Dokumentation

Udbyder	:	sms1919.dk
Service	:	push
Version	:	v1.21

Indholdsfortegnelse

Versionshistorik 3

Før du kalder API'en 4

- Oprettelse af API-nøgler 4
- Godkendelse af IP-adresse 4

Afsendelse af sms 5

- WapPush-parameter 6
- Mms-parametre 6
  - Eksempel på content 6
  - Eksempel på smil 6
- Binære beskeder 7
  - Eksempel på binær besked 7
- Valgfrie værdier 8
- Extended reply 9
- Callbacks 10

Få en liste over dine bestilte smser 12

- Værdier 12
- Retursvar 12

Slet forudbestilte smser 13

- Værdier 13

Kodeeksempler for push i ASP 14

- Geteksempel 14
- Posteksempel 14

Kodeeksempler for push i PHP 15

- Eksempel ved brug af PEAR Request 15

Kodeeksempler for smsqueue i ASP 16

- Geteksempel 16
- Posteksempel 16

Kodeeksempler for smsqueue i PHP 17

- Eksempel ved brug af PEAR Request 17

Kodeeksempler for dequeuesms i ASP 18

- Geteksempel 18
- Posteksempel 18

Kodeeksempler for dequeuesms i PHP 19

- Eksempel ved brug af PEAR Request 19

Versionshistorik

Dokumentationen bliver løbende opdateret, og her finder du datoer og kort beskrivelse af hvad der er blevet opdateret.

Dato og version	Ændringer
11. december, 2008 v1.00	Original udgivelse af sms1919-dokumentationen
1. marts, 2009 v1.01	Tilføjede php-eksempler
18. marts, 2009 v1.02	Tilføjede mulighed for extended reply fra APIen med messageid, m.v.
23. marts, 2009 v1.03	SMSC er nu et valgfrit parameter.
17. april, 2009 v1.04	Sikker SSL-krypteret forbindelse er etableret
22. april, 2009 v1.05	Afsnit om MMS-beskeder tilføjet Afsnit om Wap-Push tilføjet
26. august, 2009 v1.06	Nu mulighed for flere modtagere i et kald Understøttelse af flashbeskeder
1. september, 2009 v1.07	Understøttelse af binære beskeder
7. april, 2010 v1.08	Tilføjede mulighed for extended reply fra APIen ved afsendelse til flere modtagere Tilføjede statuskoder for MMS Tilføjede statuskoder for SMS
21. april, 2010 v1.09	Bemærkelse vedr. afsender-navn og iPhones tilføjet
29. februar, 2012 v1.10	Tilføjede encoding til php-eksempel og rettede fejl ved flash-parameter
10. september, 2015 v1.20	Opdatering af autoriseringsmetode (API-nøgle) samt kodeeksempler Opdatering af API-url (Krav om SSL mod api.sms1919.dk) Tilføjet afsnit om oprettelse af API-nøgler Tilføjet afsnit om godkendelse af IP-adresser
29. september, 2015 v1.21	Tilføjet mulighed for ændre standard encoding Opdaterede php-eksempel til at benytte sig af encoding-parameter

Før du kalder API'en

Inden du kalder API'en første gang er der et par ting der skal gøres. Først skal du oprette en API-nøgle. Denne kan du roligt give til 3.-part da nøglen ikke giver adgang til sms1919.dk, men udelukkende kan bruges til at kalde API'en med.

Herefter skal du godkende den IP-adresse som serveren som benytter API'en kalder fra. Herefter er du klar til at gå i gang med at benytte dig af API'en på sms1919.dk. Vi har her lavet et par hurtige forklaringer til hvordan du kommer igennem de to steps:

Oprettelse af API-nøgler

For at øge sikkerheden anbefaler vi at man benytter sig af API-nøgler når man bruger API'en. Du kan oprette lige så mange API-nøgler du vil og kan deaktivere dem midlertidigt eller slette dem igen så der ikke længere er adgang til API'en fra den pågældende nøgle.

Du opretter dine egne API-nøgler på sms1919.dk under "Indstillinger" og "API-nøgler". Vælg en nøgle af typen "Adgangsnøgle til API" og tryk på tilføj. Herefter fil nøglen blive tilføjet listen over dine API-nøgler og du kan frit bruge den til at kalde API'en med:



Godkendelse af IP-adresse

Når du kalder API'en hos sms1919.dk skal du sørge for at den IP-adresse du kalder fra er godkendt, ellers vil dit API-kald blive afvist med fejlen: "Invalid host". Godkendelsen kan du lave selv på dit login hvis du går ind under "Indstillinger" og "IP-sikkerhed":



Afsendelse af sms

Her ser du en oversigt over de værdier der skal sendes til vores API. Størstedelen af værdierne er påkrævet for at sende en sms. De skal sendes som enten `post` eller `get` til følgende adresse:

https://api.sms1919.dk/rpc/push/

Følgende værdier skal inkluderes:

Navn	Værdi	Type	Eksempel
apikey	Den API-nøgle du udsteder på dit login	Tekststreng	acf3da33(…)329fac3
recipient	Modtageren af beskeden, inklusiv landkode. Ønskes det at samme besked sendes til flere modtagere, kan de angives kommasepareret i dette felt.	Tal	4520304050 *eller* 4520304050,462030405060, 4530405060,4540506070
text	Den tekst som smsen skal indeholde. Bemærk at der takseres per påbegyndt besked. Beskeder deles som følger pga. måden de skal encodes på: 1-160 tegn: 1 besked 161-306 tegn: 2 beskeder 307-459 tegn: 3 beskeder Smser over 1530 vil blive afvist af systemet da ikke alle terminaler understøtter dette.	Tekststreng	Hej! Hvordan går det?

WapPush-parameter

Følgende parameter inkluderes derudover såfremt beskeden ønskes leveret som et Wap-Push link:

Navn	Værdi	Type	Eksempel
wapurl	URL til den hjemmeside/service/fil du ønsker brugeren skal have tilsendt	Tekststreng	http://site.dk/login

Mms-parametre

Følgende værdier skal udover sms-parametrene, inkluderes såfremt du ønsker beskeden skal udformes som en mms-besked. Begge parametre er XML-data som hhv. bestemmer hvilke filer der skal sendes, og hvordan de skal tages sig ud på terminalen.

Navn	Værdi	Type
content	Listen over filer der skal sendes med mms-beskeden.	Tekststreng
smil (valgfri)	SMIL-dokument jf. http://www.w3.org/AudioVideo/	Tekststreng

Eksempel på content

Et eksempel på et content-dokument.

```
<content>
 <text name="Text01.txt">Se mit billede i denne besked!</text>
 <image name="sms.image.jpeg">http://dit-domæne.dk/sms.image.jpeg</image>
</content>
```

Eksempel på smil

Et eksempel på et SMIL-dokument

```
<smil>
 <head>
  <layout>
   <root-layout width="240" height="320" />
   <region id="Image" width="100%" height="70%" left="0%" top="30%" fit="meet" />
   <region id="Text" width="100%" height="30%" left="0%" top="0%" fit="scroll" />
  </layout>
 </head>
 <body>
  <par dur="8000ms">
   <text src="Text01.txt" region="Text" />
   <img src="sms.image.jpeg" region="Image" />
  </par>
 </body>
</smil>
```

Binære beskeder

Med binære beskeder har du en unik mulighed for at afsende specialbeskeder som f.eks. Nokia vCards og lignende. Afsendelsen af den binære besked kræver blot at du kender portnummeret der skal sendes til på telefonen, og har den encodede tekststreng som udgør selve den binære besked.

Bemærk: Porten skal defineres som en decimalværdi, så hvis du er vant til at benytte porte med HEX værdier, som 158A skal disse konverteres. I feltet *text*, hvor du normalt skriver din sms-besked, angiver du din binære besked.

Navn	Værdi	Type	Eksempel
port	Porten som beskeden skal sendes til på telefonen	Tal	2948

Eksempel på binær besked

Følgende kunne være et eksempel på de gældende parametre for en binær sms i ASP.

```
<h1>Opbygning af binær streng</h1>

<%
    ' Build Parameters
    sParameters = "apikey="     & "4bfe1cf53258cda0fd32e15bf5416b50" &
                  "&price="     & "0"                               &
                  "&recipient=" & "4512345678"                      & _
                  "&port="      & "2948"                            & _
                  "&text="      & "500601AE02056A0045C60C037777772e736d73313931392e646b" & _
                                  "000AC307200901010000000103736d73313931392e646b000101"

%>
```

Valgfrie værdier

Hos sms1919.dk bliver vi ved med at udvide med nye features.

Her er en liste over valgfrie værdier som giver dig helt nye muligheder for at tilpasse dine services til din målgruppe. Ønsker du nye features er du altid velkommen til at kontakte kundeservice med dine ideer.

Navn	Værdi	Type	Eksempel
from	Afsenderen af beskeden som kan bestå af 11 karakterer eller 22 tal. Ikke alle tegn er tilladt. Ved telefonnumre som afsender, så husk at URL-encode '+' som '%2b'.	Tekststreng	sms1919 %2b4512345678
schedule	Giver mulighed for at forudbestille beskeden ved at angive det tidspunkt beskeden skal afsendes på.	Tal	20080516201000 *Angives åååmmddttnnss*
reschedule	Ønskes beskeden genudsendt, sættes en parameter for hyppigheden her. Dette kræver at `schedule` også er sat, ellers vil værdien blive ignoreret.	Tekststreng	daily
customid	Denne værdi kan sættes i forbindelse med forudbestilte smser. Id'et gemmes og bruges ifbm. sletning af sms'en.	Tekststreng	013442:4520304050:zA835X
callback_url	Ønskes et callback med en status når beskeden er leveret, angives URLen.	Tekststreng	http://domain.com/callback.php
callback_session	Har du en session fra dit eget system du ønsker at sende med kan den angives her.	Tekststreng	48234
reply	Hvis du ønsker at modtage et detaljeret svar i XML når du sender en besked kan du sætte denne værdi til `extended`. Sender du til flere modtagere sættes den til `extendedm`.	Tekststreng	extended *eller* extendedm
smsc	Det sms-center der skal sendes til. Det kan blandt andet være 'tdc' eller 'hi3g'. Benyttes kun ifbm. Indholdstakseringer.	Tekststreng	tdc
flash	Benyttes hvis beskeden ønskes fremsendt som en flashbesked, altså en besked der ikke bliver gemt. Angives der ikke noget sættes værdien til 'no'.	Tekststreng	yes *eller* no
shortcode	Den shortcode man ønsker at benytte.	Tal	1919
price	Prisen som smsen. Beløb angives i øre.	Tal	0 *(Standard)*
encoding	Som standard forventes Latin1-karaktersæt, men dette kan ændres til UTF8 ved hjælp af denne parameter	Tekststreng	utf8

Extended reply

Som beskrevet under "valgfrie værdier" er det muligt at få et udvidet svar fra APIen ved afsendelse af beskeder. Svaret kommer i stedet for værdien `SUCCESS` og er defineret ud fra følgende skabelon såfremt der sendes til en enkelt modtager:

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<result>
  <status>SUCCESS</status>
  <messageid>1027:81241</messageid>
  <messagecount>2</messagecount>
  <callback>
    <url>http://www.domain.com/callback.php</url>
    <session>51A78CF23</session>
  </callback>
</result>
```

Såfremt du ikke har angivet nogen `callback_url` vil sektionen med callback ikke være inkluderet. Sender du til flere modtagere, ser svar-XML'en en anelse anderledes ud. Bemærk, callback vil ikke blive vist i svar-XML'en uanset om den er sat eller ej, ved benyttelse af denne funktion.

Svar-XML'en for en besked til flere modtagere kan se således ud:

```
<?xml version="1.0" encoding="iso-8859-1"?>
<result>
  <message>
    <status>SUCCESS</status>
    <messageid>8:1927</messageid>
    <messagecount>1</messagecount>
    <recipient>4512345678</recipient>
  </message>
  <message>
    <status>SUCCESS</status>
    <messageid>8:1928</messageid>
    <messagecount>1</messagecount>
    <recipient>4512345679</recipient>
  </message>
</result>
```

Kolonnen `<messageid>` vil oftes oplevels fortløbende, men der er ingen garanti såfremt der sendes flere kald til API'en samtidigt.

Callbacks

Når der afsendes en besked fra systemet som der ønskes et callback på vil der blive sendt et kald til den angivede `callback_url` når vi modtager svar fra leverandøren med status på smsen. Benyttes eksemplet fra forrige side vil der blive kaldt følgende:

https://domain.com/callback.php?messageid=8:2945&session=482A&status={nedenstående_skema}

Status kan indeholde følgende værdier for indholdstakserede beskeder:

Værdi	Beskrivelse
1	Beskeden er afleveret
2	Ingen dækning hos modtager
3	Abonnenten er blacklisted
4	Abonnenten er ukendt
5	Ukendt SMSC-fejl
6	Gyldighedsperiode udløbet
7	Beskeden kan ikke leveres
8	Beskeden er annulleret
9	Beskeden er blevet slettet

Status kan indeholde følgende værdier for MMS-beskeder:

Værdi	Beskrivelse
34	Beskeden kunne ikke sendes til MMS-centralen
35	Telefonnummer eller operatør var ugyldig
36	Content-delen af MMSen var ikke gyldig
37	Beskeden blev afvist af MMS-centralen

Status kan indeholde følgende værdier for nul-takserede SMS-beskeder:

Værdi	Beskrivelse
136	Unknown subscriber
137 - 144	Service temporary not available
145	Illegal error code
146	Network time-out
147	Error in MS
148	Facility not supported
149	Unknown subscriber
150	Facility not provided
151	Call barred
152	Operation barred
153	SC congestion
154	Facility not supported
155	Absent subscriber
156	Delivery fail
157	SC congestion
158	Protocol error
159	MS not equipped
160	Unknown SC
161	SC congestion
162	Illegal MS
163	MS not a subscriber
164	Error in MS
165	SMS lower layer not provisioned
166	System fail
167	PLMN system failure
168	HLR system failure
169	VLR system failure
170	Previous VLR system failure
171	Controlling MSC system failure
172	VMSC system failure
173	EIR system failure
174	System failure
175	Unexpected data value
176	Error in address service centre
177	Invalid absolute Validity Period
178	Short message exceeds maximum
179	Unable to Unpack GSM message
180	Unable to convert IRA ALPHABET
181	Invalid validity period format
182	Invalid destination address
183	Duplicate message submit
184	Invalid message type indicator

Få en liste over dine bestilte smser

Med push-kaldet er der mulighed for at forudbestille en sms til en bestemt dato, på et bestemt tidspunkt. Med dette kald har du mulighed for at se en liste over de smser der endnu ikke er afsendt. Kaldet returnerer et XML-dokument som indeholder informationer om smserne der ligger og venter i køen.

Værdier

Her ser du en oversigt over de værdier der skal sendes til sms1919s API. Alle værdierne er påkrævet for at se en liste over forudbestilte smser. De skal sendes som enten `post` eller `get` til følgende adresse:

https://api.sms1919.dk/rpc/smsqueue/

Følgende værdier skal inkluderes:

Navn	Værdi	Type	Eksempel
apikey	Den API-nøgle du udsteder på dit login	Tekststreng	acf3da33(…)329fac3

Retursvar

Som retursvar kan du få et XML-feed der ser således ud:

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<queue>
  <sms>
    <scheduleid>717024b92794c041f63bd7693d4dee14</scheduleid>
    <customid>013442:4520304050:zA835X</customid>
    <created>11-02-2008 20:34:51</created>
    <scheduled>15-02-2008 22:00:00</scheduled>
    <shortcode>1919</shortcode>
    <smsc>TDC</smsc>
    <from>sms1919.dk</from>
    <recipient>4520304050</recipient>
    <price>0</price>
    <text>
      <![CDATA[Dette er en forudbestilt besked - til lige omkring kl. 22.00.]]>
    </text>
  </sms>
  <sms>
    <scheduleid>1b26c7dafdd83b53d52719f5f38a1767</scheduleid>
    <created>11-02-2008 20:34:58</created>
    <scheduled>15-02-2008 21:11:00</scheduled>
    <shortcode>1919</shortcode>
    <smsc>TDC</smsc>
    <from>sms1919.dk</from>
    <recipient>4520304050</recipient>
    <price>0</price>
    <text>
      <![CDATA[Dette er en anden forudbestilt besked - til omkring kl. 21.11.]]>
    </text>
  </sms>
</queue>
```

Slet forudbestilte smser

Med push-kaldet er der mulighed for at forudbestille en sms til en bestemt dato, på et bestemt tidspunkt. Med dette kald har du mulighed for at slette en sms, såfremt den stadigvæk ligger i kø og afventer at blive sendt.

For at kende det unikke id du skal bruge for at slette den enkelte sms, skal du benytte dig af kaldet smsqueue som du har læst om på forrige side.

Værdier

Her ser du en oversigt over de værdier der skal sendes til sms1919s API. Alle værdierne er påkrævet for at slette en sms. De skal sendes som enten `post` eller `get` til følgende adresse:

https://api.sms1919.dk/rpc/dequeuesms/

Følgende værdier skal inkluderes:

Navn	Værdi	Type	Eksempel
apikey	Den API-nøgle du udsteder på dit login	Tekststreng	acf3da33(…)329fac3
scheduleid serie: 1/2	Det unikke id du får fra smsqueue. Enten dette id, eller et customid skal medsendes.	Tekststreng	717024b92794c041f63bd7693d4dee14
customid serie: 2/2	Der kan også slettes ved hjælp af et customid hvis det blev sendt med da sms'en blev bestilt.	Tekststreng	013442:4520304050:zA835X

Kodeeksempler for push i ASP

Her finder du et eksempel på hvordan du benytter sms1919s gateway til at sende smser igennem ved hjælp af ASP 3.0.

Geteksempel

Dette eksempel viser hvordan man med `get`-parametre kan sende informationer til sms1919s API.

```
<h1>GET Example</h1>

<%
    ' Build Parameters
    sParameters = "apikey="     & "4bfe1cf53258cda0fd32e15bf5416b50" & _
                  "&shortcode=" & "1919"                            & _
                  "&price="     & "0"                               & _
                  "&text="      & Server.URLEncode("Her er beskeden.") &
                  "&recipient=" & "4520304050"

    ' Create Object
    set objHTTP = Server.CreateObject("MSXML2.ServerXMLHTTP")
        objHTTP.open "GET","https://api.sms1919.dk/rpc/push/?" & sParameters,false
        objHTTP.send

    ' Write Result
    Response.Write( objHTTP.responseText )

    ' Release Object
    set objHTTP = Nothing
%>
```

Posteksempel

Dette eksempel viser hvordan man med `post`-parametre kan sende informationer til sms1919s API.

```
<h1>POST Example</h1>

<%
    ' Build Parameters
    sParameters = "apikey="     & "4bfe1cf53258cda0fd32e15bf5416b50" &
                  "&shortcode=" & "1919"                            & _
                  "&price="     & "0"                               & _
                  "&text="      & Server.URLEncode("Her er beskeden.") & _
                  "&recipient=" & "4520304050"

    ' Create Object
    set objHTTP = Server.CreateObject("MSXML2.ServerXMLHTTP")
        objHTTP.open "POST","https://api.sms1919.dk/rpc/push/",false
        objHTTP.SetRequestHeader "Content-Type", "application/x-www-form-urlencoded"
        objHTTP.send sParameters

    ' Write Result
    Response.Write( objHTTP.responseText )

    ' Release Object
    set objHTTP = Nothing
%>
```

Kodeeksempler for push i PHP

Her finder du et eksempel på hvordan du benytter sms1919s gateway til at sende smser igennem ved hjælp af PHP.

Eksempel ved brug af PEAR Request

Dette eksempel viser hvordan man med "PEAR Request"[1] kan sende informationer til sms1919s API.

```
<h1>PEAR Request Example</h1>

<?php

    require_once('HTTP/Request.php');

    $req = & new HTTP_Request('https://api.sms1919.dk/rpc/push/');
    $req->setMethod(HTTP_REQUEST_METHOD_POST);
    $req->addPostData('apikey', '4bfe1cf53258cda0fd32e15bf5416b50');
    $req->addPostData('shortcode', '1919');
    $req->addPostData('price', '0');
    $req->addPostData('text', 'Her er beskeden.');
    $req->addPostData('recipient', '4520304050');
    $req->addPostData('encoding', 'utf8');

    $req->sendRequest();

    # Print the output
    echo $req->getResponseBody();

?>
```

[1] http://pear.php.net/package/HTTP_Request

Kodeeksempler for smsqueue i ASP

Her finder du et eksempel på hvordan du får en liste over forudbestilte smser igennem sms1919s gateway ved hjælp af ASP 3.0.

Geteksempel

Dette eksempel viser hvordan man med `get`-parametre kan sende informationer til sms1919s API.

```
<h1>GET Example</h1>

<%
    ' Build Parameters
    sParameters = "apikey=4bfe1cf53258cda0fd32e15bf5416b50"

    ' Create Object
    set objHTTP = Server.CreateObject("MSXML2.ServerXMLHTTP")
        objHTTP.open "GET","https://api.sms1919.dk/rpc/smsqueue/?" & sParameters,false
        objHTTP.send

    ' Write Result
    Response.Write( objHTTP.responseText )

    ' Release Object
    set objHTTP = Nothing
%>
```

Posteksempel

Dette eksempel viser hvordan man med `post`-parametre kan sende informationer til sms1919s API.

```
<h1>POST Example</h1>

<%
    ' Build Parameters
    sParameters = "apikey=4bfe1cf53258cda0fd32e15bf5416b50"

    ' Create Object
    set objHTTP = Server.CreateObject("MSXML2.ServerXMLHTTP")
        objHTTP.open "POST","https://api.sms1919.dk/rpc/smsqueue/",false
        objHTTP.SetRequestHeader "Content-Type", "application/x-www-form-urlencoded"
        objHTTP.send sParameters

    ' Write Result
    Response.Write( objHTTP.responseText )

    ' Release Object
    set objHTTP = Nothing
%>
```

Kodeeksempler for smsqueue i PHP

Her finder du et eksempel på hvordan du får en liste over forudbestilte smser igennem sms1919s gateway ved hjælp af PHP.

Eksempel ved brug af PEAR Request

Dette eksempel viser hvordan man med ”PEAR Request” kan sende informationer til sms1919s API.

```
<h1>PEAR Request Example</h1>

<?php

    require once('HTTP/Request.php');

    $req = & new HTTP_Request('https://api.sms1919.dk/rpc/smsqueue/ ');
    $req->setMethod(HTTP_REQUEST_METHOD_POST);
    $req->addPostData('apikey', '4bfe1cf53258cda0fd32e15bf5416b50');

    $req->sendRequest();

    # Print the output
    echo $req->getResponseBody();

?>
```

Kodeeksempler for dequeuesms i ASP

Her finder du et eksempel på hvordan du sletter forudbestilte smser igennem sms1919s gateway ved hjælp af ASP 3.0.

Geteksempel

Dette eksempel viser hvordan man med `get`-parametre kan sende informationer til sms1919s API.

```
<h1>GET Example</h1>

<%
    ' Build Parameters
    sParameters = "apikey=" & "4bfe1cf53258cda0fd32e15bf5416b50" & _
                  "&scheduleid=" & "717024b92794c041f63bd7693d4dee14"

    ' Create Object
    set objHTTP = Server.CreateObject("MSXML2.ServerXMLHTTP")
        objHTTP.open "GET","https://api.sms1919.dk/rpc/dequeuesms/?" & sParameters,false
        objHTTP.send

    ' Write Result
    Response.Write( objHTTP.responseText )

    ' Release Object
    set objHTTP = Nothing
%>
```

Posteksempel

Dette eksempel viser hvordan man med `post`-parametre kan sende informationer til sms1919s API.

```
<h1>POST Example</h1>

<%
    ' Build Parameters
    sParameters = "apikey=" & "4bfe1cf53258cda0fd32e15bf5416b50" & _
                  "&scheduleid=" & "717024b92794c041f63bd7693d4dee14"

    ' Create Object
    set objHTTP = Server.CreateObject("MSXML2.ServerXMLHTTP")
        objHTTP.open "POST","https://api.sms1919.dk/rpc/dequeuesms/",false
        objHTTP.SetRequestHeader "Content-Type", "application/x-www-form-urlencoded"
        objHTTP.send sParameters

    ' Write Result
    Response.Write( objHTTP.responseText )

    ' Release Object
    set objHTTP = Nothing
%>
```

Kodeeksempler for dequeuesms i PHP

Her finder du et eksempel på hvordan du sletter forudbestilte smser igennem sms1919s gateway ved hjælp af PHP

Eksempel ved brug af PEAR Request

Dette eksempel viser hvordan man med ”PEAR Request” kan sende informationer til sms1919s API.

```
<h1>PEAR Request Example</h1>

<?php

    require once('HTTP/Request.php');

    $req = & new HTTP_Request('https://api.sms1919.dk/rpc/dequeuesms/ ');
    $req->setMethod(HTTP_REQUEST_METHOD_POST);
    $req->addPostData('apikey', '4bfe1cf53258cda0fd32e15bf5416b50');
    $req->addPostData('scheduleid', '717024b92794c041f63bd7693d4dee14');

    $req->sendRequest();

    # Print the output
    echo $req->getResponseBody();

?>
```