last modified
VPN Passthrough Value	1.0	1.0

Field	Field	Field	Parameter	Size	Description
	value	type		(byte)	
Туре	0x10		100	$\sqrt{3}$	VPN Passthrough Value
Length	1		, ia.	2	
Value	$\rightarrow$	boolean	vpn_pass_through_value	CP	Indicates whether an IPSec VPN
			Val. Val.	0	passthrough is allowed; boolean value.

#### **Error codes**

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_MALFORMED_MSG	Message was not formulated correctly by the control point
	or the message was corrupted during transmission
QMI_ERR_MISSING_ARG	Some TLV was missing
QMI_ERR_INVALID_HANDLE	Mobile AP handle provided in the request is not valid, i.e., it
	is not assigned to the control point
QMI_ERR_NOT_SUPPORTED	Operation is not supported

# 3.11.3 Description of QMI\_QCMAP\_GET\_PPTP\_VPN\_PASS\_THROUGH REQ/RESP

This command queries the PPTP VPN passthrough value on the device.

## 3.12 QMI\_QCMAP\_SET\_L2TP\_VPN\_PASS\_THROUGH

Configures the Layer 2 Tunneling Protocol (L2TP) VPN passthrough setting.

**QCMAP** message ID

0x002A

**Version introduced** 

Major - 1, Minor - 0

## 3.12.1 Request - QMI\_QCMAP\_SET\_L2TP\_VPN\_PASS\_THROUGH\_REQ

Message type

Request

Sender

Control point

#### **Mandatory TLVs**

Name	Version introduced	Version last modified
Mobile AP Handle	1.0	1.0
VPN Passthrough Value	1.0	1.0

Field	Field	Field	Parameter	Size	Description
	value	type	*100	(byte)	
Туре	0x01			1	Mobile AP Handle
Length	4			2	
Value	$\rightarrow$	uint32	mobile_ap_handle	4	Handle identifying the mobile AP call
					instance.
					The value must be the handle previously
					returned by QMI_QCMAP_MOBILE_
					AP_ENABLE_REQ.
Туре	0x02			1	VPN Passthrough Value
Length	1			2	
Value	$\rightarrow$	boolean	vpn_pass_through_value	1	Indicates whether an IPSec VPN
					passthrough is allowed; boolean value.

#### **Optional TLVs**

None

## 3.12.2 Response - QMI\_QCMAP\_SET\_L2TP\_VPN\_PASS\_THROUGH\_-RESP

Message type

Response

Sender

Service

**Mandatory TLVs** 

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

**Optional TLVs** 

None

#### **Error codes**

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_MALFORMED_MSG	Message was not formulated correctly by the control point
CONT.	or the message was corrupted during transmission
QMI_ERR_MISSING_ARG	Some TLV was missing
QMI_ERR_INVALID_HANDLE	Mobile AP handle provided in the request is not valid, i.e., it
	is not assigned to the control point
QMI_ERR_INVALID_ARG	Argument is not correct
QMI_ERR_NOT_SUPPORTED	Operation is not supported

# 3.12.3 Description of QMI\_QCMAP\_SET\_L2TP\_VPN\_PASS\_THROUGH REQ/RESP

This command sets the L2TP VPN passthrough on the device. The command handler overwrites any previously configured value with the current value.

## 3.13 QMI\_QCMAP\_GET\_L2TP\_VPN\_PASS\_THROUGH

Queries the L2TP VPN passthrough setting.

**QCMAP** message ID

0x0029

**Version introduced** 

Major - 1, Minor - 0

## 3.13.1 Request - QMI\_QCMAP\_GET\_L2TP\_VPN\_PASS\_THROUGH\_REQ

Message type

Request

Sender

Control point

#### **Mandatory TLVs**

Name	Version introduced	Version last modified
Mobile AP Handle	1.0	1.0

Field	Field	Field	Parameter	Size	Description
	value	type	C 05 793	(byte)	
Туре	0x01		4400	1	Mobile AP Handle
Length	4			2	
Value	$\rightarrow$	uint32	mobile_ap_handle	4	Handle identifying the mobile AP call
					instance.
					The value must be the handle previously
					returned by QMI_QCMAP_MOBILE_
					AP_ENABLE_REQ.

#### **Optional TLVs**

## 3.13.2 Response - QMI\_QCMAP\_GET\_L2TP\_VPN\_PASS\_THROUGH\_-**RESP**

Message type

Response

Sender

Service

#### **Mandatory TLVs**

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

#### **Optional TLVs**

Name	Version introduced	Version last modified
VPN Passthrough Value	1.0	1.0

Field	Field	Field	Parameter	Size	Description
	value	type		(byte)	
Туре	0x10			$\sqrt{3}$	VPN Passthrough Value
Length	1		1 1 2	2	
Value	$\rightarrow$	boolean	vpn_pass_through_value	(P	Indicates whether an IPSec VPN
			Val. Val.	0	passthrough is allowed; boolean value.

#### **Error codes**

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_MALFORMED_MSG	Message was not formulated correctly by the control point
	or the message was corrupted during transmission
QMI_ERR_MISSING_ARG	Some TLV was missing
QMI_ERR_INVALID_HANDLE	Mobile AP handle provided in the request is not valid, i.e., it
	is not assigned to the control point
QMI_ERR_NOT_SUPPORTED	Operation is not supported

## 3.13.3 Description of QMI\_QCMAP\_GET\_L2TP\_VPN\_PASS\_THROUGH **REQ/RESP**

This command queries the L2TP VPN passthrough value on the device.

#### QMI\_QCMAP\_SET\_DYNAMIC\_NAT\_ENTRY\_TIMEOUT 3.14

Sets the Network Address Translation (NAT) entry timeout.

**QCMAP** message ID

0x002C

**Version introduced** 

Major - 1, Minor - 0

#### Request - QMI\_QCMAP\_SET\_DYNAMIC\_NAT\_ENTRY\_TIMEOUT\_-3.14.1 **REQ**

Message type

Request

Sender

Control point

#### **Mandatory TLVs**

	Name	10 D	Version introduced	Version last modified
Mobile AP Handle	<b>V</b> X	10 No 18	1.0	1.0
Timeout		110 05 110°	1.0	1.0

Field	Field	Field	Parameter	Size	Description
	value	type		(byte)	
Туре	0x01			1	Mobile AP Handle
Length	4			2	
Value	$\rightarrow$	uint32	mobile_ap_handle	4	Handle identifying the mobile AP call
					instance.
					The value must be the handle previously
					returned by QMI_QCMAP_MOBILE_
					AP_ENABLE_REQ.
Туре	0x02			1	Timeout
Length	2			2	
Value	$\rightarrow$	uint16	timeout	2	NAT entry timeout.

#### **Optional TLVs**

None

## 3.14.2 Response - QMI\_QCMAP\_SET\_DYNAMIC\_NAT\_ENTRY\_-TIMEOUT\_RESP

Message type

Response

Sender

Service

**Mandatory TLVs** 

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

**Optional TLVs** 

None

#### **Error codes**

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_MALFORMED_MSG	Message was not formulated correctly by the control point
CO. 1	or the message was corrupted during transmission
QMI_ERR_MISSING_ARG	Some TLV was missing
QMI_ERR_INVALID_HANDLE	Mobile AP handle provided in the request is not valid, i.e., it
	is not assigned to the control point
QMI_ERR_INVALID_ARG	Argument is not correct
QMI_ERR_NOT_SUPPORTED	Operation is not supported

## 3.14.3 Description of QMI\_QCMAP\_SET\_DYNAMIC\_NAT\_ENTRY\_-TIMEOUT REQ/RESP

This command sets the NAT entry timeout on the device.

#### QMI\_QCMAP\_GET\_DYNAMIC\_NAT\_ENTRY\_TIMEOUT 3.15

Queries the NAT entry timeout.

**QCMAP** message ID

0x002B

**Version introduced** 

Major - 1, Minor - 0

#### Request - QMI\_QCMAP\_GET\_DYNAMIC\_NAT\_ENTRY\_TIMEOUT\_-3.15.1 **REQ**

Message type

Request

Sender

Control point

## **Mandatory TLVs**

	Name	Vers	ion introduced	Version last modified
Mobile AP Handle	70° 30° 00° 00° 00° 00° 00° 00° 00° 00° 0	000	1.0	1.0

Field	Field	Field	Parameter	Size	Description
	value	type	410	(byte)	
Туре	0x01			1	Mobile AP Handle
Length	4			2	
Value	$\rightarrow$	uint32	mobile_ap_handle	4	Handle identifying the mobile AP call
					instance.
					The value must be the handle previously
					returned by QMI_QCMAP_MOBILE_
					AP_ENABLE_REQ.

**Optional TLVs** 

## 3.15.2 Response - QMI\_QCMAP\_GET\_DYNAMIC\_NAT\_ENTRY\_-**TIMEOUT RESP**

Message type

Response

Sender

Service

#### **Mandatory TLVs**

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

#### **Optional TLVs**

Name	Version introduced	Version last modified	
Timeout	1.0	1.0	

Field	Field	Field	Parameter	Size	Description
	value	type		(byte)	
Туре	0x10			37.7	Timeout
Length	2			20	
Value	$\rightarrow$	uint16	timeout	2	Dynamic NAT entry timeout.

#### **Error codes**

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_MALFORMED_MSG	Message was not formulated correctly by the control point
	or the message was corrupted during transmission
QMI_ERR_MISSING_ARG	Some TLV was missing
QMI_ERR_INVALID_HANDLE	Mobile AP handle provided in the request is not valid, i.e., it
	is not assigned to the control point
QMI_ERR_NOT_SUPPORTED	Operation is not supported

#### 3.15.3 Description of QMI\_QCMAP\_GET\_DYNAMIC\_NAT\_ENTRY\_-**TIMEOUT REQ/RESP**

This command queries the NAT entry timeout on the device.

## 3.16 QMI\_QCMAP\_ADD\_STATIC\_NAT\_ENTRY

Adds a static NAT entry.

**QCMAP** message ID

0x002D

**Version introduced** 

Major - 1, Minor - 0

## 3.16.1 Request - QMI\_QCMAP\_ADD\_STATIC\_NAT\_ENTRY\_REQ

Message type

Request

Sender

Control point

## **Mandatory TLVs**

Name	Version introduced	Version last modified
Mobile AP Handle	1.0	1.0
SNAT Entry Configuration	1.0	1.0

Field	Field	Field	Parameter	Size	Description
	value	type	450	(byte)	
Туре	0x01			1	Mobile AP Handle
Length	4			2	
Value	$\rightarrow$	uint32	mobile_ap_handle	4	Handle identifying the mobile AP call
					instance.
					The value must be the handle previously
					returned by QMI_QCMAP_MOBILE_
					AP_ENABLE_REQ.
Туре	0x02			1	SNAT Entry Configuration
Length	9			2	
Value	$\rightarrow$	uint32	private_ip_addr	4	Private IP address.
		uint16	private_port	2	Private port.
		uint16	global_port	2	Global port.
		uint8	protocol	1	Protocol.

#### **Optional TLVs**

None

#### 3.16.2 Response - QMI\_QCMAP\_ADD\_STATIC\_NAT\_ENTRY\_RESP

#### Message type

Response

#### Sender

Service

#### **Mandatory TLVs**

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

#### **Optional TLVs**

#### **Error codes**

Optional TLVs	Ne 5ec.
None	"LILIA"
Error codes	Contain
QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_MALFORMED_MSG	Message was not formulated correctly by the control point
1613	or the message was corrupted during transmission
QMI_ERR_MISSING_ARG	Some TLV was missing
QMI_ERR_INVALID_HANDLE	Mobile AP handle provided in the request is not valid, i.e., it
· K4	is not assigned to the control point
QMI_ERR_INVALID_ARG	Argument is not correct
QMI_ERR_NOT_SUPPORTED	Operation is not supported
QMI_ERR_MAX_LIMIT_REACHED	Maximum limit was reached for the static NAT entry
QMI_ERR_DUPLICATE_ENTRY	Entry already exists

#### Description of QMI\_QCMAP\_ADD\_STATIC\_NAT\_ENTRY REQ/RESP 3.16.3

This command adds a static NAT entry.

# 3.17 QMI\_QCMAP\_DELETE\_STATIC\_NAT\_ENTRY

Deletes a static NAT entry.

**QCMAP** message ID

0x002E

**Version introduced** 

Major - 1, Minor - 0

## 3.17.1 Request - QMI\_QCMAP\_DELETE\_STATIC\_NAT\_ENTRY\_REQ

Message type

Request

Sender

Control point

## **Mandatory TLVs**

Name	Version introduced	Version last modified
Mobile AP Handle	1.0	1.0
SNAT Entry Configuration	1.0	1.0

Field	Field	Field	Parameter	Size	Description
	value	type	440	(byte)	
Туре	0x01			1	Mobile AP Handle
Length	4			2	
Value	$\rightarrow$	uint32	mobile_ap_handle	4	Handle identifying the mobile AP call
					instance.
					The value must be the handle previously
					returned by QMI_QCMAP_MOBILE_
					AP_ENABLE_REQ.
Туре	0x02			1	SNAT Entry Configuration
Length	9			2	
Value	$\rightarrow$	uint32	private_ip_addr	4	Private IP address.
		uint16	private_port	2	Private port.
		uint16	global_port	2	Global port.
		uint8	protocol	1	Protocol.

#### **Optional TLVs**

None

## 3.17.2 Response - QMI\_QCMAP\_DELETE\_STATIC\_NAT\_ENTRY\_RESP

Message type

Response

Sender

Service

#### **Mandatory TLVs**

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

#### **Optional TLVs**

None

#### **Error codes**

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_MALFORMED_MSG	Message was not formulated correctly by the control point
1613	or the message was corrupted during transmission
QMI_ERR_MISSING_ARG	Some TLV was missing
QMI_ERR_INVALID_HANDLE	Mobile AP handle provided in the request is not valid, i.e., it
EX.	is not assigned to the control point
QMI_ERR_INVALID_ARG	Argument is not correct
QMI_ERR_NOT_SUPPORTED	Operation is not supported
QMI_ERR_NO_ENTRY	Entry was not found

ortain trade

# 3.17.3 Description of QMI\_QCMAP\_DELETE\_STATIC\_NAT\_ENTRY REQ/RESP

This command deletes a static NAT entry.

## 3.18 QMI\_QCMAP\_GET\_STATIC\_NAT\_ENTRIES

Queries all static NAT entries.

**QCMAP** message ID

0x002F

**Version introduced** 

Major - 1, Minor - 0

## 3.18.1 Request - QMI\_QCMAP\_GET\_STATIC\_NAT\_ENTRIES\_REQ

Message type

Request

Sender

Control point

## **Mandatory TLVs**

Name	Version introduced	Version last modified
Mobile AP Handle	1.0	1.0

Field	Field	Field	Parameter	Size	Description
	value	type	600,00	(byte)	
Туре	0x01		440	1	Mobile AP Handle
Length	4			2	
Value	$\rightarrow$	uint32	mobile_ap_handle	4	Handle identifying the mobile AP call
					instance.
					The value must be the handle previously
					returned by QMI_QCMAP_MOBILE_
					AP_ENABLE_REQ.

#### **Optional TLVs**

## 3.18.2 Response - QMI\_QCMAP\_GET\_STATIC\_NAT\_ENTRIES\_RESP

## Message type

Response

#### Sender

Service

#### **Mandatory TLVs**

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

## **Optional TLVs**

Name	Version introduced	Version last modified
SNAT Configuration	1.0	1.0

Field	Field	Field	Parameter	Size	Description
	value	type		(byte)	_
Туре	0x10			of for	SNAT Configuration
Length	Var		33	2	
Value	$\rightarrow$	uint8	snat_config_len	\$10°	Number of sets of the following
			3, 00	(0)	elements:
			and the second	Sim	• private_ip_addr
		,	8,0° (c) (@"		• private_port
					• global_port
			20, 992		• protocol
		uint32	private_ip_addr	4	Private IP address.
		uint16	private_port	2	Private port.
		uint16	global_port	2	Global port.
		uint8	protocol	1	Protocol.

#### **Error codes**

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_MALFORMED_MSG	Message was not formulated correctly by the control point
	or the message was corrupted during transmission
QMI_ERR_MISSING_ARG	Some TLV was missing
QMI_ERR_INVALID_HANDLE	Mobile AP handle provided in the request is not valid, i.e., it
	is not assigned to the control point
QMI_ERR_INVALID_ARG	Argument is not correct
QMI_ERR_NOT_SUPPORTED	Operation is not supported

# 3.18.3 Description of QMI\_QCMAP\_GET\_STATIC\_NAT\_ENTRIES REQ/RESP

This command queries all static NAT entries. The response message contains the number of entries followed by the value of these entries sequentially.

## 3.19 QMI\_QCMAP\_SET\_DMZ

Sets the DMZ (perimeter network) IP address for the mobile AP.

#### **QCMAP** message ID

0x0030

#### **Version introduced**

Major - 1, Minor - 0

## 3.19.1 Request - QMI\_QCMAP\_SET\_DMZ\_REQ

Message type

Request

Sender

Control point

## **Mandatory TLVs**

	Name	Pylo	Versio	n introduced	Version last modified
Mobile AP Handle		30 05	5: 63:	1.0	1.0
DMZ IP Address		10 N	5	1.0	1.0

Field	Field	Field	Parameter	Size	Description
	value	type	*100	(byte)	
Туре	0x01			1	Mobile AP Handle
Length	4			2	
Value	$\rightarrow$	uint32	mobile_ap_handle	4	Handle identifying the mobile AP call
					instance.
					The value must be the handle previously
					returned by QMI_QCMAP_MOBILE_
					AP_ENABLE_REQ.
Туре	0x02			1	DMZ IP Address
Length	4			2	
Value	$\rightarrow$	uint32	dmz_ip_addr	4	DMZ IP address.

## **Optional TLVs**

## Response - QMI\_QCMAP\_SET\_DMZ\_RESP

Message type

Response

Sender

Service

#### **Mandatory TLVs**

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

#### **Optional TLVs**

#### **Error codes**

Optional TLVs	
None	Secrets
Error codes	ale sec
QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_MALFORMED_MSG	Message was not formulated correctly by the control point
	or the message was corrupted during transmission
QMI_ERR_MISSING_ARG	Some TLV was missing
QMI_ERR_INVALID_HANDLE	Mobile AP handle provided in the request is not valid, i.e., it
	is not assigned to the control point
QMI_ERR_INVALID_ARG	Argument is not correct
QMI_ERR_NOT_SUPPORTED	Operation is not supported

#### Description of QMI\_QCMAP\_SET\_DMZ REQ/RESP 3.19.3

This command sets the DMZ IP address for the mobile AP.

## 3.20 QMI\_QCMAP\_GET\_DMZ

Queries the DMZ IP address on the mobile AP.

**QCMAP** message ID

0x0032

**Version introduced** 

Major - 1, Minor - 0

## 3.20.1 Request - QMI\_QCMAP\_GET\_DMZ\_REQ

Message type

Request

Sender

Control point

## **Mandatory TLVs**

Name	Version introduced	Version last modified
Mobile AP Handle	1.0	1.0

Field	Field	Field	Parameter	Size	Description
	value	type	600	(byte)	
Туре	0x01		440	1	Mobile AP Handle
Length	4			2	
Value	$\rightarrow$	uint32	mobile_ap_handle	4	Handle identifying the mobile AP call
					instance.
					The value must be the handle previously
					returned by QMI_QCMAP_MOBILE_
					AP_ENABLE_REQ.

#### **Optional TLVs**

## 3.20.2 Response - QMI QCMAP GET DMZ RESP

#### Message type

Response

#### Sender

Service

#### **Mandatory TLVs**

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

#### **Optional TLVs**

Name	Version introduced	Version last modified
DMZ IP Address	1.0	1.0

Field	Field	Field	Parameter	Size	Description
	value	type		(byte)	_
Туре	0x10			of o	DMZ IP Address
Length	4			2	
Value	$\rightarrow$	uint32	dmz_ip_addr	N 1.40	DMZ IP address.

#### **Error codes**

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_MALFORMED_MSG	Message was not formulated correctly by the control point
	or the message was corrupted during transmission
QMI_ERR_MISSING_ARG	Some TLV was missing
QMI_ERR_INVALID_HANDLE	Mobile AP handle provided in the request is not valid, i.e., it
	is not assigned to the control point
QMI_ERR_INVALID_ARG	Argument is not correct
QMI_ERR_NOT_SUPPORTED	Operation is not supported

## 3.20.3 Description of QMI\_QCMAP\_GET\_DMZ REQ/RESP

This command queries the DMZ entry that was previously set via QMI\_QCMAP\_SET\_DMZ.

If no DMZ is set at the modem, an IP address of 0.0.0.0 is returned.

## 3.21 QMI\_QCMAP\_DELETE\_DMZ

Deletes the DMZ entry or DMZ IP address.

**QCMAP** message ID

0x0031

**Version introduced** 

Major - 1, Minor - 0

## 3.21.1 Request - QMI\_QCMAP\_DELETE\_DMZ\_REQ

Message type

Request

Sender

Control point

## **Mandatory TLVs**

Name	Version introduced	Version last modified
Mobile AP Handle	1.0	1.0

Field	Field	Field	Parameter	Size	Description
	value	type	C 60 109	(byte)	
Туре	0x01		4400	1	Mobile AP Handle
Length	4			2	
Value	$\rightarrow$	uint32	mobile_ap_handle	4	Handle identifying the mobile AP call
					instance.
					The value must be the handle previously
					returned by QMI_QCMAP_MOBILE_
					AP_ENABLE_REQ.

#### **Optional TLVs**

## Response - QMI\_QCMAP\_DELETE\_DMZ\_RESP

Message type

Response

Sender

Service

#### **Mandatory TLVs**

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

#### **Optional TLVs**

#### **Error codes**

Optional TLVs	
None	C. Ected's
Error codes	ade 5ect
QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_MALFORMED_MSG	Message was not formulated correctly by the control point
	or the message was corrupted during transmission
QMI_ERR_MISSING_ARG	Some TLV was missing
QMI_ERR_INVALID_HANDLE	Mobile AP handle provided in the request is not valid, i.e., it
	is not assigned to the control point
QMI_ERR_INVALID_ARG	Argument is not correct
QMI_ERR_NOT_SUPPORTED	Operation is not supported

#### Description of QMI\_QCMAP\_DELETE\_DMZ REQ/RESP 3.21.3

This command deletes the DMZ entry that was previously set via QMI\_QCMAP\_SET\_DMZ.

## 3.22 QMI\_QCMAP\_GET\_WWAN\_CONFIG

Queries the WWAN IP configuration.

**QCMAP** message ID

0x0033

**Version introduced** 

Major - 1, Minor - 0

## 3.22.1 Request - QMI\_QCMAP\_GET\_WWAN\_CONFIG\_REQ

Message type

Request

Sender

Control point

## **Mandatory TLVs**

	Name	Me	Versio	nintroduced	Version last modified
Mobile AP Handle		30 05	5: 63:	1.0	1.0
Address Type		- AN 65	15 m	1.0	1.0

Field	Field	Field	Parameter	Size	Description	
	value	type	V 4400	(byte)		
Туре	0x01			1	Mobile AP Handle	
Length	4			2		
Value	$\rightarrow$	uint32	mobile_ap_handle	4	Handle identifying the mobile AP call	
					instance.	
					The value must be the handle previously	
					returned by QMI_QCMAP_MOBILE_	
					AP_ENABLE_REQ.	
Туре	0x02			1	Address Type	
Length	8			2		
Value	$\rightarrow$	mask	addr_type_op	8	WWAN configuration mask values:	
					• 1 – IPv4 address	
					• 2 – IPv6 address	
					• 4 – IPv4 DNS address	
					• 8 – IPv6 DNS address	

## **Optional TLVs**

None

#### Response - QMI\_QCMAP\_GET\_WWAN\_CONFIG\_RESP 3.22.2

#### Message type

Response

## Sender

Service

## **Mandatory TLVs**

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

## **Optional TLVs**

Name	Version introduced	Version last modified
IPv4 Address	1.0	1.0
IPv6 Address	1.0	1.0
IPv4 Primary DNS Address	1.0	1.0
IPv4 Secondary DNS Address	1.0	1.0
IPv6 Primary DNS Address	1.0	1.0
IPv6 Secondary DNS Address	1.0	1.0

Field	Field	Field	Parameter	Size	Description
	value	type	410	(byte)	
Туре	0x10			1	IPv4 Address
Length	4			2	
Value	$\rightarrow$	uint32	v4_addr	4	IPv4 address.
Type	0x11			1	IPv6 Address
Length	16			2	
Value	$\rightarrow$	uint8	v6_addr	16	IPv6 address.
Туре	0x12			1	IPv4 Primary DNS Address
Length	4			2	
Value	$\rightarrow$	uint32	v4_prim_dns_addr	4	IPv4 primary DNS address.
Туре	0x13			1	IPv4 Secondary DNS Address
Length	4			2	
Value	$\rightarrow$	uint32	v4_sec_dns_addr	4	IPv4 secondary DNS address.
Туре	0x14			1	IPv6 Primary DNS Address
Length	16			2	
Value	$\rightarrow$	uint8	v6_prim_dns_addr	16	IPv6 primary DNS address.
Туре	0x15			1	IPv6 Secondary DNS Address
Length	16			2	
Value	$\rightarrow$	uint8	v6_sec_dns_addr	16	IPv6 secondary DNS address.

#### **Error codes**

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_MALFORMED_MSG	Message was not formulated correctly by the control point
	or the message was corrupted during transmission
QMI_ERR_MISSING_ARG	Some TLV was missing
QMI_ERR_INVALID_HANDLE	Mobile AP handle provided in the request is not valid, i.e., it
	is not assigned to the control point
QMI_ERR_INVALID_ARG	Argument is not correct
QMI_ERR_NOT_SUPPORTED	Operation is not supported

## 3.22.3 Description of QMI\_QCMAP\_GET\_WWAN\_CONFIG REQ/RESP

This command queries the WWAN IP configuration for the mobile AP. The command must be issued by the control point after QCMAP\_WWAN\_STATUS\_IND has indicated a successful WWAN bringup, otherwise a QMI\_ERR\_INTERNAL error is returned.

#### QMI\_QCMAP\_ENABLE\_FIREWALL\_SETTING 3.23

Enables the firewall setting.

**QCMAP** message ID

0x0034

**Version introduced** 

Major - 1, Minor - 0

#### Request - QMI\_QCMAP\_ENABLE\_FIREWALL\_SETTING\_REQ 3.23.1

Message type

Request

Sender

Control point

## **Mandatory TLVs**

	Name	Vers	sion introduced	Version last modified
Mobile AP Handle		3 20, 64.	1.0	1.0
Packets Allowed		20 00 20 TO	1.0	1.0

Field	Field	Field	Parameter	Size	Description
	value	type	4400	(byte)	
Туре	0x01			1	Mobile AP Handle
Length	4			2	
Value	$\rightarrow$	uint32	mobile_ap_handle	4	Handle identifying the mobile AP call
					instance.
					The value must be the handle previously
					returned by QMI_QCMAP_MOBILE_
					AP_ENABLE_REQ.
Туре	0x02			1	Packets Allowed
Length	1			2	
Value	$\rightarrow$	boolean	pkts_allowed	1	Packets allowed operation. Values:
					• TRUE – Packets matching the firewall
					rule are allowed
					• FALSE – Packets matching the firewall
					rule are dropped

#### **Optional TLVs**

None

## 3.23.2 Response - QMI\_QCMAP\_ENABLE\_FIREWALL\_SETTING\_RESP

#### Message type

Response

#### Sender

Service

#### **Mandatory TLVs**

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

#### **Optional TLVs**

None

#### **Error codes**

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_MALFORMED_MSG	Message was not formulated correctly by the control point
1613	or the message was corrupted during transmission
QMI_ERR_MISSING_ARG	Some TLV was missing
QMI_ERR_INVALID_HANDLE	Mobile AP handle provided in the request is not valid, i.e., it
54	is not assigned to the control point
QMI_ERR_INVALID_ARG	Argument is not correct
QMI_ERR_NOT_SUPPORTED	Operation is not supported

antain trade

# 3.23.3 Description of QMI\_QCMAP\_ENABLE\_FIREWALL\_SETTING REQ/RESP

This command enables the firewall and sets the condition whether the packets matching the firewall rule are to be allowed or dropped.

## 3.24 QMI\_QCMAP\_GET\_FIREWALL\_SETTING

Queries the firewall setting.

**QCMAP** message ID

0x0035

**Version introduced** 

Major - 1, Minor - 0

## 3.24.1 Request - QMI\_QCMAP\_GET\_FIREWALL\_SETTING\_REQ

Message type

Request

Sender

Control point

## **Mandatory TLVs**

Name	Version introduced	Version last modified
Mobile AP Handle	1.0	1.0

Field	Field	Field	Parameter	Size	Description
	value	type	600,00	(byte)	
Туре	0x01		440	1	Mobile AP Handle
Length	4			2	
Value	$\rightarrow$	uint32	mobile_ap_handle	4	Handle identifying the mobile AP call
					instance.
					The value must be the handle previously
					returned by QMI_QCMAP_MOBILE_
					AP_ENABLE_REQ.

#### **Optional TLVs**

## 3.24.2 Response - QMI\_QCMAP\_GET\_FIREWALL\_SETTING\_RESP

## Message type

Response

#### Sender

Service

#### **Mandatory TLVs**

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

## **Optional TLVs**

Name	Version introduced	Version last modified
Firewall Enabled	1.0	1.0
Packets Allowed	1.0	1.0

Field	Field	Field	Parameter	Size	Description
	value	type		(byte)	) `
Туре	0x10		674	~1\)	Firewall Enabled
Length	1		Mic	(2.0)	
Value	$\rightarrow$	boolean	firewall_enabled	(d)	Whether the firewall is enabled; boolean
			Service of the servic	Stan	value.
Туре	0x11		STO ST. CO	1	Packets Allowed
Length	1		(0) 27 14	2	
Value	$\rightarrow$	boolean	pkts_allowed	1	Whether packets are allowed; boolean
			440		value.

#### **Error codes**

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_MALFORMED_MSG	Message was not formulated correctly by the control point
	or the message was corrupted during transmission
QMI_ERR_MISSING_ARG	Some TLV was missing
QMI_ERR_INVALID_HANDLE	Mobile AP handle provided in the request is not valid, i.e., it
	is not assigned to the control point
QMI_ERR_NOT_SUPPORTED	Operation is not supported

## 3.24.3 Description of QMI\_QCMAP\_GET\_FIREWALL\_SETTING REQ/RESP

This command queries the firewall setting.



## 3.25 QMI\_QCMAP\_DISABLE\_FIREWALL\_SETTING

Disables the firewall setting.

**QCMAP** message ID

0x0036

**Version introduced** 

Major - 1, Minor - 0

## 3.25.1 Request - QMI\_QCMAP\_DISABLE\_FIREWALL\_SETTING\_REQ

Message type

Request

Sender

Control point

## **Mandatory TLVs**

Name	Version introduced	Version last modified
Mobile AP Handle	1.0	1.0

Field	Field	Field	Parameter	Size	Description
	value	type	600,00	(byte)	
Туре	0x01		440	1	Mobile AP Handle
Length	4			2	
Value	$\rightarrow$	uint32	mobile_ap_handle	4	Handle identifying the mobile AP call
					instance.
					The value must be the handle previously
					returned by QMI_QCMAP_MOBILE_
					AP_ENABLE_REQ.

#### **Optional TLVs**

## Response - QMI\_QCMAP\_DISABLE\_FIREWALL\_SETTING\_RESP

Message type

Response

Sender

Service

#### **Mandatory TLVs**

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

#### **Optional TLVs**

#### **Error codes**

Optional TLVs	
None	cectaits
Error codes	de Sec.
QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_MALFORMED_MSG	Message was not formulated correctly by the control point
	or the message was corrupted during transmission
QMI_ERR_MISSING_ARG	Some TLV was missing
QMI_ERR_INVALID_HANDLE	Mobile AP handle provided in the request is not valid, i.e., it
	is not assigned to the control point
QMI_ERR_NOT_SUPPORTED	Operation is not supported

#### Description of QMI\_QCMAP\_DISABLE\_FIREWALL\_SETTING 3.25.3 **REQ/RESP**

This command disables the firewall setting.

# 3.26 QMI\_QCMAP\_ADD\_FIREWALL\_CONFIG

Adds a firewall configuration rule.

**QCMAP** message ID

0x0037

**Version introduced** 

Major - 1, Minor - 0

## 3.26.1 Request - QMI\_QCMAP\_ADD\_FIREWALL\_CONFIG\_REQ

Message type

Request

Sender

Control point

## **Mandatory TLVs**

Name	Version introduced	Version last modified
Mobile AP Handle	1.0	1.0
Firewall Configuration	1.0	1.0

Field	Field	Field	Parameter	Size	Description
	value	type	4400	(byte)	
Туре	0x01			1	Mobile AP Handle
Length	4			2	
Value	$\rightarrow$	uint32	mobile_ap_handle	4	Handle identifying the mobile AP call
					instance.
					The value must be the handle previously
					returned by QMI_QCMAP_MOBILE_
					AP_ENABLE_REQ.
Туре	0x02			1	Firewall Configuration
Length	5			2	
Value	$\rightarrow$	uint16	start_dest_port	2	Start value of the destination port range.
		uint16	end_dest_port	2	End value of the destination port range.
		uint8	protocol	1	Protocol value.

## **Optional TLVs**

None

#### Response - QMI\_QCMAP\_ADD\_FIREWALL\_CONFIG\_RESP 3.26.2

## Message type

Response

#### Sender

Service

## **Mandatory TLVs**

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

## **Optional TLVs**

Name	Version introduced	Version last modified
Firewall Handle	1.0	1.0

Field	Field	Field	Parameter	Size	Description
	value	type	30	(byte)	
Туре	0x10			S 1	Firewall Handle
Length	4		Silver Color	2	
Value	$\rightarrow$	uint32	firewall_handle	4	Handle identifying the firewall rule.

#### **Error codes**

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_MALFORMED_MSG	Message was not formulated correctly by the control point
	or the message was corrupted during transmission
QMI_ERR_MISSING_ARG	Some TLV was missing
QMI_ERR_INVALID_HANDLE	Mobile AP handle provided in the request is not valid, i.e., it
	is not assigned to the control point
QMI_ERR_INVALID_ARG	Argument is not correct
QMI_ERR_NOT_SUPPORTED	Operation is not supported
QMI_ERR_NO_MEMORY	Maximum number of supported firewall rules was exceeded;
	cannot add any more firewall rules

# 3.26.3 Description of QMI\_QCMAP\_ADD\_FIREWALL\_CONFIG REQ/RESP

This command adds a firewall configuration rule.



#### QMI\_QCMAP\_DELETE\_FIREWALL\_CONFIG 3.27

Deletes a firewall configuration rule.

**QCMAP** message ID

0x0039

**Version introduced** 

Major - 1, Minor - 0

#### Request - QMI\_QCMAP\_DELETE\_FIREWALL\_CONFIG\_REQ 3.27.1

Message type

Request

Sender

Control point

## **Mandatory TLVs**

	Name	Pylo	Version	introduced	Version last modified
Mobile AP Handle		100 05	5.03	1.0	1.0
Firewall Handle		- Ch 100	25/111	1.0	1.0

Field	Field	Field	Parameter	Size	Description
	value	type	450	(byte)	
Туре	0x01			1	Mobile AP Handle
Length	4			2	
Value	$\rightarrow$	uint32	mobile_ap_handle	4	Handle identifying the mobile AP call
					instance.
					The value must be the handle previously
					returned by QMI_QCMAP_MOBILE_
					AP_ENABLE_REQ.
Туре	0x02			1	Firewall Handle
Length	4			2	
Value	$\rightarrow$	uint32	firewall_handle	4	Handle identifying the firewall entry.
					The value must be the handle previously
					returned by QMI_QCMAP_ADD_
					FIREWALL_CONFIG_RESP or
					QMI_QCMAP_GET_FIREWALL_
					CONFIG_RESP.

## **Optional TLVs**

None

# 3.27.2 Response - QMI\_QCMAP\_DELETE\_FIREWALL\_CONFIG\_RESP

Message type

Response

Sender

Service

## **Mandatory TLVs**

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

## **Optional TLVs**

None

#### **Error codes**

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_MALFORMED_MSG	Message was not formulated correctly by the control point
1613	or the message was corrupted during transmission
QMI_ERR_MISSING_ARG	Some TLV was missing
QMI_ERR_INVALID_HANDLE	Mobile AP handle provided in the request is not valid, i.e., it
EX.	is not assigned to the control point
QMI_ERR_INVALID_ARG	Argument is not correct
QMI_ERR_NOT_SUPPORTED	Operation is not supported

ortain Trade

# 3.27.3 Description of QMI\_QCMAP\_DELETE\_FIREWALL\_CONFIG REQ/RESP

This command deletes a firewall rule.

# 3.28 QMI\_QCMAP\_GET\_FIREWALL\_CONFIG

Queries the firewall configuration rules.

**QCMAP** message ID

0x0038

**Version introduced** 

Major - 1, Minor - 0

## 3.28.1 Request - QMI\_QCMAP\_GET\_FIREWALL\_CONFIG\_REQ

Message type

Request

Sender

Control point

## **Mandatory TLVs**

Name	Version introduced	Version last modified
Mobile AP Handle	1.0	1.0

Field	Field	Field	Parameter	Size	Description
	value	type	600,00	(byte)	
Туре	0x01		440	1	Mobile AP Handle
Length	4			2	
Value	$\rightarrow$	uint32	mobile_ap_handle	4	Handle identifying the mobile AP call
					instance.
					The value must be the handle previously
					returned by QMI_QCMAP_MOBILE_
					AP_ENABLE_REQ.

## **Optional TLVs**

None

## 3.28.2 Response - QMI\_QCMAP\_GET\_FIREWALL\_CONFIG\_RESP

N	les	sage	e tv	рe

Response

#### Sender

Service

## **Mandatory TLVs**

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

## **Optional TLVs**

Name	Version introduced	Version last modified
Firewall Configuration	1.0	1.0
	763	
	700	

Field	Field	Field	Parameter	Size	Description
	value	type		(byte)	
Туре	0x10			oT of	Firewall Configuration
Length	Var			2	
Value	$\rightarrow$	uint8	firewall_config_len	$\mathcal{N}1$	Number of sets of the following
			30	(0)	elements:
				3	firewall_handle
			8,0° (c) (@"		• start_dest_port
					• end_dest_port
			20, 992		• protocol
		uint32	firewall_handle	4	Handle identifying the firewall rule.
		uint16	start_dest_port	2	Start value of the destination port range.
		uint16	end_dest_port	2	End value of the destination port range.
		uint8	protocol	1	Protocol value.

#### **Error codes**

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_MALFORMED_MSG	Message was not formulated correctly by the control point
	or the message was corrupted during transmission
QMI_ERR_MISSING_ARG	Some TLV was missing
QMI_ERR_INVALID_HANDLE	Mobile AP handle provided in the request is not valid, i.e., it
	is not assigned to the control point
QMI_ERR_NOT_SUPPORTED	Operation is not supported

## 3.28.3 Description of QMI\_QCMAP\_GET\_FIREWALL\_CONFIG REQ/RESP

This command queries all the firewall entries. The response message contains the number of entries followed by the value of these entries sequentially.

# 3.29 QMI\_QCMAP\_STATION\_MODE\_ENABLE

Enables Station (STA) mode functionality for a mobile AP instance on the modem.

**QCMAP** message ID

0x003B

**Version introduced** 

Major - 1, Minor - 0

## 3.29.1 Request - QMI\_QCMAP\_STATION\_MODE\_ENABLE\_REQ

Message type

Request

Sender

Control point

## **Mandatory TLVs**

Name	Version introduced	Version last modified
Mobile AP Handle	1.0	1.0

Field	Field	Field	Parameter	Size	Description
	value	type	600,00	(byte)	
Туре	0x01		440	1	Mobile AP Handle
Length	4			2	
Value	$\rightarrow$	uint32	mobile_ap_handle	4	Handle identifying the mobile AP call
					instance.
					The value must be the handle previously
					returned by QMI_QCMAP_MOBILE_
					AP_ENABLE_REQ.

## **Optional TLVs**

None

#### Response - QMI QCMAP STATION MODE ENABLE RESP 3.29.2

Message	ty	pe
---------	----	----

Response

Sender

Service

#### **Mandatory TLVs**

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

## **Optional TLVs**

#### **Error codes**

Optional TLVs	
None	Secrats
Error codes	desect
QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_MALFORMED_MSG	Message was not formulated correctly by the control point
	or the message was corrupted during transmission
QMI_ERR_MISSING_ARG	Some TLV was missing
QMI_ERR_INVALID_HANDLE	Mobile AP handle provided in the request is not valid, i.e., it
	is not assigned to the control point
QMI_ERR_NOT_SUPPORTED	Operation is not supported

#### Description of QMI QCMAP STATION MODE ENABLE REQ/RESP 3.29.3

This command enables STA mode functionality at the modem for a single mobile AP instance.

After this request is successfully processed, all packet connectivity to an outside network occurs through the WLAN station. The modem routing engine appropriately handles the packet routing into and out of the modem.

# 3.30 QMI\_QCMAP\_STATION\_MODE\_DISABLE

Disables STA mode functionality for a mobile AP instance on the modem.

**QCMAP** message ID

0x003C

Version introduced

Major - 1, Minor - 0

## 3.30.1 Request - QMI\_QCMAP\_STATION\_MODE\_DISABLE\_REQ

Message type

Request

Sender

Control point

## **Mandatory TLVs**

Name	Version introduced	Version last modified
Mobile AP Handle	1.0	1.0

Field	Field	Field	Parameter	Size	Description	
	value	type	C 60 109	(byte)		
Туре	0x01		4400	1	Mobile AP Handle	
Length	4			2		
Value	$\rightarrow$	uint32	mobile_ap_handle	4	Handle identifying the mobile AP call	
					instance.	
				The value must be the handle previous		
				returned by QMI_QCMAP_MOBILE		
					AP_ENABLE_REQ.	

## **Optional TLVs**

None

#### Response - QMI\_QCMAP\_STATION\_MODE\_DISABLE\_RESP 3.30.2

Message type

Response

Sender

Service

#### **Mandatory TLVs**

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

## **Optional TLVs**

None

#### **Error codes**

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_MALFORMED_MSG	Message was not formulated correctly by the control point
	or the message was corrupted during transmission
QMI_ERR_MISSING_ARG	Some TLV was missing
QMI_ERR_INVALID_HANDLE	Mobile AP handle provided in the request is not valid, i.e., it
	is not assigned to the control point
QMI_ERR_NOT_SUPPORTED	Operation is not supported

#### Description of QMI\_QCMAP\_STATION\_MODE\_DISABLE 3.30.3 **REQ/RESP**

This command disables STA mode functionality at the modem for a single mobile AP instance. When this request has been successfully processed, the control point invokes bringing up the WWAN from the mobile AP.

# 3.31 QMI\_QCMAP\_GET\_STATION\_MODE

Queries the STA mode functionality for a mobile AP instance on the modem.

**QCMAP** message ID

0x003D

**Version introduced** 

Major - 1, Minor - 0

## 3.31.1 Request - QMI\_QCMAP\_GET\_STATION\_MODE\_REQ

Message type

Request

Sender

Control point

## **Mandatory TLVs**

Name	Version introduced	Version last modified
Mobile AP Handle	1.0	1.0

Field	Field	Field	Parameter	Size	Description
	value	type	600,00	(byte)	
Туре	0x01		440	1	Mobile AP Handle
Length	4			2	
Value	$\rightarrow$	uint32	mobile_ap_handle	4	Handle identifying the mobile AP call
					instance.
					The value must be the handle previously
					returned by QMI_QCMAP_MOBILE_
					AP_ENABLE_REQ.

## **Optional TLVs**

None

## 3.31.2 Response - QMI QCMAP GET STATION MODE RESP

## Message type

Response

#### Sender

Service

## **Mandatory TLVs**

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

## **Optional TLVs**

Name	Version introduced	Version last modified
Station Mode	1.0	1.0

Field	Field	Field	Parameter	r Size	Description
	value	type		(byte)	
Туре	0x10			of for	Station Mode
Length	1			2	
Value	$\rightarrow$	boolean	station_mode	N 10	Whether STA mode has been enabled;
				10 00 B	boolean value.

#### **Error codes**

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_MALFORMED_MSG	Message was not formulated correctly by the control point
	or the message was corrupted during transmission
QMI_ERR_MISSING_ARG	Some TLV was missing
QMI_ERR_INVALID_HANDLE	Mobile AP handle provided in the request is not valid, i.e., it
	is not assigned to the control point
QMI_ERR_NOT_SUPPORTED	Operation is not supported

## 3.31.3 Description of QMI QCMAP GET STATION MODE REQ/RESP

This command queries the STA mode functionality at the modem for a single mobile AP instance.

# 3.32 QMI\_QCMAP\_ADD\_EXTD\_FIREWALL\_CONFIG

Adds IP filter-based firewall rules (extended firewall).

**QCMAP** message ID

0x003F

**Version introduced** 

Major - 1, Minor - 1

## 3.32.1 Request - QMI\_QCMAP\_ADD\_EXTD\_FIREWALL\_CONFIG\_REQ

Message type

Request

Sender

Control point

## **Mandatory TLVs**

Name	Ma	Version introduced	Version last modified
Mobile AP Handle	10	1.1	1.1
Next Header Protocol		1.1	1.1

Field	Field	Field	Parameter	Size	Description
	value	type	4400	(byte)	
Туре	0x01			1	Mobile AP Handle
Length	4			2	
Value	$\rightarrow$	uint32	mobile_ap_handle	4	Handle identifying the mobile AP call instance.  The value must be the handle previously returned by QMI_QCMAP_MOBILE_AP_ENABLE_REQ.
Туре	0x02			1	Next Header Protocol
Length	4			2	

Field	Field	Field	Parameter	Size	Description
	value	type		(byte)	
Value	$\rightarrow$	enum	next_hdr_prot	4	IPv4/IPv6 next header protocol after the
					IP header. Values:
					• 0x01 – QCMAP_EXTD_FIREWALL_
					PROTO_TCP – Transmission Control
					Protocol
					• 0x02 – QCMAP_EXTD_FIREWALL_
					PROTO_UDP – User Datagram Protocol
					• 0x03 – QCMAP_EXTD_FIREWALL_
					PROTO_ICMP – Internet Control
					Message Protocol
					• 0x04 – QCMAP_EXTD_FIREWALL_
					PROTO_ICMP6 – Internet Control
					Message Protocol version 6
					• 0x05 – QCMAP_EXTD_FIREWALL_
					PROTO_ESP – Encapsulating Security
					Payload Protocol
					• 0x06 – QCMAP_EXTD_FIREWALL_
				,	PROTO_TCP_UDP – Transmission
				all a	Control Protocol/User Datagram
				ATT S	Protocol

## **Optional TLVs**

Name	Version introduced	Version last modified
TCP/UDP Source	1.1	1.1
TCP/UDP Destination	1.1	1.1
ICMP Type	1.1	1.1
ICMP Code	1.1	1.1
ESP Security Parameters Index	1.1	1.1
IPv4 Source Address	1.1	1.1
IPv4 Destination Address	1.1	1.1
IPv4 TOS	1.1	1.1
IPv6 Source Address	1.1	1.1
IPv6 Destination Address	1.1	1.1
IPv6 Traffic Class	1.1	1.1

Field	Field	Field	Parameter	Size	Description
	value	type		(byte)	
Туре	0x10			1	TCP/UDP Source
Length	4			2	
Value	$\rightarrow$	uint16	port	2	TCP/UDP port as specified in the
					TCP/UDP protocol (RFC 793 [S4] and
					RFC 768 [S1]).
		uint16	range	2	TCP/UDP port range as specified in the
					TCP/UDP protocol (RFC 793 [S4] and
					RFC 768 [S1]).

Field	Field	Field	Parameter	Size	Description
	value	type		(byte)	
Туре	0x11			1	TCP/UDP Destination
Length	4			2	
Value	$\rightarrow$	uint16	port	2	TCP/UDP port as specified in the
					TCP/UDP protocol (RFC 793 [S4] and
					RFC 768 [S1]).
		uint16	range	2	TCP/UDP port range as specified in the
					TCP/UDP protocol (RFC 793 [S4] and
					RFC 768 [S1]).
Туре	0x12			1	ICMP Type
Length	1			2	
Value	$\rightarrow$	uint8	icmp_type	1	ICMP type as specified in the ICMP
					specification (RFC 792 [S3]).
Туре	0x13			1	ICMP Code
Length	1			2	X <sup>5</sup>
Value	$\rightarrow$	uint8	icmp_code	1	ICMP code as specified in the ICMP
					specification (RFC 792 [S3]).
Туре	0x14			1	ESP Security Parameters Index
Length	4			2	(10
Value	$\rightarrow$	uint32	esp_spi	4	Security parameters index as specified in
				000	the ESP protocol (RFC 4303 [S7]).
Туре	0x15		6.	1	IPv4 Source Address
Length	8		Mo	2.0	
Value	$\rightarrow$	uint32	addr	4	IPv4 address as specified in the IPv4
			Service of the servic	Star	protocol specification (RFC 791 [S2]).
		uint32	subnet_mask	4	IPv4 subnet mask as specified in the IPv4
					protocol specification (RFC 791 [S2]).
Туре	0x16		30, 903,	1	IPv4 Destination Address
Length	8		410	2	
Value	$\rightarrow$	uint32	addr	4	IPv4 address as specified in the IPv4
					protocol specification (RFC 791 [S2]).
		uint32	subnet_mask	4	IPv4 subnet mask as specified in the IPv4
					protocol specification (RFC 791 [S2]).
Туре	0x17			1	IPv4 TOS
Length	2			2	
Value	$\rightarrow$	uint8	value	1	TOS value as specified in the IPv4
					protocol specification (RFC 791 [S2]).
		uint8	mask	1	IPv4 TOS mask.
Туре	0x18			1	IPv6 Source Address
Length	17			2	
Value	$\rightarrow$	uint8	addr	16	IPv6 address as specified in the IPv6
					protocol specification (RFC 2460 [S5]).
		uint8	prefix_len	1	IPv6 prefix length as specified in the
					IPv6 protocol addressing architecture
					specification (RFC 3513 [S6]).
Туре	0x19			1	IPv6 Destination Address
Length	17			2	

Field	Field	Field	Parameter	Size	Description
	value	type		(byte)	
Value	$\rightarrow$	uint8	addr	16	IPv6 address as specified in the IPv6
					protocol specification (RFC 2460 [S5]).
		uint8	prefix_len	1	IPv6 prefix length as specified in the
					IPv6 protocol addressing architecture
					specification (RFC 3513 [S6]).
Туре	0x1A			1	IPv6 Traffic Class
Length	2			2	
Value	$\rightarrow$	uint8	value	1	IPv6 traffic class value as specified in the
					IPv6 protocol specification (RFC 2460
					[S5]).
		uint8	mask	1	IPv6 traffic class mask.

# 3.32.2 Response - QMI\_QCMAP\_ADD\_EXTD\_FIREWALL\_CONFIG\_RESP

Message type

Response

Sender

Service

## **Mandatory TLVs**

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

## **Optional TLVs**

Name	Version introduced	Version last modified
Firewall handle	1.1	1.1

Field	Field	Field	Parameter	Size	Description
	value	type		(byte)	
Туре	0x10			1	Firewall handle
Length	4			2	
Value	$\rightarrow$	uint32	firewall_handle	4	Handle identifying the added firewall
					rule.

## **Error codes**

QMI_ERR_NONE	No error in request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_MALFORMED_MSG	Message was not formulated correctly by the control point
	or the message was corrupted during transmission
QMI_ERR_MISSING_ARG	Some TLV was missing

QMI_ERR_INVALID_HANDLE	Mobile AP handle provided in the request is not valid, i.e., it
	is not assigned to the control point
QMI_ERR_INVALID_ARG	Argument is not correct
QMI_ERR_NOT_SUPPORTED	Operation is not supported
QMI_ERR_NO_MEMORY	Maximum number of supported firewall rules was exceeded;
	cannot add any more firewall rules

# 3.32.3 Description of QMI\_QCMAP\_ADD\_EXTD\_FIREWALL\_CONFIG REQ/RESP

This command adds a single IP filter-based firewall rule. The control point must specify the source/destination port and range when the value of the Next Header Protocol TLV is TCP/UDP. Otherwise, a QMI\_ERR\_MISSING\_ARG error is returned.

# 3.33 QMI\_QCMAP\_GET\_EXTD\_FIREWALL\_CONFIG

Gets the firewall rules.

**QCMAP** message ID

0x0040

**Version introduced** 

Major - 1, Minor - 1

## 3.33.1 Request - QMI\_QCMAP\_GET\_EXTD\_FIREWALL\_CONFIG\_REQ

Message type

Request

Sender

Control point

## **Mandatory TLVs**

Name	Version introduced	Version last modified
Firewall Handle	1.1	1.1

Field	Field	Field	Parameter	Size	Description
	value	type	601/09	(byte)	
Туре	0x01		×100	1	Mobile AP handle
Length	4			2	
Value	$\rightarrow$	uint32	mobile_ap_handle	4	Handle identifying the mobile AP call
					instance.
					The value must be the handle previously
					returned by QMI_QCMAP_MOBILE_
					AP_ENABLE_REQ.
Туре	0x02			1	Firewall Handle
Length	4			2	
Value	$\rightarrow$	uint32	firewall_handle	4	Handle identifying the firewall entry.
					The value must be the handle previously
					returned by one of the following:
					<ul><li>QMI_QCMAP_ADD_FIREWALL_</li></ul>
					CONFIG_RESP
					<ul><li>QMI_QCMAP_GET_FIREWALL_</li></ul>
					CONFIG_RESP
					• QMI_QCMAP_ADD_EXTD_
					FIREWALL_CONFIG_RESP
					• QMI_QCMAP_GET_FIREWALL_
					CONFIG_HANDLE_LIST_RESP

## **Optional TLVs**

None

## 3.33.2 Response - QMI\_QCMAP\_GET\_EXTD\_FIREWALL\_CONFIG\_RESP

## Message type

Response

#### Sender

Service

## **Mandatory TLVs**

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

## **Optional TLVs**

Name		Version introduced	Version last modified
Next Header Protocol		×2 1.1	1.1
TCP/UDP Source		(°) (°) 1.1	1.1
TCP/UDP Destination	,i	3 3 6 1.1	1.1
ICMP Type	, la	(8) (8) 1.1	1.1
ICMP Code	X101 2	1.1	1.1
ESP Security Parameters Index	Ser De	1.1	1.1
IPv4 Source Address	11 OS 111 OS	1.1	1.1
IPv4 Destination Address	CO. W. M.	1.1	1.1
IPv4 TOS	200	1.1	1.1
IPv6 Source Address	1/1	1.1	1.1
IPv6 Destination Address		1.1	1.1
IPv6 Traffic Class		1.1	1.1

Field	Field	Field	Parameter	Size	Description
	value	type		(byte)	
Туре	0x10			1	Next Header Protocol
Length	4			2	

Field	Field	Field	Parameter	Size	Description
	value	type		(byte)	-
Value	$\rightarrow$	enum	next_hdr_prot	4	IPv4/IPv6 next header protocol after the
			•		IP header. Values:
					• 0x01 – QCMAP_EXTD_FIREWALL_
					PROTO_TCP – Transmission Control
					Protocol
					• 0x02 – QCMAP_EXTD_FIREWALL_
					PROTO_UDP – User Datagram Protocol
					• 0x03 – QCMAP_EXTD_FIREWALL_
					PROTO_ICMP – Internet Control
					Message Protocol
					• 0x04 – QCMAP_EXTD_FIREWALL_
					PROTO_ICMP6 – Internet Control
					Message Protocol for IPv6
					• 0x05 – QCMAP_EXTD_FIREWALL_
					PROTO_ESP – Encapsulating Security
					Payload Protocol
					• 0x06 – QCMAP_EXTD_FIREWALL_
					PROTO_TCP_UDP – Transmission
					Control Protocol/User Datagram
				X	Protocol
Turne	0x11			COL S	TCP/UDP Source
Type	4			2	TCF/ODF Source
Length		uint16	nout.	V V	TCD/LIDD most as specified in the
Value	$\rightarrow$	umtro	port	2	TCP/UDP port as specified in the
			Ser. 25 - 2	6	TCP/UDP protocol (RFC 793 [S4] and
		:	(C) (C)	2	RFC 768 [S1]).
		uint16	range	2	TCP/UDP port range as specified in the TCP/UDP protocol (RFC 793 [S4] and
			20, 90,		RFC 768 [S1]).
_	0-10		440	1	
Туре	0x12			1	TCP/UDP Destination
Length	4	-: /16		2	TCD/IIDD
Value	$\rightarrow$	uint16	port	2	TCP/UDP port as specified in the
					TCP/UDP protocol (RFC 793 [S4] and
					RFC 768 [S1]).
		uint16	range	2	TCP/UDP port range as specified in the
					TCP/UDP protocol (RFC 793 [S4] and
					RFC 768 [S1]).
Туре	0x13			1	ICMP Type
Length	1			2	
Value	$\rightarrow$	uint8	icmp_type	1	ICMP type as specified in the ICMP
					specification (RFC 792 [S3]).
Туре	0x14			1	ICMP Code
Length	1			2	
Value	$\rightarrow$	uint8	icmp_code	1	ICMP code as specified in the ICMP
					specification (RFC 792 [S3]).
Туре	0x15			1	ESP Security Parameters Index
Length	4			2	-

Field	Field	Field	Parameter	Size	Description
	value	type		(byte)	
Value	$\rightarrow$	uint32	esp_spi	4	Security parameters index as specified in
					the ESP protocol (RFC 4303 [S7]).
Туре	0x16			1	IPv4 Source Address
Length	8			2	
Value	$\rightarrow$	uint32	addr	4	IPv4 address as specified in the IPv4
					protocol specification (RFC 791 [S2]).
		uint32	subnet_mask	4	IPv4 subnet mask as specified in the IPv4
					protocol specification (RFC 791 [S2]).
Туре	0x17			1	IPv4 Destination Address
Length	8			2	
Value	$\rightarrow$	uint32	addr	4	IPv4 address as specified in the IPv4
					protocol specification (RFC 791 [S2]).
		uint32	subnet_mask	4	IPv4 subnet mask as specified in the IPv4
					protocol specification (RFC 791 [S2]).
Туре	0x18			1	IPv4 TOS
Length	2			2	3
Value	$\rightarrow$	uint8	value	1	TOS value as specified in the IPv4
				/	protocol specification (RFC 791 [S2]).
		uint8	mask	101	IPv4 TOS mask.
Туре	0x19			OT O	IPv6 Source Address
Length	17		34	2	
Value	$\rightarrow$	uint8	addr	16	IPv6 address as specified in the IPv6
			10	(6)	protocol specification (RFC 2460 [S5]).
		uint8	prefix_len	§ 1	IPv6 prefix length as specified in the
			Side of the		IPv6 protocol addressing architecture
			000		specification (RFC 3513 [S6]).
Туре	0x1A		30,900	1	IPv6 Destination Address
Length	17		440	2	
Value	$\rightarrow$	uint8	addr	16	IPv6 address as specified in the IPv6
					protocol specification (RFC 2460 [S5]).
		uint8	prefix_len	1	IPv6 prefix length as specified in the
					IPv6 protocol addressing architecture
					specification (RFC 3513 [S6]).
Туре	0x1B			1	IPv6 Traffic Class
Length	2			2	
Value	$\rightarrow$	uint8	value	1	IPv6 traffic class value as specified in the
					IPv6 protocol specification (RFC 2460
					[S5]).
		uint8	mask	1	IPv6 traffic class mask.

#### **Error codes**

QMI_ERR_NONE	No error in request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_MALFORMED_MSG	Message was not formulated correctly by the control point
	or the message was corrupted during transmission
QMI_ERR_MISSING_ARG	Some TLV was missing
QMI_ERR_INVALID_HANDLE	Mobile AP handle provided in the request is not valid, i.e., it
	is not assigned to the control point
QMI_ERR_NOT_SUPPORTED	Operation is not supported

# 3.33.3 Description of QMI\_QCMAP\_GET\_EXTD\_FIREWALL\_CONFIG REQ/RESP

This command gets a firewall rule associated with a single firewall handle.

# 3.34 QMI\_QCMAP\_GET\_FIREWALL\_CONFIG\_HANDLE\_LIST

Gets the handles of all the firewall rules.

**QCMAP** message ID

0x0041

**Version introduced** 

Major - 1, Minor - 1

## 3.34.1 Request - QMI\_QCMAP\_GET\_FIREWALL\_CONFIG\_HANDLE\_-LIST\_REQ

Message type

Request

Sender

Control point

## **Mandatory TLVs**

	Name	3	Version introduced	Version last modified
Mobile AP handle	Let 22	000	1.1	1.1

Field	Field	Field	Parameter	Size	Description
	value	type	440	(byte)	
Туре	0x01			1	Mobile AP handle
Length	4			2	
Value	$\rightarrow$	uint32	mobile_ap_handle	4	Handle identifying the mobile AP call
					instance.
					The value must be the handle previously
					returned by QMI_QCMAP_MOBILE_
					AP_ENABLE_REQ.

**Optional TLVs** 

None

# 3.34.2 Response - QMI\_QCMAP\_GET\_FIREWALL\_CONFIG\_HANDLE\_-LIST\_RESP

Message type

Response

Sender

Service

## **Mandatory TLVs**

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

## **Optional TLVs**

Name	Version introduced	Version last modified
Firewall Handle List	1,10	1.1

Field	Field	Field	Parameter	Size	Description
	value	type		(byte)	
Туре	0x10		10	$\sqrt{3}$	Firewall Handle List
Length	Var			20	
Value	$\rightarrow$	uint8	firewall_handle_list_len	LOP.	Number of sets of the following
			10 No. 32 00	9	elements:
			THE OF THE		firewall_handle_list
		uint32	firewall_handle_list	Var	Firewall handle list.

#### **Error codes**

QMI_ERR_NONE	No error in request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_MALFORMED_MSG	Message was not formulated correctly by the control point
	or the message was corrupted during transmission
QMI_ERR_MISSING_ARG	Some TLV was missing
QMI_ERR_INVALID_HANDLE	Mobile AP handle provided in the request is not valid, i.e., it
	is not assigned to the control point
QMI_ERR_NOT_SUPPORTED	Operation is not supported

## 3.34.3 Description of QMI\_QCMAP\_GET\_FIREWALL\_CONFIG\_-HANDLE LIST REQ/RESP

This command gets all the firewall handles associated with a single mobile AP instance.



#### QMI\_QCMAP\_CHANGE\_NAT\_TYPE 3.35

Changes the currently existing NAT type.

**QCMAP** message ID

0x0042

**Version introduced** 

Major - 1, Minor - 3

#### Request - QMI\_QCMAP\_CHANGE\_NAT\_TYPE\_REQ 3.35.1

Message type

Request

Sender

Control point

## **Mandatory TLVs**

Name	Version introduced	Version last modified
Mobile AP handle	1.3	1.3

Field	Field	Field	Parameter	Size	Description
	value	type	168 10°	(byte)	
Туре	0x01		4400	1	Mobile AP handle
Length	4			2	
Value	$\rightarrow$	uint32	mobile_ap_handle	4	Handle identifying the mobile AP call
					instance.
					The value must be the handle previously
					returned by QMI_QCMAP_MOBILE_
					AP_ENABLE_REQ.

## **Optional TLVs**

Name	Version introduced	Version last modified
NAT Type Option	1.3	1.3

Field	Field	Field	Parameter	Size	Description
	value	type		(byte)	
Туре	0x10			1	NAT Type Option
Length	4			2	

Field	Field	Field	Parameter	Size	Description
	value	type		(byte)	
Value	$\rightarrow$	enum	nat_type_option	4	NAT type specified for the NAT type
					change. Values:
					• 0x00 – QCMAP_NAT_TYPE_
					SYMMETRIC – Symmetric NAT
					• 0x01 – QCMAP_NAT_TYPE_PORT_
					RESTRICTED_CONE – Port restricted
					cone NAT

#### 3.35.2 Response - QMI\_QCMAP\_CHANGE\_NAT\_TYPE\_RESP

Message type

Response

Sender

Service

#### **Mandatory TLVs**

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

**Optional TLVs** 

None

#### **Error codes**

QMI_ERR_NONE	No error in request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_MALFORMED_MSG	Message was not formulated correctly by the control point
	or the message was corrupted during transmission
QMI_ERR_MISSING_ARG	Some TLV was missing
QMI_ERR_INVALID_HANDLE	Mobile AP handle provided in the request is not valid, i.e., it
	is not assigned to the control point
QMI_ERR_NOT_SUPPORTED	Operation is not supported

#### Description of QMI\_QCMAP\_CHANGE\_NAT\_TYPE REQ/RESP 3.35.3

This command changes the NAT type associated with a mobile AP instance. When the NAT type is changed, the old NAT table data is cleaned and all existing connections are lost. If an IPv6 handle is passed, a QMI\_ERR\_INVALID\_HANDLE error is returned.

# 3.36 QMI\_QCMAP\_GET\_NAT\_TYPE

Gets the currently enabled NAT type.

**QCMAP** message ID

0x0043

**Version introduced** 

Major - 1, Minor - 3

## 3.36.1 Request - QMI\_QCMAP\_GET\_NAT\_TYPE\_REQ

Message type

Request

Sender

Control point

## **Mandatory TLVs**

Name	Version	n introduced	Version last modified
Mobile AP handle	9 9 67	1.3	1.3

Field	Field	Field	Parameter	Size	Description
	value	type	600,00	(byte)	
Туре	0x01		440	1	Mobile AP handle
Length	4			2	
Value	$\rightarrow$	uint32	mobile_ap_handle	4	Handle identifying the mobile AP call
					instance.
					The value must be the handle previously
					returned by QMI_QCMAP_MOBILE_
					AP_ENABLE_REQ.

## **Optional TLVs**

None

## 3.36.2 Response - QMI\_QCMAP\_GET\_NAT\_TYPE\_RESP

## Message type

Response

#### Sender

Service

## **Mandatory TLVs**

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

## **Optional TLVs**

Name	Version introduced	Version last modified
Current NAT Type	1.3	1.3

Field	Field	Field	Parameter	Size	Description
	value	type		(byte)	_
Туре	0x10			oT of	Current NAT Type
Length	4			2	
Value	$\rightarrow$	enum	nat_type_option	( 4 ° )	NAT type currently on the modem.
			3, 30,	(0)	Values:
				Sim	• 0x00 – QCMAP_NAT_TYPE_
			4.0° (c)		SYMMETRIC – Symmetric NAT
					• 0x01 – QCMAP_NAT_TYPE_PORT_
			20, 993		RESTRICTED_CONE – Port restricted
			40		cone NAT

## **Error codes**

QMI_ERR_NONE	No error in request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_MALFORMED_MSG	Message was not formulated correctly by the control point
	or the message was corrupted during transmission
QMI_ERR_MISSING_ARG	Some TLV was missing
QMI_ERR_INVALID_HANDLE	Mobile AP handle provided in the request is not valid, i.e., it
	is not assigned to the control point
QMI_ERR_NOT_SUPPORTED	Operation is not supported

# 3.36.3 Description of QMI\_QCMAP\_GET\_NAT\_TYPE REQ/RESP

This command gets the currently enabled NAT type associated with a mobile AP instance. If an IPv6 handle is passed, a QMI\_ERR\_INVALID\_HANDLE is returned.



# A Call End Reasons

# A.1 Call End Reasons

Table A-1 lists the call end reasons.

Table A-1 Call end reasons

Value	Name
1	QCMAP_CER_UNSPECIFIED
2	QCMAP_CER_CLIENT_END
3	QCMAP_CER_NO_SRV
4	QCMAP_CER_FADE
5	QCMAP_CER_REL_NORMAL
6	QCMAP_CER_ACC_IN_PROG
7	QCMAP_CER_ACC_FAIL
8	QCMAP_CER_REDIR_OR_HANDOFF
9	QCMAP_CER_CLOSE_IN_PROGRESS
10	QCMAP_CER_AUTH_FAILED
11	QCMAP_CER_INTERNAL_CALL_END
500	QCMAP_CER_CDMA_LOCK
501	QCMAP_CER_INTERCEPT
502	QCMAP_CER_REORDER
503	QCMAP_CER_REL_SO_REJ
504	QCMAP_CER_INCOM_CALL
505	QCMAP_CER_ALERT_STOP
506	QCMAP_CER_ACTIVATION
507	QCMAP_CER_MAX_ACCESS_PROBE
508	QCMAP_CER_CCS_NOT_SUPP_BY_BS
509	QCMAP_CER_NO_RESPONSE_FROM_BS
510	QCMAP_CER_REJECTED_BY_BS
511	QCMAP_CER_INCOMPATIBLE
512	QCMAP_CER_ALREADY_IN_TC
513	QCMAP_CER_USER_CALL_ORIG_DURING_GPS
514	QCMAP_CER_USER_CALL_ORIG_DURING_SMS
515	QCMAP_CER_NO_CDMA_SRV
1000	QCMAP_CER_CONF_FAILED
1001	QCMAP_CER_INCOM_REJ
1002	QCMAP_CER_NO_GW_SRV
1003	QCMAP_CER_NETWORK_END
1004	QCMAP_CER_LLC_SNDCP_FAILURE
1005	QCMAP_CER_INSUFFICIENT_RESOURCES

## Table A-1 Call end reasons (cont.)

Value	Name
1006	QCMAP_CER_OPTION_TEMP_OOO
1007	QCMAP_CER_NSAPI_ALREADY_USED
1008	QCMAP_CER_REGULAR_DEACTIVATION
1009	QCMAP_CER_NETWORK_FAILURE
1010	QCMAP_CER_UMTS_REATTACH_REQ
1011	QCMAP_CER_PROTOCOL_ERROR
1012	QCMAP_CER_OPERATOR_DETERMINED_BARRING x
1013	QCMAP_CER_UNKNOWN_APN
1014	QCMAP_CER_UNKNOWN_PDP
1015	QCMAP_CER_GGSN_REJECT
1016	QCMAP_CER_ACTIVATION_REJECT
1017	QCMAP_CER_OPTION_NOT_SUPP
1018	QCMAP_CER_OPTION_UNSUBSCRIBED
1019	QCMAP_CER_QOS_NOT_ACCEPTED
1020	QCMAP_CER_TFT_SEMANTIC_ERROR
1021	QCMAP_CER_TFT_SYNTAX_ERROR
1022	QCMAP_CER_UNKNOWN_PDP_CONTEXT
1023	QCMAP_CER_FILTER_SEMANTIC_ERROR
1024	QCMAP_CER_FILTER_SYNTAX_ERROR
1025	QCMAP_CER_PDP_WITHOUT_ACTIVE_TFT
1026	QCMAP_CER_INVALID_TRANSACTION_ID
1027	QCMAP_CER_MESSAGE_INCORRECT_SEMANTIC
1028	QCMAP_CER_INVALID_MANDATORY_INFO
1029	QCMAP_CER_MESSAGE_TYPE_UNSUPPORTED
1030	QCMAP_CER_MSG_TYPE_NONCOMPATIBLE_STATE
1031	QCMAP_CER_UNKNOWN_INFO_ELEMENT
1032	QCMAP_CER_CONDITIONAL_IE_ERROR
1033	QCMAP_CER_MSG_AND_PROTOCOL_STATE_UNCOMPATIBLE
1034	QCMAP_CER_APN_TYPE_CONFLICT
1035	QCMAP_CER_NO_GPRS_CONTEXT
1036	QCMAP_CER_FEATURE_NOT_SUPPORTED
1500	QCMAP_CER_CD_GEN_OR_BUSY
1501	QCMAP_CER_CD_BILL_OR_AUTH
1502	QCMAP_CER_CHG_HDR
1503	QCMAP_CER_EXIT_HDR
1504	QCMAP_CER_HDR_NO_SESSION
1505	QCMAP_CER_HDR_ORIG_DURING_GPS_FIX
1506	QCMAP_CER_HDR_CS_TIMEOUT
1507	QCMAP_CER_HDR_RELEASED_BY_CM

## A.2 Verbose Call End Reasons

Table A-2 lists the verbose call end reasons.

Table A-2 Verbose call end reasons

Value	Name
0	QCMAP_VCER_UNSPECIFIED
MIP	
65600	QCMAP_VCER_MIP_FA_REASON_UNSPECIFIED
65601	QCMAP_VCER_MIP_FA_ADMIN_PROHIBITED
65602	QCMAP_VCER_MIP_FA_INSUFFICIENT_RESOURCES
65603	QCMAP_VCER_MIP_FA_MOBILE_NODE_AUTH_FAILURE
65604	QCMAP_VCER_MIP_FA_HA_AUTH_FAILURE
65605	QCMAP_VCER_MIP_FA_REQ_LIFETIME_TOO_LONG
65606	QCMAP_VCER_MIP_FA_MALFORMED_REQUEST
65607	QCMAP_VCER_MIP_FA_MALFOMED_REPLY
65608	QCMAP_VCER_MIP_FA_ENCAPSULATION_UNAVAILABLE
65609	QCMAP_VCER_MIP_FA_VJHC_UNAVAILABLE
65610	QCMAP_VCER_MIP_FA_REV_TUNNEL_UNAVAILABLE
65611	QCMAP_VCER_MIP_FA_REV_TUNNEL_IS_MAND_AND_T_BIT_NOT_SET
65615	QCMAP_VCER_MIP_FA_DELIVERY_STYLE_NOT_SUPP
65633	QCMAP_VCER_MIP_FA_MISSING_NAI
65634	QCMAP_VCER_MIP_FA_MISSING_HA
65635	QCMAP_VCER_MIP_FA_MISSING_HOME_ADDR
65640	QCMAP_VCER_MIP_FA_UNKNOWN_CHALLENGE
65641	QCMAP_VCER_MIP_FA_MISSING_CHALLENGE
65642	QCMAP_VCER_MIP_FA_STALE_CHALLENGE
65664	QCMAP_VCER_MIP_HA_REASON_UNSPECIFIED
65665	QCMAP_VCER_MIP_HA_ADMIN_PROHIBITED
65666	QCMAP_VCER_MIP_HA_INSUFFICIENT_RESOURCES
65667	QCMAP_VCER_MIP_HA_MOBILE_NODE_AUTH_FAILURE
65668	QCMAP_VCER_MIP_HA_FA_AUTH_FAILURE
65669	QCMAP_VCER_MIP_HA_REGISTRATION_ID_MISMATCH
65670	QCMAP_VCER_MIP_HA_MALFORMED_REQUEST
65672	QCMAP_VCER_MIP_HA_UNKNOWN_HA_ADDR
65673	QCMAP_VCER_MIP_HA_REV_TUNNEL_UNAVAILABLE
65674	QCMAP_VCER_MIP_HA_REV_TUNNEL_IS_MAND_AND_T_BIT_NOT_SET
65675	QCMAP_VCER_MIP_HA_ENCAPSULATION_UNAVAILABLE
131071	QCMAP_VCER_MIP_HA_REASON_UNKNOWN
Internal	
131273	QCMAP_VCER_INTERNAL_INTERNAL_ERROR
131274	QCMAP_VCER_INTERNAL_CALL_ENDED
131275	QCMAP_VCER_INTERNAL_INTERNAL_UNKNOWN_CAUSE_CODE
131276	QCMAP_VCER_INTERNAL_UNKNOWN_CAUSE_CODE
131277	QCMAP_VCER_INTERNAL_CLOSE_IN_PROGRESS
131278	QCMAP_VCER_INTERNAL_NW_INITIATED_TERMINATION
131279	QCMAP_VCER_INTERNAL_APP_PREEMPTED

## Table A-2 Verbose call end reasons (cont.)

Value	Name
Call manage	r
197108	QCMAP_VCER_CM_CDMA_LOCK
197109	QCMAP_VCER_CM_INTERCEPT
197110	QCMAP_VCER_CM_REORDER
197111	QCMAP_VCER_CM_REL_SO_REJ
197112	QCMAP_VCER_CM_INCOM_CALL
197113	QCMAP_VCER_CM_ALERT_STOP
197114	QCMAP_VCER_CM_ACTIVATION
197115	QCMAP_VCER_CM_MAX_ACCESS_PROBE
197116	QCMAP_VCER_CM_CCS_NOT_SUPP_BY_BS
197117	QCMAP_VCER_CM_NO_RESPONSE_FROM_BS
197118	QCMAP_VCER_CM_REJECTED_BY_BS
197119	QCMAP_VCER_CM_INCOMPATIBLE
197120	QCMAP_VCER_CM_ALREADY_IN_TC
197121	QCMAP_VCER_CM_USER_CALL_ORIG_DURING_GPS
197122	QCMAP_VCER_CM_USER_CALL_ORIG_DURING_SMS
197123	QCMAP_VCER_CM_NO_CDMA_SRV
197127	QCMAP_VCER_CM_RETRY_ORDER
197608	QCMAP_VCER_CM_CONF_FAILED
197609	QCMAP_VCER_CM_INCOM_REJ
197616	QCMAP_VCER_CM_NO_GW_SERV
197617	QCMAP_VCER_CM_NO_GPRS_CONTEXT
197618	QCMAP_VCER_CM_ILLEGAL_MS
197619	QCMAP_VCER_CM_ILLEGAL_ME
197620	QCMAP_VCER_CM_GPRS_SERV_AND_NON_GPRS_SERV_NOT_ALLOWED
197621	QCMAP_VCER_CM_GPRS_SERV_NOT_ALLOWED
197622	QCMAP_VCER_CM_MS_IDENTITY_CANNOT_BE_DERIVED_BY_THE_
	NETWORK
197623	QCMAP_VCER_CM_IMPLICITLY_DETACHED
197624	QCMAP_VCER_CM_PLMN_NOT_ALLOWED
197625	QCMAP_VCER_CM_LA_NOT_ALLOWED
197626	QCMAP_VCER_CM_GPRS_SERV_NOT_ALLOWED_IN_THIS_PLMN
197627	QCMAP_VCER_CM_PDP_DUPLICATE
197628	QCMAP_VCER_CM_UE_RAT_CHANGE
197629	QCMAP_VCER_CM_CONGESTION
197630	QCMAP_VCER_CM_NO_PDP_CONTEXT_ACTIVATED
197631	QCMAP_VCER_CM_ACCESS_CLASS_DSAC_REJECTION
198108	QCMAP_VCER_CM_CD_GEN_OR_BUSY
198109	QCMAP_VCER_CM_CD_BILL_OR_AUTH
198110	QCMAP_VCER_CM_CHG_HDR
198111	QCMAP_VCER_CM_EXIT_HDR
198112	QCMAP_VCER_CM_HDR_NO_SESSION
198113	QCMAP_VCER_CM_HDR_ORIG_DURING_GPS_FIX
198114	QCMAP_VCER_CM_HDR_CS_TIMEOUT
198115	QCMAP_VCER_CM_HDR_RELEASED_BY_CM
198118	QCMAP_VCER_CM_NO_HYBR_HDR_SRV

## Table A-2 Verbose call end reasons (cont.)

Value	Name		
198608	QCMAP_VCER_CM_CLIENT_END		
198609	QCMAP_VCER_CM_NO_SRV		
198610	QCMAP_VCER_CM_FADE		
198611	QCMAP_VCER_CM_REL_NORMAL		
198612	QCMAP_VCER_CM_ACC_IN_PROG		
198613	QCMAP_VCER_CM_ACC_FAIL		
198614	QCMAP_VCER_CM_REDIR_OR_HANDOFF		
3GPP speci	3GPP specification		
393224	QCMAP_VCER_3GPP_OPERATOR_DETERMINED_BARRING		
393241	QCMAP_VCER_3GPP_LLC_SNDCP_FAILURE		
393242	QCMAP_VCER_3GPP_INSUFFICIENT_RESOURCES		
393243	QCMAP_VCER_3GPP_UNKNOWN_APN		
393244	QCMAP_VCER_3GPP_UNKNOWN_PDP		
393245	QCMAP_VCER_3GPP_AUTH_FAILED		
393246	QCMAP_VCER_3GPP_GGSN_REJECT		
393247	QCMAP_VCER_3GPP_ACTIVATION_REJECT		
393248	QCMAP_VCER_3GPP_OPTION_NOT_SUPPORTED		
393249	QCMAP_VCER_3GPP_OPTION_UNSUBSCRIBED		
393250	QCMAP_VCER_3GPP_OPTION_TEMP_OOO		
393251	QCMAP_VCER_3GPP_NSAPI_ALREADY_USED		
393252	QCMAP_VCER_3GPP_REGULAR_DEACTIVATION		
393253	QCMAP_VCER_3GPP_QOS_NOT_ACCEPTED		
393254	QCMAP_VCER_3GPP_NETWORK_FAILURE		
393255	QCMAP_VCER_3GPP_UMTS_REACTIVATION_REQ		
393256	QCMAP_VCER_3GPP_FEATURE_NOT_SUPP		
393257	QCMAP_VCER_3GPP_TFT_SEMANTIC_ERROR		
393258	QCMAP_VCER_3GPP_TFT_SYTAX_ERROR		
393259	QCMAP_VCER_3GPP_UNKNOWN_PDP_CONTEXT		
393260	QCMAP_VCER_3GPP_FILTER_SEMANTIC_ERROR		
393261	QCMAP_VCER_3GPP_FILTER_SYTAX_ERROR		
393262	QCMAP_VCER_3GPP_PDP_WITHOUT_ACTIVE_TFT		
393297	QCMAP_VCER_3GPP_INVALID_TRANSACTION_ID		
393311	QCMAP_VCER_3GPP_MESSAGE_INCORRECT_SEMANTIC		
393312	QCMAP_VCER_3GPP_INVALID_MANDATORY_INFO		
393313	QCMAP_VCER_3GPP_MESSAGE_TYPE_UNSUPPORTED		
393314	QCMAP_VCER_3GPP_MSG_TYPE_NONCOMPATIBLE_STATE		
393315	QCMAP_VCER_3GPP_UNKNOWN_INFO_ELEMENT		
393316	QCMAP_VCER_3GPP_CONDITIONAL_IE_ERROR		
393317	QCMAP_VCER_3GPP_MSG_AND_PROTOCOL_STATE_UNCOMPATIBLE		
393327	QCMAP_VCER_3GPP_PROTOCOL_ERROR		
393328	QCMAP_VCER_3GPP_APN_TYPE_CONFLICT		

## Table A-2 Verbose call end reasons (cont.)

Value	Name	
Point-to-Point Protocol		
458753	QCMAP_VCER_PPP_TIMEOUT	
458754	QCMAP_VCER_PPP_AUTH_FAILURE	
458755	QCMAP_VCER_PPP_OPTION_MISMATCH	
458783	QCMAP_VCER_PPP_PAP_FAILURE	
458784	QCMAP_VCER_PPP_CHAP_FAILURE	
524287	QCMAP_VCER_PPP_UNKNOWN	
eHRPD		
524289	QCMAP_VCER_EHRPD_SUBS_LIMITED_TO_V4	
524290	QCMAP_VCER_EHRPD_SUBS_LIMITED_TO_V6	
524292	QCMAP_VCER_EHRPD_VSNCP_TIMEOUT	
524293	QCMAP_VCER_EHRPD_VSNCP_FAILURE	
524294	QCMAP_VCER_EHRPD_VSNCP_3GPP2I_GEN_ERROR	
524295	QCMAP_VCER_EHRPD_VSNCP_3GPP2I_UNAUTH_APN	
524296	QCMAP_VCER_EHRPD_VSNCP_3GPP2I_PDN_LIMIT_EXCEED	
524297	QCMAP_VCER_EHRPD_VSNCP_3GPP2I_NO_PDN_GW	
524298	QCMAP_VCER_EHRPD_VSNCP_3GPP2I_PDN_GW_UNREACH	
524299	QCMAP_VCER_EHRPD_VSNCP_3GPP2I_PDN_GW_REJ	
524300	QCMAP_VCER_EHRPD_VSNCP_3GPP2I_INSUFF_PARAM	
524301	QCMAP_VCER_EHRPD_VSNCP_3GPP2I_RESOURCE_UNAVAIL	
524302	QCMAP_VCER_EHRPD_VSNCP_3GPP2I_ADMIN_PROHIBIT	
524303	QCMAP_VCER_EHRPD_VSNCP_3GPP2I_PDN_ID_IN_USE	
524304	QCMAP_VCER_EHRPD_VSNCP_3GPP2I_SUBSCR_LIMITATION	
524305	QCMAP_VCER_EHRPD_VSNCP_3GPP2I_PDN_EXISTS_FOR_THIS_APN	
IPv6	65 22 113	
589825	QCMAP_VCER_IPV6_PREFIX_UNAVAILABLE	