

*Browser/OTA Provisioning  
Whitepaper*

Version 1.0



# Table of Contents

<b>TABLE OF CONTENTS .....</b>	<b>2</b>
<b>1 INTRODUCTION .....</b>	<b>3</b>
SUMMARY .....	3
REFERENCES .....	3
REVISION HISTORY .....	4
DEFINITIONS, ABBREVIATIONS, ACRONYMS .....	4
<b>2 SUPPORTED WAP OTA PARAMETERS .....</b>	<b>6</b>
<b>3 OTA PROVISIONING SECURITY MECHANISM - AUTHENTICATION .....</b>	<b>7</b>
USER HANDSET EXPERIENCE.....	8
NETWPIN AND USERNETWPIN AUTHENTICATION .....	10
<b>4 OTA PROVISIONING OF MMS AND WEB SESSION PARAMETERS .....</b>	<b>12</b>
CODE SAMPLES .....	13
VALUES PARSED BY THE CLIENT.....	14
DELETION OF PROVISIONED SESSION BY OPERATOR.....	14
OPERATING CONSTRAINTS .....	15
ERROR CONDITIONS .....	16
<b>5 WAP PROVISIONING DATA FORMAT .....</b>	<b>17</b>
MEDIA TYPE PARAMETER .....	18
STRING IN TEXT FORMAT .....	18
STRING IN TOKEN FORMAT .....	19
<i>Headers (Optional Fields)</i> .....	19
<i>Provisioning Document</i> .....	20
<i>Adapting to GSM SMS Format</i> .....	23
<i>Examples</i> .....	25
<b>APPENDIX A: PARAMETER MAPPING.....</b>	<b>28</b>
<b>APPENDIX B: COMPLIANCY MATRIX.....</b>	<b>29</b>

# 1 Introduction

## Summary

---

Over-the-air provisioning describes the ability to download and install content over a wireless network, usually on demand. This whitepaper will give a detailed explanation of the following:

- Supported WAP OTA Parameters
- OTA Provisioning Security Mechanism
- OTA Provisioning of MMS and Web Session Parameters
- WAP Provisioning Data Format
- Parameter Mapping Appendix
- Compliancy Matrix Appendix

For more high-level information on OTAP, consult the following reference:

<http://www.openmobilealliance.org/documents.html>

## References

---

- OMA Client Provisioning Version 1.1 - [http://www.openmobilealliance.org/omacopyrightNEW.asp?doc=OMA\\_Client\\_Provisioning\\_version1\\_1.zip](http://www.openmobilealliance.org/omacopyrightNEW.asp?doc=OMA_Client_Provisioning_version1_1.zip)
- Wireless Session Protocol Specification, WAP-230-WSP-20010705-a
- Digital cellular telecommunications system (Phase 2+); Technical realization of the Short Message Service (SMS), Point-to-Point (PP).\
- Provisioning Content Version 1.1, Open Mobile Alliance, OMA-WAP-ProvCont-v1\_1-20021112-C – <http://www.openmobilealliance.org>

- Provisioning Bootstrap Version 1.1, Open Mobile Alliance, OMA-WAP-ProvBoot-v1\_1-200211120-C – <http://www.openmobilealliance.org>
- Digital Cellular Telecommunication System (Phase 2+), Specification of the Subscriber Identity Module – Mobile Equipment (SIM-ME) interface (GSM11.11 version 7.2.0 Release 1998)
- Secure Hash Standard, NIST FIPS PUB 180-1, National Institute of Standards and Technology, U.S. Department of Commerce, April 1995.
- HMAC: Keyed-Hashing for Message Authentication, Krawczyk, H., Bellare, M., and Canetti, R., February 1997 – <http://www.ietf.org/rfc/rfc2104.txt>
- 3<sup>rd</sup> Generation Partnership Project, Technical Specification Group Core Network; Mobile radio interface layer 3 specification, Core Network Protocols – Stage 3 (Release 4), v 4.5.0 (2001-12)

## Revision History

---

Version	Date	Name	Reason
0.1	October 20, 2003	Adam Grabowski (aag101)	Initial Draft
1.0	November 3, 2003	Adam Grabowski (aag101)	Baseline

## Definitions, Abbreviations, Acronyms

---

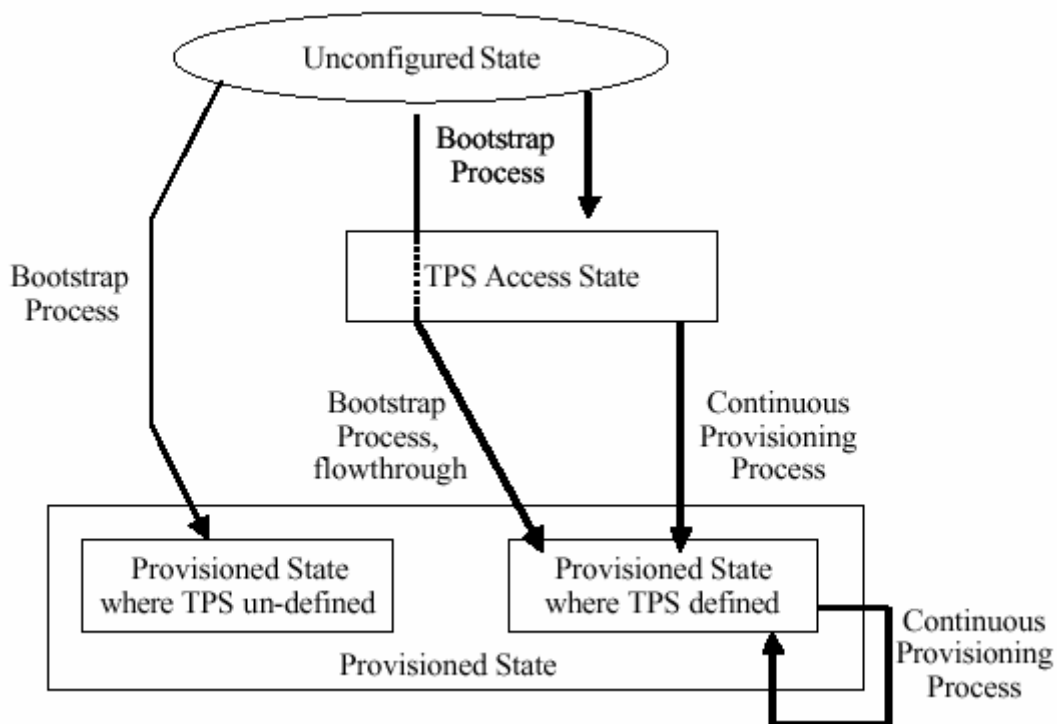
Acronym	Description
OTA	Over-the-Air
IMSI	International Mobile Subscriber Identity
SEC	Security
MAC	Message Authentication Code
SIM	Subscriber Information Module

WBXML	Wireless Binary Extensible Markup Language
TID	Transaction Identifier
OMA	Open Mobile Alliance
MT	Mobile Terminated

# 2

## Supported WAP OTA Parameters

The WAP provisioning framework specifies mechanisms to take a terminal from an unconfigured to a fully configured state. Refer to the process diagram below.



Motorola will support the Bootstrap Process to take a handset from the “Unconfigured State” directly to the “Provisioned State where TPS is undefined” state. Continuous provisioning will not be supported and therefore, it is imperative that security parameters (such as a user pin) are set to prevent unauthorized provisioning documents from configuring the phone.

# 3

## OTA Provisioning Security Mechanism – Authentication

Whether authentication is required and the method in which it is employed is determined by the gateway sending the provisioning message. This message contains an optional parameter ('sec method') which determines which authentication method is used.

SEC Value	Authentication Method
Not present	No authentication
(0) NETWPIN	Authentication Method #1
(1) USERPIN	Authentication Method #2
(2) USERNETWPIN	Authentication Method #2
(3) USERPINMAC	Authentication Method #2

The following will be used to define the SEC value for authentication methods:

- USERPIN – Shared secret between the server and client . User enters the value from handset editor prompted upon receiving the provisioning message.
- NETWPIN -- Shared secret between the server and client (IMSI). The shared secret is that the IMSI information can be obtained from the SIM card and then subsequently used in calculating a MAC value that is then compared to the one sent along with the provisioning document.
- USERNETWPIN -- Shared secret between the server and client. The shared secret is NETWPIN appended with USERPIN.

- USERPINMAC -- MAC authentication of document, and USERPIN as shared Secret.

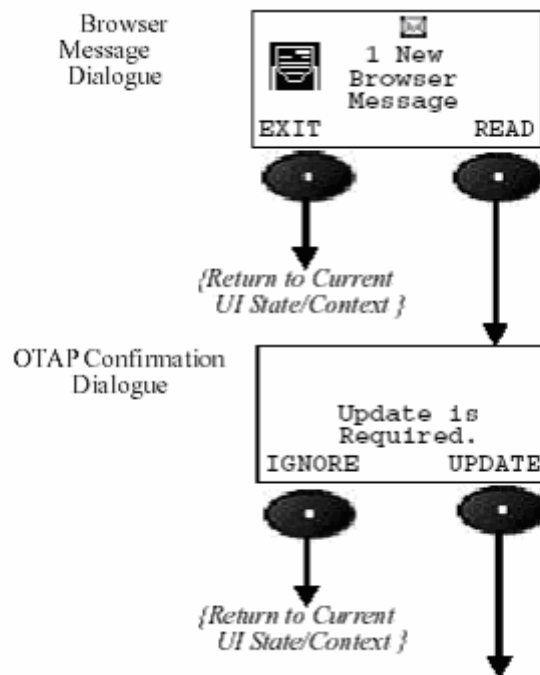
The following will be used to define the Authentication Method:

- Authentication Method # 1 -- This bootstrap message authentication is performed automatically and does not require user enter any PIN.
- Authentication Method # 2 -- This message authentication method requires the mobile handset to create a “Secure Number” Editor prompting the user to enter an authentication PIN (given to the user via some “out-of-band” method).

## User Handset Experience

---

The user will always see the following two screens upon receiving an OTAP message:

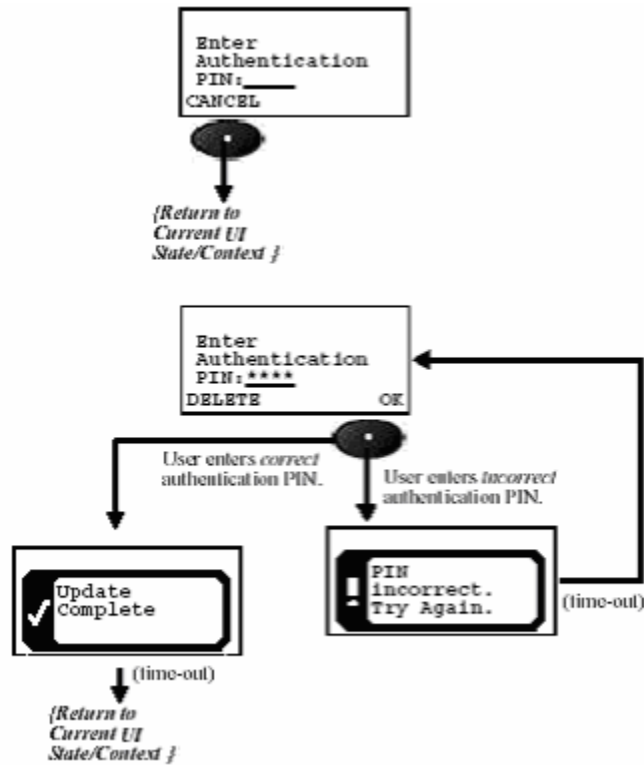


The following are the four possible experiences that may exist for the user:

- Valid Digest in case of NETWPIN



- Valid Digest in case of USERPIN or USERNETWPIN



- Invalid Digest



- Unauthenticated Message



⋮ (time-out)

{Return to  
Current UI  
State/Context}

## NETWPIN and USERNETWPIN Authentication

---

Motorola implementation is based on WAP specification in the Provisioning Bootstrap reference. When NETWPIN or USERNETWPIN is used, the IMSI is used as the network specific shared secret. When this authentication is used as the input to the MAC calculation, the IMSI is on semi-octet representation as defined in the GSM11.11 reference. The following are examples of implementation on a Motorola handset:

### 1. IMSI contains 15 digits (normally)

For example, assume IMSI number is 310170212226432.

The semi-octet representation will be the following:

0011 ?001	//0x31 or 0x39
0000 0001	// 0x01
0111 0001	// 0x71
0010 0000	// 0x20
0010 0001	// 0x21
0010 0010	// 0x22
0100 0110	// 0x46
0010 0011	// 0x23

Here ? represents “parity bit” [3GPP 24.008], in this case since there is 15 digits in the IMSI, an odd number, so the parity bit is “1”, therefore the first byte is 0x39.

In hex format, it will be 39 01 71 20 21 22 46 23, this will be the value used as the key (together with the WBXML encoded document as the data input) for the HMAC calculation [HMAC], based on the SHA-1 algorithm [SHA], in the case of NETWPIN. The output of the HMAC calculation (20 bytes) is further encoded as a string of 80 hexadecimal digits where each pair of consecutive digits represent a byte, resulting in a 40 bytes MAC value.

In case of USERNETWPIN, for example, the PIN is 1234, the value used as the key for the MAC calculation will be 39 01 71 20 21 22 46 23 31 32 33 34.

### 2. IMSI contains less than 15 digits (unlikely)

For example, assume IMSI number is 310170212226, it will be padding with FFF. The semi-octet representation will be the following:

```
0011 ?001          //0x31 or 0x39
0000 0001    // 0x01
0111 0001    // 0x71
0010 0000    // 0x20
0010 0001    // 0x21
0010 0010    // 0x22
1111 0110    // 0xF6
1111 1111    // 0xFF
```

Here ? represents “parity bit” [3GPP 24.008], in this case since there is 12 digits in the IMSI, an even number, so the parity bit is “0”, therefore the first byte is 0x31.

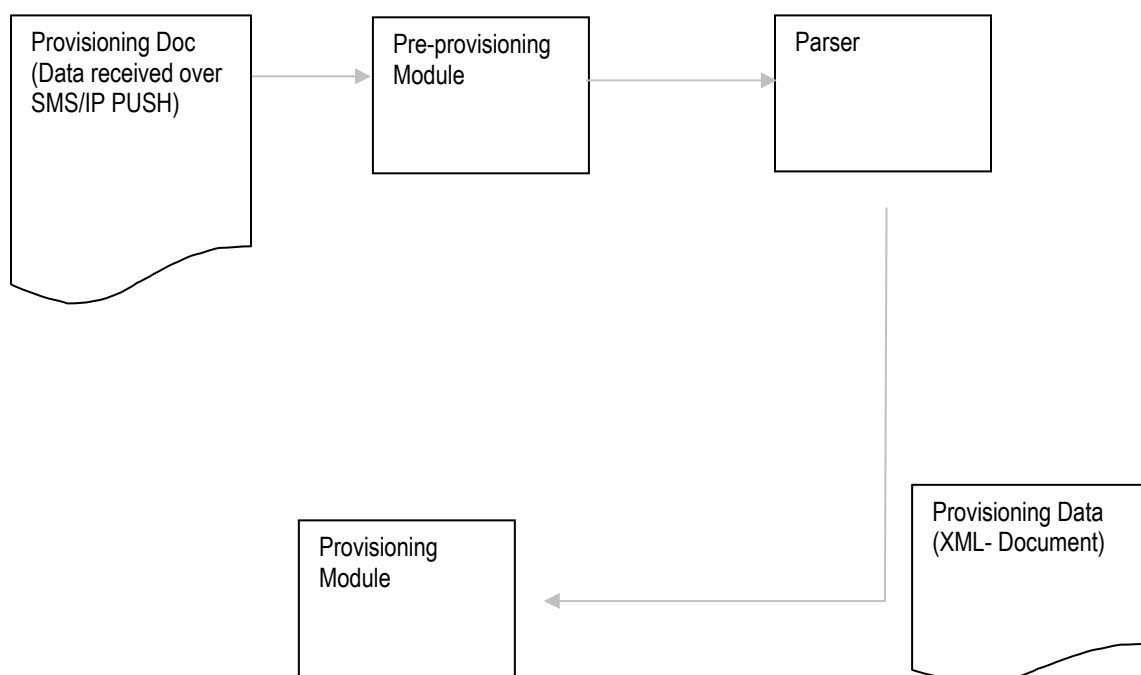
In hex format, it will be 31 01 71 20 21 22 F6 FF, but the last 0xFF is **NOT** used in the MAC calculation. Only 31 01 71 20 21 22 F6 will be the value used as the key for the MAC calculation in the case of NETWPIN.

In the case of USERNETWPIN, for example, the PIN is 1234, the value used as the key for the MAC calculation will be 31 01 71 20 21 22 F6 31 32 33 34.

# 4

## OTA Provisioning of MMS and Web Session Parameters

The existing architecture (see figure below) requires the parser to parse the received provisioning document into an XML document, allowing the provisioning module to update the corresponding NVM parameters in the phone, currently web session parameters.



To allow MMS parameters to be provisioned over the air, there will be some changes required to existing modules. The changes include the following:

- Enhanced parser to understand the MMS Application specific data.
- Enhanced provisioning module to configure the corresponding NVM parameters (feature Ids). For example, MMS Server Info.
- Enhanced provisioning module to support operator OTA deletion.

Apart from the above enhancements, Motorola will ensure conformance to the TD242 document. The TD 242 document contains a new characteristic called "APPLICATION" which defines the access parameters for a particular application. Motorola will support specific parameters from this characteristic that are required for MMS provisioning support.

## Code Samples

Below is an example of a WML document for receiving the MMS Specific parameters. In this example, the MMS server information and the connection parameters (Proxy Information and APN) are provided. An extra characteristic (APPLICATION) is added for the MMS Server Information (APP ID="w4") to the original WAP provisioning document. The assumption is that all information received will be only used for MMS.

```
<?xml version="1.0"?>
<!DOCTYPE wap-provisioningdoc PUBLIC "-//WAPFORUM//DTD PROV 1.0//EN"
"http://www.wapforum.org/DTD/prov.dtd">
<wap-provisioningdoc>
<characteristic type="BOOTSTRAP">
  <parm name="PROXY-ID" value="sdp.xyz.com"/>
  <parm name="NAME" value="xyz"/>
</characteristic>
<characteristic type="PXLOGICAL">
  <parm name="PROXY-ID" value="sdp.xyz.com"/>
  <parm name="NAME" value="MOT MMS"/>
  <parm name="STARTPAGE" value="http://xyz.co.uk/mobile"/>
  <characteristic type="PXPHYSICAL">
    <parm name="PHYSICAL-PROXY-ID" value="Gateway 1"/>
    <parm name="PXADDR" value="217.171.129.2"/>
    <parm name="PXADDRTYPE" value="IPV4"/>
    <parm name="TO-NAPID" value="NAP 1"/>
    <characteristic type="PORT"
      <parm name="PORTNBR" value="8799"/>
    </characteristic>
  </characteristic>
</characteristic>
<characteristic type="NAPDEF">
  <parm name="NAPID" value="NAP 1"/>
  <parm name="BEARER" value="GSM-GPRS"/>
  <parm name="NAME" value="XYZ GPRS"/>
  <parm name="NAP-ADDRESS" value="xyz.co.uk"/>
  <parm name="NAP-ADDRTYPE" value="APN"/>
</characteristic>
<characteristic type="APPLICATION">
  <parm name="APPID" value="w4"/>
  <parm name="NAME" value="MOT MMS"/>
  <parm name="ADDR" value="http://mms.um.xyz.co.uk:10021/mmsc"/>
</characteristic>
</wap-provisioningdoc>
```

## Values parsed by the client

---

The following describes the values parsed by the client with regards to the example above:

- Create a web session with name "MOT MMS" with the following parameters:
  - GW IP1: "Gateway IP 1:"
  - Port: "Port 1:"
  - APN: "GPRS APN:"
  - Server Address: "Server Name:"
- Create new MMS Server information with name "MOT MMS" with the Server Address <http://mms.um.xyz.co.uk:10021/mmsc>

The Motorola client will always use the same name (as received in the application characteristics) for MMSC and the corresponding web session entry. This way, Motorola can extend support to multiple servers and tie them to corresponding connection parameters. Note the "PROXY-ID" (shown as "sdp.xyz.com") under BOOTSTRAP and PXLOGICAL needs to be the same.

The document will still be processed if the APPLICATION characteristic is missing and the web session will be stored (if new) or updated (if exists) as usual, and the MMS server name will be updated accordingly (if exists).

Each OTA provisioned Web Session is uniquely identified by its TPS Name. The TPS Name is the NAME parameter that is sent in the BOOTSTRAP characteristic of a provisioning document. The operator can control the provisioned parameters by using the same bootstrap name in subsequent provisioning documents. If the BOOTSTRAP characteristic is missing, and the provisioning document is valid, then this is treated as a document with an empty TPS name. If a Web Session with the given TPS Name does not exist, a unique OTA Web Session will be created corresponding to that TPS Name. If a Web Session with the given TPS Name already exists, the data for this Web Session will be overwritten with the information from the received provisioning document. The BOOTSTRAP information is needed if the operator wants to update or delete the session (web and MMS) later.

## Deletion of Provisioned Session by Operator

---

To delete a session by service provider, the OTAP provider needs to send a session with the correct TPS Name (the NAME field in the BOOTSTRAP) and leave the body blank. The following is an example of an operator deleting the session provisioned previously according to the document above.

```
<?xml version="1.0"?>
<!DOCTYPE wap-provisioningdoc PUBLIC "-//WAPFORUM//DTD PROV 1.0//EN"
"http://www.wapforum.org/DTD/prov.dtd">
<wap-provisioningdoc>
<characteristic type="BOOTSTRAP">
```

```
<parm name="NAME" value="xyz"/>
</characteristic>
</wap-provisioningdoc>
```

The "NAME" must exactly match the one previously provisioned (example above). The previously provisioned session (web session and MMS session) matched with the current TPS "NAME" will be deleted. If this deletion is successful, the phone will display "Update Complete." If the deletion fails for any reason (e.g. the session does not exist), the phone will display "Update Failed."

If the original provisioning document does not include the BOOTSTRAP info, the operator will be able to delete the session by sending a provisioning document containing only the BOOTSTRAP characteristic with the NAME parameter set to value "" (empty string).

## Operating Constraints

---

The following is a list of operating implementation constraints:

- Upon receipt of a MMS provisioning document, a new entry will be created for MMSC and Web Session.
- The same name will be used for MMSC and Web Session.
- No two web sessions can have the same TPS. If an entry exists with the same TPS name, it will be overwritten.
- A maximum of 3 MMSC settings can be stored at anytime in the phone (Flexed & Provisioned).
- The last provisioned entry will be the selected MMSC.
- The user will have the option of selecting a MMSC from the available list. The corresponding web session will be used in conjunction with the MMSC.
- The MMSC information can be edited if the "MMS\_SERVER\_INFO\_EDIT\_AVAILABLE" field is set; similarly, it can be deleted if the "MMS\_SERVER\_INFO\_DELETE\_AVAILABLE" field is set.
- The user is able to delete provisioned web session entries if the "OTA\_SESSION\_DELETE\_AVAILABLE" feature id is set.
- The OTA provisioned web session is read only (i.e. the user can not edit). To edit the content, the user must make a copy and edit the copied version.
- The OTA web session (and the corresponding MMSC session if applicable) can be deleted/updated by the service provider. However, it can only overwrite existing OTAP sessions.
- If a web session is deleted by the user or by OTA, an event will be sent from Web Session SP to MMS. If MMS has a session with the same name, then it will delete it. The reverse situation (i.e., the user deletes a MMS session) will have no affect on the web sessions since the web session can be in use for GPRS "Always-on" feature, Browser, or KJava Networking.

- The user is not allowed to name a session with an already existing name of OTA type. If the handset receives an OTAP session with a name that already exists, the name will be appended with a number "1, 2, 3, etc" to identify it uniquely. If the name is already at max length, the last letter will be cut off in order to append the number "1...9".

## Error Conditions

---

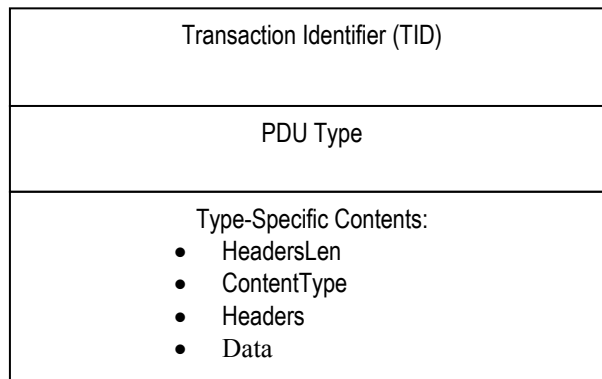
If any error occurs during provisioning, the user will be notified. The following is a list of possible error conditions:

- The provisioning document is incomplete or erroneous.
- The delete operation fails (e.g. the session has been deleted by the user before).
- No space for creating a new entry is available. A total of 10 web sessions are available, 3 of which can be MMSC entries.

# 5

# WAP Provisioning Data Format

This section describes WAP provisioning information sent using the WAP Push protocol over the GSM SMS bearer. WAP WSP defines the generic Protocol Data Unit (PDU) shown below:



- TID: uint8 will be present for connectionless WSP PDUs according to the WAP WSP specification. The value of 0x01 will be used in this example.
- PDU Type: uint8 will have the value of 0x06 for PUSH (assigned in WAP WSP).
- Type-Specific Contents: provisioning document that must include HeadersLen, Content Type, Headers, Data, as defined below.
  - HeadersLen – value of 0x5A will be used for our first example which contains no headers, just Media-Type parameter. If the HeadersLen does contain the headers, its length will be accounted for here as well.
  - ContentType – media type parameter
  - Headers – optional parameter
  - Data – actual provisioning document

# Media Type Parameter

---

According to the WAP Provisioning Content specification, security information is “transported as parameters to the media type *application/vnd.wap.connectivity-wbxml*.” The overall media-type parameter is formatted as follows:

- Mime-type; SEC=x; MAC=...
  - Mime-type: this will be *application/vnd.wap.connectivity-wbxml*
  - SEC: security mechanism. In this example, use the value 0x31 ‘1’ – USERPIN
  - MAC: this is calculated using the WBXML document as the data and the user pin as the key for the HMAC calculation based on the SHA-1 algorithm. From the WAP Provisioning Bootstrap specification, “The output of the HMAC (M=HMAC-SHA(K,A)) calculation is encoded as a string of hexadecimal digits where each pair of consecutive digits represents a byte . . . This calculation is repeated in the ME when checking the validity of the MAC.”

# String in Text Format

---

The following yields the string in text format (to be used in example):

```
application/vnd.wap.connectivity-wbxml;MAC=...;SEC=1
```

For this example, the User pin is “12345678”

## Hexadecimal dump:

```
# application/vnd.wap.connectivity-wbxml
61 70 70 6C 69 63 61 74 69 6F 6E 2F 76 6E 64 2E
77 61 70 2E 63 6F 6E 6E 65 63 74 69 76 69 74 79
2D 77 62 78 6D 6C

# ;
3B

# MAC=
4D 41 43 3D

# SHA1 HASH: A4F0 23B8 82F1 F70C D8E2 C17C A398 64CF 9681 7B42
41 34 46 30 32 33 42 38 38 32 46 31 46 37 30 43
44 38 45 32 43 31 37 43 41 33 39 38 36 34 43 46
39 36 38 31 37 42 34 32

# ;
3B
```

```
# SEC=1
53 45 43 3D 31

# NULL (End of string)
00
```

## String in Token Format

---

As an alternative, one could use token-format for the media-type parameter to save bytes. Since the Mime-type, SEC, and MAC are well known, they can be encoded as a short integer using the corresponding known assigned numbers.

The assigned number for application/vnd.wap.connectivity-wbxml is 0x36. This number will become b6 after short-integer encoding.

The assigned number for SEC is 0x11, it will become 0x91 after short-integer encoding. See reference [Wireless Specification Protocol] table 38. In this case, USERPIN (1) will be encoded as 0x81.

The assigned number for MAC is 0x12, it will become 0x92 after short-integer encoding. See reference [Wireless Specification Protocol] table 38.

The MAC value (40 bytes) remains the same.

### Hexadecimal dump of this will be:

```
# application/vnd.wap.connectivity-wbxml
b6

# SEC = 1
91 81

# MAC
92

# SHA1 HASH: A4F0 23B8 82F1 F70C D8E2 C17C A398 64CF 9681 7B42
same as before

# NULL (End of string for the encoded MAC value)
00
```

This token-format string will be demonstrated later in example #2.

## Headers (Optional Fields)

---

The operator may add headers following content type as part of Media-type parameters. In example #2, only one field, called "from" field (assigned number 0x15, and therefore

short-integer encoded as 0x95) which is used to specify sender's email address is included. If the sender's email is [abc@mot.com](mailto:abc@mot.com), the Hexadecimal dump will be the following:

```
# From
95

# abc@mot.com
61 62 63 40 6D 6F 74 2E 63 6F 6D

# NULL (End of string)
00
```

## Provisioning Document

---

In this provisioning example, the following WML document is used:

```
<?xml version="1.0"?>
<!DOCTYPE wap-provisioningdoc PUBLIC "-//WAPFORUM//DTD PROV 1.0//EN"
"http://www.wapforum.org/DTD/prov.dtd">

<wap-provisioningdoc>
<characteristic type="PXLOGICAL">
  <parm name="PROXY-ID" value="sdp.our.com"/>
  <parm name="NAME" value="HelloWorld"/>
  <parm name="STARTPAGE"
value="http://sdp.our.com/IUT/homepage.htm"/>
  <characteristic type="PXPHYSICAL">
    <parm name="PHYSICAL-PROXY-ID" value="Gateway 1"/>
    <parm name="PXADDR" value="111.31.222.46"/>
    <parm name="PXADDRTYPE" value="IPV4"/>
    <parm name="TO-NAPID" value="NAP 1"/>
    <characteristic type="PORT">
      <parm name="PORTNBR" value="80"/>
    </characteristic>
  </characteristic>
  <characteristic type="PXPHYSICAL">
    <parm name="PHYSICAL-PROXY-ID" value="Gateway 2"/>
    <parm name="PXADDR" value="222.31.111.46"/>
    <parm name="PXADDRTYPE" value="IPV4"/>
    <parm name="TO-NAPID" value="NAP 1"/>
    <characteristic type="PORT">
      <parm name="PORTNBR" value="80"/>
    </characteristic>
  </characteristic>
</characteristic>

<characteristic type="NAPDEF">
  <parm name="NAPID" value="NAP 1"/>
  <parm name="BEARER" value="GSM-GPRS"/>
  <parm name="NAME" value="OUR GPRS"/>
  <parm name="NAP-ADDRESS" value="our.net"/>
  <parm name="NAP-ADDRTYPE" value="APN"/>
</characteristic>
```

```
</characteristic>
</wap-provisioningdoc>
```

The following table shows the translation into WBXML:

Token String	Description
03	WBXML version 1.3 ,see [WBXML1] chapter 5.4
0B	The public identifier for “-/WAPFORUM//DTD PROV 1.0//EN”
6A	Character set UTF-8. MIBEnum value is 106 which 6A in hex
00	String table length.
45	wap-provisioningdoc with no version
C65101	Characteristic (06) PXLOGICAL (51) with attribute and content ended with an end of the attribute list.
871506	Parameter (07) PROXY-ID (15) with attribute but no content. The attribute is the value of 06.
037364702E6F75722E636F6D0001	The value is inline string (03) “ <a href="http://sdp.our.com">sdp.our.com</a> ” ending with the end of the string (00) and end of the attribute list (01).
870706	Parameter (07) NAME (07) with attribute but no content. The attribute is the value of 06.
0348656C6C6F576F726C640001	The value is inline string (03) “ <a href="#">HelloWorld</a> ” ending with the end of the string (00) and end of the attribute list (01).
871C06	Parameter (07) STARTPAGE (1C) with attribute but no content. The attribute is the value of 06.
03687474703A2F2F7364702E6F75722E636F6D2F4955542F686F6D65706167652E68746D0001	The value is inline string (03) “ <a href="http://sdp.our.com/IUT/homepage.htm">http://sdp.our.com/IUT/homepage.htm</a> ” ending with the end of the string (00) and end of the attribute list (01).
C65201	Characteristic (06) PXPHYSICAL (52) with attribute and content ended with an end of the attribute list.
872F06	Parameter (07) PHYSICAL-PROXY-ID (2F) with attribute but no content. The attribute is the value of 06.
034761746577617920310001	The value is inline string (03) “ <a href="#">Gateway 1</a> ” ending with the end of the string (00) and end of the attribute list (01).
872006	Parameter (07) PXADDR (20) with attribute but no content. The attribute is the value of 06.
033131312E33312E3232322E34360001	The value is inline string (03) “ <a href="#">111.31.222.46</a> ” ending with the end of the string (00) and end of the attribute list (01).
8721068501	Parameter (07) PXADDRTYPE (21) with attribute but no content. The attribute is the value of 06. The value of the attribute is “IP4” (85) with end of the attribute list.
872206	Parameter (07) TO-NAPID (22) with attribute but no content. The attribute is the value of 06.
034E415020310001	The value is inline string (03) “ <a href="#">NAP 1</a> ” ending with the end of the string (00)

	and end of the attribute list (01).
C65301	Characteristic (06) PORT (53) with attribute and content ended with an end of the attribute list.
872306	Parameter (07) PORTNBR (23) with attribute but no content. The attribute is the value of 06.
0338300001	The value is inline string (03) "80" ending with the end of the string (00) and end of the attribute list (01).
01	End of the characteristic PORT.
01	End of the characteristic PXPHYSICAL.
C65201	Characteristic (06) PXPHYSICAL (52) with attribute and content ended with an end of the attribute list.
872F06	Parameter (07) PHYSICAL-PROXY-ID (2F) with attribute but no content. The attribute is the value of 06.
034761746577617920320001	The value is inline string (03) "Gateway 2" ending with the end of the string (00) and end of the attribute list (01).
872006	Parameter (07) PXADDR (20) with attribute but no content. The attribute is the value of 06.
033232322E33312E3131312E34360001	The value is inline string (03) "222.31.111.46" ending with the end of the string (00) and end of the attribute list (01).
8721068501	Parameter (07) PXADDRTYPE (21) with attribute but no content. The attribute is the value of 06. The value of the attribute is "IP4" (85) with end of the attribute list.
872206	Parameter (07) TO-NAPID (22) with attribute but no content. The attribute is the value of 06.
034E415020310001	The value is inline string (03) "NAP 1" ending with the end of the string (00) and end of the attribute list (01).
C65301	Characteristic (06) PORT (53) with attribute and content ended with an end of the attribute list.
872306	Parameter (07) PORTNBR (23) with attribute but no content. The attribute is the value of 06.
0338300001	The value is inline string (03) "80" ending with the end of the string (00) and end of the attribute list (01).
01	End of the characteristic PORT.
01	End of the characteristic PXPHYSICAL.
01	End of the characteristic PXLOGICAL
C65501	Characteristic (06) NAPDEF (55) with attribute and content ended with an end of the attribute list.
871106	Parameter (07) NAPID (11) with attribute but no content. The attribute is the value of 06.
034E415020310001	The value is inline string (03) "NAP 1" ending with the end of the string (00) and end of the attribute list (01).
871006AB01	Parameter (07) BEARER (10) with attribute but no content. The attribute is

	the value of 06. The value of attribute is GSM-GPRS (AB).
870706	Parameter (07) NAME (07) with attribute but no content. The attribute is the value of 06.
034F555220475052530001	The value is inline string (03) "OUR GPRS" ending with the end of the string (00) and end of the attribute list (01).
870806	Parameter (07) NAP-ADDRESS (08) with attribute but no content. The attribute is the value of 06.
036F75722E6E65740001	The value is inline string (03) "our.net" ending with the end of the string (00) and end of the attribute list (01).
8709068901	Parameter (07) NAP-ADDRTYPE (09) with attribute but no content. The attribute is the value of 06. The value of the attribute is "APN" (89) with the end of the attribute list.
01	End of the characteristic NAPDEF.
01	End of the wap-provisioningdoc.

The following is the hexadecimal dump of the WBXML file:

```

03 0B 6A 00 45 C6 51 01 87 15 06 03 73 64 70 2E
6F 75 72 2E 63 6F 6D 00 01 87 07 06 03 48 65 6C
6C 6F 57 6F 72 6C 64 00 01 87 1C 06 03 68 74 74
70 3A 2F 2F 73 64 70 2E 6F 75 72 2E 63 6F 6D 2F
49 55 54 2F 68 6F 6D 65 70 61 67 65 2E 68 74 6D
00 01 C6 52 01 87 2F 06 03 47 61 74 65 77 61 79
20 31 00 01 87 20 06 03 31 31 31 2E 33 31 2E 32
32 32 2E 34 36 00 01 87 21 06 85 01 87 22 06 03
4E 41 50 20 31 00 01 C6 53 01 87 23 06 03 38 30
00 01 01 01 C6 52 01 87 2F 06 03 47 61 74 65 77
61 79 20 32 00 01 87 20 06 03 32 32 32 2E 33 31
2E 31 31 31 2E 34 36 00 01 87 21 06 85 01 87 22
06 03 4E 41 50 20 31 00 01 C6 53 01 87 23 06 03
38 30 00 01 01 01 01 C6 55 01 87 11 06 03 4E 41
50 20 31 00 01 87 10 06 AB 01 87 07 06 03 4F 55
52 20 47 50 52 53 00 01 87 08 06 03 6F 75 72 2E
6E 65 74 00 01 87 09 06 89 01 01 01

```

## Adapting to GSM SMS Format

The following is an explanation of the header format for each message:

1. SMSC Number – HelloWorld SMSC

Example: 112233008000

- Length: (number of bytes including TON/NPI) 07
- TON/NPI: 91
- MSISDN: 11 22 33 00 08 00 FF FF FF FF FF

2. TPDU Length (variable – one byte, the sum of all remaining items)

Value = xx (varies)

3. SMS Deliver TPDU (one byte)

Value = 40

- TP-MTI: (bits 1 and 0) '00' SMS-DELIVER
- TP-MMS: (bit 2) '0' more messages are waiting (1<sup>st</sup> and 2<sup>nd</sup> message)
- TP-RP: (bit 7) '0' reply path NOT present
- TP-UDHI: (bit 6) '1' User Data contains a header
- TP-SRI: (bit 5) '0' Status report will not be returned
- Not used: (bits 4 and 3) '00'

4. TP-OA (variable – the originating address)

Example: Number 0123987654

- Length: (number of digits excluding TON/NPI) 0A
- TON/NPI: 81
- MSISDN: 10 32 89 67 45

6. TP-PID

Value: 00

- Usage: (bits 7 and 6) '00' Assigns bits 0...5 as defined below
- Telematic networking: (bit 5) '0' no interworking
- Telematic device: (bits 4-0) '00000' not used since bit 5 is '0'

7. TP-DCS

Value: F5

- Coding group bits: (bits 7-4) '1111'
- Reserved: (bit 3) '0'
- Message coding: (bit 2) '1' 8-bit data
- Message class: (bits 1 and 0) '01' class 1, default meaning

8. TP-SCTS (7 bytes)

Value: 10 20 72 61 54 85 00 (example)

9. TP-UDL

Value: 8C (maximum 140 bytes)

#### 10. TP-UD

A. TP-UDH – 12-byte header for each message

Example: 0B 05 04 0B 84 00 00 00 03 00 03 01

- UDH Length: 0B
- PORT Addressing
  - IE Type: 05 16-bit port addressing
  - IE Length: 04
  - Destination Port: 0B 84 (2948 – push connectionless port)
  - Source Port : 00 00 (0000 – does not matter)
- SAR
  - IE Type: 00 SAR
  - IE Length: 03
  - Datagram Reference Number: 00
  - Total number of segments: 03 (three segments)
  - Current segment number: 01 (first segment)

B. TP-UD – Binary data (WBXML file) Maximum 128 bytes

#### Hexadecimal dump for each message:

The total length of our provisioning data is 369 bytes:

- TID – 1 byte
- PDU Type – 1 byte
- Header Length – 1 byte
- Content Type Length – 90 bytes
- Headers – 0 bytes
- Provisioning document – 276 bytes

Since each message can only carry 128 bytes of data, this will require at least three SMS messages. We will illustrate this in the next section.

## Examples

---

#### Example #1 – text format (in first segment)

- *Message 1*

```
07 91 11 22 33 00 80 00 FF FF FF FF FF // SMSC
9E // TPDU LENGTH
```

```

40 // SMS Deliver TPDU (more messages waiting)
0A 81 10 32 89 67 45 // TP-OA
00 // TP-PID
F5 // TP-DCS
10 20 72 61 54 85 00 // TP-SCTS
8C //TP-UDL
0B 05 04 0B 84 00 00 00 03 00 03 01 //TP-UDH
// TP-UD
01 // TID
06 // UNIT PUSH
5A // HEADER LENGTH
// MEDIA-TYPE PARAMETER
61 70 70 6C 69 63 61 74 69 6F 6E 2F 76 6E 64 2E
77 61 70 2E 63 6F 6E 6E 65 63 74 69 76 69 74 79
2D 77 62 78 6D 6C 3B 53 45 43 3D 31 3B 4D 41 43
3D 41 34 46 30 32 33 42 38 38 32 46 31 46 37 30
43 44 38 45 32 43 31 37 43 41 33 39 38 36 34 43
46 39 36 38 31 37 42 34 32 00
// WBXML (BYTES 1 - 35 / 276)
03 0b 03 00 45 c6 51 01 87 15 06 03 73 64 70 2e
6f 75 72 2e 63 6f 6d 00 01 87 07 06 03 48 65 6c
6c 6f 57

```

- *Message 2*

```

07 91 11 22 33 00 80 00 FF FF FF FF FF // SMSC
9E // TPDU LENGTH
40 // SMS Deliver TPDU (more messages waiting)
0A 81 10 32 89 67 45 // TP-OA
00 // TP-PID
F5 // TP-DCS
10 20 72 61 54 85 00 // TP-SCTS
8C //TP-UDL
0B 05 04 0B 84 00 00 00 03 00 03 02 //TP-UDH
// TP-UD
// WBXML (BYTES 36 - 163 / 276)
6f 72 6c 64 00 01 87 1c 06 03 68 74 74 70 3a 2f
2f 73 64 70 2e 6f 75 72 2e 63 6f 6d 2f 49 55 54
2f 68 6f 6d 65 70 61 67 65 2e 68 74 6d 00 01 c6
52 01 87 2f 06 03 47 61 74 65 77 61 79 20 31 00
01 87 20 06 03 31 31 31 2e 33 31 2e 32 32 32 2e
34 36 00 01 87 21 06 85 03 00 01 87 22 06 03 4e
41 50 20 31 00 01 c6 53 01 87 23 06 03 38 30 00
01 01 01 c6 52 01 87 2f 06 03 47 61 74 65 77 61

```

- *Message 3*

```

07 91 11 22 33 00 80 00 FF FF FF FF FF // SMSC
8F // TPDU LENGTH
44 // SMS Deliver TPDU (no more messages waiting)
0A 81 10 32 89 67 45 // TP-OA
00 // TP-PID
F5 // TP-DCS
10 20 72 61 54 85 00 // TP-SCTS
7D //TP-UDL
0B 05 04 0B 84 00 00 00 03 00 03 03 //TP-UDH
// TP-UD

```

5  
WAP Provisioning Data Format

```
// WBXML (BYTES 164 - 276 / 276)
79 20 32 00 01 87 20 06 03 32 32 32 2e 33 31 2e
31 31 31 2e 34 36 00 01 87 21 06 85 03 00 01 87
22 06 03 4e 41 50 20 31 00 01 c6 53 01 87 23 06
03 38 30 00 01 01 01 01 c6 55 01 87 11 06 03 4e
41 50 20 31 00 01 87 10 06 ab 03 00 01 87 07 06
03 4f 55 52 20 47 50 52 53 00 01 87 08 06 03 6f
75 72 2e 6e 65 74 00 01 87 09 06 89 03 00 01 01
01
```

Example #2 – token format (in first segment), and including headers

```
• Message 1
07 91 11 22 33 00 80 00 FF FF FF FF FF // SMSC
80 // TPDU LENGTH
40 // SMS Deliver TPDU (more messages waiting)
0A 81 10 32 89 67 45 // TP-OA
00 // TP-PID
F5 // TP-DCS
10 20 72 61 54 85 00 // TP-SCTS
6E //TP-UDL
0B 05 04 0B 84 00 00 00 03 00 03 01 //TP-UDH
// TP-UD
01 // TID
06 // UNIT PUSH
3C // Length for content-type and HEADERS
1F 2D // Content type value length given as "Length-quote Length"

// MEDIA-TYPE PARAMETER
B6 // for application/vnd.wap.connectivity-wbxml
91 81 // for SEC and USERPIN (1)
92 // MAC
41 34 46 30 32 33 42 38 38 32 46 31 46 37 30 43
44 38 45 32 43 31 37 43 41 33 39 38 36 34 43 46
39 36 38 31 37 42 34 32 00
95 // From field in headers
61 62 63 40 6D 6F 74 2E 63 6F 6D 00 // abc@mot.com

// WBXML (BYTES 1 - 35 / 276)
03 0b 03 00 45 c6 51 01 87 15 06 03 73 64 70 2e
6f 75 72 2e 63 6f 6d 00 01 87 07 06 03 48 65 6c
6c 6f 57
```

---

**NOTE:** 43 bytes are saved compared with the method used in example #1. Message 2 and Message 3 are the same as that in example #1.

---

# Appendix A: Parameter Mapping

Prompt	Description	OMA Parameter
Name	Name of the Internet Setup	PXLOGICAL / NAME
Homepage	The homepage for this session	PXLOGICAL / STARTPAGE
Gateway IP 1	Primary proxy address	PXPHYSICAL / PXADDR
Port 1	Port number to be used for proxy 1	PORT <sup>1</sup>
Domain 1	The domain covered by proxy 1	DOMAIN <sup>1</sup>
Service Type 1	The service offered by Port 1	SERVICE <sup>1</sup>
Gateway IP 2	Secondary proxy address	PXPHYSICAL / PXADDR
Port 2	Port number to be used for proxy 2	PORT <sup>1</sup>
Domain 2	The domain covered by proxy 2	DOMAIN <sup>1</sup>
Service Type 2	The service offered by Port 2	SERVICE <sup>1</sup>
DNS 1	Primary DNS IP Address	NAPDEF / DNS-ADDR
DNS 2	Secondary DNS IP Address	NAPDEF / DNS-ADDR
CSD Timeout	CSD Linger Timer	NAPDEF / LINGER
CSD No. 1	Primary phone number according to E164 Scheme	NAPDEF / NAP-ADDRESS
User Name 1	User Name for CSD No. 1 login	NAPAUTHINFO / AUTHNAME
Password 1	Password for CSD No. 1 login	NAPAUTHINFO / AUTHSECRET
Speed (Bps) 1	Speed for CSD No. 1	NAPDEF / LINKSPEED
Line Type 1	Call Type for CSD No. 1	NAPDEF / NAP-ADDRTYPE
CSD No. 2	Secondary phone number according to E164 Scheme	NAPDEF / NAP-ADDRESS
User Name 2	User Name for CSD No. 2 login	NAPAUTHINFO / AUTHNAME
Password 2	Password for CSD No. 2 login	NAPAUTHINFO / AUTHSECRET
Speed (Bps) 2	Speed for CSD No. 2	NAPDEF / LINKSPEED
Line Type 2	Call Type for CSD No. 2	NAPDEF / NAP-ADDRTYPE
GPRS APN	APN for Packet Data Access	NAPDEF / NAP-ADDRESS
User Name	User name for CHAP/PAP GPRS APN login	NAPAUTHINFO / AUTHNAME
Password	Password for CHAP/PAP GPRS APN login	NAPAUTHINFO / AUTHSECRET

<sup>1</sup>Present in either PXLOGICAL or PXPHYSICAL – PXPHYSICAL has higher priority.

# Appendix B: Compliance Matrix

Please note the following compliance matrix is for products using the Motorola Internet Browser (MIB) version 2.2. Products include (but are not limited to) the following:

- E380
- V300
- V500
- V600
- A835

Character Set and Encoding				
Item #	Function	Ref	Status	Motorola
ProvCont-CSE-C-001	UTF-8 Encoding	4.7	M	Y
ProvCont-CSE-C-002	Character entities	4.7	M	Y

Content Format and Tokenization				
Item #	Function	Ref	Status	Motorola
ProvCont-CO-C-001	Support for the WAP-Provisioning-doc DTD	4.1	M	Y
ProvCont-CO-C-002	Support for the WAP-Provisioning-doc DTD or textual form (text/vnd.wap.connectivity-xml)	4.2	O	N
ProvCont-CO-C-003	Support for the WAP-Provisioning-doc DTD in tokenized form (application/vnd.wap.connecti	7	M	Y Up to 10 provisioning profiles are supported.

	vity.wbxml)			
ProvCont-CO-C-004	Support for media type parameter MAC	4.3	M	Y
ProvCont-CO-C-005	Support for media type parameter SEC	4.3	M	Y

Elements and Attributes				
Item #	Function	Ref	Status	Motorola
ProvCont-CEA-C-001	Support for the element wap-provisioningdoc	4.4	M	Y
ProvCont-CEA-C-002	Support for the element characteristic	4.5	M	Y
ProvCont-CEA-C-003	Support for the element parm	4.6	M	Y
ProvCont-CEA-C-004	Support for the wap-provisioningdoc attribute "version"	4.4	M	Y
ProvCont-CEA-C-005	Support for the characteristic attribute "type"	4.5	M	Y
ProvCont-CEA-C-006	Support for the parm attribute "name"	4.6	M	Y
ProvCont-CEA-C-007	Support for the parm attribute "value"	4.6	M	Y

Characteristics				
Item #	Function	Ref	Status	Motorola
ProvCont-CC-C-001	Support for the characteristic PXLOGICAL	4.5.1	M	Y Up to 10 PXLOGICALs are supported.
ProvCont-CC-C-002	Support for the characteristic PXPHYSICAL	4.5.2	M	Y Two characteristics per PXLOGICAL will be supported.

Appendix B:  
Compliance Matrix

ProvCont-CC-C-003	Support for the characteristic PXAUTHINFO	4.5.3	O	N
ProvCont-CC-C-004	Support for the characteristic NAPDEF	4.5.5	M	Y Two CSD and one GPRS NAPDEF per provisioning document will be supported.
ProvCont-CC-C-005	Support for the characteristic NPAUTHINFO	4.5.6	O	Y
ProvCont-CC-C-006	Support for the characteristic PORT	5.2 4.5.4	M	Y
ProvCont-CC-C-007	Support for the characteristic VALIDITY	4.5.7	O	N
ProvCont-CC-C-008	Support for the characteristic BOOTSTRAP	4.5.8	O	Y We don't support continuous provisioning
ProvCont-CC-C-00	Support for the characteristic CLIENTIDENTITY	4.5.9	O	N
ProvCont-CC-C-010	Support for the characteristic VENDORCONFIG	4.5.10	O	N
ProvCont-CC-C-011	Support for the characteristic APPLICATION	4.5.11	O	Y Limited use for MMS
ProvCont-CC-C-012	Support for the characteristic APPADDR	4.5.12	O	N
ProvCont-CC-C-013	Support for the characteristic APPAUTH	4.5.13	O	N
ProvCont-CC-C-014	Support for the characteristic RESOURCE	4.5.14	O	N
ProvCont-CC-C-015	Support for the characteristic ACCESS	4.5.15	O	N

Characteristic PXLOGICAL				
Item #	Function	Ref	Status	Motorola

ProvCont-CPL-C-001	Support for the parm PROXY-ID	4.6.1	M	Y
ProvCont-CPL-C-002	Support for the parm PROXY-PROVIDER-ID	4.6.1	O	N
ProvCont-CPL-C-003	Support for the parm NAME	4.6.1	M	Y
ProvCont-CPL-C-004	Support for the parm DOMAIN	4.6.1	M	Y
ProvCont-CPL-C-005	Support for the parm TRUST	4.6.1	O	N
ProvCont-CPL-C-006	Support for the parm MASTER	4.6.1	O	N
ProvCont-CPL-C-007	Support for the parm STARTPAGE	4.6.1	M	Y
ProvCont-CPL-C-008	Support for the parm BASAUTH-ID	4.6.1	M	N The preferred approach is to have this entered by the user to do proxy authentication based on a challenge from the proxy.
ProvCont-CPL-C-009	Support for the parm BASAUTH-PW	4.6.1	M	
ProvCont-CPL-C-010	Support for the parm WSP-VERSION	4.6.1	O	N
ProvCont-CPL-C-011	Support for the parm PUSHENABLED	4.6.1	O	N
ProvCont-CPL-C-012	Support for PORT characteristic within PXLOGICAL	4.5.4	M	Y
ProvCont-CPL-C-013	Support for multiple PORT characteristics within PXLOGICAL	4.5.4	O	N
ProvCont-CPL-C-014	Support for parm PROXY-PW	4.6.1	O	N
ProvCont-CPL-C-015	Support for parm PPGAUTH-TYPE	4.6.1	O	N

Appendix B:  
Compliance Matrix

ProvCont-CPL-C-016	Support for parm PULLENABLED	4.6.1	O	N
--------------------	------------------------------	-------	---	---

Characteristic PXPHYSICAL				
Item #	Function	Ref	Status	Motorola
ProvCont-CPP-C-001	Support for the parm PHYSICAL-PROXY-ID	4.6.2	M	Y
ProvCont-CPP-C-002	Support for the parm PXADDR	4.6.2	M	Y (Motorola Browser supports only IPV4)
ProvCont-CPP-C-003	Support for the parm PXADDRTYPE	4.6.2	M	Y
ProvCont-CPP-C-004	Support for the parm TO-NAPID	4.6.2	M	Y
ProvCont-CPP-C-005	Support for the parm DOMAIN	4.6.2	O	Y
ProvCont-CPP-C-006	Support for the parm WSP-VERSION	4.6.2	O	N
ProvCont-CPP-C-007	Support for the parm PUSHENABLED	4.6.2	O	N
ProvCont-CPP-C-008	Support for the TO-NAPID value "INTERNET"	4.6.2	O	N
ProvCont-CPP-C-009	Support for the PXADDRTYPE value "IPV4"	4.6.2	O	Y
ProvCont-CPP-C-010	Support for the PXADDRTYPE value "IPV6"	4.6.2	O	N
ProvCont-CPP-C-011	Support for the PXADDRTYPE value "E164"	4.6.2	O	N
ProvCont-CPP-C-012	Support for the PXADDRTYPE value "ALPHA"	4.6.2	O	N
ProvCont-CPP-C-013	Support for PORT characteristics within PXPHYSICAL	4.5.4	M	Y

ProvCont- CPP-C-014	Support for multiple PORT characteristics within PXPHYSICAL	4.5.4	O	N
ProvCont- CPP-C-015	Support for multiple TO-NAPID within one PXPHYSICAL	4.6.2	O	Y However, the NAPIDs are applied to all characteristics within the PXLOGICAL document. Not per PXPHYSICAL characteristic.
ProvCont- CPP-C-016	Support for the parm PXADDR-FQDN	4.6.2	O	N
ProvCont- CPP-C-017	Support for the parm PULLENABLED	4.6.2	O	N

Characteristic PXAUTHINFO				
Item #	Function	Ref	Status	Motorola
ProvCont- CPA-C-001	Support for the parm PXAUTH-TYPE	4.6.3	O	N
ProvCont- CPA-C-002	Support for the parm PXAUTH-ID	4.6.3	O	N
ProvCont- CPA-C-003	Support for the parm PXAUTH-PW	4.6.3	O	N
ProvCont- CPA-C-004	Support for PXAUTH-TYPE value "HTTP-BASIC"	4.6.3	O	N
ProvCont- CPA-C-005	Support for PXAUTH-TYPE value "HTTP-DIGEST"	4.6.3	O	N
ProvCont- CPA-C-006	Support for PXAUTH-TYPE value "WTLS-SS"	4.6.3	O	N

Characteristic PORT				
Item #	Function	Ref	Status	Motorola
ProvCont-CP- C-001	Support for the parm PORTNBR	4.6.4	M	Y
ProvCont-CP- C-002	Support for the parm SERVICE	4.6.4	M	Y

Appendix B:  
Compliance Matrix

ProvCont-CP-C-003	Support for SERVICE value "CL-WSP"	4.6.4	O	Y (These parameters are defined only for WAP Pull)
ProvCont-CP-C-004	Support for SERVICE value "CO-WSP"	4.6.4	O	Y (These parameters are defined only for WAP Pull)
ProvCont-CP-C-005	Support for SERVICE value "CL-SEC-WSP"	4.6.4	O	Y (These parameters are defined only for WAP Pull)
ProvCont-CP-C-006	Support for SERVICE value "CO-SEC-WSP"	4.6.4	O	Y (These parameters are defined only for WAP Pull)
ProvCont-CP-C-007	Support for SERVICE value "CO-SEC-WTA"	4.6.4	O	N
ProvCont-CP-C-008	Support for SERVICE value "CL-SEC-WTA"	4.6.4	O	N
ProvCont-CP-C-009	Support for SERVICE value "OTA-HTTP-TO"	4.6.4	O	N
ProvCont-CP-C-010	Support for SERVICE value "OTA-HTTP-TLS-TO"	4.6.4	O	N
ProvCont-CP-C-011	Support for SERVICE value "OTA-HTTP-PO"	4.6.4	O	Y
ProvCont-CP-C-012	Support for SERVICE value "OTA-HTTP-TLS-PO"	4.6.4	O	Y

Characteristic NAPDEF				
Item #	Function	Ref	Status	Motorola
ProvCont-CND-C-001	Support for the parm NAPID	4.6.5	M	Y
ProvCont-CND-C-002	Support for the parm BEARER	4.6.5	M	Y
ProvCont-CND-C-003	Support for the parm NAME	4.6.5	M	Y

ProvCont-CND-C-004	Support for the parm INTERNET	4.6.5	O	N
ProvCont-CND-C-005	Support for the parm NAP-ADDRESS	4.6.5	M	Y
ProvCont-CND-C-006	Support for the parm NAP-ADDRTYPE	4.6.5	M	Y
ProvCont-CND-C-007	Support for the parm CALLTYPE	4.6.5	O	Y
ProvCont-CND-C-008	Support for the parm LOCAL-ADDR	4.6.5	O	N
ProvCont-CND-C-009	Support for the parm LOCAL-ADDRTYPE	4.6.5	O	N
ProvCont-CND-C-010	Support for the parm LINKSPEED	4.6.5	O	Y
ProvCont-CND-C-011	Support for the parm DNLINKSPEED	4.6.5	O	N
ProvCont-CND-C-012	Support for the parm LINGER	4.6.5	O	Y
ProvCont-CND-C-013	Support for the parm DELIVERY-ERR-SDU	4.6.5	O	N
ProvCont-CND-C-014	Support for the parm DELIVERY-ORDER	4.6.5	O	N
ProvCont-CND-C-015	Support for the parm TRAFFIC-CLASS	4.6.5	O	N
ProvCont-CND-C-016	Support for the parm MAX-SDU-SIZE	4.6.5	O	N
ProvCont-CND-C-017	Support for the parm MAX-BITRATE-UPLINK	4.6.5	O	N
ProvCont-CND-C-018	Support for the parm MAX-BITRATE-DNLINK	4.6.5	O	N
ProvCont-CND-C-019	Support for the parm RESIDUAL-BER	4.6.5	O	N

Appendix B:  
Compliance Matrix

ProvCont-CND-C-020	Support for the parm SDU-ERROR-RATIO	4.6.5	O	N
ProvCont-CND-C-021	Support for the parm TRAFFIC-HANDL-PRIO	4.6.5	O	N
ProvCont-CND-C-022	Support for the parm TRANSFER-DELAY	4.6.5	O	N
ProvCont-CND-C-023	Support for the parm GUARANTEED-BITRATE-UPLINK	4.6.5	O	N
ProvCont-CND-C-024	Support for the parm GUARANTEED-BITRATE-DNLINK	4.6.5	O	N
ProvCont-CND-C-025	Support for multiple BEARER within one NAPDEF	4.6.5	O	N
ProvCont-CND-C-026	Support for NAP-ADDRTYPE value "IPV4"	4.6.5	O	N
ProvCont-CND-C-027	Support for NAP-ADDRTYPE value "IPV6"	4.6.5	O	N
ProvCont-CND-C-028	Support for NAP-ADDRTYPE value "E164"	4.6.5	O	Y
ProvCont-CND-C-029	Support for NAP-ADDRTYPE value "ALPHA"	4.6.5	O	N
ProvCont-CND-C-030	Support for NAP-ADDRTYPE value "APN"	4.6.5	O	Y
ProvCont-CND-C-031	Support for NAP-ADDRTYPE value "SCODE"	4.6.5	O	N
ProvCont-CND-C-032	Support for NAP-ADDRTYPE value "TETRA-ITSI"	4.6.5	O	N
ProvCont-CND-C-033	Support for NAP-ADDRTYPE value "MAN"	4.6.5	O	N
ProvCont-CND-C-034	Support for CALLTYPE value "ANALOG-MODEM"	4.6.5	O	Y

ProvCont-CND-C-035	Support for CALLTYPE value "V.120"	4.6.5	O	N
ProvCont-CND-C-036	Support for CALLTYPE value "V.110"	4.6.5	O	N
ProvCont-CND-C-037	Support for CALLTYPE value "X.31"	4.6.5	O	N
ProvCont-CND-C-038	Support for CALLTYPE value "BIT-TRANSPARENT"	4.6.5	O	N
ProvCont-CND-C-039	Support for CALLTYPE value "DIRECT-ASYNCHRONOUS-DATA-SERVICE"	4.6.5	O	Y
ProvCont-CND-C-040	Support for LOCAL-ADDRTYPE value "IPV4"	4.6.5	O	N
ProvCont-CND-C-041	Support for LOCAL-ADDRTYPE value "IPV6"	4.6.5	O	N
ProvCont-CND-C-042	Support for the parm DNS-ADDR	4.6.5	O	Y
ProvCont-CND-C-043	Support for the parm MAX-NUMRETRY	4.6.5	O	N
ProvCont-CND-C-044	Support for the parm FIRST-RETRYTIMEOUT	4.6.5	O	N
ProvCont-CND-C-045	Support for the parm REREGTHRESHOLD	4.6.5	O	N
ProvCont-CND-C-046	Support for the parm T-BIT	4.6.5	O	N

<b>Bearers supported within NAPDEF Characteristics</b>				
Item #	Function	Ref	Status	Motorola
ProvCont-CBS-C-001	Support for the BEARER value "GSM-USSD"	4.6.5	O	N
ProvCont-CBS-C-002	Support for the BEARER value "GSM-SMS"	4.6.5	O	N (These parameters are defined only for WAP Pull)

Appendix B:  
Compliance Matrix

				over SMS)
ProvCont-CBS-C-003	Support for the BEARER value "ANSI-136-GUTS"	4.6.5	O	Y
ProvCont-CBS-C-004	Support for the BEARER value "IS-95-CDMA-SMS"	4.6.5	O	N (These parameters are defined only for WAP Pull over SMS)
ProvCont-CBS-C-005	Support for the BEARER value "IS-95-CDMA-CSD"	4.6.5	O	Y
ProvCont-CBS-C-006	Support for the BEARER value "IS-95-CDMA-PACKET"	4.6.5	O	Y
ProvCont-CBS-C-007	Support for the BEARER value "ANSI-136-CSD"	4.6.5	O	Y
ProvCont-CBS-C-008	Support for the BEARER value "ANSI-136-GPRS"	4.6.5	O	Y
ProvCont-CBS-C-009	Support for the BEARER value "GSM-CSD"	4.6.5	O	Y
ProvCont-CBS-C-010	Support for the BEARER value "GSM-GPRS"	4.6.5	O	Y
ProvCont-CBS-C-011	Support for the BEARER value "AMPS-CDPD"	4.6.5	O	N
ProvCont-CBS-C-012	Support for the BEARER value "PDC-CSD"	4.6.5	O	N
ProvCont-CBS-C-013	Support for the BEARER value "PDC-PACKET"	4.6.5	O	N
ProvCont-CBS-C-014	Support for the BEARER value "IDEN-SMS"	4.6.5	O	N
ProvCont-CBS-C-015	Support for the BEARER value "IDEN-CSD"	4.6.5	O	N
ProvCont-CBS-C-016	Support for the BEARER value "IDEN-PACKET"	4.6.5	O	N

ProvCont-CBS-C-017	Support for the BEARER value "FLEX/REFLEX"	4.6.5	O	N
ProvCont-CBS-C-018	Support for the BEARER value "PHS-SMS"	4.6.5	O	N
ProvCont-CBS-C-019	Support for the BEARER value "PHS-CSD"	4.6.5	O	N
ProvCont-CBS-C-020	Support for the BEARER value "TETRA-SDS"	4.6.5	O	N
ProvCont-CBS-C-021	Support for the BEARER value "TETRA-PACKET"	4.6.5	O	N
ProvCont-CBS-C-022	Support for the BEARER value "ANSI-136-GHOST"	4.6.5	O	N
ProvCont-CBS-C-023	Support for the BEARER value "MOBITEX-MPAK"	4.6.5	O	N
ProvCont-CBS-C-024	Support for the BEARER value "CDMA2000-1X-SIMPLE-IP"	4.6.5	O	N
ProvCont-CBS-C-025	Support for the BEARER value "CDMA2000-1X-MOBILE-IP"	4.6.5	O	N

<b>Characteristic NAPAUTHINFO</b>				
<b>Item #</b>	<b>Function</b>	<b>Ref</b>	<b>Status</b>	<b>Motorola</b>
ProvCont-CAN-C-001	Support for the parm AUTHTYPE	4.6.6	O	Y
ProvCont-CAN-C-002	Support for the parm AUTHNAME	4.6.6	O	Y
ProvCont-CAN-C-003	Support for the parm AUTHSECRET	4.6.6	O	Y
ProvCont-CAN-C-004	Support for AUTHTYPE value "PAP"	4.6.6	O	Y
ProvCont-CAN-C-005	Support for AUTHTYPE value "CHAP"	4.6.6	O	Y

Appendix B:  
Compliance Matrix

ProvCont-CAN-C-006	Support for AUTHTYPE value "MD5"	4.6.6	O	N
ProvCont-CAN-C-007	Support for parm AUTH-ENTITY value	4.6.6	O	N
ProvCont-CAN-C-008	Support for parm SPI	4.6.6	O	N
ProvCont-CAN-C-009	Support for AUTH-ENTITY value "AAA"	4.6.6	O	N
ProvCont-CAN-C-010	Support for AUTH-ENTITY value "HA"	4.6.6	O	N

<b>Characteristic Validity</b>				
Item #	Function	Ref	Status	Motorola
ProvCont-CV-C-001	Support for the parm COUNTRY	4.6.7	O	N
ProvCont-CV-C-002	Support for the parm NETWORK	4.6.7	O	N
ProvCont-CV-C-003	Support for the parm SID	4.6.7	O	N
ProvCont-CV-C-004	Support for the parm SOC	4.6.7	O	N
ProvCont-CV-C-005	Support for the parm VALIDUNTIL	4.6.7	O	N
ProvCont-CV-C-006	Support for multiple MNC in NETWORK value field	4.6.7	O	N
ProvCont-CV-C-007	Support for multiple SID in SID value field	4.6.7	O	N

<b>Characteristic BOOTSTRAP</b>				
Item #	Function	Ref	Status	Motorola
ProvCont-CB-C-001	Support for the parm PROVURL	4.6.8	O	N

ProvCont-CB-C-002	Support for the parm CONTEXT-ALLOW	4.6.8	O	N
ProvCont-CB-C-003	Support for the parm PROXY-ID	4.6.8	O	Y This value needs to be the same as that under PXLOGICAL
ProvCont-CB-C-004	Support for parm NETWORK	4.6.8	O	N
ProvCont-CB-C-005	Support for parm COUNTRY	4.6.8	O	N
ProvCont-CB-C-006	Support for parm NAME	4.6.8	O	Y

<b>Characteristic CLIENTIDENTITY</b>				
Item #	Function	Ref	Status	Motorola
ProvCont-CID-C-001	Support for parm CLIENT-ID	4.6.9	O	N

<b>Characteristic VENDORCONFIG</b>				
Item #	Function	Ref	Status	Motorola
ProvCont-CVC-C-001	Support for parm NAME	4.6.10	O	N
ProvCont-CVC-C-002	Support for other parameters than NAME	4.6.10	O	N

<b>Characteristic APPLICATION</b>				
Item #	Function	Ref	Status	Motorola
ProvCont-CAP-C-001	Support for parm APPID	4.6.11	O	Y So far only for MMS
ProvCont-CAP-C-002	Support for parm PROVIDER-ID	4.6.11	O	N
ProvCont-CAP-C-003	Support for parm NAME	4.6.11	O	Y

Appendix B:  
Compliance Matrix

ProvCont-CAP-C-004	Support for parm AACCEPT	4.6.11	O	N
ProvCont-CAP-C-005	Support for parm APROTOCOL	4.6.11	O	N
ProvCont-CAP-C-006	Support for parm TO-PROXY	4.6.11	O	N
ProvCont-CAP-C-007	Support for parm TO-NAPID	4.6.11	O	N
ProvCont-CAP-C-008	Support for parm ADDR	4.6.11	O	Y

<b>Characteristic APPADDR</b>				
Item #	Function	Ref	Status	Motorola
ProvCont-CAA-C-001	Support for parm ADDR	4.6.12	O	N
ProvCont-CAA-C-002	Support for parm ADDRTYPE	4.6.12	O	N

<b>Characteristic APPAUTH</b>				
Item #	Function	Ref	Status	Motorola
ProvCont-CAU-C-001	Support for parm AAUTHLEVEL	4.6.13	O	N
ProvCont-CAU-C-002	Support for parm AAUTHTYPE	4.6.13	O	N
ProvCont-CAU-C-003	Support for parm AAUTHNAME	4.6.13	O	N
ProvCont-CAU-C-004	Support for parm AAUTHSECRET	4.6.13	O	N
ProvCont-CAU-C-005	Support for parm AAUTHDATA	4.6.13	O	N

<b>Characteristic RESOURCE</b>				
Item #	Function	Ref	Status	Motorola

ProvCont-CRE-C-001	Support for parm URI	4.6.14	O	N
ProvCont-CRE-C-002	Support for parm NAME	4.6.14	O	N
ProvCont-CRE-C-003	Support for parm AACCEPT	4.6.14	O	N
ProvCont-CRE-C-004	Support for parm AAUHTHYPE	4.6.14	O	N
ProvCont-CRE-C-005	Support for parm AAUHTHNAME	4.6.14	O	N
ProvCont-CRE-C-006	Support for parm AAUHTHSECRET	4.6.14	O	N
ProvCont-CRE-C-007	Support for parm AAUHTHDATA	4.6.14	O	N
ProvCont-CRE-C-008	Support for parm STARTPAGE	4.6.14	O	N

<b>Characteristic ACCESS</b>				
<b>Item #</b>	<b>Function</b>	<b>Ref</b>	<b>Status</b>	<b>Motorola</b>
ProvCont-CAC-C-001	Support for parm RULE	4.6.15	O	N
ProvCont-CAC-C-002	Support for parm APPID	4.6.15	O	N
ProvCont-CAC-C-003	Support for parm PORTNBR	4.6.15	O	N
ProvCont-CAC-C-004	Support for parm DOMAIN	4.6.15	O	N
ProvCont-CAC-C-005	Support for parm TO-NAPID and/or parm TO-PROXY	4.6.15	O	N
ProvCont-CAC-C-006	Support for parm TO-NAPID	4.6.15	O	N

Appendix B:  
Compliance Matrix

ProvCont-CAC-C-007	Support for parm TO-PROXY	4.6.15	O	N
--------------------	---------------------------	--------	---	---

Minimum Length of Parameter Fields				
Item #	Function	Ref	Status	Motorola
ProvCont-MLP-C-001	Support for minimum length of parm NAME	5.1	M	Y
ProvCont-MLP-C-002	Support for minimum length of parm NAP-ADDRESS	5.1	O	Y
ProvCont-MLP-C-003	Support for minimum length of parm AUTHNAME	5.1	O	Y
ProvCont-MLP-C-004	Support for minimum length of parm AUTHSECRET	5.1	O	Y
ProvCont-MLP-C-005	Support for minimum length of parm PROXY-ID	5.1	O	Y
ProvCont-MLP-C-006	Support for minimum length of parm DOMAIN	5.1	O	Y
ProvCont-MLP-C-007	Support for minimum length of parm PROVURL	5.1	O	N
ProvCont-MLP-C-008	Support for minimum length of parm PXAUTH-ID	5.1	O	N
ProvCont-MLP-C-009	Support for minimum length of parm PXAUTH-PW	5.1	O	N
ProvCont-MLP-C-010	Support for minimum length of parm STARTPAGE	5.1	M	Y
ProvCont-MLP-C-011	Support for minimum length of parm BASAUTH-ID	5.1	M	N
ProvCont-MLP-C-012	Support for minimum length of parm BASAUTH-PW	5.1	M	N
ProvCont-MLP-C-013	Support for minimum length of parm PXADDR	5.1	M	Y

ProvCont-MLP-C-014	Support for minimum length of parm LINKSPEED	5.1	O	Y
ProvCont-MLP-C-015	Support for minimum length of parm DNLINKSPEED	5.1	O	N
ProvCont-MLP-C-016	Support for minimum length of parm LINGER	5.1	O	Y
ProvCont-MLP-C-017	Support for minimum length of parm VALIDUNTIL	5.1	O	N
ProvCont-MLP-C-018	Support for minimum length of parm PHYSICAL PROXY-ID	5.1	M	Y
ProvCont-MLP-C-019	Support for minimum length of parm NAPID	5.1	M	Y
ProvCont-MLP-C-020	Support for minimum length of parm CLIENT-ID	5.1	O	N
ProvCont-MLP-C-021	Support for minimum length of parm PROXY-PROVIDER-ID	5.1	O	N
ProvCont-MLP-C-022	Support for minimum length of parm PXADDR-FQDN	5.1	O	N
ProvCont-MLP-C-023	Support for minimum length of parm PROXY-PW	5.1	O	N
ProvCont-MLP-C-024	Support for minimum length of parm DNS-ADDR	5.1	O	N
ProvCont-MLP-C-025	Support for minimum length of parm APPID	5.1	O	N
ProvCont-MLP-C-026	Support for minimum length of parm PROVIDER-ID	5.1	O	N
ProvCont-MLP-C-027	Support for minimum length of parm ADDR	5.1	O	N
ProvCont-MLP-C-028	Support for minimum length of parm APROTOCOL	5.1	O	N
ProvCont-MLP-C-029	Support for minimum length of parm AAUTHNAME	5.1	O	N

Appendix B:  
Compliance Matrix

ProvCont-MLP-C-030	Support for minimum length of parm AAUTHSECRET	5.1	O	N
ProvCont-MLP-C-031	Support for minimum length of parm AACCEPT	5.1	O	N
ProvCont-MLP-C-032	Support for minimum length of parm URI	5.1	O	N
ProvCont-MLP-C-033	Support for minimum length of parm REREG-THRESHOLD	5.1	O	N
ProvCont-MLP-C-034	Support for minimum length of parm RULE	5.1	O	N



MOTOROLA and the Stylized M Logo are registered in the U.S. Patent & Trademark Office. All other product or service names are the property of their respective owners. Java and all other Java-based marks are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and other countries.

© Motorola, Inc. 2002.