REST APIs

1. PrivateServer REST APIs

PrivateServer provides some web services in a RESTful way.

1.1. **REST Request specification**

Base request url: https://server.com/rest/{api_version}/

Only HTTPS protocol is allowed.

The “api_version” is a number (integer) that specify the API version used by the client. This document is about Api version 1.

1.2. **REST Request authentication**

REST APIs are available under strict access contro. Users of REST APIs should have REST role (ADMIN role is not required and discouraged).

Administrator should create a specific user/password: this credential are required to authenticate REST requests. PrivateServer use a simplified version of Amazon S3 REST authentication protocol (http://docs.amazonwebservices.com/AmazonS3/latest/dev/RESTAuthentication.html).

For every request the REST client must calculate an HMAC-SHA1 of the request body, using the password provided. For example, using a pseudo-code

```
Signature = Base64( HMAC-SHA1(UTF-8-Encoding-Of(PrivateServer Rest Password, Request body)) );
```

PrivateServer Rest password must be sent with SHA-1 hash, in HEX (lowercase) format, for example “a94a8fe5ccb19ba61c4c0873d391e987982fbbd3” for password “test”

This signature must be placed in an HTTP header named “x-privateserver-auth” with this format:

```
x-privateserver-auth = PrivateServer Rest username : Signature
```

Every request body to sign must include the Date HTTP header, as the first body line.

Request	Request body to sign	HTTP request example	Request body to sign example

HTTP POST	Date HTTP header +
	
	HTTP post request
		`POST /rest/1/account/ HTTP/1.1`
		
		`Host: server.com`
		
		`Date: Tue, 27 Mar 2007 19:42:41 +0000`
		
		`x-privateserver-auth: restUser: hcicpDDvL9SsO6AkvxqmIWkmOuQ=`
		
		`owner=Mario Rossi`
		
		`description=Mario Rossi personal account`
		
		`phone_number=%2B393334455678`
		
		`email=mario.rossi@privatewave.com`
		
		`security_model=s`
			`Tue, 27 Mar 2007 19:42:41 +0000\n`
			
			`owner=Mario Rossi\n`
			
			`description=Mario Rossi personal account\n`
			
			`phone_number=+393334455678\n`
			
			`email=mario.rossi@privatewave.com\n`
			
			`security_model=s`

HTTP GET with parameters	Date HTTP header +
	
	HTTP get parameters, printed out as a list
		`GET /rest/1/account/?params=1&foo=3 HTTP/1.1`
		
		`Host: server.com`
		
		`Date: Tue, 27 Mar 2007 19:42:41 +0000`
		
		`x-privateserver-auth: restUser: hcicpDDvL9SsO6AkvxqmIWkmOuQ=`
			`Tue, 27 Mar 2007 19:42:41 +0000\n`
			
			`params=1\n`
			
			`foo=3`

HTTP GET without parameters	Empty string	`GET /rest/1/account/ HTTP/1.1`
		`Host: server.com`
		`Date: Tue, 27 Mar 2007 19:42:41 +0000`
		`x-privateserver-auth: restUser: hcicpDDvL9SsO6AkvxqmIWkmOuQ=`
			`Tue, 27 Mar 2007 19:42:41 +0000`

1.3. **Error response format**

JSON format of error response

```
{
   "rest_errors":
   [
     {
        "error_code"    : 1,
        "error_message" : "Unable to find user"
     }
     {
        "error_code"    : 5,
        "error_message" : "Invalid email"
     }
   ]
}
```

Error_code: a numeric error code identifier - for future use

Error_message: a human-readable error description

1.4. **Date-Time format**

The timestamp that PrivateServer sent to the REST client must have the following format:

“YYYY-MM-DD HH:MM:SS” UTC time

2. **Web Services**

2.1. **Account creation**

HTTP POST {base request}account/create

2.1.1. HTTP POST PARAMETERS

Parameter name	Validation	Min length	Max lenght	Example
owner	Alphanumerical string with spaces and underscore - hyphen Regexp : [a-z0-9_-]*{4,50}	4	50	owner=Mario Rossi
description	Free text	10	100	description=Mario Rossi personal account
phone_number	Phone number in international format, without spaces and brackets. The phone number must start with a “+” followed by international code and by the number. Regexp : +[0-9]*{8,20}	8	20	phone_number=+393334455678
email	Only valid email (see RFC 2822)	(see RFC 2822)	(see RFC 2822)	email=mario.rossi@privatewave.com

security_model=s	“security_model” acceptable value: • “s” - SDES protocol • “z” - ZRTP protocol • “sz” - Double protocol, SDES have higher priority • “zs” - Double protocol, ZRTP have higher priority	1	2	security_model=s

2.1.2. HTTP RESPONSE

- JSON representation of user data
- [MANDATORY] “provisioning_data” contain the provisioning link token and validity.

HTTP CODE		BODY

201
 	
CREATED
	All ok	<pre>{
		  "owner" : "Mario Rossi",
		  "description" : "Mario Rossi personal Account",
		  "phone_number" : "+393334455678",
		  "email" : "mario.rossi@privatewave.com",
		  "status" : "enabled",
		  "security_model" : "s",
		  "provisioning_data" :
		  {
		    "token" : "3e29ca992780c",
		    "creation_time" : "2012-01-26 18:17:48"
		    "expiry_time" : "2012-01-27 18:17:48"
		  }
		}</pre>

400
 	
BAD REQUEST
	Missing parameter or validation error or account already exists	<pre>{
		  "rest_errors":
		  [
		    {
		      "error_code" : 5,
		      "error_message" : "Invalid email"
		    }
		  ]
		}</pre>

401 UNAUTHORIZED	Bad credential	Empty body
405 METHOD NOT ALLOWED	Not an HTTP POST	Empty body

500
 	
INTERNAL ERROR
	Server error	<pre>{
		  "rest_errors":
		  [
		    {
		      "error_code" : 5,
		      “error_message" : "Exception..."
		    }
		  ]
		}</pre>

2.2. **Disable - enable account**

HTTP POST {base request}account/status/

2.2.1. HTTP POST PARAMETERS

Parameter name	Validation	Min length	Max lenght	Example
phone_number	Phone number in international format, without spaces and brackets. The phone number must start with a “+” followed by international code and by the number. Regexp : +[0-9]*{8,20}	8	20	phone_number=+393334455678
status	Only “disabled” or “enabled”	N.A.	N.A.	status=disabled
description	Free text	10	100	description=User doesn’t work for PrivateWave

2.2.2. HTTP RESPONSE

- JSON representation of user data
- [OPTIONAL] “provisioning_data” contain the provisioning link token and validity, if provisioning link is yet available

HTTP CODE		BODY

200
 	
OK
	All ok	<pre>{
		  "owner" : "Mario Rossi",
		  "description" : "Mario Rossi personal Account",
		  "phone_number" : "+393334455678",
		  "email" : "mario.rossi@privatewave.com",
		  "status" : "disabled",
		  "security_model" : "s",
		}</pre>

400
 	
BAD REQUEST
	Missing paremeter or validation error	<pre>{
		  "rest_errors":
		  [
		    {
		      "error_code" : 1,
		      "error_message" : "no description data"
		    }
		  ]
		}</pre>

401 UNAUTHORIZED	Bad credential	Empty body
404 NOT FOUND	User not found	Empty body
405 METHOD NOT ALLOWED	Not an HTTP POST	Empty body

500
 	
INTERNAL ERROR
	Server error	<pre>{
		  "rest_errors":
		  [
		    {
		      "error_code" : 5,
		      “error_message" : "Exception..."
		    }
		  ]
		}</pre>

2.3. **Request activation link re-issue**

HTTP POST {base request}account/send_configuration_link

2.3.1. HTTP POST PARAMETERS

Parameter name	Validation	Min length	Max length	Example
phone_number	Phone number in international format, without spaces and brackets. The phone number must start with a “+” followed by international code and by the number. Regexp : +[0-9]*{8,20}	8	20	phone_number=+393334455678
description	Free text	10	100	description=User doesn't receive activation link

2.3.2. HTTP RESPONSE

- JSON representation of user data
- [MANDATORY] “provisioning_data” contain the new provisioning link token and validity

HTTP CODE		BODY

200
 	
OK
	All ok	<pre>{
		  "owner" : "Mario Rossi",
		  "description" : "Mario Rossi personal Account",
		  "phone_number" : "+393334455678",
		  "email" : "mario.rossi@privatewave.com",
		  "status" : "enabled",
		  "security_model" : "s",
		  "provisioning_data" :
		  {
		    "token" : "3e29ca992780c",
		    "creation_time" : "2012-01-26 18:17:48"
		    "expiry_time" : "2012-01-27 18:17:48"
		   }
		}</pre>

400
 	
BAD REQUEST
	Missing paremeter or validation error	<pre>{
		  "rest_errors":
		  [
		    {
		      "error_code" : 1,
		      "error_message" : "Invalid phone number"
		    }
		  ]
		}</pre>

401 UNAUTHORIZED	Bad credential	Empty body
404 NOT FOUND	User not found	Empty body
405 METHOD NOT ALLOWED	Not an HTTP POST	Empty body

500
 	
INTERNAL ERROR
	Server error	{
		"rest_errors":
		[
		{
		"error_code" : 5,
		“error_message" : "Exception..."
		}
		]
		}

2.4. **List users**

HTTP GET {base request}account/list

2.4.1. HTTP RESPONSE

- JSON representation of user list
- [OPTIONAL] “provisioning_data” contain the new provisioning link token and validity

HTTP CODE		**BODY**

200
 	
OK
	All ok	{
		“user_list” :
		[
		{
		"owner" : "Mario Rossi",
		"description" : "Mario Rossi personal Account",
		"phone_number" : "+393334455678",
		"email" : "mario.rossi@privatewave.com",
		"status" : "enabled",
		"security_model" : "s",
		"provisioning_data" :
		{
		"token" : "3e29ca992780c",
		"creation_time" : "2012-01-26 18:17:48"
		"expiry_time" : "2012-01-27 18:17:48"
		}
		}
		...
		]
		}

401 UNAUTHORIZED	Bad credential	Empty body
405 METHOD NOT ALLOWED	Not an HTTP GET	Empty body

500
 	
INTERNAL ERROR
	Server error	{
		"rest_errors":
		[
		{
		"error_code" : 5,
		“error_message" : "Exception..."
		}
		]
		}

Events data SQL views