Linode.com API - v1.0

Linode.com

February 3, 2009

Contents

List of Figures

Listings

List of Tables

1 Basics

The API is reached at *https://api.linode.com/api/*. The API is interacted with via standard HTTP GET or POST requests with data passed in via form fields and returned in a user specified format.

An API key may be generated/retrieved from the "My Profile" link within the members' site. API keys are user specific and any restrictions of the associated user are enforced.

See Listing 1 and Listing 2 for a sample minimal request and associated response.

Listing 1: Sample Request

```
GET /api/?api_key=cakeisgood&action=bogus HTTP/1.1
```

Listing 2: Sample Response

```
{
    "ACTION":           "bogus",
    "ERRORARRAY":       [{
        "ERRORMESSAGE":     "Method Not Found",
        "ERRORCODE":        666.0
    }],
    "DATA":             {},
}
```

1.1 The Request

Parameters are sent as standard form data in *name=value* pairs via your choice of HTTP POST or GET requests. Field names are case insensitive. With the exception of a three special fields, field names and their value differ based on the method called.

The special field *action* contains the method you wish to invoke, *api_key* contains your API key, and *resultFormat* contains the format for the response. See Table 1 for a list of available formats.

Table 1: Available Response Formats

JSON WDDX	Machine Readable
human	Human readable HTML tables

1.2 The Response

If a specific format for the response is not requested, JSON will be used. There is a top level hash of key-value pairs. Table 2 contains an overview of the object and samples of each type follow.

Table 2: The Response Object

Key	Value
Action	The method called that generated this response
ERRORARRAY *may be empty*	Contains an Array Each element contains a hash with two keys: *ERRORMESSAGE* and *ERRORARRAY*
DATA *may be empty*	Contains a Hash Format varies based on called method

1.2.1 JSON

Listing 2 shows an example JSON formatted response for an error. The listing has had additional white-space added for readability, the actual response will contain no extraneous space. Listing 3 contains a full JSON response. JSON is the default response format.

Listing 3: Sample JSON Response

```
{
  "Action":              "domainGet",
  "ERRORARRAY":          [],
  "DATA":{
      "TTL_SEC":         0,
      "REFRESH_SEC":     0,
      "DOMAIN":          "testdomain.com",
      "DOMAINID":        23,
      "TYPE":            "master",
      "RETRY_SEC":       0,
      "SOA_EMAIL":       "caker@linode.com",
      "STATUS":          1
    },
}
```

1.2.2 WDDX

Listing 4 contains a sample WDDX response.

1.2.3 Human

Figure 1 shows a sample human readable response. This response format is intended primarily for debugging.

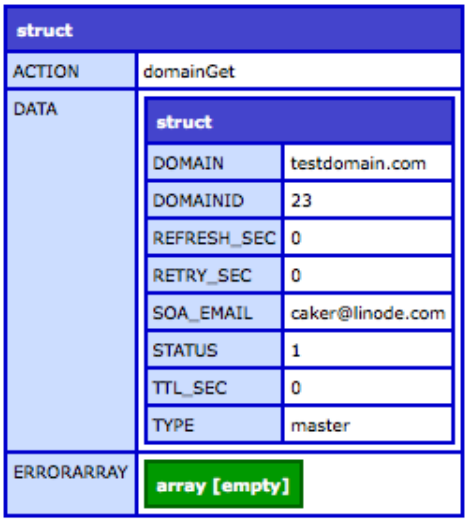
1.3 Batching Requests

Multiple requests may be batched into a single transaction. To send a batch request use *action=batch* and format your requests as normal with *action* and associated parameters passed as an array of hashes in JSON or WDDX

Listing 4: Sample WDDX Response

```
<wddxPacket version='1.0'>
<header/>
<data><struct>
  <var name='ACTION'><string>domainGet</string></var>
  <var name='ERRORARRAY'><array length='0'></array></var>
  <var name='DATA'>
    <struct>
        <var name='TTL_SEC'><number>0.0</number></var>
        <var name='REFRESH_SEC'><number>0.0</number></var>
        <var name='DOMAIN'><string>testdomain.com</string></var>
        <var name='DOMAINID'><number>23.0</number></var>
        <var name='TYPE'><string>master</string></var>
        <var name='RETRY_SEC'><number>0.0</number></var>
        <var name='SOA_EMAIL'><string>caker@linode.com</string></var>
        <var name='STATUS'><number>1.0</number></var>
    </struct>
  </var>
</struct></data>
</wddxPacket>
```

Figure 1: Sample Human Readable Response



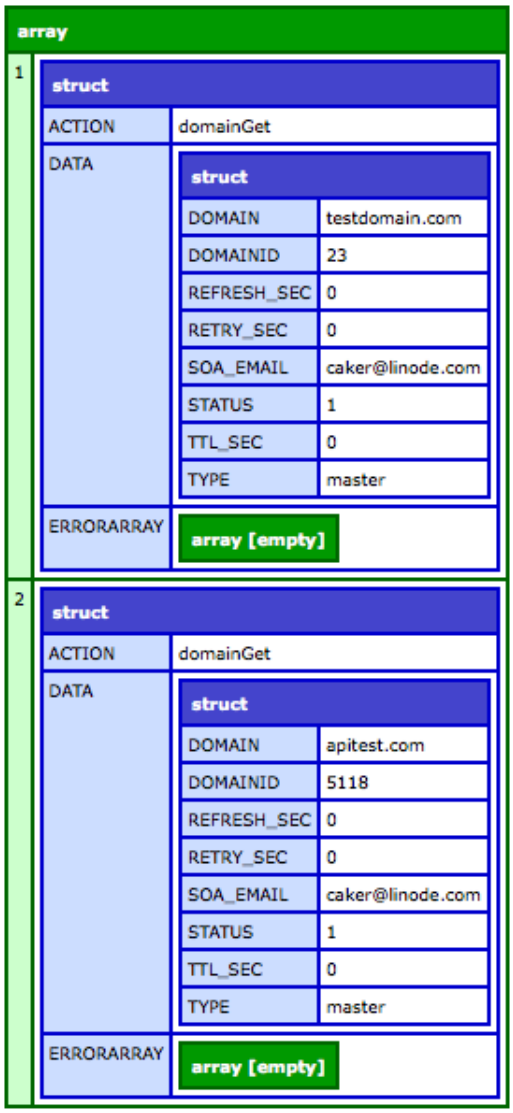
format in the *requestArray* field. Listing 5 contains a sample batch request with line breaks and URL decoding to aid in readability.

The response will be an array of replies matching the non-batched format.

Listing 5: Sample Batch Request

```
GET /api/?API_KEY=<key>&action=batch&ResultFormat=human&requestArray=[
        {"action":"domainGet","DomainID":23},
        {"action":"domainGet","DomainID":5118.0}
]
```

Figure 2: Sample Human Readable Batch Response



array

1

struct

ACTION	domainGet
DATA	struct
DOMAIN	testdomain.com
DOMAINID	23
REFRESH_SEC	0
RETRY_SEC	0
SOA_EMAIL	caker@linode.com
STATUS	1
TTL_SEC	0
TYPE	master
ERRORARRAY	array [empty]

2

struct

ACTION	domainGet
DATA	struct
DOMAIN	apitest.com
DOMAINID	5118
REFRESH_SEC	0
RETRY_SEC	0
SOA_EMAIL	caker@linode.com
STATUS	1
TTL_SEC	0
TYPE	master
ERRORARRAY	array [empty]

2 DNS Manager Methods

2.1 Overview

Brief overview of available methods:

domainList Lists the domains visible.

domainGet Retrieve the details for a specific domain.

domainSave Create or update a domain within the DNS manager.

domainDelete Delete a domain within the DNS manager.

domainResourceList Lists the resource records associated with a particular domain.

domainResourceGet Retrieve the details for a specific resource record.

domainResourceSave Create or update a resource record within the DNS manager.

domainResourceDelete Delete a resource record within the DNS manager.

Each domain, or zone, in the system has an associated status field. Valid values are listed in Table 15.

2.2 domainList

Description Lists the domains visible.

Parameters none

Returns An array, one row per domain, of hashes containing data for each visible domain.

Table 3: domainList Returned Fields

Field	Description
DOMAINID	The unique ID for this domain
DOMAIN	The domain's name *e.g. linode.com*
TYPE	*master* or *slave*
STATUS	Domain's status (see Table 15)
SOA_EMAIL	SOA email address for the domain
REFRESH_SEC	*refresh* value for the domain in seconds
RETRY_SEC	*retry* value for the domain in seconds
TTL_SEC	*ttl* value for the domain in seconds

2.3 domainGet

Description Retrieve the details for a specific domain.

Parameters *DomainID*

Returns A hash containing data for the requested domain in the format described in Table 3.

2.4 domainSave

Description Create or update a domain within the DNS manager.

Parameters See Table 5

Returns The ID of the created or modified domain.

2.5 domainDelete

Description Delete a domain within the DNS manager.

Parameters DomainID

Returns DomainID

2.6 domainResourceList

Description Lists the resource records associated with a particular domain.

Parameters *DomainID*

Returns An array, one row per record, of hashes containing data for each resource record in the domain. Only the fields from Table 9 that make sense for a particular resource type are returned.

2.7 domainResourceGet

Description Retrieve the details for a specific resource record.

Parameters *ResourceID*

Returns A hash containing the data for the requested resouce record. Only the fields from Table 11 that make sense for a particular resource type are returned.

2.8 domainResourceSave

Description Create or update a resource record within the DNS manager.

Parameters See Table 12

Returns The ID of the created domain.

Table 4: domainGet Parameters

Field	Required	Description
DomainID	**Yes**	The unique ID for the domain you wish to retrieve.

Table 5: domainSave Parameters

Field	Required	Description
DomainID	**Yes**	The unique ID for the domain you wish to update, 0 to create a new domain.
Domain	**Yes**	The domain's name *e.g. linode.com*
Type	**Yes**	*master* or *slave*
Status	**Yes**	Domain's status (see Table 15)
SOA_Email	No[a]	SOA email address for the domains
Master_IPs	No[b]	IP(s) of masters, semicolon separated, required if *Type* is slave, ignored otherwise
Refresh_Sec	No	*refresh* value for the domain in seconds, 0 for default
Retry_Sec	No	*retry* value for the domain in seconds, 0 for default
TTL_Sec	No	*ttl* value for the domain in seconds, 0 for default

[a] required for *master* domains
[b] required for *slave* domains

Table 6: domainSave Returned Fields

Field	Description
DOMAINID	The unique ID of the created or modified domain

Table 7: domainDelete Parameters

Field	Required	Description
DomainID	**Yes**	Unique identifier for the domain you wish to delete.

Table 8: domainResourceList Parameters

Field	Required	Description
DomainID	**Yes**	The unique ID for the domain you wish to list the resources from.

Table 9: domainResourceList Returned Fields

Field	Description
RESOURCEID	Unique identifier for this record
DOMAINID	Unique identifier for the domain this record belongs to
NAME	This record's name, may be empty
TYPE	RR Type, one of *NS*, *MX*, *A*, *AAAA*, *CNAME*, *TXT*, or *SRV*
TARGET	IP or name this record points to
PRIORITY	*MX* record priority
TTL_SEC	*TTL* for this record in seconds, 0 for domain default
WEIGHT	Weight for *SRV* RRs
PORT	Port for *SRV* RRs

Table 10: domainResourceGet Parameters

Field	Required	Description
ResourceID	**Yes**	The unique ID of the resource record

Table 11: domainResourceGet Returned Fields

Field	Description
RESOURCEID	Unique identifier for this record
DOMAINID	Unique identifier for the domain this record belongs to
NAME	This record's name, may be empty
TYPE	RR Type, one of *NS*, *MX*, *A*, *AAAA*, *CNAME*, *TXT*, or *SRV*
TARGET	IP or name this record points to
PRIORITY	*MX* record priority
TTL_SEC	*TTL* for this record in seconds, 0 for domain default
WEIGHT	Weight for *SRV* RRs
PORT	Port for *SRV* RRs

Table 12: domainResourceSave Parameters

Field	Required	Description
ResourceID	**Yes**	Unique identifier for the record you wish to update, 0 to create a new record.
DomainID	**Yes**	Unique identifier for the domain this record belongs to
Name	**Yes**	This records name, may be empty
Type	**Yes**	RR Type, one of *NS*, *MX*, *A*, *AAAA*, *CNAME*, *TXT*, or *SRV*
Target	**Yes**	IP or name this record points to
Priority	No[a]	*MX* record priority
TTL_Sec	No	*TTL* for this record in seconds, 0 for domain default
Weight	No[b]	Weight for *SRV* RRs
Port	No[c]	Port for *SRV* RRs
Protocol	No	Protocol for *SRV* records, defaults to *_udp*

[a] Required for *MX* records
[b] Required for *SRV* records
[c] Required for *SRV* records

Table 13: domainResourceSave Returned Fields

Field	Description
RESOURCEID	The unique ID of the created or modified record

2.9 domainResourceDelete

Description Delete a resource record within the DNS manager.

Parameters ResourceID

Returns ResourceID

Table 14: domainResourceDelete Parameters

Field	Required	Description
ResourceID	**Yes**	Unique identifier for the record you wish to delete.

Appendices

Table 15: Values for Domain Status

Value	WWW Desc	Description
0	Disabled	Domain is not being served
1	Active	Domain is active
2	Edit Mode	Domain being served but changes not rendered
3	Has Errors	There are currently errors in the rendered zonefile