



# Sunwire SolSwitch API

for Application Providers using JSON/HTTP

## Table of Contents

<b>Sunwire SolSwitch API</b> .....	<b>6</b>
<b>Call Functions</b> .....	<b>7</b>
Add Header .....	7
Example.....	7
Hold Call .....	8
Example.....	8
New Call .....	9
Example.....	10
Outbound Fax.....	11
Example.....	11
Transfer Call .....	13
Example.....	13
Transmit DTMF.....	14
Example.....	14
<b>Call Reporting</b> .....	<b>15</b>
Agent Stats .....	15
Example.....	16
Call Recording .....	17
Example.....	17
Caller ID Request.....	19
Example.....	19
CDR Export .....	21
Example.....	22
List Active Calls.....	24
Example.....	24
Live Queue Stats .....	26

Example.....	26
<b>System Management .....</b>	<b>28</b>
911 Info .....	28
Example.....	28
Add DID .....	29
Example.....	29
Create Customer .....	30
Example.....	31
Check Time Rule.....	32
Example.....	32
DID Report.....	33
Example.....	33
Find Me Follow Me Details .....	34
Example.....	34
Firewall Management Suite .....	35
/api/firewall/view .....	35
/api/firewall/add/ .....	37
/api/firewall/edit/ .....	39
/api/firewall/delete/ .....	41
Get Customer Information.....	42
Example.....	42
Monitoring Notifications.....	44
Example.....	44
Remove DID .....	45
Example.....	45
Scheduled Routing Details .....	46
Example.....	46

SIP Account Management Suite.....	47
/api/sipaccounts/view/ .....	47
/api/sipaccounts/add/ .....	49
/api/sipaccounts/edit/ .....	52
/api/sipaccounts/delete/ .....	54
Toggle Monitoring.....	55
Example.....	55
Toll Billing Info.....	56
Toll Rates.....	57
Example.....	57
Toll Reporting.....	58
Example.....	59
Trunk Management Suite .....	61
/api/trunks/view/.....	61
/api/trunks/add/ .....	63
/api/trunks/edit/.....	65
/api/trunks/delete/.....	67
<b>Error Messages .....</b>	<b>68</b>
API .....	68
CallID .....	68
Call.....	68
Channel .....	68
Customer.....	68
Date.....	68
Destination.....	68
DID.....	69
Extension.....	69

Enabled .....	69
Fax .....	69
File .....	69
IP .....	69
Key.....	70
Recording .....	70
Time Rules.....	70
Tone .....	70
Unknown.....	70
<b>Route Types.....</b>	<b>71</b>
ID, RouteType.....	71
<b>Example API Calls.....</b>	<b>72</b>
PHP CDR Export Example Call .....	72
CommandLine cURL List Active Calls Example Call .....	73
Python Scheduled Routing Details Example Call .....	74

## Sunwire SolSwitch API

This document outlines the API interface used for interacting with SolSwitch systems through outside applications. The functions available and the guidelines for using them are described below. The API functions use JSON for communication and follow the following guidelines:

The application provider will provide a delivery agent that can send JSON over HTTP as content-type application/json.

For security reasons, the API requires either a system generated API key to be used or a specific IP address to be given permission to connect. For additional security, both measures can be enabled.

The JSON document must be posted to the API URL as the contents of the HTTP POST, i.e. the HTTP POST will deliver the document as content-type application/json and not as an embedded file inside an encoded form. This means that the delivery agent will need to access the HTTP POST content directly instead of using encoded form methods or tools for encoding name-value pairs.

All API URLs will be formatted as follows, where 'solswitch' is the IP or domain of the desired system, and 'function' is the name of the function being called (as provided below):

**solswitch/api/function/**

Any questions related to the use and functionality of the API can be directed to [software@sunwire.ca](mailto:software@sunwire.ca)

## Call Functions

### Add Header

Insert a custom header into an upcoming outbound call.

**Add Header URL:** solswitch/api/addheader/

The Request must contain the following attributes:

<b>Key</b>	API key for connection to the system. Required field if an API key has been enabled.
<b>srcnumber</b>	The source number to apply the new custom header to. Optional if <b>dstnumber</b> is set. If call originating from an extension, srcnumber should be the extension number.
<b>dstnumber</b>	The destination number to apply the new custom header to. Optional if <b>srcnumber</b> is set.
<b>header-name</b>	The name of the custom header to be applied.
<b>value</b>	The value of the custom header.

The Response will contain the following attributes:

<b>Status</b>	'OK' or 'FAIL'
<b>Reason</b>	Reason for fail status (Only if status is FAIL). A list of error messages and explanations can be found at the end of this document.

### Example

<p><b>Request</b></p> <pre>POST /api/addheader HTTP/1.1 Content-Type: application/json Content-Length: 124 {   "Key": "FzeKKeEKSIUXa7ZRMx3FciZwXG2FW8jd",   "srcnumber": "123",   "header-name": "ExampleHeader",   "value": "ExampleValue" }</pre>	<p><b>Response (SUCCESS)</b></p> <pre>{   "Status": "OK" }</pre> <p><b>Response (FAIL)</b></p> <pre>{   "Status": "FAIL",   "Reason": "Missing required field destination number or source number "</pre>
---	---

## Hold Call

The Hold Call function is used to put an active call on hold.

**Hold Call URL:** solswitch/api/holdcall/

The Request must contain the following attributes:

<b>Channel</b>	Channel of an active call to be put on hold, as returned by 'List Active Calls'.
<b>Customer</b>	ID of the customer the function should run on, should be a numeric value. Optional field, if not provided the system will default to CustID 1.
<b>Key</b>	API key for connection to the system. Required field if an API key has been enabled.

The Response will contain the following attributes:

<b>Status</b>	'OK' or 'FAIL'
<b>Reason</b>	Reason for fail status (Only if status is FAIL). A list of error messages and explanations can be found at the end of this document.

## Example

<p><b>Request</b></p> <pre>POST /api/holdcall HTTP/1.1 Content-Type: application/json Content-Length: 96 {   "Channel": "SIP/Example2-000003ee",   "Customer": "4",   "Key": "FzeKKeEKSIUXa7ZRMx3FciZwXG2FW8jd" }</pre>	<p><b>Response (SUCCESS)</b></p> <pre>{   "Status": "OK" }</pre> <p><b>Response (FAIL)</b></p> <pre>{   "Status": "FAIL",   "Reason": "Unknown Customer ID" }</pre>
---	---



### New Call

The New Call function is used to generate a call from a user’s extension to a specified destination.

**New Call URL:** solswitch/api/newcall/

The Request must contain the following attributes:

<b>Extension</b>	The extension number of the user on the system. Extensions are numeric values, 3 to 5 digits in length.
<b>Destination</b>	The destination the call should be connected to. Destination is a numeric value.
<b>Customer</b>	ID of the customer the function should run on, should be a numeric value. Optional field, if not provided the system will default to CustID 1.
<b>Key</b>	API key for connection to the system. Required field if an API key has been enabled.

The Request may contain the following attributes:

<b>CustomDVDData</b>	<p>An array containing key value pairs for substitution when using a DigitValidation as the Extension attribute.</p> <p>When processing replacements we use \${} to determine variable names. For example to replace a variable URL within the request you would use \${URL} to denote where to place that data and pass an array containing the key URL with some value.</p>
<b>CustomDVDigits</b>	<p>An array containing special instruction for handling digits being pressed when using a DigitValidation as the Extension attribute. The digit is the index of the array with the value containing the instructions. All attribute values are passed through substitution for the CustomDVDData array. Setting the URL attribute to “https://\${URL}/api/” would replace \${URL} with whatever you set in the above array.</p> <p><u>Attributes</u></p> <p><b>URL</b> – URL to use in place of the set URL for the Digit Validation</p> <p><b>Method</b> – POST or GET</p> <p><b>Auth</b> – Type of authentication to use. For example “Header” authorization would expect AuthInfo to contain “type” and “credentials”</p> <p><b>AuthInfo</b> – Dependant on the Auth type above</p>

	<p><b>Response</b> – A special response to send if the request to the URL was successful</p> <p><b>CustomData</b> – An array containing data to be sent to the URL instead of the regular call data</p> <p><b>Headers</b> – An array of headers to add to the request</p> <p><b>NoURL</b> – Boolean value determining whether or not to try to call out to the URL of the DigitValidation</p>
--	---

The Response will contain the following attributes:

<b>Status</b>	'OK' or 'FAIL'
<b>CallID</b>	ID of the connected call (On OK return status).
<b>Channel</b>	Source channel of the connected call (On OK return status).
<b>Reason</b>	Reason for fail status (Only if status is FAIL). A list of error messages and explanations can be found at the end of this document.

Example

<p><b>Request</b></p> <pre>POST /api/newcall HTTP/1.1 Content-Type: application/json Content-Length: 109 {   "Extension": "123",   "Destination": "17055551234",   "Customer": "4",   "Key": "FzeKKeEKSIUXa7ZRMx3FcizwXG2FW8jd" }</pre>	<p><b>Response (SUCCESS)</b></p> <pre>{   "Status": "OK",   "CallID": "758923",   "Channel":   "SIP/Example2-000004fb" }</pre> <p><b>Response (FAIL)</b></p> <pre>{   "Status": "FAIL",   "Reason": "Error with Key" }</pre>
---	--

## Outbound Fax

The Outbound Fax function is used to send a fax to a specified number.

**Outbound Fax URL:** solswitch/api/fax/

The Request may contain the following attributes:

<b>Customer</b>	ID of the customer the function should run on, should be a numeric value. Optional field, if not provided the system will default to CustID 1.
<b>Key</b>	API key for connection to the system. Required field if an API key has been enabled.
<b>To</b>	The fax number to send to. No '-'s allowed
<b>From</b>	The fax number to send from. No '-'s allowed. Must be set up as a FaxToEmail DID.
<b>File</b>	Location of the .pdf file to send. This file must be accessible from the SolSwitch.

The Response will contain the following attributes:

<b>Status</b>	'OK' or 'FAIL'
<b>Reason</b>	Reason for fail status (Only if status is FAIL). A list of error messages and explanations can be found at the end of this document.
<b>Status</b>	The only thing we return with this is whether or not the fax was queued properly.

## Example

<b>Request</b> POST /api/fax HTTP/1.1 Content-Type: application/json Content-Length: 74 { "Customer": "4", "Key": "FzeKKeEKSIUXa7ZRMx3FciZwXG2FW8jd", "To": "7055553482", "From": "7055551234", "File": "http://path.to.file/faxMe.pdf" }
<b>Response (SUCCESS)</b> { "Status": "OK" }

**Response (FAIL)**

```
{  
  "Status": "FAIL",  
  "Reason": "You must upload a file to fax"  
}
```

## Transfer Call

The Transfer Call function is used to transfer a current active call to a new destination.

**Transfer Call URL:** solswitch/api/transfercall/

The Request must contain the following attributes:

<b>Channel</b>	Channel of the call to be transferred to a new destination, as returned by 'List Active Calls'.
<b>Destination</b>	The destination the call should be transferred to. Destination is a numeric value.
<b>Customer</b>	ID of the customer the function should run on, should be a numeric value. Optional field, if not provided the system will default to CustID 1.
<b>Key</b>	API key for connection to the system. Required field if an API key has been enabled.

The Response will contain the following attributes:

<b>Status</b>	'OK' or 'FAIL'
<b>Reason</b>	Reason for fail status (Only if status is FAIL). A list of error messages and explanations can be found at the end of this document.

## Example

<p><b>Request</b></p> <pre>POST /api/transfercall HTTP/1.1 Content-Type: application/json Content-Length: 124 {   "Channel": "SIP/Example2-000003ee",   "Destination": "17055551234",   "Customer": "4",   "Key": "FzeKKeEKSIUXa7ZRMx3FciZwXG2FW8jd" }</pre>	<p><b>Response (SUCCESS)</b></p> <pre>{   "Status": "OK" }</pre> <p><b>Response (FAIL)</b></p> <pre>{   "Status": "FAIL",   "Reason": "IP address not allowed" }</pre>
--	--

### Transmit DTMF

The Transmit DTMF function is used to send a DTMF tone to a specified channel.

**Transmit DTMF URL:** solswitch/api/dtmf/

The Request must contain the following attributes:

<b>Tone</b>	The DTMF tone to be transmitted, accepted values are: <b>0-9 * #</b>
<b>Channel</b>	Channel of an active call where the tones should be sent. This would be the channel of the desired recipient, as returned by the 'List Active Calls' function.
<b>Customer</b>	ID of the customer the function should run on, should be a numeric value. Optional field, if not provided the system will default to CustID 1.
<b>Key</b>	API key for connection to the system. Required field if an API key has been enabled.

The Response will contain the following attributes:

<b>Status</b>	'OK' or 'FAIL'
<b>Reason</b>	Reason for fail status (Only if status is FAIL). A list of error messages and explanations can be found at the end of this document.

### Example

<p><b>Request</b></p> <pre>POST /api/dtmf HTTP/1.1 Content-Type: application/json Content-Length: 107 {   "Tone": "#1",   "Channel": "SIP/Example-000003ee",   "Customer": "4",   "Key": "FzeKKeEKSIUXa7ZRMx3FciZwXG2FW8jd" }</pre>	<p><b>Response (SUCCESS)</b></p> <pre>{   "Status": "OK" }</pre> <p><b>Response (FAIL)</b></p> <pre>{   "Status": "FAIL",   "Reason": "Unknown Channel" }</pre>
---	---

## Call Reporting

### Agent Stats

The Agent Stats function returns all relevant information for Agent busy status.

**Agent Stats URL:** solswitch/api/agentstats/

The Request may contain the following attributes:

<b>Customer</b>	ID of the customer the function should run on, should be a numeric value. Optional field, if not provided the system will default to CustID 1.
<b>Key</b>	API key for connection to the system. Required field if an API key has been enabled.
<b>Extension</b>	Optional. If provided will only return information related to given extension.
<b>Date</b>	Optional. If provided will only return information related to given date. If no date is provided, return will default to the current date. Format: YYYY-MM-DD
<b>DateStart</b>	Conditionally Optional. If provided, <b>DateEnd</b> is required. This will limit the returned data to be within <b>DateStart</b> and <b>DateEnd</b> . Format: YYYY-MM-DD
<b>DateEnd</b>	Conditionally Optional. If provided, <b>DateStart</b> is required. This will limit the returned data to be within <b>DateStart</b> and <b>DateEnd</b> . Format: YYYY-MM-DD

The Response will contain the following attributes:

<b>Status</b>	'OK' or 'FAIL'
<b>Reason</b>	Reason for fail status (Only if status is FAIL). A list of error messages and explanations can be found at the end of this document.
<b>Extension IDs</b>	Details on the individual extensions (On OK return status)  <u>Attributes</u> <b>ActionIDs</b> – Array of all actions performed by the extension
<b>ActionIDs</b>	Array of all actions performed by the extension  <u>Attributes</u> <b>Action</b> – The new status of the extension <b>DateTime</b> – The datetime this action was done

	<p><b>DateTimeUnix</b> – The unixtime the action was done</p> <p><b>ReasonCode</b> – The reason code used to enable busy/paused status if one was required</p>
--	--

Example

<b>Request</b>	<pre>POST /api/agentstats HTTP/1.1 Content-Type: application/json Content-Length: 74 {   "Customer": "4",   "Key": "FzeKKeEKSIUXa7ZRMx3FciZwXG2FW8jd",   "Date": "2018-05-30" }</pre>
<b>Response (SUCCESS)</b>	<pre>{   "Status": "OK",   "220": {     "978398": {       "Action": "In",       "DateTime": "2018-05-30 00:00:00",       "DateTimeUnix": 1527652800,       "ReasonCode": ""     },     "978400": {       "Action": "Paused",       "DateTime": "2018-05-30 11:28:35",       "DateTimeUnix": 1527694115,       "ReasonCode": "10"     },     "978401": {       "Action": "Out",       "DateTime": "2018-05-30 23:59:59",       "DateTimeUnix": 1527739199,       "ReasonCode": ""     }   } }</pre>
<b>Response (FAIL)</b>	<pre>{   "Status": "FAIL",   "Reason": "No matching end time" }</pre>



### Call Recording

The Call Recording function is used to retrieve the URL(s) for the audio file(s) of a recorded call. A single call could have multiple recordings if the call went to multiple places on the system (ex, if the call was transferred to a second extension).

**Call Recording URL:** solswitch/api/recording/

The Request must contain the following attributes:

<b>CallID</b>	Unique ID of the call to retrieve recordings for
<b>Customer</b>	ID of the customer the function should run on, should be a numeric value. Optional field, if not provided the system will default to CustID 1.
<b>Key</b>	API key for connection to the system. Required field if an API key has been enabled.

The Response will contain the following attributes:

<b>Status</b>	'OK' or 'FAIL'
<b>Recording</b>	<p>Array of call recordings and relevant information.</p> <p><u>Attributes</u></p> <p><b>StartTime</b> – Time that this portion of the call connected</p> <p><b>EndTime</b> – Time that this portion of the call ended</p> <p><b>Duration</b> – Duration of this portion of the call</p> <p><b>SourceName</b> – Name of the caller</p> <p><b>SourceNumber</b> – Extension or number of caller</p> <p><b>SourceChannel</b> – Channel of the source of the call</p> <p><b>DestinationNumber</b> – Extension or number the call was connected to</p> <p><b>DestinationChannel</b> – Channel of the destination of the call</p> <p><b>DialedNumber</b> – Actual number dialed for this portion of the call</p> <p><b>URL</b> – URL to download the MP3 file of the recording</p>
<b>Reason</b>	Reason for fail status (Only if status is FAIL). A list of error messages and explanations can be found at the end of this document.

### Example

<p><b>Request</b></p> <pre>POST /api/recording HTTP/1.1 Content-Type: application/json Content-Length: 78</pre>
---

```
{
  "CallID": "62135",
  "Customer": "4",
  "Key": "FzeKKeEKSIUXa7ZRMx3FciZwXG2FW8jd"
}

Response (SUCCESS)
{
  "Status": "OK"
  "Recording": [
    {
      "StartTime": "2018-01-20 14:00:00",
      "EndTime": "2018-01-20 14:01:23",
      "Duration": "00:01:23",
      "SourceName": "Example Caller",
      "SourceNumber": "7055551234",
      "SourceChannel": "SIP/Example3-000006e1",
      "DestinationNumber": "456",
      "DestinationChannel": "SIP/Example2-000003ee",
      "DialedNumber": "7052221234",
      "URL": "solswitch/download-
file.php?id=621351&sc=a7ebe5fa6b5c110d9be13d23fefaa256"
    },
    {
      "StartTime": "2018-01-20 14:01:23",
      "EndTime": "2018-01-20 14:14:35",
      "Duration": "00:13:12",
      "SourceName": "Example Caller",
      "SourceNumber": "7055551234",
      "SourceChannel": "SIP/Example3-000006e1",
      "DestinationNumber": "654",
      "DestinationChannel": "SIP/Example-000003f2",
      "DialedNumber": "654",
      "URL": "solswitch/download-
file.php?id=721892&sc=c56afe7890d5c64b784cadeffe45acb45"
    },
  ]
}

Response (FAIL)
{
  "Status": "FAIL",
  "Reason": "Unknown CallID"
}
```

### Caller ID Request

The Caller ID Request function returns all relevant information about an ongoing call.

**Caller ID Request URL:** solswitch/api/calleridrequest/

The Request may contain the following attributes:

<b>Customer</b>	ID of the customer the function should run on, should be a numeric value. Optional field, if not provided the system will default to CustID 1.
<b>Key</b>	API key for connection to the system. Required field if an API key has been enabled.
<b>Extension</b>	Extension number of the person you'd like to get the Caller ID information from.

The Response will contain the following attributes:

<b>Status</b>	'OK' or 'FAIL'
<b>Reason</b>	Reason for fail status (Only if status is FAIL). A list of error messages and explanations can be found at the end of this document.
<b>DailedNumber</b>	This is the DID the calling party dialed to begin the call
<b>IVROption</b>	If the person calling came in through an IVR this will be the option selected. Otherwise it will be -1
<b>CallerNumber</b>	This is the caller ID number of the calling party
<b>CallerName</b>	This is the caller ID name of the calling party. If no caller ID name was supplied this will be the same as the <b>CallerNumber</b>

### Example

<p><b>Request</b></p> <pre>POST /api/calleridrequest HTTP/1.1 Content-Type: application/json Content-Length: 74 {   "Customer": "4",   "Key": "FzeKKeEKSIUXa7ZRMx3FciZwXG2FW8jd",   "Extension": "220" }</pre>
<p><b>Response (SUCCESS)</b></p> <pre>{   "Status": "OK",   "DialedNumber": "7055551234",   "IVROption": "3",</pre>

```
"CallerNumber": "7055554321",  
"CallerName": "Test Caller"  
}  
Response (FAIL)  
{  
  "Status": "FAIL",  
  "Reason": "Invalid extension number given"  
}
```

## CDR Export

The CDR Export function returns all relevant information about calls between 2 dates.

**CDR Export URL:** solswitch/api/cdreexport/

The Request may contain the following attributes:

<b>Customer</b>	ID of the customer the function should run on, should be a numeric value. Optional field, if not provided the system will default to CustID 1.
<b>Key</b>	API key for connection to the system. Required field if an API key has been enabled.
<b>DateStart</b>	This will limit the returned data to be within <b>Date Start</b> and <b>Date End</b> . Format: YYYY-MM-DD
<b>DateEnd</b>	This will limit the returned data to be within <b>Date Start</b> and <b>Date End</b> . Format: YYYY-MM-DD

The Response will contain the following attributes:

<b>Status</b>	'OK' or 'FAIL'
<b>Reason</b>	Reason for fail status (Only if status is FAIL). A list of error messages and explanations can be found at the end of this document.
<b>CallID</b>	This has an array of returned data sorted by CallID
<b>LegID</b>	The ID of the call Leg
<b>StartTime</b>	This is the time the call connected to the SolSwitch
<b>ConnectTime</b>	This is the time the call was connected to a second party
<b>EndTime</b>	This is the time the call was ended by either party
<b>SourceName</b>	Caller ID name of the calling party
<b>SourceNumber</b>	Caller ID number of the calling party
<b>DestinationName</b>	Caller ID name of the called party
<b>DestinationNumber</b>	Caller ID number of the called party
<b>Disposition</b>	This is what happened during that leg of the call. It will show if a call was answered, transferred, etc...
<b>DialedNumber</b>	This is the number that was dialed to reach the called party
<b>SourceType</b>	See <a href="#">RouteTypes</a> for ID values
<b>DestinationType</b>	See <a href="#">RouteTypes</a> for ID values
<b>CallCode</b>	Call Reason Code if agent was required to enter a reason for the call.

## Example

**Request**

```
POST /api/cdrexport HTTP/1.1
Content-Type: application/json
Content-Length: 74
{
  "Customer": "4",
  "Key": "FzeKKeEKSIUXa7ZRMx3FciZwXG2FW8jd",
  "DateStart": "2018-05-15 00:00:00",
  "DateEnd": "2018-05-15 23:59:59"
}
```

**Response (SUCCESS)**

```
{
  "Status": "OK",
  "669932": [{
    "LegID": "669932",
    "StartTime": "2018-05-15 11:07:45",
    "ConnectTime": "",
    "EndTime": "2018-05-15 11:07:47",
    "SourceName": "Test Phone #1",
    "SourceNumber": "220",
    "DestinationName": "Test Phone #2",
    "DestinationNumber": "290",
    "Disposition": "NOANSWER",
    "DialedNumber": "290",
    "SourceType": "7",
    "DestinationType": "7",
    "CallCode": ""
  },
  {
    "LegID": "669933",
    "StartTime": "2018-05-15 11:31:42",
    "ConnectTime": "2018-05-15 11:31:44",
    "EndTime": "2018-05-15 11:31:51",
    "SourceName": "Test Phone #1",
    "SourceNumber": "7055551234",
    "DestinationName": "Test Phone #2",
    "DestinationNumber": "290",
    "Disposition": "ANSWERED",
    "DialedNumber": "7055553584",
    "SourceType": "15",
    "DestinationType": "7",
    "CallCode": ""
  }
  ]
}
```

**Response (FAIL)**

```
{  
  "Status": "FAIL",  
  "Reason": "No matching end time"  
}
```

**\*\* note if a call has multiple legs it will show as an array of legs. Otherwise it will be a single object**

### List Active Calls

The List Active Calls function returns all relevant information of current active calls.

**List Active Calls URL:** solswitch/api/listactivecalls/

The Request may contain the following attributes:

<b>Customer</b>	ID of the customer the function should run on, should be a numeric value. Optional field, if not provided the system will default to CustID 1.
<b>Key</b>	API key for connection to the system. Required field if an API key has been enabled.

The Response will contain the following attributes:

<b>Status</b>	'OK' or 'FAIL'
<b>Reason</b>	Reason for fail status (Only if status is FAIL). A list of error messages and explanations can be found at the end of this document.
<b>Calls</b>	<p>Details on the individual calls (On OK return status)</p> <p><u>Attributes</u></p> <p><b>CallID</b> – The unique ID of the call</p> <p><b>SourceNumber</b> – The phone number or extension of the source of the call</p> <p><b>SourceChannel</b> – The call channel connected to the source of the call</p> <p><b>DestinationNumber</b> – The phone number or extension of the destination of the call</p> <p><b>DestinationChannel</b> – The call channel connected to the destination of the call</p> <p><b>Duration</b> – Length of the call in format HH:MM:SS</p>

### Example

<p><b>Request</b></p> <pre>POST /api/listactivecalls HTTP/1.1 Content-Type: application/json Content-Length: 74 {   "Customer": "4",   "Key": "FzeKKeEKSIUXa7ZRMx3FciZwXG2FW8jd" }</pre>
<p><b>Response (SUCCESS)</b></p> <pre>{   "Status": "OK",</pre>



```
"Calls": [  
  {  
    "CallID": "667890",  
    "SourceNumber": "7055551234",  
    "SourceChannel": "SIP/Example-000003f2",  
    "DestinationNumber": "123",  
    "DestinationChannel": "SIP/Example2-000003ee",  
    "Duration": "00:01:23"  
  },  
  {  
    "CallID": "667891",  
    "SourceNumber": "345",  
    "SourceChannel": "SIP/Example3-000006e1",  
    "DestinationNumber": "7055551234",  
    "DestinationChannel": "SIP/Example4-000004fe",  
    "Duration": "00:00:46"  
  },  
]  
}  
Response (FAIL)  
{  
  "Status": "FAIL",  
  "Reason": "IP does not have access"  
}
```

Live Queue Stats

Insert a custom header into an upcoming outbound-call

**Add Header URL:** solswitch/api/livequeuestat/

The Request must contain the following attributes:

<b>Key</b>	API key for connection to the system. Required field if an API key has been enabled.
<b>customer</b>	ID of the customer the function should run on, should be a numeric value. Optional field, if not provided the system will default to CustID 1.
<b>queueID</b>	ID of the Queue the function should run on, should be numeric value. Optional field, if not provided the system will default to All Queues

The Response will contain the following attributes:

<b>Status</b>	'OK' or 'FAIL'
<b>Ext</b>	Extension of the Queue
<b>Queue</b>	Name of the Queue
<b>Calls</b>	Current Calls of that queue
<b>Callback</b>	Number of callers waiting to be called back
<b>Avail.Agnts</b>	Available Agents for that queue
<b>Abandoned</b>	Number of calls Abandoned
<b>WaitTime</b>	Current smallest wait time
<b>EstimatedWaitTime</b>	Average Wait time for that queue
<b>Voicemail</b>	Current amount of voicemail
<b>ServiceLevel</b>	Service Level of the queue at that moment
<b>Reason</b>	Reason for fail status (Only if status is FAIL). A list of error messages and explanations can be found at the end of this document.

Example

```

Request
POST /api/livequeuestat HTTP/1.1
Content-Type: application/json
Content-Length: 124
{
  "Key": "FzeKKeEKSwXG2FW8jd",
  "customer": "1",
  "queueID": "7"
}
    
```

**Response (SUCCESS)**

```
[
  {
    "Ext": "222",
    "Queue": "JVQueue",
    "Calls": "0",
    "Callback": "0",
    "Avail.Agnts": "2",
    "Answered": "0",
    "Abandoned": "0",
    "WaitTime": "00:00:00",
    "EstimatedWaitTime": "00:00:07",
    "Voicemail": "0",
    "ServiceLevel": "0.0000"
  },
  {will repeat for every available/requested Queue}
  {
    "Status": "OK"
  }
]
```

**Response (FAIL)**

```
{
  "Status": "FAIL",
  "Reason": "No access to customer number 1"
}
```

## System Management

### 911 Info

List the current 911 information for each DID on the SolSwitch.

**911 Info URL:** solswitch/api/911info/

The Request must contain the following attributes:

<b>Key</b>	API key for connection to the system. Required field if an API key has been enabled.
------------	--

The Response will contain the following attributes:

<b>Status</b>	'OK' or 'FAIL'
<b>Data</b>	Array of 911 information
<b>Reason</b>	Reason for fail status (Only if status is FAIL). A list of error messages and explanations can be found at the end of this document.

### Example

<p><b>Request</b></p> <pre>POST /api/911info HTTP/1.1 Content-Type: application/json Content-Length: 124 {   "Key": "FzeKKeEKSIUXa7ZRMx3FciZwXG2FW8jd" }</pre>	<p><b>Response (SUCCESS)</b></p> <pre>{   "Status": "OK",   "Data" : {     "custnum" : "12345",     "did" : "1234567894",     "streetnumber" : "850",     "streetname" : "Fake St",     "unit" : "2",     "city" : "Sudbury",     "province" : "ON",     "country" : "Canada",     "postalcode" : "P3B1Y4"   },   { Continues for every entry } }</pre> <p><b>Response (FAIL)</b></p> <pre>{   "Status": "FAIL",   "Reason": "IP address not allowed" }</pre>
--	---

## Add DID

Add a new DID to the SolSwitch

**Transfer Call URL:** solswitch/api/adddid/

The Request must contain the following attributes:

<b>Key</b>	API key for connection to the system. Required field if an API key has been enabled.
<b>DID</b>	The new DID to add

The Response will contain the following attributes:

<b>Status</b>	'OK' or 'FAIL'
<b>Reason</b>	Reason for fail status (Only if status is FAIL). A list of error messages and explanations can be found at the end of this document.

## Example

<b>Request</b> POST /api/adddid HTTP/1.1 Content-Type: application/json Content-Length: 124 { "Key": "FzeKKeEKSIUXa7ZRMx3FciZwXG2FW8jd", "DID": "7051234567" }	<b>Response (SUCCESS)</b> { "Status": "OK" } <b>Response (FAIL)</b> { "Status": "FAIL", "Reason": "Invalid DID provided" }
---	---

### Create Customer

The Create Customer function will create a customer account on the SolSwitch using the supplied information and return the new customer ID.

**Create Customer URL:** solswitch/api/createcustomer/

The Request may contain the following attributes:

<b>Key</b>	API key for connection to the system. Required field if an API key has been enabled.
<b>RouteGroupID</b>	The ID of the default route group to be assigned
<b>TollPlanID</b>	The ID of the toll plan to apply to this customer
<b>MaxSIPAccounts</b>	The maximum number of SIP accounts to allow the customer to use
<b>MaxTrunks</b>	The maximum number of trunks this customer can use at once
<b>CustomerNumber</b>	The internal customer's number. To be used for toll billing if you need a specific customer number
<b>FirstName</b>	First name of the customer
<b>LastName</b>	Last name of the customer
<b>Company</b>	Optional. The company the customer represents.
<b>AddressStreetNumber</b>	Optional. Address street number
<b>AddressStreetName</b>	Optional. Address street name
<b>AddressUnit</b>	Optional. Address unit number
<b>PostalCode</b>	Optional. Postal code (L#L#L#)
<b>City</b>	Optional. City the company is located in
<b>Province</b>	Optional. Province code (ON)
<b>Country</b>	Optional. Country of residence
<b>PhoneNumber</b>	Optional. Phone number without any '-'s

The Response will contain the following attributes:

<b>Status</b>	'OK' or 'FAIL'
<b>Reason</b>	Reason for fail status (Only if status is FAIL). A list of error messages and explanations can be found at the end of this document.
<b>CustomerID</b>	This is a 0 or 1 boolean representation of the state

## Example

**Request**

```
POST /api/createcustomer HTTP/1.1
Content-Type: application/json
Content-Length: 74
{
  "Customer": "4",
  "Key": "FzeKKeEKSIUXa7ZRMx3FciZwXG2FW8jd",
  "RouteGroupID": "7",
  "TollPlanID": "2",
  "MaxSIPAccounts": "2",
  "MaxTrunks": "4",
  "CustomerNumber": "123456789",
  "FirstName": "Test",
  "LastName": "Name"
}
```

**Response (SUCCESS)**

```
{
  "Status": "OK",
  "CustomerID": "127"
}
```

**Response (FAIL)**

```
{
  "Status": "FAIL",
  "Reason": "Missing required field FirstName"
}
```

### Check Time Rule

Used to find the current status (Open or Closed) of a time rule on the system.

**Check Time Rule URL:** solswitch/api/checktimerule/

The Request may contain the following attributes:

<b>Customer</b>	ID of the customer the function should run on, should be a numeric value. Optional field, if not provided the system will default to CustID 1.
<b>Key</b>	API key for connection to the system. Required field if an API key has been enabled.
<b>ID</b>	The time rule's ID

The Response will contain the following attributes:

<b>Status</b>	'OK' or 'FAIL'
<b>Reason</b>	Reason for fail status (Only if status is FAIL). A list of error messages and explanations can be found at the end of this document.
<b>TimeRuleStatus</b>	This is a 0 or 1 boolean representation of the state
<b>TimeRuleText</b>	This is the text form of the above (0 – Open, 1 – Closed)

### Example

<p><b>Request</b></p> <pre>POST /api/checktimerule HTTP/1.1 Content-Type: application/json Content-Length: 74 {   "Customer": "4",   "Key": "FzeKKeEKSIUXa7ZRMx3FciZwXG2FW8jd",   "ID": "132" }</pre>
<p><b>Response (SUCCESS)</b></p> <pre>{   "Status": "OK",   "TimeRuleStatus": "0",   "TimeRuleText": "Open" }</pre> <p><b>Response (FAIL)</b></p> <pre>{   "Status": "FAIL",   "Reason": "Time rule not found" }</pre>



### DID Report

Pull a list of DIDs on the system with their current customer and route information.

**DID Report URL:** solswitch/api/didreport/

The Request must contain the following attributes:

<b>Key</b>	API key for connection to the system. Required field if an API key has been enabled.
------------	--

The Response will contain the following attributes:

<b>Status</b>	'OK' or 'FAIL'
<b>Reason</b>	Reason for fail status (Only if status is FAIL). A list of error messages and explanations can be found at the end of this document.
<b>DID</b>	DID number
<b>IRID</b>	Internal Routing ID
<b>CustID</b>	ID of the customer sending post request
<b>PrimaryRoute</b>	Used for Load balancing. 0 is default in most cases, or 1 if priority routing is set up for that customer and server.

### Example

<p><b>Request</b></p> <pre>POST /api/didreport HTTP/1.1 Content-Type: application/json Content-Length: 124 {   "Key": "FzeKKeEKSIUXa7ZRMx3FciZwXG2FW8jd" }</pre>	<p><b>Response (SUCCESS)</b></p> <pre>{   "0": {     "DID": "7058053584",     "IRID": "182",     "CustID": "1",     "PrimaryRoute": 1   },   { Continues for every entry },   "Status": "OK" }</pre> <p><b>Response (FAIL)</b></p> <pre>{   "Status": "FAIL",   "Reason": "Error with Key" }</pre>
--	--

### Find Me Follow Me Details

Pass in a Find Me Follow Me ID to retrieve Name, Numbers, OverflowRouteID, OverflowRouteType, Recording, AutoAnswer, StepTime, and Extension linked to the ID.

**Add Header URL:** solswitch/api/findmefollowme/

The Request must contain the following attributes:

<b>Key</b>	API key for connection to the system. Required field if an API key has been enabled.
<b>ID</b>	FindMeFollowMe ID to retrieve data for

The Response will contain the following attributes:

<b>Status</b>	'OK' or 'FAIL'
<b>Reason</b>	Reason for fail status (Only if status is FAIL). A list of error messages and explanations can be found at the end of this document.

### Example

<p><b>Request</b></p> <pre>POST /api/findmefollowme/ HTTP/1.1 Content-Type: application/json {   "Key":   "YGLbZZaJWVMJfOXHCTIEADFGIRLbHZAd",   "ID": "5" }</pre>	<p><b>Response (SUCCESS)</b></p> <pre>{   "Status": "OK",   "ID": "5",   "Name": "dave-test",   "Numbers": [     "1112223333",     "2223334444",     "3334445555"   ],   "OverflowRouteID": "148",   "OverflowRouteType": "12",   "Recording": "0",   "AutoAnswer": "0",   "StepTime": "25",   "Extension": "9991" }</pre> <p><b>Response (FAIL)</b></p> <pre>{   "Status": "FAIL",   "Reason": "No Records Found." }</pre>
---	---

Firewall Management Suite

/api/firewall/view

This endpoint will list the available firewall rules on the system. There will also be an optional ID value that will display information about just the requested firewall rule.

The Request may contain the following attributes:

<b>Key</b>	API key for connection to the system. Required field if an API key has been enabled.
<b>ID</b>	The ID of the specific firewall rule to view (Overrides the ViewType option below to Full)
<b>ViewType</b>	Can be set to Full or Partial to control how much data will be returned: <b>Full</b> – Each listing will be provided with as much information as possible <b>Partial</b> – Each listing will be provided with and ID and Name <i>Default - Partial</i>

The Response will contain the following attributes:

<b>Status</b>	'OK' or 'FAIL'
<b>Results</b>	An array with each row containing the following:
<b>ID</b>	ID of the firewall rule. To be used with other endpoints
<b>Name</b>	Name of the firewall rule
<b>Position</b>	The order priority of the firewall rule. Lower numbered items will be checked before higher numbered ones
<b>RuleType</b>	This will be Allow or Deny
<b>AddressType</b>	This will be Address, Subnet or Country
<b>Address</b>	If AddressType is Address or Subnet this will contain the IP address or IP subnet of the rule
<b>Country</b>	If the AddressType is Country this will contain the country code of the selected country (Canada = CA, United Kingdom = UK, etc)
<b>Web</b>	This will be set to 1 if the rule's filtering is applied to web traffic
<b>SysAdmin</b>	This will be set to 1 if the rule's filtering is applied to web traffic on SysAdmin accounts
<b>Provisioning</b>	This will be set to 1 if the rule's filtering is applied to provisioning traffic

	<b>SIP</b>	This will be set to 1 if the rule's filtering is applied to SIP traffic
--	------------	---

*Example*

<p><b>Request</b></p> <pre>POST /api/firewall/view/ HTTP/1.1 Content-Type: application/json {   "Key":   "YGLbZZaJWVMJfOXHCTIEADFGIRLbHZAd",   "ID": "5" }</pre>	<p><b>Response (SUCCESS)</b></p> <pre>{   "Status": "OK",   "Results": [     {       "ID": "5",       "Name": "Test name"     }   ] }</pre> <p><b>Response (FAIL)</b></p> <pre>{   "Status": "FAIL",   "Reason": "No Records Found." }</pre>
--	--

/api/firewall/add/

This endpoint will add a new rule to the firewall in the given position, or at the end if no position is given.

The Request must contain the following attributes:

<b>Key</b>	API key for connection to the system. Required field if an API key has been enabled.
<b>Name</b>	Name of the firewall rule
<b>RuleType</b>	This will be Allow or Deny
<b>AddressType</b>	This will be Address, Subnet or Country

The Request may contain the following attributes:

<b>Position</b>	The order priority of the firewall rule. This will shift all positions down so entering 3 for example will increment the current rules of priority 3 or more by 1. If omitted this will default to the highest possible position
<b>Address</b>	If AddressType is Address or Subnet this must contain the IP address or IP subnet of the rule [xxx.xxx.xxx.xxx for address and xxx.xxx.xxx.xxx/yy for subnet]
<b>Country</b>	If the AddressType is Country this must contain the country code of the selected country [Canada = CA, United Kingdom = UK, etc]
<b>Web</b>	Boolean value [0 – Disabled on web traffic, 1 – Enabled on web traffic] <b>Default 1</b>
<b>SysAdmin</b>	Boolean value [0 – Disabled on SysAdmin web traffic, 1 – Enabled on SysAdmin web traffic] <b>Default 1</b>
<b>Provisioning</b>	Boolean value [0 – Disabled on provisioning traffic, 1 – Enabled on provisioning traffic] <b>Default 1</b>
<b>SIP</b>	Boolean value [0 – Disabled on SIP traffic, 1 – Enabled on SIP traffic] <b>Default 1</b>

The Response will contain the following attributes:

<b>Status</b>	'OK' or 'FAIL'
<b>ID</b>	The ID of the new firewall rule

*Example*

<b>Request</b> POST /api/firewall/add/ HTTP/1.1 Content-Type: application/json { "Key": "YGLbZZaJWVMJfOXHCTIEADFGIRLbHZAd", "Name": "Test Rule", "RuleType": "Allow", "AddressType": "Address" "Address": "127.0.0.1" }	<b>Response (SUCCESS)</b> { "Status": "OK", "ID": "4" } <b>Response (FAIL)</b> { "Status": "FAIL", "Reason": "No Records Found." }
---	--

/api/firewall/edit/

This endpoint will allow editing of existing firewall rules.

The Request must contain the following attributes:

<b>Key</b>	API key for connection to the system. Required field if an API key has been enabled.
<b>ID</b>	The ID of the rule to be edited

The Request may contain the following attributes:

<b>Name</b>	The new name of the firewall rule
<b>RuleType</b>	This will be Allow or Deny
<b>AddressType</b>	This will be Address, Subnet or Country
<b>Position</b>	The order priority of the firewall rule. This will shift all positions down so entering 3 for example will increment the current rules of priority 3 or more by 1. If omitted this will default to the highest possible position
<b>Address</b>	If AddressType is Address or Subnet this must contain the IP address or IP subnet of the rule [xxx.xxx.xxx.xxx for address and xxx.xxx.xxx.xxx/yy for subnet]
<b>Country</b>	If the AddressType is Country this must contain the country code of the selected country [Canada = CA, United Kingdom = UK, etc]
<b>Web</b>	Boolean value [0 – Disabled on web traffic, 1 – Enabled on web traffic] <b>Default 1</b>
<b>SysAdmin</b>	Boolean value [0 – Disabled on SysAdmin web traffic, 1 – Enabled on SysAdmin web traffic] <b>Default 1</b>
<b>Provisioning</b>	Boolean value [0 – Disabled on provisioning traffic, 1 – Enabled on provisioning traffic] <b>Default 1</b>
<b>SIP</b>	Boolean value [0 – Disabled on SIP traffic, 1 – Enabled on SIP traffic] <b>Default 1</b>

The Response will contain the following attributes:

<b>Status</b>	'OK' or 'FAIL'
---------------	----------------

*Example*

<b>Request</b> POST /api/firewall/edit/ HTTP/1.1 Content-Type: application/json { "Key": "YGLbZZaJWVMJfOXHCTIEADFGIRLbHZAd", "ID": "4", "Name": "Test Rule Renamed" }	<b>Response (SUCCESS)</b> { "Status": "OK" } <b>Response (FAIL)</b> { "Status": "FAIL", "Reason": "No Records Found." }
---	--



/api/firewall/delete/

This endpoint will allow deletion of existing firewall rules.

The Request must contain the following attributes:

<b>ID</b>	The ID of the rule to be deleted
-----------	----------------------------------

The Response will contain the following attributes:

<b>Status</b>	'OK' or 'FAIL'
---------------	----------------

*Example*

<b>Request</b> POST /api/firewall/delete/ HTTP/1.1 Content-Type: application/json { "Key": "YGLbZZaJWVMJfOXHCTIEADFGIRLbHZAd", "ID": "4" }	<b>Response (SUCCESS)</b> { "Status": "OK" } <b>Response (FAIL)</b> { "Status": "FAIL", "Reason": "No Records Found." }
---	--

### Get Customer Information

The Get Customer Information function will return all information related to a customer.

**Get Customer Information URL:** solswitch/api/getcustomerinformation/

The Request may contain the following attributes:

<b>Key</b>	API key for connection to the system. Required field if an API key has been enabled.
<b>CustomerNumber</b>	The CustomerNumber on the account. Not the CustID, but the customer number entered when creating the account. Usually links with an external CRM.

The Response will contain the following attributes:

<b>Status</b>	'OK' or 'FAIL'
<b>Reason</b>	Reason for fail status (Only if status is FAIL). A list of error messages and explanations can be found at the end of this document.
<b>ID</b>	The CustID on the SolSwitch. Use this to preform more API calls
<b>firstname</b>	First name of the customer
<b>lastname</b>	Last name of the customer
<b>company</b>	The company the customer represents.
<b>streetnumber</b>	Address street number
<b>streetname</b>	Address street name
<b>unit</b>	Address unit number
<b>postalcode</b>	Postal code (L#L#L#)
<b>city</b>	City the company is located in
<b>province</b>	Province code (ON)
<b>country</b>	Country of residence
<b>telephone</b>	Phone number without any '-'s

### Example

<p><b>Request</b></p> <pre>POST /api/getcustomerinformation HTTP/1.1 Content-Type: application/json Content-Length: 74 {   "Key": "FzeKKeEKSIUXa7ZRMx3FciZwXG2FW8jd",   "CustomerNumber": "123456789" }</pre>
---

**Response (SUCCESS)**

```
{
  "Status": "OK",
  "ID": "127",
  "firstname": "Test",
  "lastname": "Name",
  "company": "Sunwire",
  "streetnumber": "850",
  "streetname": "Barrydowne Road",
  "unit": "303",
  "postalcode": "P3A3T7",
  "city": "Sudbury",
  "province": "ON",
  "country": "Canada",
  "telephone": "7055551234"
}
```

**Response (FAIL)**

```
{
  "Status": "FAIL",
  "Reason": "No results returned"
}
```

## Monitoring Notifications

When monitoring is enabled on an extension, through Toggle Monitor, event notifications will be POSTed to the notification URL (Configured in SolSwitch settings).

Notifications will be sent on incoming and outgoing calls to monitored extensions, for the following events:

- Call Incoming
- Call Connected
- Call Disconnected

The notifications will contain the following attributes:

<b>Event</b>	The event that triggered the notification (Incoming, Connected, Disconnected).
<b>CallID</b>	The unique ID of the call
<b>Customer</b>	ID of the customer for the notifying extension.
<b>SourceNumber</b>	The phone number or extension of the source of the call
<b>SourceChannel</b>	The call channel connected to the source of the call
<b>DestinationNumber</b>	The extension of the destination of the call
<b>DestinationChannel</b>	The call channel connected to the destination of the call
<b>DialedNumber</b>	The actual number dialed by the source of the call

## Example

### Request

```
POST /example HTTP/1.1
Content-Type: application/json
Content-Length: 200
{
  "Event": "Incoming",
  "CallID": "843597",
  "Customer": "4",
  "SourceNumber": "7055551234"
  "SourceChannel": "SIP/Example-000003ee",
  "DestinationNumber": "123",
  "DestinationChannel": "SIP/Example-0004f2be",
  "DialedNumber": "7055216777"
}
```

### Remove DID

Delete a DID from the SolSwitch.

**Transfer Call URL:** solswitch/api/removedid/

The Request must contain the following attributes:

<b>Key</b>	API key for connection to the system. Required field if an API key has been enabled.
<b>DID</b>	The DID to remove

The Response will contain the following attributes:

<b>Status</b>	'OK' or 'FAIL'
<b>Reason</b>	Reason for fail status (Only if status is FAIL). A list of error messages and explanations can be found at the end of this document.

### Example

<p><b>Request</b></p> <pre>POST /api/removedid HTTP/1.1 Content-Type: application/json Content-Length: 124 {   "Key": "FzeKKeEKSIUXa7ZRMx3FciZwXG2FW8jd",   "DID": "7051234567" }</pre>	<p><b>Response (SUCCESS)</b></p> <pre>{   "Status": "OK" }</pre> <p><b>Response (FAIL)</b></p> <pre>{   "Status": "FAIL",   "Reason": "Invalid DID provided" }</pre>
---	--

### Scheduled Routing Details

Pass a Route ID to return Route Type, InternalRouteID, Extension, Name, and Default Route at present moment in time if between DateTo and DateFrom, TimeTo and TimeFrom in ScheduledRoutingOptions table.

**Add Header URL:** solswitch/api/scheduledrouting/

The Request must contain the following attributes:

<b>Key</b>	API key for connection to the system. Required field if an API key has been enabled.
<b>ID</b>	The ID of the scheduled route you want data for if there's a scheduled route for the current time. If there isn't – the default route is returned.

The Response will contain the following attributes:

<b>Status</b>	'OK' or 'Fail'
<b>Reason</b>	If the ID is non numerical or there's no records found for the ID.

### Example

<p><b>Request</b></p> <pre>POST /api/scheduledrouting/ HTTP/1.1 Content-Type: application/json {   "Key":   "YGLbZZaJWVMJfOXHCTIEADFGIRLbHZAd",   "ID": "2" }</pre>	<p><b>Response (SUCCESS)</b></p> <pre>{   "Status": "OK",   "Name": "Test 2",   "Extension": "101",   "RouteExt": "9696",   "RouteType": "7",   "IRID": "68",   "RouteName": "dave-ext" }</pre> <p><b>Response (FAIL)</b></p> <pre>{   "Status": "FAIL",   "Reason": "No data found." }</pre>
---	---

## SIP Account Management Suite

/api/sipaccounts/view/

This endpoint will list the available SIP accounts on the system. There will also be an optional ID value that will display information about just the requested SIP account.

The Request may contain the following attributes:

<b>Key</b>	API key for connection to the system. Required field if an API key has been enabled.
<b>ID</b>	The ID of the specific SIP account to view (Overrides the ViewType option below to Full)
<b>ViewType</b>	Can be set to Full or Partial to control how much data will be returned: <b>Full</b> – Each listing will be provided with as much information as possible <b>Partial</b> – Each listing will be provided with and ID, Username and Description <b>Default - Partial</b>

The Response will contain the following attributes:

<b>Status</b>	'OK' or 'FAIL'
<b>Results</b>	An array with each row containing the following:
<b>ID</b>	ID of the SIP account. To be used with other endpoints
<b>CustID</b>	The Customer ID for the SIP account
<b>Description</b>	The given description of the SIP account
<b>Username</b>	SIP account username for static and dynamic SIP accounts
<b>Host</b>	SIP account host for static SIP accounts
<b>NAT</b>	Boolean [1 – Yes, 0 – No]
<b>Type</b>	[static, dynamic, register]
<b>DTMFMode</b>	[rfc2833, inband]
<b>Port</b>	SIP port, typically 5060
<b>DeleteBlock</b>	Boolean [1 – Yes, 0 – No]
<b>NotifyEmail</b>	Email address
<b>FaxCodec</b>	Boolean [1 – Yes, 0 – No]
<b>VoiceCodec</b>	Boolean [1 – Yes, 0 – No]
<b>VoiceCodec1</b>	Boolean [1 – Yes, 0 – No]
<b>VoiceCodec2</b>	Boolean [1 – Yes, 0 – No]
<b>VoiceCodec3</b>	Boolean [1 – Yes, 0 – No]

<b>VoiceCodec4</b>	Boolean [1 – Yes, 0 – No]
<b>Debug</b>	Boolean [1 – Yes, 0 – No]
<b>RegisterUsername</b>	Username for Register type
<b>RegisterName</b>	Name / BTN for register type
<b>RegisterDomain</b>	Domain for register type
<b>TransportType</b>	TCP, UDP, SRTP
<b>G722Codec</b>	Boolean [1 – Yes, 0 – No]
<b>G729Codec</b>	Boolean [1 – Yes, 0 – No]
<b>DynamicFeatures</b>	Boolean [1 – Yes, 0 – No]
<b>SIPPostAPI</b>	Boolean [1 – Yes, 0 – No]
<b>ACLAllow</b>	String containing the ACL allow list
<b>ACLDeny</b>	String containing the ACL Deny list
<b>AllowInternational</b>	Boolean [1 – Yes, 0 – No]
<b>SIPNotifyAPI</b>	Boolean [1 – Yes, 0 – No]
<b>StayInAudioPath</b>	Boolean [1 – Yes, 0 – No]
<b>Active</b>	Boolean [1 – Yes, 0 – No]

*Example*

<p><b>Request</b>          POST /api/sipaccounts/view/ HTTP/1.1          Content-Type: application/json  <pre>{   "Key":   "YGLbZZaJWVMJfOXHCTIEADFGIRLbHZAd",   "ID": "5" }</pre></p>	<p><b>Response (SUCCESS)</b>  <pre>{   "Status": "OK",   "Results": [     {       "ID": "5",       "Name": "Test name",       "Description": "Test Desc"     }   ] }</pre></p> <p><b>Response (FAIL)</b>  <pre>{   "Status": "FAIL",   "Reason": "No Records Found." }</pre></p>
--	--



/api/sipaccounts/add/

This endpoint will add a new SIP account

The Request must contain the following attributes:

<b>Key</b>	API key for connection to the system. Required field if an API key has been enabled.
<b>CustID</b>	The Customer ID for the SIP account
<b>Description</b>	The given description of the SIP account
<b>Username</b>	SIP account username for static and dynamic SIP accounts
<b>Type</b>	[static, dynamic, register]

If Type is set to static the following attributes are required:

<b>Password</b>	Password for static and dynamic SIP accounts
<b>Host</b>	SIP account host for static SIP accounts

If Type is set to dynamic the following attributes are required:

<b>Password</b>	Password for static and dynamic SIP accounts
-----------------	--

If Type is set to register the following attributes are required:

<b>RegisterUsername</b>	Username for Register type
<b>RegisterPassword</b>	Password for Register type
<b>RegisterName</b>	Name / BTN for register type
<b>RegisterDomain</b>	Domain for register type

The Request may contain the following attributes:

<b>NAT</b>	Boolean [1 – Yes, 0 – No] <b>Default 1</b>
<b>DTMFMode</b>	[rfc2833, inband] <b>Default rfc2833</b>
<b>Port</b>	SIP port, typically 5060 <b>Default 5060</b>
<b>DeleteBlock</b>	Boolean [1 – Yes, 0 – No] <b>Default 0</b>
<b>NotifyEmail</b>	Email address
<b>FaxCodec</b>	Boolean [1 – Yes, 0 – No]

	<b>Default 0</b>
<b>VoiceCodec</b>	Boolean [1 – Yes, 0 – No] <b>Default 0</b>
<b>VoiceCodec1</b>	Boolean [1 – Yes, 0 – No] <b>Default 0</b>
<b>VoiceCodec2</b>	Boolean [1 – Yes, 0 – No] <b>Default 0</b>
<b>VoiceCodec3</b>	Boolean [1 – Yes, 0 – No] <b>Default 0</b>
<b>VoiceCodec4</b>	Boolean [1 – Yes, 0 – No] <b>Default 0</b>
<b>Debug</b>	Boolean [1 – Yes, 0 – No] <b>Default 0</b>
<b>TransportType</b>	TCP, UDP, SRTP <b>Default UDP</b>
<b>G722Codec</b>	Boolean [1 – Yes, 0 – No] <b>Default 0</b>
<b>G729Codec</b>	Boolean [1 – Yes, 0 – No] <b>Default 0</b>
<b>DynamicFeatures</b>	Boolean [1 – Yes, 0 – No] <b>Default 1</b>
<b>SIPPostAPI</b>	Boolean [1 – Yes, 0 – No] <b>Default 0</b>
<b>ACLAllow</b>	String containing the ACL allow list
<b>ACLDeny</b>	String containing the ACL Deny list
<b>AllowInternational</b>	Boolean [1 – Yes, 0 – No] <b>Default 0</b>
<b>SIPNotifyAPI</b>	Boolean [1 – Yes, 0 – No] <b>Default 0</b>
<b>StayInAudioPath</b>	Boolean [1 – Yes, 0 – No] <b>Default 1</b>
<b>Active</b>	Boolean [1 – Yes, 0 – No] <b>Default 1</b>

The Response will contain the following attributes:

<b>Status</b>	'OK' or 'FAIL'
<b>ID</b>	The ID of the newly created SIP Account

*Example*

<b>Request</b> POST /api/sipaccounts/add/ HTTP/1.1 Content-Type: application/json { "Key": "YGLbZZaJWVMJfOXHCTIEADFGIRLbHZAd", "Description": "Test SIP", "CustID": "1", "Username": "TestSIPAcc", "Type": "dynamic" }	<b>Response (SUCCESS)</b> { "Status": "OK", "ID": "4" } <b>Response (FAIL)</b> { "Status": "FAIL", "Reason": "No Records Found." }
--	--

/api/sipaccounts/edit/

This endpoint will edit an existing SIP account

The Request must contain the following attributes:

<b>Key</b>	API key for connection to the system. Required field if an API key has been enabled.
<b>ID</b>	The ID of the SIP account to be edited

The Request may contain the following attributes:

<b>CustID</b>	The Customer ID for the SIP account
<b>Description</b>	The given description of the SIP account
<b>Username</b>	SIP account username for static and dynamic SIP accounts
<b>Password</b>	Password for static and dynamic SIP accounts
<b>Host</b>	SIP account host for static SIP accounts
<b>NAT</b>	Boolean [1 – Yes, 0 – No]
<b>Type</b>	[static, dynamic, register]
<b>DTMFMode</b>	[rfc2833, inband]
<b>Port</b>	SIP port, typically 5060
<b>DeleteBlock</b>	Boolean [1 – Yes, 0 – No]
<b>NotifyEmail</b>	Email address
<b>FaxCodec</b>	Boolean [1 – Yes, 0 – No]
<b>VoiceCodec</b>	Boolean [1 – Yes, 0 – No]
<b>VoiceCodec1</b>	Boolean [1 – Yes, 0 – No]
<b>VoiceCodec2</b>	Boolean [1 – Yes, 0 – No]
<b>VoiceCodec3</b>	Boolean [1 – Yes, 0 – No]
<b>VoiceCodec4</b>	Boolean [1 – Yes, 0 – No]
<b>Debug</b>	Boolean [1 – Yes, 0 – No]
<b>RegisterUsername</b>	Username for Register type
<b>RegisterPassword</b>	Password for Register type
<b>RegisterName</b>	Name / BTN for register type
<b>RegisterDomain</b>	Domain for register type
<b>TransportType</b>	TCP, UDP, SRTP
<b>G722Codec</b>	Boolean [1 – Yes, 0 – No]
<b>G729Codec</b>	Boolean [1 – Yes, 0 – No]
<b>DynamicFeatures</b>	Boolean [1 – Yes, 0 – No]

<b>SIPPostAPI</b>	Boolean [1 – Yes, 0 – No]
<b>ACLAllow</b>	String containing the ACL allow list
<b>ACLDeny</b>	String containing the ACL Deny list
<b>AllowInternational</b>	Boolean [1 – Yes, 0 – No]
<b>SIPNotifyAPI</b>	Boolean [1 – Yes, 0 – No]
<b>StayInAudioPath</b>	Boolean [1 – Yes, 0 – No]
<b>Active</b>	Boolean [1 – Yes, 0 – No]

The Response will contain the following attributes:

<b>Status</b>	'OK' or 'FAIL'
---------------	----------------

*Example*

<p><b>Request</b></p> <pre>POST /api/sipaccounts/edit/ HTTP/1.1 Content-Type: application/json {   "Key":   "YGLbZZaJWVMJfOXHCTIEADFGIRLbHZAd",   "ID": "4",   "Description": "Test Trunk Renamed" }</pre>	<p><b>Response (SUCCESS)</b></p> <pre>{   "Status": "OK" }</pre> <p><b>Response (FAIL)</b></p> <pre>{   "Status": "FAIL",   "Reason": "No Records Found." }</pre>
--	---

/api/sipaccounts/delete/

This endpoint will allow deletion of existing SIP accounts.

The Request must contain the following attributes:

<b>Key</b>	API key for connection to the system. Required field if an API key has been enabled.
<b>ID</b>	The ID of the trunk to be deleted

The Response will contain the following attributes:

<b>Status</b>	'OK' or 'FAIL'
---------------	----------------

*Example*

<b>Request</b> POST /api/sipaccounts/delete/ HTTP/1.1 Content-Type: application/json { "Key": "YGLbZZaJWVMJfOXHCTIEADFGIRLbHZAd", "ID": "4" }	<b>Response (SUCCESS)</b> { "Status": "OK" } <b>Response (FAIL)</b> { "Status": "FAIL", "Reason": "No Records Found." }
--	--

### Toggle Monitoring

The toggle monitoring function allows event monitoring to be turned on or off for extensions. Monitoring can also be enabled/disabled for extensions directly through the SolSwitch web interface.

When enabled, notifications will be sent to the notification URL (Configured in SolSwitch settings) when the extension receives an incoming or outgoing call. The SolSwitch portal allows for greater customization of what events are sent to the notification URL

**Toggle Monitoring URL:** solswitch/api/monitor/

The Request must contain the following attributes:

<b>Extension</b>	The extension number to adjust monitoring for. Extensions are numeric values, 3 to 5 digits in length.
<b>Enabled</b>	Whether monitoring should be enabled or disabled. Accepted values are 1 and 0, where 1 = Enabled and 0 = Disabled.
<b>Customer</b>	ID of the customer the function should run on, should be a numeric value. Optional field, if not provided the system will default to CustID 1.
<b>Key</b>	API key for connection to the system. Required field if an API key has been enabled.

The Response will contain the following attributes:

<b>Status</b>	'OK' or 'FAIL'
<b>Reason</b>	Reason for fail status (Only if status is FAIL). A list of error messages and explanations can be found at the end of this document.

### Example

<b>Request</b> POST /api/monitor HTTP/1.1 Content-Type: application/json Content-Length: 107 <pre>{   "Extension": "123",   "Enabled": "1",   "Customer": "4",   "Key": "FzeKKeEKSIUXa7ZRMx3FciZwXG2FW8jd" }</pre>	<b>Response (SUCCESS)</b> <pre>{   "Status": "OK" }</pre> <b>Response (FAIL)</b> <pre>{   "Status": "FAIL",   "Reason": "Unknown   Extension Number" }</pre>
--	--



### Toll Billing Info

This is used to sync two SolSwitch's toll billing rates and related toll settings. This is generally used when deploying multiple independent sets of SolSwitch systems which will be using the same billing settings.

Only a SolSwitch can use this API call as it involves accessing the database tables and comparing entries. For more information see the **Advanced Tools > Toll Management > Toll Sync** on the SolSwitch web portal.



### Toll Rates

List the current toll rate locations for a certain toll plan.

**Toll Rates URL:** solswitch/api/tollrates/

The Request must contain the following attributes:

<b>Key</b>	API key for connection to the system. Required field if an API key has been enabled.
<b>tollPlanID</b>	The ID of the toll rate list to search on.

The Request may contain the following attribute:

<b>tollPrefix</b>	Including this will only return rates that the given prefix matches
-------------------	---

The Response will contain the following attributes:

<b>Status</b>	'OK' or 'FAIL'
<b>Array</b>	Data from the toll rates
<b>Reason</b>	Reason for fail status (Only if status is FAIL). A list of error messages and explanations can be found at the end of this document.

### Example

<p><b>Request</b></p> <pre>POST /api/tollrates HTTP/1.1 Content-Type: application/json Content-Length: 124 {   "Key": "FzeKKeEKSIUXa7ZRMx3FciZwXG2FW8jd",   "tollPlanID": "11" }</pre>	<p><b>Response (SUCCESS)</b></p> <pre>{   "Status": "OK",   {     "Prefix" : "1705",     "Area" : "Sudbury, ON",     "Value" : "0.03"   },   { Continues for every entry } }</pre> <p><b>Response (FAIL)</b></p> <pre>{   "Status": "FAIL",   "Reason": "IP address not allowed" }</pre>
--	--

### Toll Reporting

Load a report of tolls for the specified time period – either as a per-customer summary, or as a per-call breakdown for a single customer.

If API permissions are set to allow access to a single customer, the report will always pull the single customer breakdown for that customer.

**Toll Reporting URL:** solswitch/api/tollreport/

The Request must contain the following attributes:

<b>Key</b>	API key for connection to the system. Required field if an API key has been enabled.
<b>Customer</b>	ID of the customer the function should run on, should be a numeric value. Optional field, if not provided the system will default to CustID 1.
<b>DateStart</b>	This will limit the returned data to be within <b>Date Start</b> and <b>Date End</b> . Format: YYYY-MM-DD OR Format: YYYY-MM-DD HH:MM:SS
<b>DateEnd</b>	This will limit the returned data to be within <b>Date Start</b> and <b>Date End</b> . Format: YYYY-MM-DD OR Format: YYYY-MM-DD HH:MM:SS

The Request may contain the following attribute:

<b>CustNum</b>	Customer Number (as configured in customer settings).  When left empty, report will run as a per-customer summary if access is granted to all customers. When provided, report will be per-call breakdown for the specified customer.
----------------	---

The Response will contain the following attributes:

<b>Status</b>	'OK' or 'FAIL'
<b>Array</b>	Data from the toll report
<b>Reason</b>	Reason for fail status (Only if status is FAIL). A list of error messages and explanations can be found at the end of this document.

## Example

<p><b>Request</b></p> <pre>POST /api/tollreport HTTP/1.1 Content-Type: application/json Content-Length: 162 {   "Key": "1234",   "DateStart": "2024-07-01",   "DateEnd": "2024-08-01" }</pre>	<p><b>Response (SUCCESS)</b></p> <pre>{   "Status": "OK",   "0": {     "CustNum": "5678",     "Company": "Main",     "Area": "Canada - AB",     "RateCost": "0.0300",     "CallCost": "0.32"   },   "1": {     "CustNum": "5678",     "Company": "Main",     "Area": "Canada - BC",     "RateCost": "0.0300",     "CallCost": "0.08"   } }</pre> <p><b>Response (FAIL)</b></p> <pre>{   "Status": "FAIL",   "Reason": "IP address not allowed" }</pre>
<pre>POST /api/tollreport HTTP/1.1 Content-Type: application/json Content-Length: 179 {   "Key": "1234",   "DateStart": "2024-07-01",   "DateEnd": "2024-08-01",   "CustNum": "5678" }</pre>	<p><b>Response (SUCCESS)</b></p> <pre>{   "Status": "OK",   "0": {     "Company": "Main",     "Area": "Toll Free",     "RateCost": "0.0500",     "CallCost": "0.06",     "CallID": "3753672",     "StartTime": "2024-07-16 10:43:35",     "Length": "68",     "SrcNumber": "4167769547",     "DstNumber": "18004614219"   },   "1": {     "Company": "Main",     "Area": "Toll Free",     "RateCost": "0.0500",     "CallCost": "0.03",     "CallID": "3754222",     "StartTime": "2024-07-16 10:45:07",     "Length": "30",     "SrcNumber": "4167769547",     "DstNumber": "18004614219"   } }</pre>

	<pre>} <b>Response (FAIL)</b> {   "Status": "FAIL",   "Reason": "IP address not allowed" }</pre>
--	--

## Trunk Management Suite

/api/trunks/view/

This endpoint will list the available trunks on the system. There will also be an optional ID value that will display information about just the requested trunk.

The Request may contain the following attributes:

<b>Key</b>	API key for connection to the system. Required field if an API key has been enabled.
<b>ID</b>	The ID of the specific trunk to view (Overrides the ViewType option below to Full)
<b>ViewType</b>	Can be set to Full or Partial to control how much data will be returned: <b>Full</b> – Each listing will be provided with as much information as possible <b>Partial</b> – Each listing will be provided with and ID, Name and Description <i>Default - Partial</i>

The Response will contain the following attributes:

<b>Status</b>	'OK' or 'FAIL'
<b>Results</b>	An array with each row containing the following:
<b>ID</b>	ID of the trunk. To be used with other endpoints
<b>Name</b>	Name of the firewall rule
<b>Description</b>	The order priority of the firewall rule. Lower numbered items will be checked before higher numbered ones
<b>TrunkType</b>	1 – SIP Account, 2 – Line Card
<b>SIPAccountsID</b>	ID of the SIP Account tied to this trunk
<b>ChannelGroup</b>	0 - 9 Line Card Channel Group
<b>ChannelOrder</b>	Outbound call order for Line Card trunk g – Sequential Ascending G – Sequential Descending r – Round Robin Ascending R – Round Robin Descending
<b>UseDIDTable</b>	Boolean [1 – Yes, 0 – No]
<b>UseRouteGroup</b>	Boolean [1 – Yes, 0 – No]
<b>RouteGroupID</b>	The ID of the route group to be assigned
<b>AddPrefix</b>	Prefix to be added to calls
<b>StripDigit</b>	Number of digits to strip from DIDs
<b>CallLimit</b>	0 – Unlimited, # - Max calls on trunk

	<b>DefaultDID</b>	
	<b>UseLocalRouting</b>	Boolean [1 – Yes, 0 – No]
	<b>AccessibleSID</b>	Boolean [1 – Yes, 0 – No]
	<b>Record</b>	Boolean [1 – Yes, 0 – No]
	<b>ApplyToll</b>	Boolean [1 – Yes, 0 – No]
	<b>UseE164</b>	Boolean [1 – Yes, 0 – No]
	<b>CanConf</b>	Boolean [1 – Yes, 0 – No]
	<b>UnlimitedOutbound</b>	Boolean [1 – Yes, 0 – No]
	<b>AutoLearnRoutes</b>	Boolean [1 – Yes, 0 – No]
	<b>AutoLearnAPIKey</b>	Blank string or API key
	<b>AutoLearnWeight</b>	Boolean [1 – Yes, 0 – No]
	<b>AutoLearnValidated</b>	Boolean [1 – Yes, 0 – No]

*Example*

<p><b>Request</b></p> <pre>POST /api/trunks/view/ HTTP/1.1 Content-Type: application/json {   "Key":   "YGLbZZaJWVMJfOXHCTIEADFGIRLbHZAd",   "ID": "5" }</pre>	<p><b>Response (SUCCESS)</b></p> <pre>{   "Status": "OK",   "Results": [     {       "ID": "5",       "Name": "Test name",       "Description": "Test Desc"     }   ] }</pre> <p><b>Response (FAIL)</b></p> <pre>{   "Status": "FAIL",   "Reason": "No Records Found." }</pre>
--	--

/api/trunks/add/

This endpoint will add a new trunk

The Request must contain the following attributes:

<b>Key</b>	API key for connection to the system. Required field if an API key has been enabled.
<b>Name</b>	Name of the firewall rule
<b>Description</b>	The order priority of the firewall rule. Lower numbered items will be checked before higher numbered ones
<b>TrunkType</b>	1 – SIP Account, 2 – Line Card
<b>SIPAccountsID</b>	ID of the SIP Account tied to this trunk

The Request may contain the following attributes:

<b>ChannelGroup</b>	0 - 9 Line Card Channel Group
<b>ChannelOrder</b>	Outbound call order for Line Card trunk g – Sequential Ascending G – Sequential Descending r – Round Robin Ascending R – Round Robin Descending A blank string can be provided if not using the LineCard trunk type
<b>UseDIDTable</b>	Boolean [1 – Yes, 0 – No] <b>Default 0</b>
<b>UseRouteGroup</b>	Boolean [1 – Yes, 0 – No] <b>Default 0</b>
<b>RouteGroupID</b>	The ID of the route group to be assigned
<b>AddPrefix</b>	Blank string or prefix to be added to calls
<b>StripDigit</b>	Number of digits to strip from DIDs
<b>CallLimit</b>	0 – Unlimited, # - Max calls on trunk <b>Default 0</b>
<b>UseLocalRouting</b>	Boolean [1 – Yes, 0 – No] <b>Default 0</b>
<b>AccessibleSID</b>	Boolean [1 – Yes, 0 – No] <b>Default 0</b>
<b>Record</b>	Boolean [1 – Yes, 0 – No] <b>Default 0</b>
<b>ApplyToll</b>	Boolean [1 – Yes, 0 – No]

	<b>Default 0</b>
<b>UseE164</b>	Boolean [1 – Yes, 0 – No] <b>Default 0</b>
<b>CanConf</b>	Boolean [1 – Yes, 0 – No] <b>Default 0</b>
<b>UnlimitedOutbound</b>	Boolean [1 – Yes, 0 – No] <b>Default 0</b>
<b>AutoLearnRoutes</b>	Boolean [1 – Yes, 0 – No] <b>Default 0</b>
<b>AutoLearnAPIKey</b>	Blank string or API key <b>Default 0</b>
<b>AutoLearnWeight</b>	Integer value for the weight of this trunk for auto learning <b>Default 23</b>
<b>AutoLearnValidated</b>	Boolean [1 – Yes, 0 – No] <b>Default 0</b>

The Response will contain the following attributes:

<b>Status</b>	'OK' or 'FAIL'
<b>ID</b>	The ID of the newly created trunk

*Example*

<b>Request</b> POST /api/trunks/add/ HTTP/1.1 Content-Type: application/json <pre>{   "Key":   "YGLbZZaJWVMJfOXHCTIEADFGIRLbHZAd",   "Name": "Main Trunk",   "Description": "Main outbound trunk",   "SIPAccountsID": "4",   "TrunkType": "1" }</pre>	<b>Response (SUCCESS)</b> <pre>{   "Status": "OK",   "ID": "7" }</pre> <b>Response (FAIL)</b> <pre>{   "Status": "FAIL",   "Reason": "No Records Found." }</pre>
--	--



/api/trunks/edit/

This endpoint will edit an existing trunk

The Request must contain the following attributes:

<b>Key</b>	API key for connection to the system. Required field if an API key has been enabled.
<b>ID</b>	The ID of the trunk to be edited

The Request may contain the following attributes:

<b>Name</b>	Name of the firewall rule
<b>Description</b>	The order priority of the firewall rule. Lower numbered items will be checked before higher numbered ones
<b>TrunkType</b>	1 – SIP Account, 2 – Line Card
<b>SIPAccountsID</b>	ID of the SIP Account tied to this trunk
<b>ChannelGroup</b>	0 - 9 Line Card Channel Group
<b>ChannelOrder</b>	Outbound call order for Line Card trunk g – Sequential Ascending G – Sequential Descending r – Round Robin Ascending R – Round Robin Descending A blank string can be provided if not using the LineCard trunk type
<b>UseDIDTable</b>	Boolean [1 – Yes, 0 – No]
<b>UseRouteGroup</b>	Boolean [1 – Yes, 0 – No]
<b>RouteGroupID</b>	The ID of the route group to be assigned
<b>AddPrefix</b>	Blank string or prefix to be added to calls
<b>StripDigit</b>	Number of digits to strip from DIDs
<b>CallLimit</b>	0 – Unlimited, # - Max calls on trunk
<b>UseLocalRouting</b>	Boolean [1 – Yes, 0 – No]
<b>AccessibleSID</b>	Boolean [1 – Yes, 0 – No]
<b>Record</b>	Boolean [1 – Yes, 0 – No]
<b>ApplyToll</b>	Boolean [1 – Yes, 0 – No]
<b>UseE164</b>	Boolean [1 – Yes, 0 – No]
<b>CanConf</b>	Boolean [1 – Yes, 0 – No]
<b>UnlimitedOutbound</b>	Boolean [1 – Yes, 0 – No]
<b>AutoLearnRoutes</b>	Boolean [1 – Yes, 0 – No]

<b>AutoLearnAPIKey</b>	Blank string or API key
<b>AutoLearnWeight</b>	Boolean [1 – Yes, 0 – No]
<b>AutoLearnValidated</b>	Boolean [1 – Yes, 0 – No]

The Response will contain the following attributes:

<b>Status</b>	'OK' or 'FAIL'
---------------	----------------

*Example*

<p><b>Request</b></p> <pre>POST /api/trunks/edit/ HTTP/1.1 Content-Type: application/json {   "Key":   "YGLbZZaJWVMJfOXHCTIEADFGIRLbHZAd",   "ID": "4",   "Name": "Test Trunk Renamed" }</pre>	<p><b>Response (SUCCESS)</b></p> <pre>{   "Status": "OK" }</pre> <p><b>Response (FAIL)</b></p> <pre>{   "Status": "FAIL",   "Reason": "No Records Found." }</pre>
--	---

/api/trunks/delete/

This endpoint will allow deletion of existing trunks.

The Request must contain the following attributes:

<b>Key</b>	API key for connection to the system. Required field if an API key has been enabled.
<b>ID</b>	The ID of the trunk to be deleted

The Response will contain the following attributes:

<b>Status</b>	'OK' or 'FAIL'
---------------	----------------

*Example*

<p><b>Request</b></p> <pre>POST /api/trunks/delete/ HTTP/1.1 Content-Type: application/json {   "Key":   "YGLbZZaJWVMJfOXHCTIEADFGIRLbHZAd",   "ID": "4" }</pre>	<p><b>Response (SUCCESS)</b></p> <pre>{   "Status": "OK" }</pre> <p><b>Response (FAIL)</b></p> <pre>{   "Status": "FAIL",   "Reason": "No Records Found." }</pre>
--	---

## Error Messages

Below are error messages that can be received from the API, and their reasons.

### API

<b>Invalid API function</b>	The function being called does not exist
-----------------------------	--

### CallID

<b>Invalid CallID</b>	The call ID provided is not valid (must be a positive integer value).
<b>Unknown CallID</b>	The provided call ID does not match any calls on record

### Call

<b>Call did not start</b>	New call unable to start or unable to retrieve call information.
---------------------------	--

### Channel

<b>Invalid Channel</b>	The channel field isn't set
<b>Unknown Channel</b>	The provided channel does not match any active calls

### Customer

<b>Invalid Customer ID</b>	The customer ID provided is not valid (must be a positive integer value).
<b>Unknown Customer ID</b>	No customer exists with the given ID.
<b>No access to customer number #</b>	Your IP/API Key does not have access rights to the given customer ID.

### Date

<b>No matching end time</b>	Start time provided but no end time
<b>Invalid date format</b>	Provided format does not match the required

### Destination

<b>Invalid Destination</b>	The destination number isn't set or isn't numeric
----------------------------	---

## DID

<b>Invalid DID Provided</b>	The DID is not a numeric entry. Must not contain "(, ")" or "-"
-----------------------------	---

## Extension

<b>Invalid Extension Number</b>	The extension number isn't set or isn't numeric.
<b>Unknown Extension Number</b>	The provided extension number does not exist on the specified customer.

## Enabled

<b>Invalid Enabled Value</b>	Enabled value not set or is an invalid character. Accepted inputs are 1 or 0.
------------------------------	---

## Fax

<b>Cannot send fax, no destination number</b>	The To variable was not set, or was not set to a valid number
<b>Cannot send fax, no source number</b>	The From variable was not set, or was not set to a valid number
<b>From phone number provided is not a valid fax to email account.</b>	The From variable does not point to a valid FaxToEmail account

## File

<b>You must upload a file to fax</b>	No File variable was passed to the function
<b>Error converting file for faxing. If this problem persists, please contact the system administrator.</b>	Error in the converting process of the .pdf. Make sure no non-standard encoding is being used when rendering the .pdf.
<b>Only PDFs are accepted, type Detected: &lt;FileType&gt;</b>	The File variable was not a .pdf file.

## IP

<b>IP does not have access</b>	The IP being used to connect does not have permission to connect to the API function.
--------------------------------	---

## Key

<b>Error with Key</b>	The API key provided is invalid or does not have access to the specified API function.
-----------------------	--

## Recording

<b>Call not recorded</b>	The requested call was not recorded. This could be due to recording not being set up for the involved parties, or one of the involved parties having privacy enabled.
<b>Recording not generated</b>	The requested call was recorded, but the system is still working on generating the recording. This will generally be completed within a couple minutes of the call ending.

## Time Rules

<b>Time rule not found</b>	There is no time rule with the supplied ID on the requested customer.
----------------------------	---

## Tone

<b>Invalid Tone</b>	Tone not set or contains invalid characters. Accepted characters are: *, #, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9
---------------------	--

## Unknown

<b>Missing required field &lt;Field&gt;</b>	A required parameter was not passed to the API. Check documentation on the function you're trying to use.
<b>No results to display</b>	No data was returned from a function that should always return data (Eg /api/getcustomerinformation)

## Route Types

Below is a table describing the route types for SourceType and DestinationType values.

### ID, RouteType

<b>1</b>	Queue
<b>2</b>	FaxToEmail
<b>3</b>	Voicemail
<b>4</b>	VirtualExtension
<b>5</b>	RingGroup
<b>6</b>	IVR
<b>7</b>	Extension
<b>8</b>	Recording
<b>9</b>	MeetMe
<b>10</b>	PageGroup
<b>11</b>	RouteGroup
<b>12</b>	FollowMe
<b>13</b>	ParkingLot
<b>14</b>	VoIP Line
<b>15</b>	Trunk
<b>16</b>	QueueCallback
<b>17</b>	TimeRule
<b>18</b>	Sol-Cast
<b>19</b>	NotifyQueue
<b>20</b>	ScheduledRouting
<b>21</b>	FaxSwitch
<b>22</b>	DV
<b>23</b>	Hold

## Example API Calls

Below are some example calls of how to use and access SolSwitch API.

### PHP CDR Export Example Call

## Request ##	## Response ##
<pre> &lt;? \$array = array(     "Key" =&gt; "YOUR_KEY_HERE",     "DateStart" =&gt; "2021-02-16",     "DateEnd" =&gt; "2021-02-16");  \$json = json_encode(\$array, 128);  \$url = "http://SS_IP_HERE/api/cdrexpert/";  \$curl = curl_init(); curl_setopt(\$curl, CURLOPT_POST, 1); curl_setopt(\$curl, CURLOPT_URL, \$url); curl_setopt(\$curl, CURLOPT_RETURNTRANSFER, TRUE); curl_setopt(\$curl, CURLOPT_HTTPHEADER, Array('Content-Type: application/json')); curl_setopt(\$curl, CURLOPT_POSTFIELDS, \$json); \$result = curl_exec(\$curl); curl_close(\$curl);  print \$result; </pre>	<pre> {     "LegID":"6166",     "StartTime":"2021-08-04 10:43:18",     "ConnectTime":"",     "EndTime":"2021-08-04-10:43:20",     "SourceName":"test-ext2",     "SourceNumber":"9697",     "DestinationName":"test-mitel-ext",     "DestinationNumber":"9696",     "Disposition":"NOANSWER",     "DialedNumber":"9696",     "SourceType":"7",     "DestinationType":"7",     "CallCode":"" } </pre>



## CommandLine cURL List Active Calls Example Call

**JSON file contents:**

```
{
  "Customer": "1",
  "Key": "YOUR_KEY_HERE"
}
```

**cURL command to run:**

```
curl -s -header 'Content-Type: application/json' -request POST -data-binary "@./YOUR_JSON_FILE.json"
http://SS_IP_HERE/api/listactivecalls | jq
```

**Results:**

```
{
  "Status": "OK",
  "Calls":
  [{
    "CallID": "25169817",
    "SourceNumber": "490",
    .....,
    "Duration": "00:00:12"
  }]
}
```

## Python Scheduled Routing Details Example Call

```
##### Request #####  
import requests  
  
url = "http://SS_IP_HERE/api/scheduledrouting/"  
headers = {"Content-Type": "application/json", "Content-Length": "74"}  
jsonBody = {  
    "Customer": "1",  
    "Key": "YOUR KEY HERE",  
    "ID": "2"  
}  
  
response = requests.request("POST", url, headers=headers, json=jsonBody)  
  
if response.status_code == 200:  
    print(response.text)  
elif response.status_code == 404:  
    print('**Not found.**')  
##### Response #####  
{  
    "Status": "OK",  
    "Name": "Test 2",  
    "Extension": "102",  
    "RouteExt": "9991",  
    "RouteType": "12",  
    "IRID": "5",  
    "RouteName": "test-ext"  
}
```