整合式服務平台 – MMS 閘道介面

IMSP - MMS Gateway Interface

Version 2.0

2019/04/08

更新履歷

版本	時間	更新內容
1.0	2005/10/31	初稿
1.1	2007/04/18	修改單次傳送 MMS 至多可攜 20 組受話號碼
1.2	2008/09/01	新增三組錯誤代碼：-6、-8、-9
1.3	2008/11/22	新增一組錯誤代碼：-10
1.4	2013/05/29	新增 SSL 介面，移除 login 之 http GET 方法介面、及說明未來將移除 eaif 之相關服務。相關 http GET 範例改為 https GET 用法。
1.5	2013/08/14	修正 http port 8080 至正確的 port 8008。
1.6	2014/05/09	說明 code 3 Submitting failed 之常見的原因。
1.7	2016/01/15	限制使用 SSL 連線、移除 EAIF 相關服務、補充一些 MMS 說明、及新增 JSP 程式範例。
2.0	2019/04/08	新增 change_passwd 變更密碼介面。

目錄

1. 說明 5

2. 簡介 5

3. Protocol 說明 6

3.1 測試帳號(Login Test) 8

3.2 傳送 MMS 命令 – 1 (send_mms) 10

3.4 查詢 MMS 命令 – 1 (query) 16

3.5 查詢 MMS 命令 – 2 (query_batch) 19

3.6 擷取 MMS 命令 (get_mime) 21

3.7 變更密碼命令(change_passwd) 23

4. 其它說明 26

4.1 關於 MMS 訊息 26
4.2 其他注意事項 26
4.3 常見問題集 27
5. 附錄 29
5.1 GET_MIME 之 MMS 規格說明 29
5.2 JSP 傳送 Mixed MMS 程式範例 31
5.3 JSP 傳送 SMIL MMS 程式範例 41

1. 說明

IMSP MMS 閘道介面提供客戶端以 HTTP protocol 從 Internet 至 MMSC 的介接。本文件提供客戶端程式與 IMSP MMS Gateway 連接時之資料傳遞格式說明及其他應注意的事項。

2. 簡介

IMSP MMS Gateway 為一 Web Application，提供客戶端以 HTTP protocol 與 IMSP MMS Gateway 連接，目前 IMSP MMS Gateway 所提供的介面有：發送 MMS 訊息至手機、查詢傳送訊息的狀態及收取手機上傳至特碼之 MMS 訊息等。客戶端傳送到 IMSP MMS Gateway 的參數均是以 HTTPs Post 方式傳送；IMSP MMS Gateway 對於客戶端的請求則是以 HTTP Respond 方式傳回一個 HTML page，請求的執行結果包括結果代碼及描述均在 HTML Body 中。

註 1：本 IMSP MMS 服務內定只提供客戶發送及查詢簡訊權限。若需要設定允入 IP 清單等，請至本公司另申請帳號之相關權限。

註 2：目前 IMSP HTTP SSL 連線已於 2015 年 10 月停用 SSL v3，而本公司隨時會依最新資安要求，需要時會調高 SSL 加密等級。當客戶端使用 JAVA 版應用程式，建議使用較新的 JDK 1.7 (含) 以上版本，以便支援其 TLS 或更高安全等級連線服務，以免因 JDK 版本過舊而可能無法使用過時已不安全的 SSL 連線。建議客戶端直接採用最高等級 TLS v1.2 安全連線。

JDK 1.6：須更新至 update 111 支援 TLS v1.1，至 update 121 另支援 TLS v1.2，內定啟用 TLS v1.0；
JDK 1.7：支援 TLS 1.1 及 TLS 1.2，內定啟用 TLS v1.0；
JDK 1.8：支援 TLS 1.1 及 TLS 1.2，內定啟用 TLS v1.2。

在 Java 程式中可經由 SSLContext ssl_ctx = SSLContext.getInstance("TLSv1.2"); 指定 TLS v1.2 安全加密連線。可參考後述 5.2 及 5.3 章節範例。

3. Protocol 說明

目前 IMSP MMS Gateway 支援六種 SSL request interface，分別是 login、send_mms、query、query_batch、get_mime、及 change_passwd。Gateway 在執行後會將結果傳回給 client。執行結果包括一個數字的結果代號 (return code) 和一個字串的描述 (description)，兩者以
 隔開。return code = 0 代表 request 執行成功。return code > 0 代表 IMSP MMS Server 成功的執行了 request，但是依照不同的 request 可能產生成功以外的結果 (請參考下面各個 request 的說明)。若 return code < 0 則代表 IMSP MMS Gateway 無法成功地完成 request，除了每個 request 都有各自的失敗代碼外，下表列出了所有的 request 都有可能產生的 return code 和處理方式。

Return code	失敗原因	處理方式
-1	IMSP MMS Gateway 收到了一個錯誤的 HTTP packet。	請檢查您送到 IMSP MMS Gateway 的 HTTP request packet 有沒有錯誤，最常見的錯誤為編碼錯誤，例如說送出了%7 這種編碼(正確為%07)。
-2	某個參數錯誤	請參考 description 的敘述檢查您送到 IMSP MMS Gateway 的參數是否正確。通常會造成這種錯誤的原因包括： 1. IMSP MMS Gateway 沒有收到一個必須要有的參數(也有可能是參數名稱錯誤。提醒您系統處理參數的名稱大小寫是有分別的)； 2. 參數格式錯誤(例如 IMSP MMS Gateway 預期收到 PLMN address，但是您送了一個 e-mail address)； 3. 參數未作 x-www-form-urlencoded 編碼或是編碼錯誤等。
-3	資料庫錯誤	當您遇到這種錯誤，請和我們的服務人員聯絡，謝謝。
-4	認證失敗	IMSP MMS Gateway 無法認證您的身分，可能原因包括： 1. 您在參數列中的帳號密碼有誤，請確定做任何 request 時都需帶帳號密碼。 1. 請參考 "3.1 測試帳號"，利用瀏覽器直接鍵入網址 https://imsp.emome.net:4443/imsp/mms/servlet/login?account=xxxxx&password=xxxxx 來測試帳號是否有誤。
-19	未知的錯誤	IMSP MMS Server 發生了未知的錯誤，請和我們的技術人員聯絡，謝謝。

3.1 測試帳號(Login Test)

<table>
<tr><td>命令</td><td colspan="6">說明</td><td>備註</td></tr>
<tr><td rowspan="5">要求</td><td>URL Path</td><td colspan="5">https://imsp.emome.net:4443/imsp/mms/servlet/login</td><td rowspan="4"></td></tr>
<tr><td>HTTP Method</td><td colspan="5">POST 或 GET</td></tr>
<tr><td rowspan="3">Parameters</td><td>名稱</td><td>描述</td><td>資料型態</td><td>資料長度限制</td><td>初值設定</td></tr>
<tr><td>account</td><td>帳號</td><td>ASCII 字串</td><td>10</td><td>無</td></tr>
<tr><td>password</td><td>密碼</td><td>ASCII 字串</td><td>10</td><td>無</td></tr>
<tr><td>回應</td><td colspan="6"><html> <header></header> <body> {return code}
 {description}
 </body> </html></td><td rowspan="4"></td></tr>
<tr><td>Return Code</td><td colspan="2">Description</td><td>失敗原因</td><td colspan="2">處理方式</td></tr>
<tr><td>0</td><td colspan="2">Login succeed from {your ip}</td><td>您的帳號是可登入的。</td><td colspan="2"></td></tr>
<tr><td>-21</td><td colspan="2">Login failed</td><td>您傳送給 MMS Gateway 的帳號或密碼不正確。</td><td colspan="2">請檢查您送給 MMS Gateway 的帳號和密碼是否正確，請注意帳號和密碼是有區分大小寫的。</td></tr>
</table>

	-22	Illegal IP Address	您的來源 IP 是不被允許的。	請從登錄的 IP Address 連接 MMS Gateway。(可洽本公司設定您的固定式 IP 清單，以強化您的帳號安全)	
	-23	Account disabled	這個帳號已停用。	請聯絡我們的管理人員。	

範例一：測試帳號成功

傳送：

https://imsp.emome.net:4443/imsp/mms/servlet/login?account=1xxxx&password=xxxxx

回應：

<html> <header></header> <body>0
 Login succeed from xx.xx.xx.xx
 </body> </html>

範例二：測試帳號失敗

傳送：

https://imsp.emome.net:4443/imsp/mms/servlet/login?account=1xxxx&password=xxxxx

回應：

<html> <header></header> <body>-21
 Login failed
 </body> </html>

3.2 傳送 MMS 命令 –1 (send_mms)

命令	說明						備註
要求	URL Path	https://imsp.emome.net:4443/imsp/mms/servlet/send_mms					URL 編碼字串是以 x-www-form-url encoded 方式編碼的字串
	HTTP Method	POST (不支援 GET)					
	Parameters	名稱	描述	資料型態	資料長度限制	初值設定	
		account	帳號	ASCII 字串	10	無	
		password	密碼	ASCII 字串	10	無	
		from_addr	如下註 1	[0~9]數字串	35	無	
		from_addr_type	如下註 2	0 或 1	1	0	
		to_addr	如下註 3	[0~9]數字串	35	無	
		to_addr_type	如下註 4	0 或 1	1	0	
		cp_id	如下註 5	[0~9]數字串	3	000	
		subject	如下註 6	URL 編碼字串	40	無	
		expire_offset	如下註 7	[0~9]數字串	無	0	
		message_type	如下註 8	0 或 1	1	0	
		presentation_id	如下註 9	ASCII 字串	100	000	

		no_contents	如下註 10	[0~9]數字串	無	無	
		contentXid	如下註 11	ASCII 字串	100	無	
		contentXtype	如下註 12	ASCII 字串	50	無	
		contentX	如下註 13	URL 編碼字串	無	無	
回應	Response—<html> <header></header> <body> {return code} {description} </body> </html>						
	Return Code	Description	失敗原因	處理方式			
	0	一個數字形的 message id，此 message id 在查詢訊息狀態時需要傳給 MMS Gateway。	此筆訊息已成功接收，MMS Gateway 將會為您發送此訊息到所有的 to address。				
	1	一個(message id)之後接著多個(to address:fail reason)，視您的 MMS 訊息中有幾個失敗的 to address 而定。未列在此處的 to address 表示已成功接收，其訊息代號為 message id。	無法為您傳送到部分的 to address，最常見的原因有您填的 to address type 不是 PLMN，或是您的帳號無此權限。	請參考 fail reason 重送正確的 to address。			

	-5	Content Size 過大	MMS 所含 Content Size 過大(content size 總合超過 capacity 限制)。	請刪減 content 內容。每支手機可接受的 MMS size 並不一定，建議至大不要超過 100K，Gateway 所檢查的 content size 並不是針對收方手機的能力，而只是一個概括的檢查。	
	-6	Prepaid System Error	預付式系統發生錯誤	企客預付式系統或放心講客戶扣款失敗，可能原因為餘額不足或預付式系統介接異常。	
	-7	To_addrs Exceed	受話號碼(to_addr)超過限制，目前限制 20 組。	請修正每次 request 至多帶 20 組號碼(號碼間用逗號隔開)，分多次傳送。	
	-8	Can't determine customer type	系統無法查詢客戶屬性。	MMS Gateway 無法查詢客戶屬性。	

	-9	Can't determine msisdn type	系統無法查詢計費/受訊號碼之網內/外資訊	MMS Gateway 無法查詢計費/受訊號碼之網內/外資訊	
	-10	Service not provisioned in prepaid system	系統禁止此特碼使用預付式扣款服務	MMS Gateway 禁止此特碼使用預付式扣款服務。若有疑問請洽行通特碼 PM 人員。	
	-41	Permission Denied	您的帳號無此權限。	請不要再發送這筆訊息，若您需要提昇權限，請聯絡服務人員。	

註 1. from address：這通 MMS 訊息的發送者的號碼，此欄位不包含 /TYPE=PLMN 字串。由於此欄位關係到收費對象，請謹慎填寫。

註 2. from address type：發送者號碼 (from address) 的種類，MMS Server 可以接受兩種之一發送者號碼。當此欄為 0 時代表 from address 是一個手機門號；當此欄為 1 的時候代表 from address 是一個 emome 代碼，系統會自動轉換至對應的門號。故不管此欄位設定為 0 或 1，受訊端手機看到的 MMS 發訊方號碼會是行動電話號碼。

註 3. to address：這通 MMS 訊息的發送對象，此欄位不包含 /TYPE=PLMN 字串，可以接受 09 開頭、886 開頭、+886 開頭三種格式 (皆指國內台灣門號)。若您有足夠的權限，此欄可以接受其他電信業者的門號。此

欄可以包含一個以上的門號，以 ASCII 的逗號 (,) 隔開，但至多不超過 100 個門號，每一個 to address 都要記一通 MMS 傳送費用。

註 4. to address type：發送對象號碼 (to address) 的種類，MMS Gateway 可以接受兩種發送對象號碼，當此欄為 0 時代表 to address 是一個手機門號，當此欄為 1 的時候代表 to address 是一個 emome 代碼。為保持回溯相容，若沒有傳送這個參數，系統自動內定此參數為 0(發送對象為手機門號)。請注意若此參數為 1 時 (發送對象為 emome 代碼)，to address 只能接受一個號碼，否則系統會回應參數錯誤。

註 5. cp id：若您是為其他 content provider 代發，請在此欄填入 content provider 的代碼。若否，請在此欄填入 000。

註 6. subject：這通 MMS 訊息的 subject，以 UTF-8 作為編碼方式(可含有中文)，不可為空字串。以 java 程式為例，即 URLEncoder.encode(strSubject.getBytes("UTF-8"))

註 7. expire offset：這通 MMS 訊息何時(幾分鐘後)會失效。MMS 訊息失效的時間以此訊息進入 MMS Gateway 的時間為準，以分鐘為單位。若此欄位為 0 ，則使用 MMSC 的預設值。

註 8. message type：這通 MMS 訊息的訊息種類，如果該訊息為 multipart mixed，此欄設為 0；如果該訊息為 multipartrelated (含 SMIL)，此欄設為 1。

NOTE：要在 Android 手機達到最多相容性，建議使用 SMIL，即其值為 1，可不需另指定 presentation id 及 content SMIL 內容。

註 9. presentation id：若這一通 MMS Message 是一個包含有 SMIL 的 message 的話，請在這一欄填入 SMIL content 的 content id。若 server 沒有收到這個參數，則這個參數會被設為 "<0000>" 這個字串。

註 10. number of contents：這通 MMS 訊息包含的 content 數量。

註 11. contentXid：X 為一個數字，第一個 content 為 content0id，第二個為 content1id，其餘類推。contentXid

的個數必須符合 number of content(也就是說要有 content0id~content(number of content-1)id)。此 content id 為一個字串，主要用途是在 message type 為 multipart related 時作為 SMIL presentation 的參考對象。

註 12. contentXtype : X 為一個數字，第一個 content 為 content0type，第二個為 content1type，其餘類推。ContentXtype
的個數必須符合 number of content(也就是說要有 content0type~content(number of content-1)type)。此參數指出 content 的格式，例如 text/plain 等。

註 13. contentX : X 為一個數字，第一個 content 為 content0，第二個為 content1，其餘類推。ContentX 的個數必須符合 number of content (也就是說要有 content0~content(number of content-1))。若該 content 的 content type 為 text/* 的話，請用 UTF-8 編碼送出；即以 java 程式為例，即 URLEncoder.encode(strText.getBytes("UTF-8"))。其他圖檔及聲音檔請直接讀出 byte[] 資料，再經由 URLEncoder.encode() 送出。

可參考後述 5.2 及 5.3 章節之 JSP 傳送 MMS 程式範例說明。

範例一：發送單一門號

傳送：

https://imsp.emome.net:4443/imsp/mms/servlet/get_mime?account=1xxxx&password=xxxxx

回應：

範例一：發送多個門號

傳送：

https://imsp.emome.net:4443/imsp/mms/servlet/get_mime?account=1xxxx&password=xxxxx

回應：

3.3 查詢 MMS 命令 –1 (query)

命令	說明						備註
要求	URL Path	https://imsp.emome.net:4443/imsp/mms/servlet/query					
	HTTP Method	POST (不支援 GET)					
	Parameters	名稱	描述	資料型態	資料長度限制	初值設定	
		account	帳號	ASCII 字串	10	無	
		password	密碼	ASCII 字串	10	無	
		message_id	如下註 1	[0~9]數字串	10	無	
		to_address_type	如下註 2	0 或 1	1	0	

<table>
<tr><td></td><td></td><td>to_address</td><td></td><td>[0~9]數字串</td><td>35</td><td>無</td><td></td></tr>
<tr><td>回應</td><td colspan="6"><html> <header></header> <body> {return code}
 {description}
 </body> </html>
<table>
<tr><th>Return Code</th><th>Description</th><th>失敗原因</th><th>處理方式</th></tr>
<tr><td>0</td><td>Initial state</td><td>MMS Server 已成功接收此筆訊息並處理中。</td><td>〔視為發訊中〕</td></tr>
<tr><td>1</td><td>Submitting</td><td>MMS Server 正在轉送此筆訊息。</td><td>〔視為發訊中〕</td></tr>
<tr><td>2</td><td>Submitted to MMSC</td><td>此筆訊息已成功轉送至 MMSC。</td><td>〔視為發訊中〕</td></tr>
<tr><td>3</td><td>Submitting failed</td><td>此筆訊息無法轉送至 MMSC。</td><td>〔視為發訊失敗〕大部份情況為受訊端手機未啟用 GPRS 上網功能等，使得無法傳送 MMS 至受訊端。</td></tr>
<tr><td>4</td><td>Message pending</td><td>此筆訊息正等待接收端接收。</td><td>〔視為發訊中〕</td></tr>
</table></td><td></td></tr>
</table>

	5	Message delivered	此筆訊息已成功發送至接收端。	〔視為發訊成功〕	
	6	Message expired	此筆訊息無法在時間內發送至接收端。	〔視為發訊失敗〕	
	7	Message rejected	接收端拒絕接受此筆訊息。	〔視為發訊失敗〕	
	8	Unknown state	此筆訊息狀態不明。	〔視為無法得知〕	
	-31	Message not found	無法找到您要找的訊息。	請確定您所傳送的 Message id 和 to address 是否正確。	

註 1. message id：MMS message 的 message id，由 MMS Server 在成功的 MMS 傳送請求時傳回。to address : MMS message 的 to address，只包含 address，不包含 /TYPE=PLMN 等字串。

註 2. to address type：發送對象號碼 (to address) 的種類，MMS Gateay 可以接受兩種發送對象號碼，當此欄為 0 時代表 to address 是一個手機門號，當此欄為 1 的時候代表 to address 是一個 emome 代碼。為保持回溯相容，若沒有傳送這個參數，系統自動內定此參數為 0 (發送對象為手機門號)。

範例一：查詢單一門號結果

傳送：

https://imsp.emome.net:4443/imsp/mms/servlet/get_mime?account=1xxxx&password=xxxxx

回應：

<html> <header></header> <body>1
 No MO message found
 </body> </html>

範例一：查詢多個門號之一結果

傳送：

https://imsp.emome.net:4443/imsp/mms/servlet/get_mime?account=1xxxx&password=xxxxx

回應：

<html> <header></header> <body>1
 No MO message found
 </body> </html>

3.4 查詢 MMS 命令 – 2 (query_batch)

命令	說明						備註
要求	URL Path	https://imsp.emome.net:4443/imsp/mms/servlet/query_batch (SSL)					與 query 不同的是，query_batch 會傳回此 msgid 的所有訊息狀態 (如果受訊是多方)，query_batch 不適合用在 Emome 平
	HTTP Method	POST (不支援 GET)					
	Parameters	名稱	描述	資料型態	資料長度限制	初值設定	
		account	帳號	ASCII 字串	10	無	
		password	密碼	ASCII 字串	10	無	

<table>
<tr><td></td><td></td><td>message_id</td><td>如下註 1</td><td>[0~9]數字串</td><td>10</td><td>無</td><td>台。</td></tr>
<tr><td rowspan="4">回應</td><td colspan="6"><html> <header></header> <body> {return code}
 {description}
 </body> </html></td><td rowspan="4"></td></tr>
<tr><td>Return Code</td><td colspan="2">Description</td><td>失敗原因</td><td colspan="2">處理方式</td></tr>
<tr><td>> 0</td><td colspan="2">$msisdn,$status,$desc;{$msisdn,...} 如註 2.</td><td>成功查詢到 n 筆資料</td><td colspan="2"></td></tr>
<tr><td>-31</td><td colspan="2">Message not found</td><td>無法找到您要找的訊息。</td><td colspan="2">請確定您所傳送的 Message id 是否正確。</td></tr>
</table>

註 1. message id：MMS message 的 message id，由 MMS Gateway 在成功的 MMS 傳送請求時傳回。to address : MMS message 的 to address，只包含 address，不包含/TYPE=PLMN 等字串。

註 2. Return code 表示總共查詢到幾筆資料，查到的資料會以 msisdn,status,desc 這樣的 pattern 顯示，二筆資料間以”;” 隔開，status 及 desc 可參考 3.4 節。底下為一 example，表成功查詢到兩筆資料：
<html><header></header><body>2
0934444444,7,Message rejected;0937044059,2, Submitted to MMSC
 </body></html>

範例一：查詢批次發送只有一個門號結果

傳送：

https://imsp.emome.net:4443/imsp/mms/servlet/get_mime?account=1xxxx&password=xxxxx

回應：

<html> <header></header> <body>1
 No MO message found
 </body> </html>

範例二：查詢批次發送有多個門號結果

傳送：

https://imsp.emome.net:4443/imsp/mms/servlet/get_mime?account=1xxxx&password=xxxxx

回應：

<html> <header></header> <body>1
 No MO message found
 </body> </html>

3.5 擷取 MMS 命令 (get_mime)

命令	說明						備註
要求	URL Path	https://imsp.emome.net:4443/imsp/mms/servlet/get_mime					從 Http Response code 檢查是否有未收之 MMS。
	HTTP Method	POST (不支援 GET)					
	Parameters	名稱	描述	資料型態	資料長度限制	初值設定	
		account	帳號	ASCII 字串	10	無	
		password	密碼	ASCII 字串	10	無	

<table>
<tr><td rowspan="3">回應</td><td colspan="4"><html> <header></header> <body> {return code}
 {description}
 </body> </html></td><td rowspan="3">若有屬於您的帳號的 mms 的話，http response code 為 200，content 則為一個 binary 的 eaif /mime packet。若沒有屬於您的帳號的 mms 的話或是發生了其他的錯誤的話，http response code 為 202</td></tr>
<tr><td>Return Code</td><td>Description</td><td>失敗原因</td><td>處理方式</td></tr>
<tr><td>1</td><td>No MO message found</td><td>沒有您的 MMS 訊息</td><td></td></tr>
</table>

註： get_mime 所傳回的是符合 MIME (Multipurpose Internet Mail Extensions)格式之 MMS 封包，MIME 有 RFC 標準，decoder 取得也較方便，將 Server 所回傳之封包存成 ".eml" 的檔案即可使用 mail tools (如 outlook express) 瀏覽 MMS 內容，相當方便，在〈附錄一〉裡有更多的說明，網站上也有相關的 Java 程式範可參考，其網址為 http://imsp.emome.net:8008/imsp_home/mms_cp_help.htm。

範例一：收 MMS 無資料時

傳送：

https://imsp.emome.net:4443/imsp/mms/servlet/get_mime?account=1xxxx&password=xxxxx

回應：

<html> <header></header> <body>1
 No MO message found
 </body> </html>

範例二：收 MMS 有資料時

傳送：

https://imsp.emome.net:4443/imsp/mms/servlet/get_mime?account=1xxxx&password=xxxxx

回應：

3.6 變更密碼命令(change_passwd)

命令	說明						備註
要求	URL Path	https://imsp.emome.net:4443/imsp/mms/servlet/change_passwd					
	HTTP Method	POST or GET					
	Parameters	名稱	描述	資料型態	資料長度限制	初值設定	
		account	帳號	ASCII 字串	8	無	
		password	原密碼	ASCII 字串	8	無	

<table>
<tr><td></td><td></td><td>newcode</td><td>新密碼</td><td>ASCII 字串</td><td>8</td><td>無</td><td></td></tr>
<tr><td>回應</td><td colspan="6">➢ 變更密碼成功
<html><header></header> <body> 0
 Successful
 </body> </html>
➢ 變更密碼失敗
<html> <header></header> <body> {return code}
 {description}
 </body> </html></td><td></td></tr>
<tr><td></td><td colspan="2">Return Code</td><td colspan="2">Description</td><td>失敗原因</td><td>處理方式</td><td></td></tr>
<tr><td></td><td colspan="2">0</td><td colspan="2">Successful</td><td>變更密碼成功。</td><td>下次請改用 newcode 給予的新密碼進行介接服務。
註：當完成密碼變更後，請至多等待一小時，使整個 IMSP 系統套用您的新密碼。</td><td></td></tr>
<tr><td></td><td colspan="2">-19</td><td colspan="2">System error (reason)</td><td>系統故障，可能會包含額外資訊描述。</td><td>請聯絡技術人員。</td><td></td></tr>
<tr><td></td><td colspan="2">-21</td><td colspan="2">Login failed</td><td>帳號或是密碼錯誤</td><td></td><td></td></tr>
<tr><td></td><td colspan="2">-22</td><td colspan="2">Illegal IP Address</td><td>你的登入 IP 未在系統註冊</td><td></td><td></td></tr>
</table>

	-23	Account disabled	帳號已停用		
	-33	use https only	不允許用 http 變更密碼	請改用 https 變更密碼	
	-34	Failed to change password	變更密碼作業沒有成功。	新密碼須包括英文、數字、及特殊符號，至少八碼，至多 20 碼。	

<範例 1>

傳送：

https://imsp.emome.net:4443/imsp/mms/servlet/change_passwd?account=1xxxx&password=xxxxx&newcode=xxxxx

回應：

<html> <header></header> <body>0
 Successful
 </body> </html>

<範例 2>

傳送：

https://imsp.emome.net:4443/imsp/mms/servlet/change_passwd?account=1xxxx&password=xxxxx&newcode=123456

回應：

<html> <header></header> <body>-34
 Failed to change password
</body> </html>

4. 其它說明

4.1 關於 MMS 訊息

- 若發送方號碼帶的是 emome subscriber number (EMOME 平台專用) 的話，請記得把 from address type 設為 1，否則會造成錯誤。若您使用的是 send_eaif command 的話，請在 EAIF packet 裡的 from address 欄填入 emome subscriber number，並把 address type 設為 PLMN，和 semd_mms command 一樣，請記得把 from address type 設為 1。
- 在 3GPP 的 standard 中並沒有規範手機必須支援的聲音格式，經過實測，各家手機的支援格式也各不相同。發送之前請自行針對設定的手機作測試，以免得到無法預期的結果。

4.2 其他注意事項

- IMSP MMS 的功能為傳送 MMS 訊息到指定的門號上，沒有包含網頁展示等其他功能。
- 基於 http protocol 傳送上的需要，所有的 parameter 必須以 x-www-form-urlencoded 方式編碼，詳細編碼方式請參閱 http://www.w3.org/TR/html4/，或參考說明網頁上的 Java 範例。
- 請注意 parameter name 全部為小寫。

4.3 常見問題集

A. 無法 login?

請確定下列事項 : 1. 是否由註冊的 IP address 連接，若否的話，server 會擋掉連線。2. Parameter 參數名稱有區分大小寫。3. Parameter 參數值不包含大括號 (上面說明的大括號只是為了清楚表達用)。4. 用正確的帳號和密碼連接。

B. 收到 http packet error 的錯誤訊息？

server 收到一個錯誤編碼的 http packet，最常見的原因是收到%7 之類的字串(正確字串應該是%07)，URL encoded 編碼請參考在 sample code 裡的 String UrlEncode(byte[]) 函式。

C. 收到 eaif packet error 的錯誤訊息？

要不是 server 沒有收到 eaif_packet 這個 parameter，就是 server 收到了一個無法解譯的 eaif packet。

D. send_mms request 的 presentation id 是作什麼的？

若這一通 MMS Message 是一個包含有 SMIL 的 message 的話，請在這一欄填入 SMIL content 的 content id。若 server 沒有收到這個參數，則這個參數會被設為 "<0000>" 這個字串。經過測試某些手機如 Nokia 7650 會自動尋找 SMIL content，而某些手機如 Sharp i98 則需要這個參數來指明 SMIL content。為了安全起見，建議您不管手機型和為何，都指

定這個參數，以得到最大的相容性。

5. 附錄

5.1 GET_MIME 之 MMS 規格說明

請參考 Java 範例程式中的 "example_mime.eml" (https://imsp.emome.net:4443/imsp_home/mms_cp_help.htm 範例程式中可取得)，用文字編輯模式開啟。

```
Date: 2005/03/08 13:52:53 CST                    //  手機上載 MMS 的時間
To: 44441
From: 0000013969                                 //  上載的手機號碼/Emome-id <註一>
Msgid: 96625                                     //    message id,檔案名稱會以此命名
Subject: =?utf-8?B?6ZaD5YWJ5LiA6Jmf?=           //   “閃光一號”，編碼規格詳見<註二>
Content_number: 3
Mime-version: 1.0
Content-type: multipart/mixed; boundary="----=_CHT_MMS_BOUNDARY.96625"

------=_CHT_MMS_BOUNDARY.96625                   // boundary
Content-type: application/smil
Content-transfer-encoding: base64                // content  一律使用 base64
Content-disposition: attachment; filename="96625_0.smil"
Content-id: "<t0>

PHNtaWw+CiA8aGVhZD4KICA8bGF5b3V0PgogICA8cm9vdC1sYXlvdXQgaGVpZ2h0PSIxMjciIHdp
ZHRoPSIxMTEiLz4KICAgPHJlZ2lvbiBpZD0iSW1hZ2UiIHRvcD0iNTAlIiBsZWZ0PSIwJSIgaGVp
Z2h0PSI1MCUiIHdpZHRoPSIxMDAlIi8+CiAgIDxyZWdpb24gaWQ9IlRleHQiIHRvcD0iMCUiIGxl
ZnQ9IjAlIiBoZWlnaHQ9IjUwJSIgd2lkdGg9IjEwMCUiLz4KICA8L2xheW91dD4KIDwvaGVhZD4K
CTxib2R5PgoJCTxwYXIgZ….(以下略)
```

------=_CHT_MMS_BOUNDARY.96625
Content-type: text/plain
Content-transfer-encoding: base64
Content-disposition: attachment; filename="96625_1.txt"
Content-id: t1.txt

6YCZ5piv6ZaD5YWJ5LiA6Jmf

“這是閃光一號” (utf-8 編碼,base64) <註三>

------=_CHT_MMS_BOUNDARY.96625
Content-type: image/jpeg
Content-transfer-encoding: base64
Content-disposition: attachment; filename="96625_2.jpg"
Content-id: picture014.jpg

/9j/4QEQRXhpZgAASUkqAAgAAAAGABoBBQABAAAAVgAAABsBBQABAAAAXgAAACgBAwABAA
AAAgAAADIBAgAUAAAAZgAAABMCAwABAAAAAQAAAGmHBAABAAAAegAAAAAAAABIAAA
AAQAAAEgAAAABAAAAMjAwNTowMzowODoxMzo0OTo0NQAIAACQBwAEAAAAMDIxMAOQAgA
UAAAA4AAAAASQAgAUAAAA9AAAAAGRBwAEAAAAAQIDAACgBwAEAAAAMDEwMAGgAwAB
AAAAAQAAAAKgAwABAAAA4AEAAAOgAwABAAAAgAIAAAAAAAAyMDA1OjAzOjA4OjEzOjQ5Oj
Q1ADIwMDU6MDM6MDg6MTM6NDk6NDUA/9sAhAAGBAUFBQQGBQUFBwYGBwkPCgkICAkSDQ4
LDxYTFxcVExUVG (…..以下略)

------=_CHT_MMS_BOUNDARY.96625-- // 結尾的 boundary

註 1. From 有兩種表示方法，一為手機號碼(如 0937xxxxxx)，一為 emome-subid (如 000300xxxx)，會收到手機號碼或是 emome-subid 是可設定的，若發現收到的種類不是預期希望，請聯絡行通服務人員進行設定。

註 2. SUBJECT 編碼方式符合 RFC 2047 標準，即 'encoded-word'，舉 "=?utf-8?B?6ZaD5YWJ5LiA6Jmf?=" 為例，一開始的 "=?" 及結尾的 "?=" 為 encoded-word 的開始及結尾，然後兩個 "?" 將剩餘部份切成三部份，第一部份指明文字編碼方式，即 "utf-8"，第二部份的 "B" 表示 Base64，第三部份即真正的內文部份。MMS 的 subject 一律以 UTF-8 及 Base64 編碼。

註 3. TEXT 皆以 UTF-8 編碼，包括 SUBJECT 及內文部份 (UTF-8 編碼後再以 Base64 編碼)。

- MIME 格式可參考 RFC 2045、RFC 2046、RFC 2047，手動解碼時，請注意換行為 CRLF，ASCII 分別為 13、10。

5.2 JSP 傳送 Mixed MMS 程式範例

以下以 jsp 檔為例，說明如何經由 HTTPS 傳送 Mixed MMS 訊息，即類似 email 附加檔功能。

```
<%@ page contentType="text/html; charset=utf-8" pageEncoding="utf-8"%>
<%@ page
    import="java.io.BufferedReader,java.io.InputStreamReader,java.io.RandomAccessFile"%>
```

```
<%@ page
    import="java.net.URL,java.net.HttpURLConnection,java.io.DataOutputStream"
    import="java.io.FileInputStream,java.io.IOException"
    import="java.net.URLEncoder"%>
<%@ page
    import="javax.net.ssl.HostnameVerifier,javax.net.ssl.HttpsURLConnection"
    import="javax.net.ssl.SSLContext,javax.net.ssl.TrustManager"%>
<%@ page
    import="javax.net.ssl.SSLPeerUnverifiedException,javax.net.ssl.SSLSession"
    import="javax.net.ssl.X509TrustManager"%>
<%@ page
    import="java.security.SecureRandom,java.security.cert.Certificate"
    import="java.security.cert.X509Certificate"%>
<%!
    private String readTextFile(String strFilePath) {
        BufferedReader reader = null;
        StringBuffer sb = new StringBuffer();

        try {
            String strFileCharset = "UTF-8"; //文字檔請用 UTF-8 編碼儲存

            reader = new BufferedReader(new InputStreamReader(new FileInputStream(strFilePath),
                    strFileCharset));
```

```
            int c;

            while ((c = reader.read()) != -1) {
                sb.append((char) c);
            }
        } catch (Exception e) {
            e.printStackTrace();
        } finally {
            if (reader != null) {
                try {
                    reader.close();
                } catch (Exception e) {
                }
                reader = null;
            }
        }

        return sb.toString();
    }

    private byte[] ReadBinaryContentFile(String strFilePath) {
        RandomAccessFile rf = null;
```

```
byte[] buf = null;

try {
    rf = new RandomAccessFile(strFilePath, "r");

    int nFileSize = (int) rf.length();
    buf = new byte[nFileSize];

    for (int i = 0; i < nFileSize; i++) {
        buf[i] = rf.readByte();
    }
} catch (Exception e) {
    e.printStackTrace();
} finally {
    if (rf != null) {
        try {
            rf.close();
        } catch (Exception e) {
        }
        rf = null;
    }
}
```

```
    return buf; //二位元格式
}

private TrustManager[] get_trust_mgr() {
    TrustManager[] certs = new TrustManager[] { new X509TrustManager() {
        public X509Certificate[] getAcceptedIssuers() {
            return null;
        }

        public void checkClientTrusted(X509Certificate[] certs, String t) {
        }

        public void checkServerTrusted(X509Certificate[] certs, String t) {
        }
    } };

    return certs;
}

// HTTPS POST request
private void sendHttpSSLPost(String strUrl, String strParameters) throws Exception {
    System.err.println("\n=============== HTTP SSL POST TEST=========================");
```

```
//JDK 1.6_111 才支援 v1.1, 1.6_121 才支援 v1.2; JDK 1.7 有支援 v1.1 及 v1.2; JDK1.8 內定啟用 v1.2, 建議使用 TLSv1.2
SSLContext ssl_ctx = SSLContext.getInstance("TLSv1.2");
TrustManager[] trust_mgr = get_trust_mgr();

ssl_ctx.init(null, // key manager
         trust_mgr, // trust manager
         new SecureRandom()); // random number generator

HttpsURLConnection.setDefaultSSLSocketFactory(ssl_ctx.getSocketFactory());

System.out.println("\nSending SSL 'POST' request to URL: \n" + strUrl);
System.out.println("POST Parameters (length=" + strParameters.length() + "): "
         + strParameters);

URL url = new URL(strUrl);
HttpsURLConnection https = (HttpsURLConnection) url.openConnection();

https.setHostnameVerifier(new HostnameVerifier() {
     public boolean verify(String host, SSLSession sess) {
         if (host.equals("imsp.emome.net")) {
              return true;
         } else {
              return false;
```

```
        }
    }
});

//add request header
https.setRequestMethod("POST");

// Send post request
https.setDoOutput(true);

DataOutputStream wr = new DataOutputStream(https.getOutputStream());

wr.writeBytes(strParameters);
wr.flush();
wr.close();

int responseCode = https.getResponseCode();

System.out.println("Response Code: " + responseCode);

BufferedReader isr = new BufferedReader(
        new InputStreamReader(https.getInputStream()/*, "UTF-8"*/));
String inputLine;
```

```
        StringBuffer response = new StringBuffer();

        while ((inputLine = isr.readLine()) != null) {
            response.append(inputLine);
        }

        isr.close();
        https.disconnect();

        System.out.println("SSL POST Result: " + response.toString());
    }%>


<%
    String account = "1xxxx";  //請填您的特碼帳號
    String password = "xxxxxxxx";  //請填您的特碼密碼
    String cp_id = "xxx";  //請填您的 CP_ID 資訊 (內定 000)
    String expire_offset = "0";  //請填訊息重送期限

    String from_addr = "09xxxxxxxx";  //請填您的發訊方號碼
    String from_addr_type = "0";
    String to_addr = "09xxxxxxxx"; //請填您的受訊方號瑪
    String to_addr_type = "0";
```

```
byte[] baSubject = new String("含兩段文字、及一張圖-SMIL").getBytes("UTF-8");
String strSubject = new String(baSubject, "ISO-8859-1");

String message_type = "0";
String presentation_id = "";

String content0id = "T0";
String content0type = "text/plain";
byte[] ba0 = new String("此為要傳送的文字內容").getBytes("UTF-8");  //若要跳列字元, 傳送文字內容可含 \r\n 或 \n 表示式
String strContent0 = new String(ba0, "ISO-8859-1");

String content1id = "T1";
String content1type = "image/jpeg";
byte[] ba1 = ReadBinaryContentFile("C:\\myMMS.jpg"); //要傳送的圖形
String strContent1 = new String(ba1, "ISO-8859-1");

String content2id = "T2";
String content2type = "text/plain";
byte[] ba2 = new String("上圖為包括一張圖。").getBytes("UTF-8");
String strContent2 = new String(ba2, "ISO-8859-1");

int no_contents = 3;
```

```
try {
    StringBuffer sbPostData = new StringBuffer();

    sbPostData.append("account=" + account);
    sbPostData.append("&password=" + password);
    sbPostData.append("&cp_id=" + cp_id);
    sbPostData.append("&expire_offset=" + expire_offset);
    sbPostData.append("&from_addr=" + from_addr);
    sbPostData.append("&from_addr_type=" + from_addr_type);

    sbPostData.append("&message_type=" + message_type);
    sbPostData.append("&to_addr=" + to_addr);
    sbPostData.append("&to_addr_type=" + to_addr_type);
    sbPostData.append("&subject=" + URLEncoder.encode(strSubject, "ISO-8859-1"));
    sbPostData.append("&presentation_id=" + presentation_id);
    sbPostData.append("&no_contents=" + Integer.toString(no_contents));

    sbPostData.append("&content0id=" + content0id);
    sbPostData.append("&content0type=" + content0type);
    sbPostData.append("&content0=" + URLEncoder.encode(strContent0, "ISO-8859-1"));

    sbPostData.append("&content1id=" + content1id);
```

```
        sbPostData.append("&content1type=" + content1type);
        sbPostData.append("&content1=" + URLEncoder.encode(strContent1, "ISO-8859-1"));

        sbPostData.append("&content2id=" + content2id);
        sbPostData.append("&content2type=" + content2type);
        sbPostData.append("&content2=" + URLEncoder.encode(strContent2, "ISO-8859-1"));

        sendHttpSSLPost("https://imsp.emome.net:4443/imsp/mms/servlet/send_mms",
                        sbPostData.toString());
    } catch (Exception e) {
        e.printStackTrace();
    }
%>
```

5.3 JSP 傳送 SMIL MMS 程式範例

類似上述 5.2 章節，除了下列程式異動、及補充之外，使可傳送 SMIL 訊息，如：依一或多個版面顯示或播放訊息。

```
:
String message_type = "1";
```

```
    String presentation_id = "T3";    //若無此客制化版面需求，可不設定該值，交由手機自行處理 SMIL MMS 顯示。

    :

    String content3id = "T3";  //若無此客制化版面需求，可不設定這些資訊。
    String content3type = "application/smil";
    byte[] ba3 = new String("<smil><head></head><body><par><text src=\"T0\"/><img src=\"T1\"/><text
src=\"T2\"/></par></body></smil>").getBytes("UTF-8");
    String strContent3 = new String(ba3, "ISO-8859-1");

    int no_contents = 4;  //若無此客制化版面需求，請設 3 表示不包括 content3 資訊。

    try {
        :

        if (no_contents == 4) {
            sbPostData.append("&content3id=" + content3id);  //若無此客制化版面需求，可不設定這些資訊。
            sbPostData.append("&content3type=" + content3type);  //(同上)
            sbPostData.append("&content3=" + URLEncoder.encode(strContent3, "ISO-8859-1"));  //(同上)
        }

        sendHttpSSLPost("https://imsp.emome.net:4443/imsp/mms/servlet/send_mms",
                    sbPostData.toString());
```

```
} catch (Exception e) {
    e.printStackTrace();
}
```