

SPECYFIKACJA TECHNICZNA

WEB XML-API

GATESMS.EU
wersja 1.1

Spis

Historia wersji dokumentu.....	3
Bezpieczeństwo.....	3
Wymagania ogólne.....	3
Mechanizm zabezpieczenia transmisji HTTP.....	3
Zasady ogólne.....	3
Definicja usług zdalnych.....	4
Wysyłanie wiadomości do aplikacji Gatesms.eu.....	5
Przykład żądania – parametr body:.....	5
Przykład odpowiedzi – dokument XML.....	5
Uwagi do działania usługi.....	6
Przesyłanie raportów doręczenia.....	6
Przykład żądania – parametr body.....	7
Przesyłanie wiadomości przychodzących.....	7
Przykład żądania.....	7
Importowanie kontaktów do bazy Gatesms.eu.....	8
Kody statusów wiadomości.....	8
Długości wiadomości sms.....	9
Dodatek A.....	10
Klasa obsługująca Web XML Api.....	10

Historia wersji dokumentu

- 1.0 - wersja podstawowa 1.03.2011
- 1.1 - opis długości wiadomości sms, oraz opcji SPEEDsms 20.12.2011

Bezpieczeństwo

Wymagania ogólne

1. Komunikacja w obu kierunkach musi uwzględniać zabezpieczenia transmisji HTTP opisane poniżej.

Mechanizm zabezpieczenia transmisji HTTP

Zabezpieczenia dla usług zdalnych dostępnych przez HTTP są niezależne od metody i formatu przesyłanych wiadomości

Zasady ogólne

Metody zdalne zwracają status http **mniejszy niż 300** w przypadku, gdy operacja się powiedzie. Zwykle będą to statusy **200** lub **204**. W przypadku, gdy operacja się nie powiedzie, usługa zwraca status błędu **większy bądź równy 400**. Dla uściślenia statusy od 400 włącznie do 500 wyłącznie oznaczają błąd w żądaniu (np. złe parametry - 400, błąd uwierzytelniania - 403, zły adres - 404). Statusy powyżej 500 włącznie oznaczają błąd po stronie serwera. Możliwe, że to samo żądanie po usunięciu usterki może zostać obsłużone poprawnie. Oznacza to, że dla statusów błędów poniżej 500 nie ma sensu ponawiać żądania, natomiast dla statusów od 500 włącznie, żądanie można ponawiać.

Cele mechanizmu bezpieczeństwa:

1. Uwierzytelnienie klienta i serwera
2. Weryfikacja poprawności i prawdziwości żądania i odpowiedzi
3. Zabezpieczenie przed "cofaniem zegarka" i wykonywaniem wielu takich samych żądań z rzędu z tym samym timestampem.

Serwer dla każdego klienta przygotowuje parę **KeyID** i **SecretKey**, gdzie **KeyID** jest identyfikatorem (*jawnym, login podany w procesie rejestracji*) a **SecretKey** jest kluczem ukrytym (*hasło, 10 znaków*). Aktualny tajny klucz można odczytać lub wygenerować po zalogowaniu w zakładce **DANE UŻYTKOWNIKA**. Obie te wartości są znane dla klienta i serwera.

Uwierzytelnianie opiera się o dwa nagłówki HTTP dodawane do żądania i odpowiedzi. Dopuszczalne jest również stosowanie nagłówka **X-BP**

1. **X-GT-Timestamp** - timestamp operacji (unix timestamp)
2. **X-GT-Auth** - dodatkowy nagłówek do uwierzytelniania.

Przykład: X-GT-Auth: [KeyID]:[Hash] gdzie Hash to przygotowany skrót wiadomości opisany poniżej.

X-GT-Auth: 1234567:098f6bcd4621d373cade4e832627b4f6.

Dla żądania

Hash = MD5(HTTP_method + URI_Path_Query + MD5_Body + Accept + X-BP-Timestamp + SecretKey)

Gdzie MD5 to funkcja haszująca a '+' oznacza konkatenację łańcuchów znaków oraz:

1. **HTTP_method** to GET, POST, PUT itp.
2. **URI_Path_Query** to ścieżka i query string czyli część URL'a od pierwszego znaku '/' np: /sms_xmlapi.php
3. **MD5_Body** - to MD5 z treści żądania o ile jest
4. **Accept** - nagłówek określający format odpowiedzi (zgodny ze standardem), przyjmuje wartości np. application/xml
5. **X-GT-Timestamp** - timestamp wykonania operacji (taki jak w nagłówku)
6. **SecretKey** - klucz klienta

Przykład:

MD5(POST/sms_xmlapi.phpZXQW345YULAccept:application/xml12844675753LQBwxTiDnv)

Strona odbierająca żądanie ma obowiązek sprawdzenia, czy **Hash** jest poprawny. Jeśli nie jest to powinien zostać zwrócony status 403.

Dla odpowiedzi

Hash = MD5(MD5_Body + Content-Type + X-GT-Timestamp + SecretKey)

gdzie **SecretKey** to klucz klienta, od którego otrzymaliśmy wiadomość

Przykład

MD5(9843f33LQd8xTiDnvGAKoc8n7pQ5qi3CW9SG584ContentType:application/xml12844675753LQBwxTiDnv)

Klient otrzymujący odpowiedź od serwera ma obowiązek sprawdzenia, czy odpowiedź jest podpisana poprawnie o ile żądanie zostało poprawnie obsłużone. Odpowiedzi o statusach ≥ 400 nie muszą być podpisane.

W przypadku błędu komunikaty błędów mogą być zamieszczone w odpowiedzi HTTP jako tekst.

Definicja usług zdalnych

Do transmisji danych wykorzystywany będzie głównie format XML. Obie strony mają obowiązek zapewnić zgodność przesyłanych dokumentów ze standardem XML. Dotyczy to w szczególności zasad używania znaków specjalnych w wiadomościach. Dokument XML przesyłany jest za pomocą parametru **body** metodą *POST*. Zawartość tego parametru przed wysłaniem należy zakodować metodą kodowania URLi, zamieniając niealfanumeryczne znaki na dwie cyfry szesnastkowe poprzedzone znakiem procenta (%), a spacje kodowane jako znak (+). Zawartość musi zostać zakodowana dopiero po obliczeniu funkcji md5. Najczęściej występujące znaki powodujące błędy w transmisji: '+' => '%2B', '&' => '%26', '#' => '%23'.

Do pobrania statusów wiadomości wymagany jest dodatkowy nagłówek **X-GT-Action: Get Status**, do importu bazy numerów **X-GT-Action: Put HLR**.

Serwer zwraca dodatkowy nagłówek np. **X-GT-Status: 002** po poprawnym zaimportowaniu paczki z wiadomościami, lub w przypadku globalnych błędów, takich jak brak odpowiednich środków na koncie, błędy serwera, dostawców sieci internet, gsm, itp. [kody statusów](#).

Nagłówki Accept i Content-Type muszą być poprawnie ustawione, np.:

Content-Type: application/x-www-form-urlencoded

Accept: application/xml

Wysyłanie wiadomości do aplikacji Gatesms.eu

Adres usługi : http://www.gatesms.eu/sms_xmlapi.php

Metoda : **POST**

Nazwa zmiennej przesyłająca dane metodą POST: **body**

Content-Type: application/x-www-form-urlencoded

Accept: application/xml

Przykład żądania – parametr **body**:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
```

```
<package test="true">
```

```
<msg pl="true" phone="48518778404" localid="123-34-AA-33" isflash="true" from="Twoja Nazwa">Przykładowa tresc 1</msg>
```

```
<msg wap="" pl="true" phone="48518778404" localid="123-34-AA-34">Tresc z dziwnymi znakami [ i ] i &quot; oraz </msg>
```

```
<msg pl="true" phone="48518778404" localid="123-34-AA-35" from="Twoja Nazwa">Tresc 3</msg>
```

```
<msg phone="48518778404" localid="123-34-AA-36">Tresc pod zly numer</msg>
```

```
<msg pl="true" phone="48518778404" localid="123-34-AA-37">Wiadomość z polskimi znakami ąśęół</msg>
```

```
</package>
```

Element **package**:

- Atrybuty opcjonalne
 - *test* - *true* oznacza symulację wysyłki; domyślnie *false*
- Elementy obowiązkowe
 - *msg* - przynajmniej jedna wiadomość do wysłania Element **msg**: Zawiera treść wiadomości do wysłania
 - Atrybuty obowiązkowe
 - *phone* - 11-cyfrowy numer telefonu do wysyłki
 - *localid* - identyfikator wiadomości w systemie klienta
 - Atrybuty opcjonalne
 - *wap* - poprawny adres (<http://www.gatesms.eu>), umożliwia przesłanie aktywnego linku - wap push (wymagane jest podanie nadawcy)
 - *pl* - oznacza możliwość używania polskich znaków; domyślnie *false*
 - *isflash* - określa, czy wysłać wiadomości jako flash sms; domyślnie *false*
 - *from* - nazwa nadawcy do 11 znaków; jeśli jest podana taka nazwa, której klient nie ma prawa używać to powoduje to błąd wysyłki, nazwy można rejestrować pisząc na adres pomoc@gatesms.eu
 - *time* - timestamp określający datę i czas wysłania wiadomości w przyszłości; domyślnie wiadomość wysyłana jest natychmiast
 - *speed* – parametr powodujący ustawianie wiadomości na początku kolejki wysyłanych smsów. W przypadku połączenia z atrybutem *from* wiadomość zostanie wysłana jako SPEEDsms z czasem dotarcia poniżej 7 sekund.

Przykład odpowiedzi – dokument XML

Identyczną odpowiedź otrzyma klient pytający o zmiany statusów. [Patrz przesyłanie raportów doręczeń.](#)

Content-Type: application/xml; charset: utf-8

Accept: application/xml

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
```

```
<report>
<status timestamp="1295959673" localid="123-34-AA-33" ok="true" globalid="452342323">002</status>
<status timestamp="1295959673" localid="123-34-AA-34" ok="true" globalid="452342324">002</status>
<status timestamp="1295959673" localid="123-34-AA-35" ok="true" globalid="452342325">002</status>
<status timestamp="1295959673" localid="123-34-AA-36" ok="false" globalid="452342326">005</status>
<status timestamp="1295959673" localid="123-34-AA-37" ok="false" globalid="452342327">005</status>
</report>
```

Element **report** zawiera listę elementów **status**. Elementów jest dokładnie tyle ile przysłanych wiadomości.

Element **status**:

- Obowiązkowo zawiera [kod statusu](#) wiadomości.
- Atrybuty obowiązkowe
 - *timestamp* - unix timestamp; czas wygenerowania statusu wiadomości
 - *localid* - identyfikator wiadomości w systemie klienta taki, jak został przysłany
 - *globalid* - identyfikator wiadomości w systemie gatesms.eu
 - *ok* - *true* jednoznacznie określa czy wiadomość jest poprawna czy błędna (*false*)
- Atrybuty dodatkowe
 - *net* - określa sieć do jakiej została wysłana wiadomość
 - *sender* – numer z jakiego została wysłana wiadomość sms

Uwagi do działania usługi

Wiadomości identyfikowane są za pomocą *localid*. Dzięki temu w jednej paczce wiadomości można wysłać wiele wiadomości pod jeden numer telefonu. Jeśli z jakiegoś powodu (np. problemy z siecią) zostanie ponownie przetransmitowana wiadomość o tym samym *localid*, to usługa ma obowiązek zwrócić prawdziwy status tej wiadomości, ale nie wysłać jej ponownie.

Jeśli usługa nie wykonała się poprawnie z powodu błędnego żądania (został zwrócony status pomiędzy 400 a 500) to przyjmujemy, że na pewno żadna wiadomość nie została wysłana. Jeśli usługa zwróciła status większy niż 500 to nie ma pewności co do losów wiadomości. Będą one transmitowane ponownie z uwzględnieniem sprawdzania *localid*.

Maksymalny rozmiar jednej paczki wiadomości wynosi 300 sztuk.

Przesyłanie raportów doręczenia

Jeżeli w zakładce *KOLEJKA/HISTORIA* dostępnej po zalogowaniu do systemu zostanie zdefiniowany adres *raportu* system będzie przysyłał wiadomości wg załączonego schematu, za każdym razem kiedy zmieni się kod statusu wiadomości.

Adres usługi : XML: *adres usługi klienta*

Przesyłana jest lista bieżących zmian statusów wiadomości. Format taki sam jak dla odpowiedzi na wysłanie wiadomości, przy czym atrybut *localid* nie jest tu wymagany. [Kody statusów w załączniku](#). Dokument XML przesyłany jest za pomocą parametru *body* metodą *POST*. Zdalna usługa odpowiada statusem HTTP 204 jeśli raport został poprawnie przyjęty. Serwer będzie ponawiał wysłanie statusów przez kolejne 60 minut w odstępach co 5 min, chyba że uzyska status http 200, 202, 204, lub napis OK. Usługa zdalna odpowie serwerowi statusem 202 wtedy gdy chce nadal oczekiwać na kod potwierdzenia 004. W przypadku otrzymania statusu większego niż 004 usługa powinna odpowiedzieć nagłówkiem 204.

Adres usługi : http://www.gatesms.eu/sms_xmlapi.php

Metoda : **POST**

Nazwa zmiennej przesyłająca dane metodą POST: **body**

Content-Type: application/x-www-form-urlencoded

Accept: application/xml

X-GT-Action: Get Status

Przykład żądania – parametr body

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<report>
  <status timestamp="1295959673" ok="true" globalid="452342323"/>
  <status timestamp="1295959673" ok="true" globalid="452342324"/>
  <status timestamp="1295959673" ok="true" globalid="452342325"/>
</report>
```

[Przykład odpowiedzi – dokument XML](#)

Przesyłanie wiadomości przychodzących

Jeżeli w zakładce *KOLEJKA/HISTORIA* dostępnej po zalogowaniu do systemu zostanie zdefiniowany adres *odpowiedzi sms* system będzie przysyłał wiadomości wg. załączonego schematu za każdym razem kiedy odbierze wiadomość sms.

Adres usługi : XML: *adres klienta*

Metoda : **POST**

Nazwa zmiennej przysyłająca dane metodą POST: **body**

Content-Type: application/x-www-form-urlencoded

Accept: application/xml

Przykład żądania

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<inpackage>
  <msg to="48500100200" time="1297413504" phone="48500123456" >jakas wiadomosc</msg>
  <msg to="48500100200" time="1297413504" phone="48500123888" >jakas wiadomosc2</msg>
</inpackage>
```

Element **inpackage** zawiera listę **msg**. Element **msg** zawiera treść wiadomości.

- Atrybuty obowiązkowe
 - *to* - numer dostępowy Gatesms, na jaki przyszła wiadomość
 - *time* - czas dostarczenia wiadomości (unix timestamp)
 - *phone* - numer telefonu klienta

Opcja niedostępna czasowo!

Przykład odpowiedzi

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<report>
  <status ok="true" localid="123-221-3-0" globalid="1234567"/>
  <status ok="true" localid="123-221-3-1" globalid="1234568"/>
</report>
```

Element **report** zawiera listę elementów **status** Element **status**

- Atrybuty obowiązkowe
 - *ok* - określa, czy wiadomość została przyjęta poprawnie
 - *localid* - identyfikator wiadomości w systemie klienta
 - *globalid* - identyfikator wiadomości w systemie GateSMS

Jeśli usługa otrzyma duplikat wiadomości odpowie tak, jakby otrzymała poprawną wiadomość

Importowanie kontaktów do bazy GateSMS.eu

Adres usługi: do http://www.gatesms.eu/sms_xmlapi.php

Metoda: POST

Content-Type: application/x-www-form-urlencoded

Accept: text/plain

X-GT-Action: Put HLR

Metoda przesyła listę numerów do zaimportowania do bazy danych w celu sprawdzenia sieci klienta przed wysyłką. Ta metoda wyjątkowo, dla oszczędności ruchu, wykorzystuje zwykły tekst do transmisji.

Przykład żądania

48602300000
48602300001
48602300002
48602300003
48602300004

Żądanie zawiera listę numerów rozdzielonych znakiem nowej linii. Numery przesyłane są za pomocą parametru **body** metodą *POST*. Metoda zwraca jako tekst liczbę przekazanych numerów. Usługa dodaje tylko te numery których nie ma w repozytorium. Duplikaty są pomijane.

Przykład odpowiedzi

5

Kody statusów wiadomości

- 000 - Brak połączenia z smsc.
- 001 - Brak autoryzacji, błędny login lub hasło.
- 002 - Wiadomość została zakolejkowana do wysłania (oczekiwanie na potwierdzenie).
- 003 - Wiadomość została wysłana do odbiorcy.
- 004 - Wiadomość odebrana przez odbiorcę (potwierdzenie odbioru).
- 005 - Błąd wiadomości.
- 006 - Numer nieaktywny.
- 007 - Błąd w dostarczeniu wiadomości.
- 008 - Wiadomość odebrana przez sms center.
- 009 - Błąd sieci GSM.
- 010 - Wiadomość wygasła z powodu niemożliwości jej dostarczenia do odbiorcy.
- 011 - Wiadomość została zakolejkowana do późniejszego wysłania.
- 012 - Błąd providera, natychmiastowy kontakt z administratorem systemu.
- 103 - Brak pola text w wiadomości lub pole text niepełne.
- 104 - Błędnie wypełnione lub brak pola nadawcy.
- 105 - Pole text jest za długie.
- 106 - Błędny lub brak pola numer.
- 107 - Błędny parametr type.
- 110 - SMSC nie obsługuje danego typu wiadomości.
- 113 - Pole text jest za długie.

- 201 - Błąd systemu, natychmiastowy kontakt z administratorem systemu.
- 202 - Niewystarczająca ilość kredytów na koncie.
- 203 - Działanie niedozwolone, natychmiastowy kontakt z administratorem systemu.
- 204 - Konto nieaktywne. Najprawdopodobniej zablokowane.
- 205 - Sieć docelowa zablokowana.
- 301 - Brak lub błędny identyfikator wiadomości.
- 500 - Błędnie wypełnione pole nadawcy lub pole text za długie.
- 600 - Brak kredytów na koncie premium dla podanego odbiorcy.
- 700 – Brak potwierdzenia zapisu rekordu.
- 800 - Deduplikator: Brak Zapisu Rekordu! Wiadomość powtórzona.
- 888 – Restart zablokowanych smsów.
- 999 – Zewnętrzna infrastruktura – status przejściowy.

Długości wiadomości sms

Tabela podziału SMS bez polskich znaków:

Ilość znaków	Długość poszczególnych części SMS	Ilość SMSów które otrzyma odbiorca	Ilość SMSów użytych do billingu klienta
1-160	1 x 160 znaków	1	1
161-304	2 x 152 znaków	1	2
305-456	3 x 152 znaków	1	3
457-608	4 x 152 znaków	1	4

* Przy wiadomościach mających więcej niż 160 znaków, treść wiadomości dzielona jest na pojedyncze SMSy gdzie każdy z nich ma 153 znaki. Pozostałe 7 znaków rezerwowane jest na informacje pozwalające scalić wiadomości w jednego długiego SMSa.

Ilość znaków	Długość poszczególnych części SMS	Ilość SMSów które otrzyma odbiorca	Ilość SMSów użytych do billingu klienta
1-70	1 x 70 znaków	1	1
71-134	2 x 67 znaków	1	2
135-201	3 x 67 znaków	1	3
202-268	4 x 67 znaków	1	4

* Przy wiadomościach mających więcej niż 70 znaków, treść wiadomości dzielona jest na pojedyncze SMSy gdzie każdy z nich ma 67 znaków. Pozostałe 3 znaków rezerwowane jest na informacje pozwalające scalić wiadomości w jednego długiego SMSa.

Dodatek A

Klasa obsługująca Web XML Api

Przykład obsługi klasy SMS_SENDER z pliku sms_xmlsender.php.

```
<?
include_once("example/sms_xmlsender.php");

//Konstruktor klasy, oraz parametry

$sms = new SMS_SENDER;
$sms->login='mojlogin@gatesms.eu';
$sms->pass='0a3e8dew45';
$sms->test='false';

//Wysyłanie paczek z wiadomościami z dynamicznym identyfikatorem localId:

$sms->localIdRandom=1;
for($z=0;$z<12;$z++) {
    $sms->to=48501123456;
    $err = $sms->AddMsg("TEST A" . $z);
    $LocalmsgId = $sms->localId;
};
$report = $sms->sendsms();
exit;

//Wysyłanie paczek z wiadomościami z własnym identyfikatorem localId:

$sms->localIdRandom=0;
for($z=0;$z<12;$z++) {
    $sms->localId='540';
    $sms->to=48601098765;
    $err = $sms->AddMsg("TEST A" . $z);
};
$report = $sms->sendsms();
exit;

//Zapytanie o statusy

for($z=100000;$z<102225;$z++) {
    $sms->globalId=$z;
    $sms->AddStat();
};
$report = $sms->GetStat();
exit;

//Dodawanie numerów do weryfikacji

$numery=array(500600300,602301605,723898549);
echo $sms->AddHlr($numery);
exit;
?>
```