



WebSitePulse API Documentation

Rev. 2.5

Table of Contents

1. Overview	3
2. Getting Started	3
3. HTTP API	4
3.1 HTTP API Overview	4
3.2 HTTP API methods	5
3.2.1 GetStatus method	5
3.2.2 GetUptime method	6
3.2.3 GetHistory method	7
3.2.4 UpdateTargetStatus method	8
3.2.5 GetDailyLog method	9
3.2.6 AccountStatus method	11
3.2.7 AccountSummary method	12
3.3 HTTP Image API	14
4. Web Service – WSDL API	15
4.1 WSDL API Overview	15
4.2 WSDL API methods	16
4.2.1 doGetStatus method	16
4.2.2 doGetUptime method	17
4.2.3 doInstantTest method	18
4.2.4 doInstantFullpageTest method	19
4.2.5 doInstantTransactionTest method	19
4.2.6 doInstantFullpageTransactionTest method	20
4.2.7 UpdateTargetStatus method	21
4.2.8 doGetSettings method	21
5. Errors	23
5.1 HTTP API Errors	27
5.2 WSDL API Errors	27
6. Contact information	29
Appendix A: Examples	30
A.1 HTTP API Examples	30
A.2 WSDL API Examples	33

1 Overview

The purpose of this document is to provide application developers with a comprehensive piece of information on how to integrate the WebSitePulse monitoring information into their applications. WebSitePulse currently provides two ways to access the monitoring data:

- HTTP – retrieve data using GET or POST HTTP requests
- Web Service (WSDL) – call the API methods using SOAP to access the monitoring data or trigger an instant test.

This document covers both the HTTP and WSDL API methods and description.

2 Getting Started

If you are an application developer or business owner and wish to use WebSitePulse monitoring data in your applications you will need an active WebSitePulse account monitoring at least one target. To protect your account the API uses a special key which in addition to your WebSitePulse user name will be used to verify the authenticity of the request. To retrieve your API key please go to

<http://www.websitepulse.com/reports/wsdllkey.php>

At this URL you will be able to view a list with the target IDs which are needed for some of the API methods. Furthermore you can configure some of the API settings like limiting the Ips that will be allowed to access the API with your username, etc. This page requires logging in with your WebSitePulse user name and password.

3 HTTP API

3.1 HTTP API overview

The HTTP API can be used to send a GET/POST HTTP request with a set of parameters and receive a set of formatted data from the WebSitePulse.com monitoring database.

The address of the HTTP API is

<http://api.websitepulse.com/textserver.php>

There are three parameters which must be sent with every request - username, key and method. You can also choose whether to receive the API response in one of the predefined formats like CSV or XML or to define a custom format using the format parameters. Here is the list of the global variables that can be used with every method:

Name	Description
username	Your WebSitePulse.com user name. This parameter is required.
key	The API access key. This parameter is required.
method	<p>The name of the method you want to call. Currently supported are:</p> <ul style="list-style-type: none">• GetStatus – returns the most recent status of the selected target(s).• GetUptime – returns a monthly uptime and response time summary• GetHistory – returns a log of all status changes during a selected period• UpdateTargetStatus – suspend or resume the monitoring of the selected target(s) <p>For more information see section 3.2</p>
format	<p>Select a predefined format for the requested data. You can choose any of the following values for this parameter:</p> <ul style="list-style-type: none">• xml – XML formatted data• csv1 – Semi-colon separated values• csv2 – Comma separated values• txt – Tab delimited data <p>You can define a custom format using the delimiter and enclose parameters listed below.</p>
delimiter	This value will be used to separate the fields in the response.

enclose	<p>Controls the quoting of the text fields. Use this parameter to avoid problems with fields containing the delimiter character. If the enclose character is found within a field content it will be duplicated.</p> <p>Here are some examples of using different values for this parameter to format the following text:</p> <p>Text containing ' (single quotes) and " (double quotes)</p> <p>enclose=" => "Text containing ' (single quotes) and "" (double quotes)"</p> <p>enclose=' => 'Text containing " (single quotes) and " (double quotes)'</p> <p>enclose= => Text containing ' (single quotes) and " (double quotes) </p>
---------	--

3.2 Methods

3.2.1 **GetStatus** – returns information from the most recent test(s) of the selected target(s).

Request parameters:

Name	Description
target	A comma separated list of target IDs. To request information for all targets use target=all
location	A comma separated list of WebSitePulse monitoring location indexes. To request information from all locations monitoring the selected target(s) use location=all

Return values:

Name	Description
label	The Target Label you specified for this target in your WebSitePulse.com configuration
location	Monitoring location name
targetType	A letter indicating the target type.
frequency	Monitoring frequency in minutes
statusCode	Status Code - one of UP, Down, Test pending.
statusMessage	The status message from the last effective check
fails	Number of failures since the target was last detected to be working properly
responseTime	The total response time in seconds calculated during the last check
lastCheck	Last check date and time in the format (HH:ii mm/dd/yy)
lastDown	Date and time of the last problem detected in the format (HH:ii mm/dd/yy)

PING statistics (optional)	
pingLoss	Packet loss percentage
pingMin	Minimum packet round-trip time
pingAvg	Average packet round-trip time
pingMax	Maximum packet round-trip time

Sample request:

<http://api.websitepulse.com/textserver.php?username=demo&key=35e6340814655582a79039dbc1cb0b65&target=7182&location=1&method=GetStatus&format=xml>

The result of this request will look like this:

```
<Result>
<item>
  <label>Model Ink Jet</label>
  <location>Houston, TX </location>
  <targetType>HTTP advanced</targetType>
  <frequency>20</frequency>
  <statusCode>UP</statusCode>
  <statusMessage>OK</statusMessage>
  <fails>0</fails>
  <responseTime>1.329</responseTime>
  <lastCheck>23:48 06/20/2006</lastCheck>
  <lastDown>07:57 06/21/2006</lastDown>
  <pingLoss>0.00</pingLoss>
  <pingMin>0.000</pingMin>
  <pingAvg>0.000</pingAvg>
  <pingMax>0.000</pingMax>
</item>
</Result>
```

3.2.2 **GetUptime** – returns a monthly uptime and response time summary

Request parameters:

Name	Description
target	A comma separated list of target IDs. To request information for all targets use target=all
location	A comma separated list of WebSitePulse monitoring location indexes. To request information from all locations monitoring the selected target(s) use location=all
startdate	The beginning of the time period for which you want to receive information in YYYYMM format (“200607” is July 2006).
enddate	The end of the time period for which you want to receive information in YYYYMM format (“200607” is July 2006).

Return values:

Name	Description
label	The Target Label you specified for this target in your WebSitePulse.com control panel
location	Monitoring location name
checks	Total checks made during the selected period.
fails	Total number of fails detected during the selected period.
resptime	Average response time for all checks for the selected period in seconds.
uptimepercent	Uptime % for the selected period.
downtime	Estimated downtime for the selected period in seconds.

Sample request:

<http://api.websitepulse.com/textserver.php?username=demo&key=35e6340814655582a79039dbc1cb0b65&target=7182&location=1&startdate=200606&enddate=200606&method=GetUptime&format=xml>

The result of this request will look like this:

```
<Result>
<item>
  <label>Model Ink Jet</label>
  <location>Houston, TX </location>
  <checks>570</checks>
  <fails>2</fails>
  <resptime>2.1878</resptime>
  <uptimepercent>99.6491</uptimepercent>
  <downtime>2975</downtime>
</item>
</Result>
```

3.2.3 *GetHistory* – returns a log of all status changes during a selected period. This method can be used to check whether an error has occurred during a selected period, the type of problem detected and the time it took for the target to recover.

Request parameters:

Name	Description
target	A comma separated list of target IDs. To request information for all targets use target=all
location	A comma separated list of WebSitePulse monitoring location indexes. To request information from all locations monitoring the selected target(s)

	use location=all
startdate	The beginning of the time period for which you want to receive information in YYYYMMDDhhmmss format ("20060701000000" is 1 st of July 2006, 12h a.m.).
enddate	The end of the time period for which you want to receive information in YYYYMMDDhhmmss format ("20060701000000" is 1 st of July 2006, 12h a.m.).

Return values:

Name	Description
label	The Target Label you specified for this target in your WebSitePulse.com configuration
location	Monitoring location name
status	Status of the target - one of UP, Down, N/A (target was suspended).
statustype	A more detailed status message describing the problem detected
starttime	The start time of the status period (MM/DD/YYYY hh:mm)
endtime	The end time of the status period (MM/DD/YYYY hh:mm)
duration	Estimated duration

Sample request:

<http://api.websitepulse.com/textserver.php?username=demo&key=35e6340814655582a79039dbc1cb0b65&target=7182&location=1&startdate=20060601000000&enddate=20060620035959&method=GetHistory&format=xml>

The result of this request will look like this:

```
<Result>
<item>
  <label>Model Ink Jet</label>
  <location>Houston, TX </location>
  <startdate>June 1, 2006 00:00:00</startdate>
  <enddate>June 20, 2006 03:59:59</enddate>
  <statushistory>
    <row>
      <status>UP</status>
      <statustype>OK</statustype>
      <starttime>06/12/2006 23:20</starttime>
      <endtime>06/20/2006 08:57</endtime>
      <duration>7d 9h 37m</duration>
    </row>
  </statushistory>
</item>
</Result>
```


3.2.4 UpdateTargetStatus – suspend or resume the monitoring of the selected target(s). When a target is suspended your account value will be updated accordingly and the time this target was not monitored will be deducted from your monthly invoice. If you have activated the respective notifications in your account settings you will be notified by email for every change in the status of the targets in your account or the current account value. Since this method may affect the overall monitoring of your targets we strongly recommend you to limit the API usage to certain IP addresses and secure your username and access key to avoid any service interruptions.

Request parameters:

Name	Description
target	A comma separated list of target IDs. To request information for all targets use target=all
action	Should be one of : activate – resumes the monitoring of the selected targets suspend – suspend the selected targets.

Return values:

Name	Description
Status	A message describing the outcome of the executed API request

Sample request:

<http://api.websitepulse.com/textserver.php?username=demo&key=35e6340814655582a79039dbc1cb0b65&target=7182&action=suspend&method=UpdateTargetStatus&format=xml>

The result of this request will look like this:

```
<Status>
The monitoring of all selected targets was suspended successfully.
</Status>
```

3.2.5 GetDailyLog – retrieves the monitoring data for a 24 hours period for all selected targets and monitoring locations. Includes information of all tests performed during the selected period like response times, PING information and target status.

Request parameters:

Name	Description
target	A comma separated list of target IDs. To request information for all targets use target=all

location	A comma separated list of WebSitePulse monitoring location indexes. To request information from all locations monitoring the selected target(s) use location=all
startdate	The beginning of the time period for which you want to receive information in YYYYMMDDhhmmss format ("20060701000000" is 1 st of July 2006, 12h a.m.). If omitted the API will return all monitoring data starting from 12 am of the current day. Please note that this date will be used according to your targets timezone.

Return values:

Name	Description
label	The Target Label you specified for this target in your WebSitePulse.com configuration
location	Monitoring location name
targetType	A letter indicating the target type.
frequency	Monitoring frequency in minutes
fails	Number of failures since the target was last detected to be working properly
totalUptime	The total uptime percentage since the target was added for monitoring
Test results – for each test the following information will be available. Some of the fields are available for specific target types only	
date	Date and time when the test was performed
status	Detected target status
responseTime	Total response time
dnsTime	DNS lookup time
connectTime	Connection time
redirectTime	Redirects time – time needed to perform any HTTP/S redirects before reaching the final page. Currently available for HTTP/S targets only.
firstbyteTime	The time used by the monitored server to process the request and start serving the requested data
lastbyteTime	The time needed to download the page content
pingLoss	Packet loss percentage
pingMin	Minimum packet round-trip time
pingAvg	Average packet round-trip time

pingMax	Maximum packet round-trip time
size	Size in bytes of the downloaded page content. Currently available for HTTP/S performance targets only.
md5hash	MD5 hash of the downloaded page content. Currently available for HTTP/S performance targets only.

Sample request:

<http://api.websitespulse.com/textserver.php?username=demo&key=35e6340814655582a79039dbc1cb0b65&target=7182&location=1&method=GetDailyLog&format=xml>

The result of this request will look like this:

```
<Result>
  <item>
    <label>Model HTTP server</label>
    <location>Houston, TX </location>
    <targetType>HTTP advanced</targetType>
    <frequency>20</frequency>
    <fails>0</fails>
    <totalUptime>99.75</totalUptime>
    <tests>
      <test>
        <date>08/10/2007 00:02</date>
        <status>OK</status>
        <responseTime>0.923</responseTime>
        <dnsTime>0.053</dnsTime>
        <connectTime>0.050</connectTime>
        <redirectTime>0.000</redirectTime>
        <firstbyteTime>0.718</firstbyteTime>
        <lastbyteTime>0.103</lastbyteTime>
      </test>
      ...
    </tests>
  </item>
</Result>
```

3.2.6 AccountStatus – returns a summary of the status of all monitored targets in your account.

Request parameters:

Name	Description
target	Optional. A comma separated list of target IDs. To request information for all targets use target=all
location	Optional. A comma separated list of WebSitePulse monitoring location indexes. To request information from all locations monitoring the selected target(s) use location=all

Return values:

Name	Description
Total	Total number of targets
OK	Number of targets that are currently OK
Failed	Number of targets which report failure from all monitoring locations
Warnings	Number of targets which report a failure from some of the locations only
Suspend	Number of suspended targets

Sample request:

<http://api.websitepulse.com/textserver.php?username=demo&key=35e6340814655582a79039dbc1cb0b65&method=AccountStatus&format=xml>

The result of this request will look like this:

```
<Result>
  <total>8</total>
  <ok>7</ok>
  <failed>1</failed>
  <suspended>0</suspended>
  <warnings>0</warnings>
</Result>
```

3.2.7 AccountSummary – returns a list of all targets in your account including their respective settings.

Request parameters:

Name	Description
target	A comma separated list of target IDs. To request information for all targets use target=all

Return values:

Name	Description
label	The Target Label you specified for this target in your WebSitePulse.com configuration
type	Type of the monitored target – for example HTTP performance.
frequency	The interval in minutes at which tests are being performed
forced	Shows whether “Forced monitoring on error” is enabled and if yes, the forced monitoring interval in minutes
locations	The number of WebSitePulse locations selected to monitor this target
traceroute	Whether the “Traceroute on error” option is enabled or not
keeplogs	The time in months during which the detailed monitoring data will be available in your reports.

takeover	Whether the "Takeover" option is enabled or not
cache	The caching level for the specific target. Defines when the page content is cached on our servers and whether to generate screenshots once a day or one every error detected.
steps	The number of steps in a Web or Fullpage transaction. Will display n/a for all other target types
price	The current monitoring price for the selected target
status	Status of the target – will display Active if the target is currently being monitored or Suspended if not.

Sample request:

<http://api.websitepulse.com/textserver.php?username=demo&key=35e6340814655582a79039dbc1cb0b65&target=7182&method=AccountSummary&format=xml>

The result of this request will look like this:

```
<Result>
  <item>
    <label>Model HTTP server</label>
    <type>HTTP advanced</type>
    <frequency>20</frequency>
    <forced>n/a</forced>
    <locations>3</locations>
    <traceroute>Yes</traceroute>
    <keeplogs>1</keeplogs>
    <takeover>No</takeover>
    <cache>Level 1</cache>
    <steps>n/a</steps>
    <price>7.00</price>
    <status>Suspended</status>
  </item>
</Result>
```

3.3 HTTP API – Request Image

This is a special HTTP API which instead of the monitoring data and statistics will return a different image representing this data in graphical format. This API is intended to make easier the embedding of WebSitePulse.com monitoring data in web applications.






The address for requests to the HTTP Image API is

<http://api.websitepulse.com/img.php>

Request parameters:

Name	Description
username	Your WebSitePulse.com user name. This parameter is required.
key	The API access key. This parameter is required.
target	A WebsitePulse target ID. This parameter is required.
method	Reserved for future use. Currently the only available method is GetStatus .
location	A WebSitePulse monitoring location index. This parameter is required.
size	A keyword indicating the size of the image that should be returned. Use 16, "s" or "small" to request a image sized 16x16 pixels , or 32, "l", "large" to request a image with dimmensions 32x32 pixels.

Responses:

Image	Description
	The target is OK
	There was an error detected but it was not confirmed by a second monitoring location
	A failure has been found and confirmed
	Our monitoring agent have detected problems with this target for two or more consecutive tests.
	Cannot determine the target status due to one of the following reasons: <ul style="list-style-type: none"> the target is suspended or doesn't exist; the target is not monitored by the selected location; no tests were performed since the last target update the credentials provided were invalid

4 Web Service (WSDL)

4.1 WSDL API overview

Web Services Description Language provides a model and an XML format for describing Web services. WSDL enables separation of the description from the functionality of the WebSitePulse service. The description of the functionality that can be used with WebSitePulse's API is described in the WSDL file (<http://api.websitepulse.com/wsp.wsdl>).

Many programming languages now understand WSDL and can use this file to automatically invoke the Web Service. For example, the WSDL can be imported into .NET, converted into Java code using Apache Axis WSDL2Java, or used directly by Perl SOAP::Lite. Sample applications written in PERL and PHP (using the NuSOAP class) are available at the WebSitePulse WSDL pack (<http://www.websitepulse.com/wSDL/wSDL.zip>).

There are three parameters which must be sent with every request - username, key and location. Here is the list of the global variables that can be used with every method:

Name	Description
username	Your WebSitePulse.com user name. This parameter is required.
key	The API access key. This parameter is required.
location	The index of the WebSitePulse.com monitoring location you want to query. A full list of WebSitePulse location IDs can be found here: http://www.websitepulse.com/services/network.php – the number next to the location label is the Location ID.

4.2 Methods

4.2.1 doGetStatus – This method will return a set of monitored targets with their most recent status.

Request parameters:

Name	Description
targetType	A => All targets B => Basic type E => Advanced type F => Performance type I => Full-page type T => Performance Transactions M => Email Round-trip targets X => Fullpage Transactions
targetID	The ID or a list of IDs separated by comma for WebSitePulse targets.

	Set to 0 to show all targets from selected type.
--	--

Return values:

Name	Description
targetType	a single letter indicating the target type
label	the label you specified for this target in your WebSitePulse.com configuration
frequency	Monitoring interval in minutes.
fails	Number of failures since the last OK status.
StatusMessage	currently not available. Will show the status message from the last effective check
lastCheck	Last check date & time in the format (HH:ii mm/dd/yy)
lastDown	Date & time the target was last detected DOWN in the format (HH:ii mm/dd/yy)

4.2.2 doGetUptime – The method generates Uptime statistics for a selected period.

Request parameters:

Name	Description
targetType	A => All targets B => Basic type E => Advanced type F => Performance type I => Full-page type T => Performance Transactions M => Email Round-trip targets X => Fullpage Transactions
targetID	The ID or a list of IDs separated by comma for WebSitePulse targets. Set to 0 to show all targets of selected type.
startmonth	The index of the start month (1-January, 2 - February, ...) you want statistics for. Leave empty or set to 0 to get the information for the current month.
startyear	Leave empty or set to 0 to get the information for the current year. Enter a four-digit year otherwise.
endmonth	The index of the end month (1-January, 2 - February, ...) you want statistics for. Leave empty or set to 0 to get summarized uptime information from the starting period till present.
endyear	Leave empty or set to 0 to get the information till the current year. Enter a four-digit year otherwise.

Return values:

Name	Description
targetType	a letter indicating the target type
label	the label you specified for this target in your WebSitePulse.com configuration
checks	Total number of checks made during the selected period
fails	Total number of failures detected during the selected period
respTime	Average Response Time for the selected period.
uptimePercent	Uptime percent for the selected period.
downtime	Total downtime for selected period in seconds.

4.2.3 *dolInstantTest* – Tests a target from the specified location and returns the check results. The function serves Basic, Advanced and Performance target types.

Request parameters:

Name	Description
targetID	A target ID or a list of comma separated target IDs. Set to 0 to show all targets of selected type.

Return values:

Name	Description
targetType	A string indicating the target type
status	The status message returned by the monitoring agent
respTime	Response Time in seconds
dnsTime	DNS time in seconds
connectTime	Connection time in seconds
firstByte	First byte time
lastByte	Last byte time
size	Size of the downloaded page content. Available for HTTP/S performace targets only
md5hash	MD5 hash of the downloaded page content. Available for HTTP/S performace targets only
PING statistics (optional)	
pingLoss	Packet loss percentage

pingMin	Minimum packet round-trip time
pingAvg	Average packet round-trip time
pingMax	Maximum packet round-trip time

4.2.4 doInstantFullpageTest – Tests a Fullpage target from the specified location and returns the check results.

Request parameters:

Name	Description
targetID	A target ID or a list of comma separated target IDs. Set to 0 to show all targets of selected type.

Return values:

Name	Description
targetType	A letter indicating the target type
status	The status message returned by the monitoring agent
respTime	Response Time in seconds
dnsTime	DNS time in seconds
connectTime	Connection time in seconds
firstByte	Time in seconds before the first byte of the server response is received
lastByte	Time in seconds needed to complete the download
totalRefs	Total number of tested references
failedRefs	Failed references
imagesTested	Number of images tested
framesTested	Number of frames tested
otherInline	Number of other inline components tested. This number includes all CSS and JS files and the number of embeded objects found on the tested page
linkTested	Number of tested links
otherExt	Number of other external references like HTML forms etc
totalBytes	Total bytes received during the test (excluding HTML headers)
PING statistics (optional)	
pingLoss	Packet loss percentage
pingMin	Minimum packet round-trip time
pingAvg	Average packet round-trip time
pingMax	Maximum packet round-trip time

4.2.5 *doInstantTransactionTest* – Tests a transaction from the specified location and returns detailed results for every step in the transaction.

Request parameters:

Name	Description
targetID	A target ID or a list of comma separated target IDs. Set to 0 to show all targets of selected type.

Return values:

Name	Description
targetType	A string indicating the target type
status	The status message returned by the monitoring agent
respTime	Response Time in seconds
dnsTime	DNS time in seconds
connectTime	Connection time in seconds
firstByte	Time in seconds before the first byte of the server response is received
lastByte	Time in seconds needed to complete the download
stepNumber	Number of the step in the transaction
stepLabel	Step label as specified in your WebSitePulse control panel
stepStatus	Status message from the check of this step
stepRespTime	Total Time in seconds to perform the test of this step
stepSize	Size of the downloaded content
stepMd5	MD5 hash of the downloaded content
stepdnsTime	DNS time for the current step
stepconnectTime	Connect time for the current step
stepfirstByte	Time in seconds to receive the First Byte during the current step test
steplastByte	Time in seconds to download the contents of the current step
PING statistics (optional)	
stepPingLoss	Packet loss percentage
stepPingMin	Minimum packet round-trip time
stepPingAvg	Average packet round-trip time
stepPingMax	Maximum packet round-trip time

4.2.6 doInstantFullpageTransactionTest – Tests a fullpage transaction from the specified location and returns detailed results for every step in the transaction.

Request parameters:

Name	Description
targetID	A target ID or a list of comma separated target IDs. Set to 0 to show all targets of selected type.

Return values:

Name	Description
targetType	A string indicating the target type
status	The status message returned by the monitoring agent
respTime	Response Time in seconds
dnsTime	DNS time in seconds
connectTime	Connection time in seconds
firstByte	Time in seconds before the first byte of the server response is received
lastByte	Time in seconds needed to complete the download
stepNumber	Number of the step in the transaction
stepLabel	Step label as specified in your WebSitePulse control panel
stepStatus	Status message from the check of this step
stepRespTime	Total Time in seconds to perform the test of this step
stepSize	Size of the downloaded content
stepMd5	MD5 hash of the downloaded content
totalRefs	Total number of tested references
failedRefs	Failed references
imagesTested	Number of images tested
framesTested	Number of frames tested
otherInline	Number of other inline components tested. This number includes all CSS and JS files and the number of embeded objects found on the tested page
totalBytes	Total bytes received during the test (excluding HTML headers)
PING statistics (optional)	
stepPingLoss	Packet loss percentage
stepPingMin	Minimum packet round-trip time
stepPingAvg	Average packet round-trip time

stepPingMax	Maximum packet round-trip time
-------------	--------------------------------

4.2.7 *UpdateTargetStatus* – suspend or resume the monitoring of the selected target(s).

Request parameters:

Name	Description
targetType	A => All targets B => Basic type E => Advanced type F => Performance type I => Full-page type T => Performance Transactions M => Email Round-trip targets X => Fullpage Transactions
targetID	A target ID or a list of comma separated target IDs. Set to 0 to show all targets of selected type.
action	Should be one of : activate – resumes the monitoring of the selected targets suspend – suspend the selected targets.

Return values:

Name	Description
Status	A message describing the outcome of the executed API request

4.2.8 *doGetSettings* – get target settings of selected target(s).

Name	Description
targetType	A => All targets B => Basic type E => Advanced type F => Performance type I => Full-page type T => Performance Transactions M => Email Round-trip targets X => Fullpage Transactions
targetID	A target ID or a list of comma separated target IDs. Set to 0 to show all targets of selected type.

showBasicSettings	Boolean: TRUE – include basic target settings FALSE – don't include basic target settings
showAdvancedSettings	Boolean: TRUE – include advanced target settings FALSE – don't include advanced target settings
showMonitoringLocations	Boolean: TRUE – include monitoring locations info FALSE – don't include monitoring locations info

Return values:

Name	Description
Basic Settings	
label	Target label.
targetType	Target type string.
frequency	Monitoring interval in minutes.
gmtOffset	Target's GMT offset.
pingThreshold	Ping threshold value in ms. Applies to PING targets only.
timeout	Total target timeout.
maxRefs	Number of references to check (Fullpage targets and transactions)
keepLogs	Keep log details in months.
useTraceroute	Traceroute information on connection problems.
useTakeover	Configured takeover.
forcedMonitoring	Forced monitoring interval.
locationsNum	Number of locations currently monitoring a target.
snapshotError	Create snapshot on error.
screenshotError	Create screenshot on error.
isActive	A boolean value indicating whether the target is currently monitored or not.
Common Settings	
stepNumber	Indicates the consecutive number of a step in a transaction
stepLabel	Step Label (in transactions)
protocol	Protocol
host	Monitored host
requestPage	Request Page
requestCustom	Custom request

port	Port
usePing	Use ping.
useSSL	Use SSL.
requestMethod	Request method.
authUser	Username for authentication.
authPass	Password for authentication.
authDomain	Authenticate to domain.
queryString	Query String
lookupName	String to lookup for (DNS targets only).
lookupType	Lookup type (DNS targets only).
requestTimeout	Timeout for a single URL. This defines the maximum time to wait while downloading a step in performance transaction or a single page component with fullpage monitoring.
pageloadTimeout	This timeout will apply to the time needed to download all components in a page with fullpage monitoring. When the monitoring depth is higher than 0 this timeout will also apply to the time needed to test all subsequent pages.
Fullpage Settings (FullPage targets and transactions)	
Images	How to check images
Frames	How to check page frames
otherInline	How to check other inline components
links	How to check links
otherExternal	How to check other components which are not automatically loaded
reuseConnection	Whether to reuse opened TCP connection.
depthLevel	Page components depth level.
includeURLs	A list of URLs which may not be present in the page content but should be tested as well.
excludeURLs	A list of URLs which will not be tested even if they are present in the page content.
Page Content	
keywordNotFound	Keywords that should be present in the page content.
keywordsFound	Keywords that should not be present in the page content like error messages etc.
checkFileSize	Whether to check file size.
enableCompression	Whether to use compression.
MD5checksum	Compare a hash calculated from page content.

compareContent	How to compare page content.
comparePercent	Alert if content differs more than this percent.
Browser Emulation	
userAgent	What user agent string to be used.
cookies	Specifies custom cookies
acceptLanguage	What accept-language header field to be used.
metaRefresh	Follow meta refresh tags.
redirectJS	Follow JavaScript redirects.
Misc Settings	
bandwidthLimit	Use bandwidth limit (in kbps)
delayStep	Set delay in seconds before requesting the next step (transactions)
Email Transaction Settings	
smtpLabel	SMTP server label
smtpHost	SMTP host that will be used to deliver the test message. Could be empty if the MX records will be used.
smtpPort	Which port to use when connecting to the SMTP server
useMX	Whether to retrieve the list of SMTP servers to use from the hosts' MX records
destEmail	The email which will receive the test messages
fromEmail	The test messages will be sent using this email in the From: header.
smtpUser	The monitoring agent will authenticate to the SMTP server with this username.
smtpPassword	The monitoring agent will authenticate to the SMTP server with this password.
smtpResend	Try to resend the test message using the primary SMTP server if all previous attempts were unsuccessful
secondaryMX	Use this host as a secondary SMTP server
smtpSSL	SMTP server requires SSL
smtpPing	Whether to ping the primary SMTP server
smtpAuth	The authentication method which will be used with your SMTP server
recvLabel	Label for the server from which the test message should be receive
recvHost	The hostname/IP address of the email server
recvProtocol	Protocol – POP3 or IMAP
recvPort	The port on which the email server resides
recvUser	POP3/IMAP username

recvPass	POP3/IMAP password
recvPing	Whether to ping the email server
recvSSL	Whether to use SSL with the email server
recvWaitTime	Total time to wait before the message is reported undeliverable.

5 Errors

5.1. HTTP API

You've exceeded the maximum number of requests.

This error occurs when the user exceeds the limit of 1000 requests per day.

Missing or invalid access key

The client must specify a **key** parameter. You can obtain your key from <http://www.websitepulse.com/reports/wsdkey.php> The page requires logging in with your WebSitePulse user name and password.

Invalid key.

You have entered a wrong key.

Invalid username.

You have entered an invalid **username**.

Invalid API method.

The HTTP API currently accepts only the following methods: **doGetStatus**, **doGetUptime** and **doGetHistory**. See section 3.2 of this document for detailed description of each method.

Target(s) not specified.

You must sent a **target** parameter containing a comma separated list of target IDs or the special keyword "all". A full list of the targets in your account with their respective IDs can be found here:

<http://www.websitepulse.com/reports/wsdkey.php>

Location(s) not specified.

You must specify a **location** parameter. The **location** should contain a comma separated list of location IDs or the special keyword "all". A full list of WebSitePulse monitoring locations can be found here:

<http://www.websitepulse.com/services/network.php> – the location ID is the number next to the location label.

The delimiter cannot be the same as the enclose string.

To avoid ambiguity you cannot use the same string for both **delimiter** and **enclose** parameters. You should resend you request with different values fro those parameters.

No data found.

This error means that there's no data matching your criteria. Try to change the **startdate** and **enddate** parameters to specify different period or **location**.

5.2. WSDL API

Must supply a valid username and key.

You have entered an invalid **username** or **key**. You can obtain your key from <http://www.websitespulse.com/reports/wsdkey.php> The page requires logging in with your WebSitePulse user name and password.

Invalid key

The client must specify a **key** parameter. You can obtain your key from <http://www.websitespulse.com/reports/wsdkey.php> The page requires logging in with your WebSitePulse user name and password.

Invalid username

You have entered an invalid **username**.

Invalid target type.

Valid target types are:

- A => All targets
- B => Basic type
- E => Advanced type
- F => Performance type
- I => Full-page type
- T => Transactions
- M => Mail transaction

Invalid location ID.

You must specify a valid location ID. A full list of WebSitePulse location IDs can be found here: <http://www.websitespulse.com/services/network.php> – the location ID is the number next to the location label.

You've exceeded the maximum number of requests.

With the WSDL API every user can make up to 100 requests per day. If you want to use the WSDL API please send a request to support@websitespulse.com

Target is suspended and cannot be checked.

The **doInstantTest** methods cannot be used on suspended targets. You should login to your WebSitePulse.com control panel and activate the target before you can run this test,

Error connecting to remote server.

This error occurs when our WSDL server cannot fetch the test results from our monitoring location. If this error persists, please report it at support@websitespulse.com

6 Contact Information

Customer Support (available 24/7)

Email: support@websitepulse.com

Toll Free: 1-888-WSPULSE (977-8573), option 2

International: 1-407-380-1600

Billing and Payments (Mon-Fri 9:00-17:00 EST)

Email: support@websitepulse.com

Toll Free: 1-888-WSPULSE (977-8573), option 3

International: 1-407-380-1600

Marketing Communications (Mon-Fri 9:00-17:00 EST)

Name: Ognian (Oggie) Dimov

E-mail: dimoff@websitepulse.com

Toll Free: 1-888-WSPULSE (977-8573), ext. 808

International: 1-407-380-1600, ext. 808

Mailing Address

WebSitePulse

674 Saxon Boulevard,

Deltona, Florida 32725

USA

To send a fax to any department please fax to:

Toll Free: 1-888-977-8573

International: 1-407-380-1600

APPENDIX A: Examples

A1 HTTP API Examples:

1.1. GetStatus Method

1.1.1. Sample use – the following URL will request the most recent status of one monitored target from our Houston, TX monitoring location.

<http://api.websitepulse.com/textserver.php?username=demo&key=35e6340814655582a79039dbc1cb0b65&target=7182&location=1&method=GetStatus&format=xml>

XML Result:

```
<Result>
<item>
  <label>Target Label</label>
  <location>Houston, TX </location>
  <targetType>HTTP advanced</targetType>
  <frequency>20</frequency>
  <statusCode>UP</statusCode>
  <statusMessage>OK</statusMessage>
  <fails>0</fails>
  <responseTime>1.329</responseTime>
  <lastCheck>23:48 06/20/2006</lastCheck>
  <lastDown>07:57 06/21/2006</lastDown>
  <pingLoss>0.00</pingLoss>
  <pingMin>0.000</pingMin>
  <pingAvg>0.000</pingAvg>
  <pingMax>0.000</pingMax>
</item>
</Result>
```

1.1.2 Using in PHP code:

```
<?php
$url="http://api.websitepulse.com/textserver.php?";
$url.="username=demo";
$url.="&key=35e6340814655582a79039dbc1cb0b65";
$url.="&method=GetStatus";
$url.="&target=7182";
$url.="&location=1";
$url.="&format=csv1";
$handle = fopen($url, "r");

$resultLabels = array("label", "location", "targetType", "frequency", "statusCode",
"statusMessage", "fails", "responseTime", "lastCheck", "lastDown", "pingLoss",
"pingMin", "pingAvg", "pingMax");

while (($data = fgetcsv($handle, 1000, ";")) !== FALSE) {
    for ($i=0; $i<sizeof($resultLabels); $i++){
        echo $resultLabels[$i]. " => ".$data[$i]. "<br />\n";
    }
}
fclose($handle); ?>
```

1.2. GetUptime Method:

1.2.1. Sample use – The following URL will retrieve the summarized uptime statistics for May – June 2006 for a single target. It will show only the results from our Houston, TX monitoring location.

<http://api.websitepulse.com/textserver.php?username=demo&key=35e6340814655582a79039dbc1cb0b65&target=7182&location=1&startdate=200605&enddate=200606&method=GetUptime&format=xml>

XML Result:

```
<Result>
<item>
  <label>Target Label</label>
  <location>Houston, TX </location>
  <checks>570</checks>
  <fails>2</fails>
  <resptime>2.1878</resptime>
  <uptimepercent>99.6491</uptimepercent>
  <downtime>2975</downtime>
</item>
</Result>
```

1.2.2. Using in PHP code:

```
<?php
$url="http://api.websitepulse.com/textserver.php?";
$url.="username=demo";
$url.="&key=35e6340814655582a79039dbc1cb0b65";
$url.="&method=GetUptime";
$url.="&target=7182";
$url.="&location=1";
$url.="&startdate=200606";
$url.="&enddate=200606";
$url.="&format=csv1";

$handle = fopen($url, "r");

$resultLabels = array("label","location", "checks", "fails", "resptime",
"uptimepercent", "downtime");

while (($data = fgetcsv($handle, 1000, ";")) !== FALSE) {
    for ($i=0; $i<sizeof($resultLabels); $i++){
        echo $resultLabels[$i]. " => ".$data[$i]. "<br />\n";
    }
}

fclose($handle);
?>
```

1.3. GetHistory Method:

1.3.1. Sample use - The following URL will retrieve all status changes recorded during the period between May 1, 2006 and June 1 2006. It will show only the results from our Houston, TX monitoring location.

<http://api.websitepulse.com/textserver.php?username=demo&key=35e6340814655582a79039dbc1cb0b65&target=7182&location=1&startdate=2006-05-01&enddate=2006-06-01&method=GetHistory&format=xml>

XML Result:

```
<Result>
<item>
  <label>Target Label</label>
  <location>Houston, TX </location>
  <startdate>June 1, 2006 00:00:00</startdate>
  <enddate>June 20, 2006 03:59:59</enddate>
  <statushistory>
    <row>
      <status>UP</status>
      <statustype>OK</statustype>
      <starttime>06/12/2006 23:20</starttime>
      <endtime>06/20/2006 08:57</endtime>
      <duration>7d 9h 37m</duration>
    </row>
  </statushistory>
</item>
</Result>
```

1.3.2. Using in PHP code:

```
<?php
$url="http://api.websitepulse.com/textserver.php?";
$url.="username=demo";
$url.="&key=35e6340814655582a79039dbc1cb0b65";
$url.="&method=GetHistory";
$url.="&target=7182";
$url.="&location=1";
$url.="&startdate=2006-05-01";
$url.="&enddate=2006-06-01";
$url.="&format=csv1";
$handle = fopen($url, "r");

$resultLabels = array("status", "statusType", "starttime", "endtime", "duration");

while (($data = fgetcsv($handle, 1000, ";")) !== FALSE) {
    for ($i=0; $i<sizeof($resultLabels); $i++){
        echo $resultLabels[$i]. " => ".$data[$i]. "<br />\n";
    }
}

fclose($handle);
?>
```


2. WSDL Examples:

2.1. doGetStatus used in PHP code. Note: This example uses the NuSphere's **nusoap.php** library to handle SOAP requests.

```
<?php
require_once('nusoap.php');
$type=strtoupper($type);
$username="wspdemo";
$key="cabf996fb29c55809dbba11ba99a7e09";
//Please refer to section 4.1 Required
$location=1;
/*
A - all targets, B - port monitoring only, E - Advanced targets
F - Performance targets, I - Fullpage tests,
T - Transactions, M - Mail transaction
*/
if (strstr("ABEFITM",$type)===FALSE)$type="A";
/*
Target ID - check www.websitepulse.com for details or use 0 to show all targets of a
selected type
*/
$targetid=0;
/*This is the location of the WSDL file*/
$wsdlfile="wsp.wsdl";
//Proxy Information
$proxyserver="";//Put the name of your proxy server here
$proxyport="";//Put the port to connect to your proxy server here
$parameters=array($username,$key,$location,$type,$targetid);
$soapclient = new soapclient($wsdlfile,'wsdl');
if($proxyserver && $proxyport)$soapclient->setHTTPProxy($proxyserver,$proxyport);
$result=$soapclient->call('doGetStatus',$parameters);
//UNCOMMENT THE NEXT LINE IF YOU WANT DEBUG INFORMATION
//echo "<xmp>".$soapclient->debug_str."</xmp><br>\n";
if ($soapclient->fault){
    echo "Error: ".$soapclient->faultstring."<br>\n";
}
else{
    if(sizeof($result["resultElements"])){
        echo "<table border=1 cellspacing=0 cellpadding=5> <tr align=center>
        <td><b>Label</b></td><td><b>Frequency</b></td><td><b>Fails</b></td><td><b>Estimated
        downtime</b></td><td><b>Status Code</b></td> <td><b>Status</b></td><td><b>Last
        Check</b></td></tr>\n";
        for ($i=0;$i<sizeof($result["resultElements"]);$i++){
            echo "<tr>";
            echo "<td>".$result["resultElements"][$i]["targetType"]."</td>";
            echo "<td>".$result["resultElements"][$i]["label"]."</td>";
            echo "<td align=right>".$result["resultElements"][$i]["frequency"]."</td>";
            echo "<td>".$result["resultElements"][$i]["fails"]."</td>";
            echo "<td>".($result["resultElements"][$i]["fails"] *
            $result["resultElements"][$i]["frequency"])."</td>";
            echo "<td>".$result["resultElements"][$i]["statusCode"]."</td>";
            echo "<td>".$result["resultElements"][$i]["statusMessage"]."</td>";
            echo "<td>".$result["resultElements"][$i]["lastCheck"]."&nbsp;</td>";
            echo "</tr>\n";
        }
        echo "</table>";
    }
    else echo "<b><font color='Red'>No targets found!</font></b>";//Should not happen
}
?>
```

2.2. doGetUptime method used with PHP. Note: This example uses the NuSphere's *nusoap.php* library to handle SOAP requests.

```
<?php
require_once('nusoap.php');
$type=strtoupper($type);
$username="wspdemo";
$key="cabf996fb29c55809dbba11ba99a7e09";
//Please refer to section 4.1 Required
$location=1;
/*
A - all targets, B - port monitoring only, E - Advanced targets
F - Performance targets, I - Fullpage tests,
T - Transactions, M - Mail transaction
*/
if (strstr("ABEFITM",$type)===FALSE)$type="A";
/*
Target ID - check www.websitepulse.com for details or use 0 to show all targets of
a selected type
*/
$targetid=0;
$month=3;//leave empty for the current month
$year=2003;//leave empty for the current year
/*This is the location of the WSDL file*/
$wsdlfile="wsp.wsdl";
//Proxy Information
$proxyserver="";//Put the name of your proxy server here
$proxyport="";//Put the port to connect to your proxy server here
$parameters=array($username,$key,$location,$type,$targetid,$month,$year);
$soapclient = new soapclient($wsdlfile,'wsdl');
if($proxyserver && $proxyport)$soapclient->setHTTPProxy($proxyserver,$proxyport);
$result=$soapclient->call('doGetUptime',$parameters);
//UNCOMMENT THE NEXT LINE IF YOU WANT DEBUG INFORMATION
//echo "<xmp>".debug_str."</xmp><br>\n";
if ($soapclient->fault){
    echo "Error: ".$soapclient->faultstring."<br>\n";
}
else{
    if(sizeof($result["UptimeElements"])){
        echo "<table border=1 cellpadding=5><tr align=center>
        <td><b>Label</b></td><td><b>Checks</b></td><td><b>Fails</b></td> <td><b>Average
        Response Time</b></td><td><b>Uptime %</b></td></tr>\n";
        for ($i=0;$i<sizeof($result["UptimeElements"]);$i++){
            echo "<tr>";
            echo "<td>".$result["UptimeElements"][$i]["targetType"]."</td>";
            echo "<td>".$result["UptimeElements"][$i]["label"]."</td>";
            echo "<td align=right>".$result["UptimeElements"][$i]["checks"]."</td>";
            echo "<td>".$result["UptimeElements"][$i]["fails"]."</td>";
            echo "<td>".$result["UptimeElements"][$i]["respTime"]."</td>";
            echo "<td>".$result["UptimeElements"][$i]["uptimePercent"]." %</td>";
            echo "</tr>\n";
        }
        echo "</table>";
    }
    else echo "<b><font color='Red'>No targets found!</font></b>";//Should not happen
}??
```

2.3. doInstantTest method used with PHP. Note: This example uses the NuSphere's *nusoap.php* library to handle SOAP requests.

```
<?php
require_once('nusoap.php');
$username="wspdemo";
$key="cabf996fb29c55809dbba11ba99a7e09";
//Please refer to section 4.1 Required
$location=4;
/*
A - all targets, B - port monitoring only, E - Advanced targets
F - Performance targets, I - Fullpage tests,
T - Transactions, M - Mail transaction
*/
$type="b";
/*
Target ID - check www.websitepulse.com for details or use 0 to show all targets of
a selected type
*/
$targetid=131164;
/*This is the location of the WSDL file*/
$wsdlfile="wsp.wsdl";
//Proxy Information
$proxyserver=""; //Put the name of your proxy server here
$proxyport=""; //Put the port to connect to your proxy server here
$parameters=array($username,$key,$location,$type,$targetid);
$soapclient = new soapclient($wsdlfile,'wsdl');
if($proxyserver && $proxyport)$soapclient->setHTTPProxy($proxyserver,$proxyport);
$result=$soapclient->call('doInstantTest',$parameters);
//UNCOMMENT THE NEXT LINE IF YOU WANT DEBUG INFORMATION
//echo "<xmp>".$soapclient->debug_str."</xmp><br>\n";
if ($soapclient->fault){
    echo "Error: ".$soapclient->faultstring."<br>\n";
}
else{
    echo "<table border=1 cellpadding=5>";
    echo "<tr><td><b>Label</b></td><td>".$result["label"]."</td>";
    echo "<tr><td><b>Target Type</b></td><td>".$result["targetType"]."</td></tr>\n";
    echo "<tr><td><b>Status</b></td><td>".$result["status"]."</td></tr>\n";
    echo "<tr><td><b>Consecutive Fails</b></td><td>".$result["fails"]."</td></tr>\n";
    echo "<tr><td><b>Response Time</b></td><td>".$result["respTime"]."</td></tr>\n";
    if(isset($result["pingLoss"]))echo "<tr><td><b>PING Loss</b></td><td>".$result["pingLoss"]."</td></tr>\n";
    if(isset($result["pingMin"]))echo "<tr><td><b>PING Min</b></td><td>".$result["pingMin"]."</td></tr>\n";
    if(isset($result["pingAvg"]))echo "<tr><td><b>PING Avg</b></td><td>".$result["pingAvg"]."</td></tr>\n";
    if(isset($result["pingMax"]))echo "<tr><td><b>PING Max</b></td><td>".$result["pingMax"]."</td></tr>\n";
    echo "</table>";
}
?>
```

2.4. doUpdateTargetStatus method used with PHP. Note: This example uses the NuSphere's *nusoap.php* library to handle SOAP requests.

```
<?php
require_once( 'nusoap.php' );
$username="wspdemo";
$key="cabf996fb29c55809dbba11ba99a7e09";
/*
Please refer to section 4.2.6 Required
A - all targets, B - port monitoring only, E - Advanced targets
F - Performance targets, I - Fullpage tests,
T - Transactions, M - Mail transaction
*/
$type="b";
/*
Target ID - check www.websitepulse.com for details or use 0 to show all targets of
a selected type
*/
$targetid=131164;
/*
action - activate or suspend
*/
$action="suspend";
/*This is the location of the WSDL file*/
$wsdlfile="wsp.wsdl";
//Proxy Information
$proxyserver="";//Put the name of your proxy server here
$proxyport="";//Put the port to connect to your proxy server here
$parameters=array($username,$key,$location,$type,$targetid,$action);
$soapclient = new soapclient($wsdlfile,'wsdl');
if($proxyserver && $proxyport)$soapclient->setHTTPProxy($proxyserver,$proxyport);
$result=$soapclient->call('doUpdateTargetStatus',$parameters);
//UNCOMMENT THE NEXT LINE IF YOU WANT DEBUG INFORMATION
//echo "<xmp>".$soapclient->debug_str."</xmp><br>\n";
if ($soapclient->fault){
    echo "Error: ".$soapclient->faultstring."\n";
}
else{
    echo "<table border=1 cellspacing=0 cellpadding=5>";
    echo "<tr><td><b>Status</b></td><td>".$result."</td>";
    echo "</table>";
}

?>
```