Zipwhip API Developer Reference
Provision Phone Numbers

Version 1.0, March 2017

Zipwhip, Inc

1501 1st Avenue South

Seattle, WA 98134

Phone: 1-855-ZIPWHIP (1-855-947-9447)

Email: info@zipwhip.com

For Customer Use Only

Version History

Version	Version Date	Authors	Notes
Version 1.0	March 1, 2017	Alan Capps, James Lapic, and Scott Johnson	First release

Contents

1.0 Provisioning API Summary

This document describes how to use the HTTP Version of Zipwhip's Provisioning API to create your own provisioning process/application. You can use the API-based process/application to perform high-volume provisioning for adding text-messaging service to existing telephone numbers. This document also includes reference material for the commands and parameters used to provision phone numbers and create/manage accounts.

1.1 Introduction

There are two Zipwhip tools that you can use to complete provisioning tasks:

- The HTTP Provisioning API, which gives programmers the tools to create their own provisioning solutions. This version of the API is better suited to high-volume provisioning.
- The Dashboard API, which is a web-based application with a graphic user interface (GUI) designed to enable non-programmers to create/manage accounts and provision phone numbers.

These tools support the two types of customer accounts that require provisioning capabilities: Platform and Non-Platform.

Non-Platform account customers use Zipwhip's messaging gateway services in conjunction with their own applications and systems. Non-Platform account customers cannot use the full set of features available through Zipwhip's cloud-based end-user applications.

Platform account customers can use Zipwhip's Dashboard and Web/Desktop messaging applications (platforms). The Dashboard and messaging applications use cloud resources, which gives them more features (custom signatures, auto-reply, keywords, etc.) than the HTTP API.

Provisioning Tasks: HTTP vs Dashboard

Task	HTTP	Dashboard
Eligible	Yes	Yes
Provision	Yes	Yes
Status	Yes	Yes
Manage	No	Yes
Delete Account	Yes	Yes

1.2 Provisioning guidelines

Zipwhip requires that all customers who use the Provisioning tools secure proper authorization for any number that they provision. Please refer to your Zipwhip service agreement for details about authorization to provision phone numbers.

1.3 Provisioning tasks overview

There are several tasks you can complete using the Provisioning API. The following is a complete list of tasks:

- Verify that the telephone number is eligible or ineligible to be provisioned.
- Provision an eligible telephone number.
- Manage the provisioned line.
- Set up a new line (platform specific) and account.
- Notify the user.
- Delete an account.
- View the service and routing status.

2.0 Web Portal (Zipwhip Dashboard)

You can use Zipwhip's web portal, the Zipwhip Dashboard, to create and manage accounts, and provision telephone numbers. The Dashboard has an easy-to-use interface that makes provisioning and managing lines simple and efficient.

2.1 Supported Dashboard calls:

The Dashboard can issue the following calls:

- **add** sets up a new service
- **manage** changes details of an existing service

2.1.1 Table: Production location

System	Location
Production	https://provision.zipwhip.com

3.0 HTTP Provisioning API

You can use Zipwhip's HTTP Provisioning API to create your own automated provisioning solutions. Upon request, Zipwhip sends you an API key, which then must be included in all interactions with the HTTP Provisioning API. All web calls must be made in the form of an HTTP GET. Input values are passed as standard key value pairs via HTTP parameters. Make sure that all passed values are URL encoded.

Zipwhip requires that all customers who use the Provisioning tools secure proper authorization from the end user of record for each telephone number provisioned. Failure to secure proper authorization can lead to termination of your service agreement with Zipwhip. Refer to your Zipwhip service agreement for specific contractual requirements regarding provisioning telephone numbers.

3.1 Supported API calls

The Zipwhip Provisioning API supports both primary and secondary calls. The primary calls are those included in all versions of the API. Secondary calls are those that are included as options that may require an additional purchase.

3.1.1 Primary API calls:

You can use the following primary calls with the Provisioning API:

- **eligible** verifies that telephone number is/is not eligible to be provisioned
- **add** sets up a new service for an eligible telephone number
- **status** shows the service and routing status of a specific entry assigned to the customer account
- **update** changes details of an existing service
- **delete** removes the service

These primary calls give you the ability to perform all of the tasks required to provision and set up text messaging service for any provisioned phone number.

3.1.2 Secondary API calls

The secondary calls available with the Provisioning API may require an additional purchase. The following are the secondary calls that you can use:

- **stop/unstop** allows you to view a report that identifies the wireless phone numbers that have requested that they stop receiving text messages. It also identifies numbers that have requested to restart receiving messages. This call is currently available only to Non-Platform users.
- **enum** allows you to view details about carrier assignment. This call requires an additional purchase. Zipwhip recommends that you decide if you need this call prior to purchase. This call is available to both platform and non-platform users.

- **session key** acquires an authentication key from Zipwhip. This call is available only to platform users and should be used with Zipwhip's Messaging API. Upon each request, this call creates a new, non-expiring session key.

  Note: You must create a session key for each phone number that you provision.

The secondary calls give you the ability to add specific end-user or admin-level tasks to your text-messaging service.

3.1.3 Table: API location

The API is versioned by the date of release and does not change until the next versioned release.

Note: This API is different from the Zipwhip Messaging API.

System	Location
Production	https://provision.zipwhip.com/api/20140925

3.2 Provisioning API tasks

There are several tasks you can complete using the Provisioning API. The following is a complete list of tasks:

- Verify that a landline or toll free telephone number is eligible or ineligible to be provisioned
- Provision an eligible telephone number
- View the provision/routing status
- Manage provisioned line
- Set up new line (platform specific) and customer account
- Notify the user
- Delete an account

4.0 Verify a telephone number is eligible

You can verify that the number you want to provision is eligible as a stand-alone task or as part of the provisioning process. This section describes verifying eligibility as a stand-alone task. For information about completing this task as part of the provisioning process, see **Section 5 Provisioning process start to finish**.

Before any telephone number can be provisioned for text-message traffic, you must verify that the number is eligible. To verify that a phone number is eligible, you can use the **provision/eligible** call.

The eligible call is generally used to perform the first step in the provisioning process. However, it can be used for other applications, such as building a list of phone number that are eligible to be provisioned or displaying routing information for an eligible number. The eligible call can also be used as a standalone task.

Important! If you are provisioning multiple telephone numbers, then you must use a separate eligible call for each number.

The eligible call tries to verify that the landline or toll free telephone number that you submit meets the following criteria:

- Conforms to e.164 format. For example, +12067772340.
- Is an active phone number that currently services inbound and outbound voice traffic.
- Is not already provisioned in the Zipwhip system.

When you enter a phone number to be verified, there are five possible results:

- **Eligible**, which means that the phone number is a valid landline or toll free number that can be provisioned.
- **Ineligible**, which means that the phone number is not a valid landline/toll free number and cannot be provisioned.
- **Invalid**, which means that you did not type a 10-digit phone number.
- **Provisioned**, which means that the phone number is already provisioned.
- **Number not assigned to your organization**, which means that the number is already provisioned in the Zipwhip system.

Note: If you believe that the ineligible status for the phone number in question is an error, please contact Zipwhip customer support. Text or Call (855) 947-9447, or email Support@zipwhip.com.

4.1 Verify call examples

The following examples show successful and unsuccessful attempts to verify phone numbers.

4.1.1 Example: Provision eligible call

```
GET /api/20140925/provision/eligible HTTP/1.1
Host: https://provision.zipwhip.com

?api_key=&phone_number=
```

4.1.2 Example: Response for eligible number

```
{
    "eligible": "true",
    "error": "false",
    "status_code": 100,
    "transaction_id": "3f009aa8-3e88-c69f-3e48-bc42f3594c89"
}
```

4.1.3 Example: Response for ineligible mobile number

```
{
    "eligible": "false",
    "error": "false",
    "status_code": "111",
    "status_desc": "Ineligible - Mobile Number",
    "transaction_id": "8c63823c-3fa1-3345-7c9a-30199323ec08"
}
```

4.1.4 Example: Response for number already provisioned by your organization

```
{
    "eligible": "false",
    "error": "false",
    "status_code": "113",
    "status_desc": "Ineligible - Number already provisioned by your Org",
    "transaction_id": "3c4bee29-7fff-25ed-064f-0620d6bd4c45"
}
```

4.1.5 Example: Response for number already provisioned by another organization

```
{
    "eligible": "false",
    "error": "false",
    "status_code": "112",
    "status_desc": "Ineligible - Number already provisioned by another Org",
    "transaction_id": "bd021b09-0c78-ba38-73c0-c87bc82d2e78"
}
```

4.1.6 Table: Eligible call input parameters

Name	Required	Type	Description
`api_key`	Yes	String (24)	Authentication token. You get the token when you sign up for API service from Zipwhip.
`phone_number`	Yes	String (12)	The phone number you want to identify as eligible or ineligible. The number must be formatted in full e.164 format. Example: +18005555555

4.1.7 Table: Eligible call output parameters

Name	Required	Type	Description
`transaction_id`	No	String (34)	A unique value that is assigned to each specific eligibility request. This value is used to track the request.
`status_code`	No	Integer (3)	Status code that is used to identify the current state of the eligibility request. For example, there are/should be status codes for success, in-process, and failure.
`status_desc`	No	String (255)	Natural language description of the status code. For example, "The number requested is already provisioned by another company."
`error`	No	Boolean	Response that specifies if the request status returned is an error state.
`eligible`	Yes	string	True or False value indicates if the phone number eligible/ineligible to be provisioned.

5.0 Provisioning process start to finish

To successfully provision a telephone number for text-message traffic, you must do more than issue the **add** call. While this is the call that specifically sends the provision request to the primary voice carrier, there are other steps (and associated calls) that must be completed to provide text-messaging service. The following are the steps Zipwhip recommends you complete to successfully provision a phone number for text-message traffic:

1. Verify that the number is eligible (eligible call).

2. Send the provision request (add call).

3. Optional: Track the provision request/routing status (status call).

4. Set up user login and customer account in your system.

   Important! For Platform users, use web hooks to notify your platform of a new message event on your managed lines. Platform users have a Zipwhip Cloud Account and their messages are stored in Zipwhip databases. They can use features such as custom signature, auto reply, and keywords.

5. Notify the end user that their account is active.

   - If you use the Platform version, then either you or Zipwhip can send the activation email message.
   - If you use the Non-Platform version, then you must send the activation email message.

5.1 Verify the telephone number is eligible

The first step in the process of provisioning a phone number is to verify that the number you want to provision is eligible to be provisioned. The eligible call is a required step in the provisioning process. When you build a step around the eligible call, you should accommodate both eligible (success) and ineligible (failure) results.

For a complete description of this step and using the provision/eligible call, see Section 4: Verify a telephone number is eligible to be provisioned.

5.2 Send the provision request

The second step of the provisioning process is sending the provision request to the primary voice carrier for the eligible telephone number. This step cannot be completed until you have verified that the telephone number being submitted for provisioning is eligible. The **provision/add** call is the specific command that sends the provision request.

The following example includes the API key (required) and phone number (required). The effective date is not required because if no date is specified, the phone number is processed immediately. You use the effective date as a method of assuring that the account is paid before the line/number is provisioned. Vendor customer ID is a field you can use to cross-reference with your existing systems and applications.

If you want to provision multiple numbers, then you must send an individual request for each number.

5.2.1 Example: Provision add call

```
$ curl https://provision.zipwhip.com/api/20140925/provision/add \
    -d api_key=lgaMfmMCJ7I9Euz7kYpIDuSF\
    -d phone_number= %2B12065551212 \
    -d effective_date=2014-05-08T17%3A35%3A16%2B00%3A00 \
    -d vendor_customer_id=Acme_0123456
```

5.2.2 Example: Provision add response

```
{
  "transaction_id": "4bdedfa2-3d37-bcea-7cde-1f7f46d23406",
  "status_code": "100",
  "status_desc": "Successful Request ",
  "error": "false",
  "effective_date": "2014-05-08T17:35:16+00:00",
  "password": "abcabc123"
}
```

The following table shows the input parameters for the provision/add call:

5.2.3 Table: Add call input parameters

Name	Required	Type	Description
`api_key`	Yes	String (24)	Authentication token. You get the token when you sign up for API service from Zipwhip.
`phone_number`	Yes	String (12)	The phone number to be provisioned and activated for service. The number must be formatted in full e.164 format. Example: +18005555555
`effective_date`	No	String (25)	The specific date and time to process the provisioning request. If no effective date is provided, then the number is provisioned immediately. Format: ISO_8601 Example: 2014-05-08T17:35:16+00:00
`vendor_customer_id`	No	String (100)	Custom ID is an optional field that customers can use to link to link with their existing systems/applications.

5.2.4 Table: Add call output parameters

Name	Required	Type	Description
transaction_id	No	String (34)	A unique value that is assigned to each specific provisioning request. This value is used to track the request.
status_code	No	Integer (3)	Status code that is used to identify the end state of the provisioning request.
status_desc	No	String (255)	Natural language description of the status code. For example, "Success means that the number has been successfully provisioned and the account is active."
error	No	Boolean	Response that specifies if the request status returned is an error state.
effective_date	No	String (255)	Specific date time that the phone number is provisioned if an effective_date parameter is specified. If no effective_date is specified, then the response includes the current time (in UTC). Format: ISO_8601 Example: 2014-05-08T17:35:16+00:00

5.3 Track the provision request/routing status

The third step in the process of provisioning a phone number is an optional step. You can use the **provision/status** call to track the status of your provision request.

You can use the status call in several different use cases. You can show the status states to the end user or hide them from the end user. You can let the end user make this call on an ad hoc basis, or build the call into a process. You can choose to display the provisioning request status in real-time or only when the status changes.

Zipwhip assigns one of three status states to your request:

- **Pending Activation** means that the provision request is still being processed.
- **Account Activated** means that the account has been successfully provisioned.
- **Unknown** means that an error has occurred.

Pending Activation means that provisioning has started but is not complete. Provisioning normally takes less than 15 minutes; however, some numbers can take up to one (1) hour to provision. If a provision request is Pending for four (4) hours, then you can contact Zipwhip Support to identify the cause for the delay.

Account Activated means that the account has been successfully provisioned. When a given number is successfully provisioned, then the number can successfully send and receive text messages. However, for Platform users, before end users can send and receive text messages, you must provide end users with a method to log in/authenticate and set up their line. When you set up the end user’s line, you give them a user interface (UI) that lets them view and respond to text messages. You can either create a custom UI or give end users access to the Zipwhip UI.

Unknown means that an error has occurred in the provisioning process. If this error occurs, then retry provisioning the given toll free number. If your retry effort fails, then you should contact Zipwhip Support.

5.3.1 Example: Provision status call

```
$ curl https://provision.zipwhip.com/api/20140925/provision/status \
    -d api_key=lgaMfmMCJ7I9Euz7kYpIDuSF\
    -d phone_number=%2B12065551212
```

5.3.2 Example: Provision status response

```
{
  "transaction_id": "4bdedfa2-3d37-bcea-7cde-1f7f46d23406",
  "status_code": "100",
  "status_desc": "Successful Request",
  "error": "false",
  "account_status": "provisioned",
  "last_updated": "2014-05-08T17:35:16+00:00"
}
```

5.3.3 Table: Status call input parameters

Name	Required	Type	Description
api_key	Yes	String (24)	Authentication token. You get the token when you sign up for API service from Zipwhip.
phone_number	Yes	String (12)	The phone number for which you want the provisioning status. The number must be formatted in full e.164 format. Example: +18005555555

5.3.4 Table: Status call output parameters

Name	Required	Type	Description
`transaction_id`	No	String (34)	A unique value that is assigned to each specific status request. This value is used to track the request.
`status_code`	No	Integer (3)	Status code that is used to identify the current status of the provisioning request.
`status_desc`	No	String (255)	Natural language description of the status code. For example, "Success means that the number has been successfully provisioned and the account is active."
`error`	No	Boolean	Response that specifies that an error occurred when processing the status request.
`account_status`	Yes	String (255)	The current state of the provisioning request.
`last_updated`	Yes	String (25)	The specific date and time that the account was last updated. Format: ISO_8601 Example: 2014-05-08T17:35:16+00:00

5.4 Set up user login and account

The fourth step in provisioning a phone number is setting up the end-user login. How you complete this step depends on whether you are using the Platform or Non-Platform version of the Zipwhip provision tools.

For Platform users, the login must include a username and password. If you use the Platform version, then you can use the Dashboard to create and send a username/password.

For Non-Platform users, the end user setup must occur through your system and must use your system to authenticate.

5.5 Notify the end user

The final step in the process of provisioning a phone number is the required step of notifying the end user that their account has been activated.

If you use the Platform version, then you can have Zipwhip send the activation email message to the end user, or send the activation message to the end user yourself. If you use the Non-Platform version, then you are responsible for sending the activation message.

If you use the Platform version, then you can use the Manage Account page to send an email to notify the end user their account is active. The email message must include the end user's username and password. This email message is a Zipwhip-branded message.

If you use the Non-Platform version, then you can use several methods to notify the end user that their account is active. You can use your existing business email system and manually send notifications, adapt your email system to send the notifications automatically, or add the notify function to the system/application that uses the Provisioning API.

6.0 Update an account

You can use the provision/update call to make changes to an existing line/account. At present, if you use the Platform version, the only parameter you can update is Vendor Customer ID.

If you use the Non-Platform version, then you define the parameters that you can update. If you want to update multiple lines/accounts, then you must update each line/account individually. You cannot update multiple lines with a single call.

6.1 Verify call examples

The following example shows a successful attempt to update the status of a phone number.

6.1.1 Example: Update call

```
$ curl https://provision.zipwhip.com/api/20140925/provision/update \
     -d api_key=lgaMfmMCJ7I9Euz7kYpIDuSF\
     -d phone_number= %2B12065551212 \
     -d effective_date=2014-05-08T17%3A35%3A16%2B00%3A00 \
     -d vendor_customer_id=Acme_0123456
```

6.1.2 Example: Update call response

```
{
  "transaction_id": "4bdedfa2-3d37-bcea-7cde-1f7f46d23406",
  "status_code": "100",
  "status_desc": "Successful Request ",
  "error": "false",
  "effective_date": "2014-05-08T17:35:16+00:00"
}
```

6.1.3 Table: Update call input parameters

Name	Required	Type	Description
`api_key`	Yes	String (24)	Authentication token. You get the token when you sign up for API service from Zipwhip.
`phone_number`	Yes	String (12)	The phone number to be updated. The number must be formatted in full e.164 format. Example: +18005555555
`effective_date`	No	String (25)	Specific date time that the phone number is updated if an effective_date parameter is specified. If no effective_date is specified, then the update occurs immediately. The time is in UTC format. Format: ISO_8601 Example: 2014-05-08T17:35:16+00:00
`vendor_customer_id`	No	String (100)	Custom ID is an optional field that customers can use to link to link with their existing systems/applications.

6.1.4 Table: Update call output parameters

Name	Required	Type	Description
`transaction_id`	No	String (34)	A unique value that is assigned to each specific provisioning request. This value is used to track the request.
`status_code`	No	Integer (3)	Status code that is used to identify the current state of the update request.
`status_desc`	No	String (255)	Natural language description of the status code.
`error`	No	Boolean	Response that specifies that an error occurred when processing the provision request.
`effective_date`	No	String (25)	Specific date time that the phone number is updated if an effective_date parameter is specified. If no effective_date is specified, then the current time is included in the response (in UTC). Format: ISO_8601 Example: 2014-05-08T17:35:16+00:00

7.0 Delete an account

You can use the **provision/delete** call to delete an existing account, which means that billing on the account stops on date the account is deleted. If you want to delete multiple phone numbers or accounts, then you must delete each line/account individually.

Important! If you're using the Platform version, when you delete an account, then the contacts associated with the account are deleted. The messages sent and received to/from the account are also deleted.

7.1 Delete call examples

The following example shows a successful attempt to delete an existing account.

7.1.1 Example: Delete call

```
$ curl https://provision.zipwhip.com/api/20140925/provision/delete \
     -d api_key=lgaMfmMCJ7I9Euz7kYpIDuSF\
     -d phone_number= %2B12065551212 \
     -d effective_date=2014-05-08T17%3A35%3A16%2B00%3A00 \
     -d reason=Customer%20cancelled%20service.
```

7.1.2 Example: Delete call response

```
{
  "transaction_id": "4bdedfa2-3d37-bcea-7cde-1f7f46d23406",
  "status_code": "100",
  "status_desc": "Successful Request",
  "error": "false",
  "effective_date": "2014-05-08T17:35:16+00:00"
}
```

7.1.3 Table: Delete call input parameters

Name	Required	Type	Description
`api_key`	Yes	String (24)	Authentication token. You get the token when you sign up for API service from Zipwhip.
`phone_number`	Yes	String (12)	The phone number to be deleted and service suspended. The number must be formatted in full e.164 format. Example: +18005555555
`effective_date`	No	String (25)	The specific date and time to delete the phone number and account. If no effective date is provided, then the number is deleted immediately and the current date and time is included in UTC format. Format: ISO_8601 Example: 2014-05-08T17:35:16+00:00
`reason`	No	String (255)	Reason that the account is cancelled.

7.1.4 Table:Delete call output parameters

Name	Required	Type	Description
transaction_id	No	String (34)	A unique value that is assigned to each specific provisioning request. This value is used to track the request.
status_code	No	Integer (3)	Status code that is used to identify the current state of the delete request.
status_desc	No	String (255)	Natural language description of the status code.
error	No	Boolean	Response that specifies if the request status returned is an error state.
effective_date	No	String (25)	The specific date and time to process the delete request. If no effective date is provided, then the number is deleted immediately. Format: ISO_8601 Example: 2014-05-08T17:35:16+00:00

Zipwhip, Inc.

1501 1st Avenue South
Seattle, WA 98134
Phone: 1-855-ZIPWHIP (1-855-947-9447)
Email: info@zipwhip.com