

Table of Contents

1 Accounts-accountId-domains-domainName:PUT	1
2 PUT /accounts/{accountId}/domains/{domainName}	2
2.1 Contents	2
2.2 Request	2
2.3 Response	3
2.4 Example of ordering .com domain	3
2.5 Example of ordering .us domain	4
2.6 See also	4
3 Accounts-accountId-transfer-domains-domainNames:PUT	5
4 PUT /accounts/{accountId}/domains/{domainName}	6
4.1 Contents	6
4.2 Request	6
4.3 Response	7
4.4 Example of ordering transfer .com domain	7
4.5 Example of ordering transfer .us domain	8
4.6 See also	9
5 Accounts-accountNumber-domains-domainName-search:GET	10
6 GET /accounts/{accountNumber}/domains/{domainName}	11
6.1 Contents	11
6.2 Request	11
6.3 Response	11
6.4 Example search for a domain product	11
6.5 See also	12
7 Accounts-accountNumber-domains-transfer-domainName:GET	13
8 GET /accounts/{accountNumber}/domains/transfers/{domainName}	14
8.1 Contents	14
8.2 Request	14
8.3 Response	14
8.4 Example of getting transfer status	15
8.5 See also	16
9 Accounts-accountNumber-domains-transfer-domainName:PUT	17
10 PUT /accounts/{accountNumber}/domains/transfers/{domainName}	18
10.1 Contents	18
10.2 Request	18
10.3 Response	18
10.4 Example of getting transfer status	19
10.5 See also	19
11 Accounts-accountNumber-domains:GET	20
12 GET /accounts/{accountNumber}/domains	21
12.1 Contents	21
12.2 Request	21
12.3 Response	21
12.4 Example search for a domain product	22
12.5 Example search for a domain product	22
12.6 See also	22
13 Domain Name Registration	23
14 Domain Name Registration API	24
14.1 Contents	24
14.2 Top level domains	24
14.3 Domain Name Availability Check	24
14.4 Domain Name Transfers	25
14.5 Ordering domain	25
14.6 Domain Name Management	25
14.7 Renew Domain	26
14.8 Domain Pricing	26
14.9 Private Registration Availability By TLD	26
14.10 See also	26
15 Domains-domainName-authInfo:GET	27
16 GET /domains/{domainName}/authInfo	28
16.1 Contents	28
16.2 Request	28
16.3 Response	28
16.4 Example of getting the domain authorization code	28
16.5 See also	28
17 Domains-domainName-authInfo:PUT	29
18 PUT /domains/{domainName}/authInfo	30
18.1 Contents	30
18.2 Request	30

Table of Contents

18 PUT /domains/{domainName}/authInfo	
18.3 Response.....	30
18.4 Example of setting the domain auth info.....	30
18.5 See also.....	31
19 Domains-domainName-autoRenew:GET.....	32
20 GET /domains/{domainName}/autoRenew.....	33
20.1 Contents.....	33
20.2 Request.....	33
20.3 Response.....	33
20.4 Example of getting the domain autoRenew status.....	33
20.5 See also.....	33
21 Domains-domainName-autoRenew:OPTIONS.....	35
22 OPTIONS /domains/{domainName}/autoRenew.....	36
22.1 Contents.....	36
22.2 Request.....	36
22.3 Response.....	36
22.4 Example of setting the domain autoRenew status.....	36
22.5 See also.....	36
23 Domains-domainName-autoRenew:PUT.....	37
24 PUT /domains/{domainName}/autoRenew.....	38
24.1 Contents.....	38
24.2 Request.....	38
24.3 Response.....	38
24.4 Example of setting the domain autoRenew status.....	38
24.5 See also.....	38
25 Domains-domainName-contactType:GET.....	39
26 GET /domains/{domainName}/{contactType}.....	40
26.1 Contents.....	40
26.2 Request.....	40
26.3 Response.....	40
26.4 Example of getting domain contact information.....	41
26.5 See also.....	41
27 Domains-domainName-contactType:PUT.....	42
28 PUT /domains/{domainName}/{contactType}.....	43
28.1 Contents.....	43
28.2 Request.....	43
28.3 Response.....	44
28.4 Example of setting domain contact information.....	44
28.5 See also.....	44
29 Domains-domainName-contactverification:GET.....	45
30 GET /domains/{domainName}/contactverification.....	46
30.1 Contents.....	46
30.2 Request.....	46
30.3 Response.....	46
30.4 Example getting contact verification status for a domain.....	46
30.5 Example sending contact verification for a domain.....	46
30.6 See also.....	47
31 Domains-domainName-details:GET.....	48
32 GET /domains/{domainName}/details.....	49
32.1 Contents.....	49
32.2 Request.....	49
32.3 Response.....	49
32.4 Examples.....	51
32.5 See also.....	53
33 Domains-domainName-hold:DELETE.....	54
34 DELETE /domains/{domainName}/hold.....	55
34.1 Contents.....	55
34.2 Request.....	55
34.3 Response.....	55
34.4 Example remove a hold from a domain.....	55
34.5 See also.....	56
35 Domains-domainName-hold:GET.....	57
36 GET /domains/{domainName}/hold.....	58
36.1 Contents.....	58
36.2 Request.....	58
36.3 Response.....	58
36.4 Example of getting the domain hold status.....	58

Table of Contents

36 GET /domains/{domainName}/hold	59
36.5 See also.....	59
37 Domains-domainName-hold:PUT.....	60
38 PUT /domains/{domainName}/hold.....	61
38.1 Contents.....	61
38.2 Request.....	61
38.3 Response.....	61
38.4 Example remove hold on a domain.....	62
38.5 See also.....	62
39 Domains-domainName-hosts:DELETE.....	63
40 DELETE /domains/{domainName}/hosts/{hostName}.....	64
40.1 Contents.....	64
40.2 Request.....	64
40.3 Response.....	64
40.4 Example of deleting a host.....	64
40.5 See also.....	64
41 Domains-domainName-hosts:GET.....	65
42 GET /domains/{domainName}/hosts.....	66
42.1 Contents.....	66
42.2 Request.....	66
42.3 Response.....	66
42.4 Example of getting the domain authorization code.....	66
42.5 See also.....	66
43 Domains-domainName-hosts:POST.....	68
44 POST /domains/{domainName}/hosts.....	69
44.1 Contents.....	69
44.2 Request.....	69
44.3 Response.....	69
44.4 Example of updating hosts.....	69
44.5 See also.....	70
45 Domains-domainName-hosts:PUT.....	71
46 PUT /domains/{domainName}/hosts.....	72
46.1 Contents.....	72
46.2 Request.....	72
46.3 Response.....	72
46.4 Example of replacing the hosts.....	73
46.5 See also.....	73
47 Domains-domainName-lock:DELETE.....	74
48 DELETE /domains/{domainName}/lock.....	75
48.1 Contents.....	75
48.2 Request.....	75
48.3 Response.....	75
48.4 Example unlock domain.....	75
48.5 See also.....	76
49 Domains-domainName-lock:GET.....	77
50 GET /domains/{domainName}/lock.....	78
50.1 Contents.....	78
50.2 Request.....	78
50.3 Response.....	78
50.4 Example of getting the domain lock status.....	78
50.5 See also.....	79
51 Domains-domainName-lock:PUT.....	80
52 PUT /domains/{domainName}/lock.....	81
52.1 Contents.....	81
52.2 Request.....	81
52.3 Response.....	81
52.4 Example lock domain.....	82
52.5 See also.....	82
53 Domains-domainName-nameservers:GET.....	83
54 GET /domains/{domainName}/nameservers.....	84
54.1 Contents.....	84
54.2 Request.....	84
54.3 Response.....	84
54.4 Example getting nameservers on a domain.....	84
54.5 See also.....	85

Table of Contents

55 Domains-domainName-nameservers:PUT	86
56 PUT /domains/{domainName}/nameservers	87
56.1 Contents	87
56.2 Request	87
56.3 Response	87
56.4 Example updating nameservers on a domain	87
56.5 Example updating nameservers using one or more name server that is not registered	87
56.6 See also	88
57 Domains-domainName-namespinner:GET	89
58 GET /domains/{domainName}/namespinner	90
58.1 Contents	90
58.2 Request	90
58.3 Response	90
58.4 Example Name Spinner	91
58.5 NameSpinner Sample Code	93
58.6 See also	93
59 Domains-domainName-namespinner:SAMPLE	94
60 SAMPLE /domains/{domainName}/namespinner	95
60.1 Contents	95
60.2 JavaScript	95
60.3 HTML	96
60.4 See also	100
61 register:GET	101
62 GET /domains/{domainName}/prices/register	102
62.1 Contents	102
62.2 Request	102
62.3 Response	102
62.4 Example of getting register prices	102
62.5 See also	103
63 renew:GET	104
64 GET /domains/{domainName}/prices/renew	105
64.1 Contents	105
64.2 Request	105
64.3 Response	105
64.4 Example of getting renew prices	105
64.5 See also	106
65 transfer:GET	107
66 GET /domains/{domainName}/prices/transfer	108
66.1 Contents	108
66.2 Request	108
66.3 Response	108
66.4 Example of getting transfer prices	108
66.5 See also	108
67 Domains-domainName-prices:GET	110
68 GET /domains/{domainName}/prices	111
68.1 Contents	111
68.2 Request	111
68.3 Response	111
68.4 Example of getting prices	111
68.5 See also	112
69 Domains-domainName-privateregistration-:GET	113
70 GET /domains/privateregistration	114
70.1 Contents	114
70.2 Request	114
70.3 Response	114
70.4 Example search for a domain product	114
70.5 See also	115
71 Domains-domainName-privateregistration:DELETE	116
72 DELETE /domains/{domainName}/privateregistration	117
72.1 Contents	117
72.2 Request	117
72.3 Response	117
72.4 Example of deleting a host	117
72.5 See also	117
73 Domains-domainName-privateregistration:GET	118

Table of Contents

74 GET /domains/{domainName}/privateregistration.....	119
74.1 Contents.....	119
74.2 Request.....	119
74.3 Response.....	119
74.4 Example of getting the domain authorization code.....	119
74.5 See also.....	120
75 Domains-domainName-privateregistration:POST.....	121
76 POST /domains/{domainName}/privateregistration.....	122
76.1 Contents.....	122
76.2 Request.....	122
76.3 Response.....	122
76.4 Example of updating e-mail address for "full" type of privateregistration.....	122
76.5 Example of downgrading "full" type of privateregistration to "light".....	123
76.6 Example of upgrading "light" type of privateregistration to "full".....	123
76.7 See also.....	123
77 Domains-domainName-renew:POST.....	124
78 POST /domains/{domainName}/renew.....	125
78.1 Contents.....	125
78.2 Request.....	125
78.3 Response.....	125
78.4 Example of ordering domain renewal.....	126
78.5 See also.....	126
79 Domains-domainName-renewTerms:GET.....	127
80 GET /domains/{domainName}/renewTerms.....	128
80.1 Contents.....	128
80.2 Request.....	128
80.3 Response.....	128
80.4 Example of getting the available renewal terms and pricing.....	128
80.5 See also.....	128
81 Domains-domainName-transfer:GET.....	130
82 GET /domains/{domainName}/transfer.....	131
82.1 Contents.....	131
82.2 Request.....	131
82.3 Response.....	131
82.4 See also.....	131
83 Domains-domainName:GET.....	132
84 GET /domains/{domainName}.....	133
84.1 Contents.....	133
84.2 Request.....	133
84.3 Response.....	133
84.4 See also.....	133
85 Domains-tlds-extparams:GET.....	135
86 Domains-tlds-registryagreements:GET.....	136
87 GET /domains/tlds/registryagreement.....	137
87.1 Contents.....	137
87.2 Request.....	137
87.3 Response.....	137
87.4 Example get registry agreements for com,net,ca.....	137
87.5 Example get registry agreements (supplying invalid tld) for com,net,ca,invalidTld.....	137
87.6 See also.....	138
88 Domains-tlds:GET.....	139
89 GET /domains/tlds.....	140
89.1 Contents.....	140
89.2 Request.....	140
89.3 Response.....	140
89.4 See also.....	140
90 OrderingDomain.....	141
90.1 Service Attributes.....	141
90.2 See also.....	142
91 OrderingTransferDomain.....	143
91.1 Service Attributes.....	144
91.2 See also.....	144

1 Accounts-accountId-domains-domainName:PUT

2 PUT /accounts/{accountId}/domains/{domainName}

Place a domain registration order for an existing account.

2.1 Contents

- 1 PUT /accounts/{accountId}/domains/{domainName}
 - ◆ 1.1 Request
 - ◇ 1.1.1 URI Parameters
 - ◇ 1.1.2 Request Headers
 - ◇ 1.1.3 Request Body
 - 1.1.3.1 Request Parameters
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - ◆ 1.3 Example of ordering .com domain.
 - ◆ 1.4 Example of ordering .us domain.
 - ◆ 1.5 See also

2.2 Request

PUT /accounts/{accountId}/domains/{domainName}

2.2.1 URI Parameters

accountId - string

The account number

domainName - string

The domain name. IE: **example.com** is a valid domain name, whereas **http://www.example.com** or **www.example.com** are not. If the domain name is invalid, or is otherwise not registerable such as for a non-existent or non-supported TLD, an error will be returned. Domain names are case insensitive. The domain name must be an open or suspended product under the authenticated franchise.

2.2.2 Request Headers

Content-Type

Required. Set this header to `application/json; charset=UTF-8`

Authorization

Required.

2.2.3 Request Body

Request Body for registering a domain

```
{
  "domainName": "{domainName}",
  "operation": "register",
  "term": {term},
  "authInfo": "{authInfo}",
  "nameservers": [ "{nameserver}", "{nameserver}" ],
  "extraParams": [{"name": "...", "value": "..."}, ...],
  "contactInfo": {
    "registrant": {
      "type": "registrant",
      "firstName": "{firstName}",
      "lastName": "{lastName}",
      "company": "{company}",
      "street1": "{street1}",
      "city": "{city}",
      "zipOrPostal": "{zipOrPostal}",
      "stateOrProvince": "{stateOrProvince}",
      "countryCode": "{countryCode}",
      "email1": "{email1}",
      "phone1": "{phone1}" },
    "administrative": {
      "type": "administrative",
      ...
    },
    ...
  },
  "billing": {
    "type": "billing",
    ...
  },
  "technical": {
    "type": "technical",
    ...
  }
}
```

2.2.3.1 Request Parameters

domainName - string

The domain name

operation - string

"register" to register a domain

term - integer

the registration term (period) in years

authInfo - *string*
 authentication info, currently not supported

nameservers - *list of strings*
 domain name servers (DNS) for the domain

extraParams - *list of extra params* (optional)
 List of additional parameters required for some TLDs. Each parameter is an object with "name" and "value" fields. For example, .us registration requires the fields "us_nexus" which can be one of the following [C11, C12, C21, C31, C32] - see [What are Domain Name Application Purpose Codes?](#) and "us_purpose" which can be one of [P1, P2, P3, P4, P5] - see [What do the Nexus Codes mean?](#) You can find more information on extra parameters in the [GET /domains/tlds/{tld}/extparams](#) documentation.

company - *string* (optional)
 The company or organization name

firstName - *string*
 The first name of the contact person

lastName - *string*
 The last name of the contact person

street1 - *string*
 The street address

street2 - *string* (optional)
 The second line of the street address (if any)

city - *string*
 The city

countryCode - *string*
 The international 2-letter country code

stateOrProvince - *string*
 The state or province. For US states use the 2-letter state code

zipOrPostal - *string*
 The postal code

email1 - *string*
 The e-mail address of the contact person

phone1 - *string*
 The phone number of the contact person in valid [E.164](#) format, e.g. +1.3209270625

2.3 Response

2.3.1 Status Code

202 Accepted
 The domain registration order has been submitted.

400 Bad Request
 The domain information is invalid.

2.3.2 Response Body

Response Body for registering domain

The response contains the order id and the link for the domain registration order.

```
{
  "orderId": "0000013dfeb0db5aa6c51427000ac81d1e6e110e",
  "links": [
    { "href": "http://coreapi01.ote.chicago.hostway:8100/orders/0000013dfeb0db5aa6c51427000ac81d1e6e110e",
      "rel": "self" } ]
}
```

2.4 Example of ordering .com domain.

PUT /accounts/acct839326/domains/ng-test-fitness-shortcut-20130412-0940.com
 Content-Type: application/json; charset=UTF-8
 Authorization: Basic ...

```
{
  "domainName": "ng-test-fitness-shortcut-20130412-0940.com",
  "operation": "register",
  "term": 1,
  "authInfo": "mysecret",
  "nameservers": [ "ns1.test.com", "ns2.test.com" ],
  "contactInfo": {
    "registrant": {
      "type": "registrant",
      "firstName": "John",
      "lastName": "Doe",
      "company": "ACME Inc.",
      "street1": "123 First Str.",
      "city": "Chicago",
      "zipOrPostal": "34567",
      "stateOrProvince": "IL",
      "countryCode": "US",
      "email1": "john.doe@test.com",
      "phone1": "+1.312.456789098"
    },
    "administrative": {
      "type": "administrative",
      ...
    },
    "billing": {
      "type": "billing",
      ...
    },
    "technical": {
      "type": "technical",
      ...
    }
  }
}
```


Response

202 Accepted

```
{
  "orderId": "0000013dfeb0db5aa6c51427000ac81d1e6e110e",
  "links": [
    { "href": "http://coreapi01.ote.chicago.hostway:8100/orders/0000013dfeb0db5aa6c51427000ac81d1e6e110e",
      "rel": "self" } ]
}
```

2.5 Example of ordering .us domain.

PUT /accounts/acct839326/domains/ng-test-Fitness-shortcut-20130919-0134.us
Content-Type: application/json; charset=UTF-8
Authorization: Basic ...

```
{
  "domainName": "ng-test-Fitness-shortcut-20130919-0134.us",
  "operation": "register",
  "term": 1,
  "authInfo": "mysecret",
  "nameservers": [ "ns1.test.com", "ns2.test.com" ],
  "extraParams": [ { "name": "us_nexus", "value": "C11" }, { "name": "us_purpose", "value": "P1" } ],
  "contactInfo": {
    "registrant": {
      "type": "registrant",
      "firstName": "John",
      "lastName": "Doe",
      "company": "ACME Inc.",
      "street1": "123 First Str.",
      "city": "Chicago",
      "zipOrPostal": "34567",
      "stateOrProvince": "IL",
      "countryCode": "US",
      "email1": "john.doe@test.com",
      "phone1": "+1.312.456789098"
    },
    "administrative": {
      "type": "administrative",
      ...
    },
    "billing": {
      "type": "billing",
      ...
    },
    "technical": {
      "type": "technical",
      ...
    }
  }
}
```

Response

202 Accepted

```
{
  "orderId": "00000141381446d465883bc9007f000000010001",
  "links": [
    { "href": "http://coreapi01.ote.chicago.hostway:8182/orders/00000141381446d465883bc9007f000000010001",
      "rel": "self" } ]
}
```

2.6 See also

- [Domain Name Registration API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

3 Accounts-accountId-transfer-domains-domainNames:PUT

4 PUT /accounts/{accountId}/domains/{domainName}

Place a domain registration order for an existing account.

4.1 Contents

- 1 PUT /accounts/{accountId}/domains/{domainName}
 - ◆ 1.1 Request
 - ◇ 1.1.1 URI Parameters
 - ◇ 1.1.2 Request Headers
 - ◇ 1.1.3 Request Body
 - 1.1.3.1 Request Parameters
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - ◆ 1.3 Example of ordering transfer .com domain.
 - ◆ 1.4 Example of ordering transfer .us domain.
 - ◆ 1.5 See also

4.2 Request

PUT /accounts/{accountId}/domains/{domainName}

4.2.1 URI Parameters

accountId - string

The account number

domainName - string

The domain name. IE: **example.com** is a valid domain name, whereas **http://www.example.com** or **www.example.com** are not. If the domain name is invalid, or is otherwise not registerable such as for a non-existent or non-supported TLD, an error will be returned. Domain names are case insensitive. The domain name must be an open or suspended product under the authenticated franchise.

4.2.2 Request Headers

Content-Type

Required. Set this header to `application/json; charset=UTF-8`

Authorization

Required.

4.2.3 Request Body

Request Body for registering a domain

```
{
  "domainName": "{domainName}",
  "operation": "transfer",
  "term": {term},
  "authInfo": "{authInfo}",
  "nameservers": [ "{nameserver}", "{nameserver}" ],
  "additionalParams": {
    "usNexusCode": "{usNexusCode}",
    "usNexusCategory": "{usNexusCategory}"
  },
  "contactInfo": {
    "registrant": {
      "type": "registrant",
      "firstName": "{firstName}",
      "lastName": "{lastName}",
      "company": "{company}",
      "street1": "{street1}",
      "city": "{city}",
      "zipOrPostal": "{zipOrPostal}",
      "stateOrProvince": "{stateOrProvince}",
      "countryCode": "{countryCode}",
      "email1": "{email1}",
      "phone1": "{phone1}" },
    "administrative": {
      "type": "administrative",
      ...
    }
  },
  "billing": {
    "type": "billing",
    ...
  },
  "technical": {
    "type": "technical",
    ...
  }
}
```

4.2.3.1 Request Parameters

domainName - *string*

The domain name

operation - *string*

"register" to register a domain

term - *integer*
the registration term (period) in years, for transfer operation value must be 1

authInfo - *string*
authentication info, currently not supported

nameservers - *list of strings*
domain name servers (DNS) for the domain

usNexusCode - *string* (optional) (required for .us)
one of the following [P1, P2, P3, P4, P5]

usNexusCategory - *string* (optional) (required for .us)
one of the following [C11, C12, C21, C31, C32]

company - *string* (optional)
The company or organization name

firstName - *string*
The first name of the contact person

lastName - *string*
The last name of the contact person

street1 - *string*
The street address

street2 - *string* (optional)
The second line of the street address (if any)

city - *string*
The city

countryCode - *string*
The international 2-letter country code

stateOrProvince - *string*
The state or province. For US states use the 2-letter state code

zipOrPostal - *string*
The postal code

email1 - *string*
The e-mail address of the contact person

phone1 - *string*
The phone number of the contact person in valid E.164 format, e.g. +1.3209270625

4.3 Response

4.3.1 Status Code

202 Accepted
The domain registration order has been submitted.

400 Bad Request
The domain information is invalid.

4.3.2 Response Body

Response Body for registering domain

The response contains the order id and the link for the domain registration order.

```
{
  "orderId": "0000013dfeb0db5aa6c51427000ac81d1e6e110e",
  "links": [
    { "href": "http://coreapi01.ote.chicago.hostway:8100/orders/0000013dfeb0db5aa6c51427000ac81d1e6e110e",
      "rel": "self" } ]
}
```

4.4 Example of ordering transfer .com domain.

Request

PUT /accounts/acct839326/domains/ng-test-fitness-shortcut-20130412-0940.com
Content-Type: application/json; charset=UTF-8
Authorization: Basic ...

```
{
  "domainName": "ng-test-fitness-shortcut-20130412-0940.com",
  "operation": "transfer",
  "term": 1,
  "authInfo": "mysecret",
  "nameservers": [ "ns1.test.com", "ns2.test.com" ],
  "contactInfo": {
    "registrant": {
      "type": "registrant",
      "firstName": "John",
      "lastName": "Doe",
      "company": "ACME Inc.",
      "street1": "123 First Str.",
      "city": "Chicago",
      "zipOrPostal": "34567",
      "stateOrProvince": "IL",
      "countryCode": "US",
      "email1": "john.doe@test.com",
      "phone1": "+1.312.456789098"
    },
    "administrative": {
      "type": "administrative",
      ...
    },
    "billing": {
      "type": "billing",
      ...
    },
    "technical": {
```

```
        "type": "technical",
        ...
    }
}
```

Response

202 Accepted

```
{
  "orderId": "0000013dfeb0db5aa6c51427000ac81d1e6e110e",
  "links": [
    { "href": "http://coreapi01.ote.chicago.hostway:8100/orders/0000013dfeb0db5aa6c51427000ac81d1e6e110e",
      "rel": "self" }
  ]
}
```

4.5 Example of ordering transfer .us domain.

Request

PUT /accounts/acct839326/domains/ng-test-fitness-shortcut-20130412-0940.us
Content-Type: application/json; charset=UTF-8
Authorization: Basic ...

```
{
  "domainName": "ng-test-fitness-shortcut-20130412-0940.us",
  "operation": "transfer",
  "term": 1,
  "authInfo": "mysecret",
  "nameservers": [ "ns1.test.com", "ns2.test.com" ],
  "additionalParams": { "usNexusCode": "P1", "usNexusCategory": "C11" },
  "contactInfo": {
    "registrant": {
      "type": "registrant",
      "firstName": "John",
      "lastName": "Doe",
      "company": "ACME Inc.",
      "street1": "123 First Str.",
      "city": "Chicago",
      "zipOrPostal": "34567",
      "stateOrProvince": "IL",
      "countryCode": "US",
      "email1": "john.doe@test.com",
      "phone1": "+1.312.456789098"
    },
    "administrative": {
      "type": "administrative",
      ...
    },
    "billing": {
      "type": "billing",
      ...
    },
    "technical": {
      "type": "technical",
      ...
    }
  }
}
```

Response

202 Accepted

```
{
  "orderId": "0000013dfeb0db5aa6c51427000ac81d1e6e110e",
  "links": [
    { "href": "http://coreapi01.ote.chicago.hostway:8100/orders/0000013dfeb0db5aa6c51427000ac81d1e6e110e",
      "rel": "self" }
  ]
}
```

4.6 See also

- [Domain Name Registration API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

5 Accounts-accountNumber-domains-domainName-search:GET

6 GET /accounts/{accountNumber}/domains/{domainName}

Search for domains in client's account. Standard paging parameters can be used. The result is in standard paging format: list of list of results and pages count. You can search by name (with wildcard), expired or expiring (within 45 days of expiration) domains, or specify date range for expiration.

6.1 Contents

- 1 GET /accounts/{accountNumber}/domains/{domainName}
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◇ 1.1.4 Request Body
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - ◆ 1.3 Example search for a domain product.
 - ◆ 1.4 See also

6.2 Request

GET /accounts/{accountNumber}/domains

6.2.1 Parameters

name Domain name or wildcard pattern (optional).
expired true|false (optional)
expiring true|false (optional)
days number of days (optional) to be used in conjunction with expiring or expired. The default value is 45 days, i.e. a domain is considered "expiring" if it expires in 45 days or less.
expiresBefore yyyy-mm-dd (optional)
expiresAfter yyyy-mm-dd (optional)
page page number (optional)
pageSize number of items per page (optional)

6.2.2 URI Parameters

accountNumber client account number
domainName - string
The domain name. IE: **example.com** is a valid domain name, whereas **http://www.example.com** or **www.example.com** are not. If the domain name is invalid, or is otherwise not registerable such as for a non-existent or non-supported TLD, an error will be returned. Domain names are case insensitive. The domain name must be an open or suspended product under the authenticated franchise.

6.2.3 Request Headers

None.

6.2.4 Request Body

None.

6.3 Response

6.3.1 Status Code

200 OK
Returns list of domains.

6.3.2 Response Body

List of list of domains and pages count.

6.4 Example search for a domain product.

Request

GET /accounts/acct839326/domains?page=1&pageSize=10&name=ng-test*&expired=false&expiring=false&expiresAfter=2014-04-11

Response

200 OK
Content-Type: application/json; charset=UTF-8


```
{
  "total_count": 1,
  "list":
  [
    {
      "status": "open",
      "createDate": "2013-02-02",
      "billingCycle": 12,
      "name": "ng-test-fitness-20130202-0605.com",
      "expirationDate": "2018-02-02"
    }
  ],
  "links":
  [
    {
      "href": "http://{host:port}/accounts/acct839326/domains?page=1&pageSize=10&filterName=ng-test-fitness-search-20131010-022746.com&fil",
      "rel": "self"
    }
  ]
}
```

6.5 See also

- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

7 Accounts-accountNumber-domains-transfer-domainName:GET

8 GET /accounts/{accountNumber}/domains/transfers/{domainName}

This call will return detailed information on a domain transfer that is in progress for the given domain name and customer number.

8.1 Contents

- 1 GET /accounts/{accountNumber}/domains/transfers/{domainName}
 - ◆ 1.1 Request
 - ◇ 1.1.1 URI Parameters
 - ◇ 1.1.2 Parameters
 - ◇ 1.1.3 Request Headers
 - ◇ 1.1.4 Request Body
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - ◇ 1.2.3 Parameters
 - ◇ 1.2.4 Status object fields
 - ◆ 1.3 Example of getting transfer status.
 - ◆ 1.4 See also

8.2 Request

GET /accounts/{accountNumber}/domains/transfers/{domainName}

8.2.1 URI Parameters

accountNumber
client account number

Template:DomainTransfer

8.2.2 Parameters

None.

8.2.3 Request Headers

Accept: application/json

8.2.4 Request Body

None.

8.3 Response

8.3.1 Status Code

200 OK
Returns status for transfer.

403 Forbidden

This customer is not authorized to view the transfer. Possibly the transfer belongs to another account.

404 Not Found

No transfer order found for this domain and account number.

500 Internal Server Error

Registry error.

8.3.2 Response Body

Status info for the transfer order.

8.3.3 Parameters

history
List of status objects (see below for the status fields)

status
Current status object with the following fields:

8.3.4 Status object fields

canBeReset
Boolean value that indicates if a transfer is in a state that can be reset. This means the transfer ran into a problem that prevents it from proceeding until corrective actions are taken and the transfer is reset. If a transfer is still in progress it cannot be reset. Nor may a transfer be reset if it has been voided.

status
Status value of the transfer order. Valid statuses are:

Status	Can be Reset	Description
--------	--------------	-------------

0	N	Transfer request created - awaiting fax
1	N	Whols information matches
2	Y	Canceled due to Whols error
3	N	Pending due to domain status
4	Y	Canceled due to domain status
5	N	Transferred and paid successfully
6	Y	Transfer incomplete charge problem
7	N	Frozen due to charge problem
8	Y	NSI rejected transfer
9	N	Awaiting auto verification of transfer request
10	Y	Unable to retrieve current domain contacts from UWWhois
11	N	Auto verification of transfer request initiated
12	N	Awaiting for auto transfer string validation
13	N	Domain awaiting transfer initiation
14	N	Domain transfer initiated and awaiting approval
15	Y	Canceled - cannot obtain domain contacts from UWWhois
16	Y	Canceled - domain contacts did not response to verification e-mails
17	Y	Canceled - domain contacts did not approve transfer of domain
18	N	Canceled - domain validation string is invalid
19	N	Canceled - Whois information provided does not match current registrant
20	N	Canceled - Domain is currently not registered and cannot be transferred
21	N	Canceled - Domain already registered in account & cannot be transferred
22	Y	Canceled - Domain is locked at current registrar
23	Y	Canceled - Transfer already initiated for this domain
24	Y	Canceled - Unable to transfer due to unknown error
25	Y	Canceled - Transfer rejected by losing registrar
26	Y	Canceled - Transfer authorization fax not received
27	Y	Canceled by customer
28	N	Fax received - awaiting registrant verification
29	N	Awaiting manual fax verification
30	N	Canceled - Domain name is invalid or is Invalid for Transfers
31	Y	Canceled - Domain is currently undergoing transfer by another Registrar
32	Y	Canceled - Invalid EPP/authorization key - Please contact current registrar to obtain correct key
33	N	Canceled - Cannot transfer domain from name-only account
34	N	Unable to complete transfer. Transfers must include a change in registrar.
35	N	Transfer request not yet submitted
36	N	Canceled - Account is not authorized to perform domain transfers
37	Y	Canceled - Domain was not retagged or not retagged in time by losing registrar
45	Y	Order cancelled

description

Description of the status.

emailAddress (optional)

E-mail address to which FOA notification was sent

statusChangeDate (optional)

The date when status has changed

authInfo

The current authInfo value being used to start the transfer at the registry. The authInfo can be updated using the reset transfer call. For example if the transfer is blocking with a status of 32, the authInfo needs to be updated, or the losing registrar must set the authInfo to the present value.

8.4 Example of getting transfer status.

Request

```
GET /accounts/acct12345/domains/transfers/test.com
```

Response

```
200 OK
Content-Type: application/json; charset=UTF-8
```

```
{
  "status": {
    "status": 7,
    "emailAddress": "joshbond@usa.net",
    "description": "Auto verification of transfer request initiated",
    "statusChangeDate": "2014-03-25T11:15:11-05:00",
    "authInfo": "auth123$000",
    "canBeReset": false
  },
  "history": [
    {
      "status": 7,
      "emailAddress": "joshbond@usa.net",
      "description": "Auto verification of transfer request initiated",
      "statusChangeDate": "2014-03-25T11:15:11-05:00",
      "authInfo": "auth123$000",
      "canBeReset": false
    }
  ]
}
```

8.5 See also

- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

9 Accounts-accountNumber-domains-transfer-domainName:PUT

10 PUT /accounts/{accountNumber}/domains/transfers/{domainName}

A domain name transfer is a long running process that may run into several problems along they way. If a problem occurs, a domain transfer will block for up to 60 days. The reset call is made to retry the transfer based on some change that will remove the impediment to the transfer's progress. Possible reasons that a transfer will block are:

- Domain status at the registry prevented the transfer: for example the domain was locked, or in redemptionPeriod
- Wrong authInfo was supplied and the registry rejected the attempt
- Domain was registered for less that 60 days at the time the transfer was attempted
- The registrar or customer rejected the transfer request (NACK)
- The current authorized contact of the domain failed to respond to the request to authorize the transfer
- The domain was already pending transfer at the registry

10.1 Contents

- 1 PUT
/accounts/{accountNumber}/domains/transfers/{domainName}
 - ◆ 1.1 Request
 - ◇ 1.1.1 URI Parameters
 - ◇ 1.1.2 Request Headers
 - ◇ 1.1.3 Request Body
 - ◇ 1.1.4 Parameters
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - ◇ 1.2.3 Parameters
 - ◆ 1.3 Example of getting transfer status.
 - ◆ 1.4 See also

10.2 Request

```
PUT /accounts/{accountNumber}/domains/transfers/{domainName}
```

10.2.1 URI Parameters

accountNumber
client account number

Template:DomainTransfer

10.2.2 Request Headers

Content-type: application/json

10.2.3 Request Body

```
{  
  'authInfo': '{authInfo}'  
}
```

10.2.4 Parameters

authInfo - string

The authInfo passed in the request body is optional. If a value is passed it must be a valid authInfo string from 6 to 16 characters in length. If no authInfo is passed the previous value will be used. In the case where the previous value was not accepted by the registry, a new authInfo should be supplied or the existing registrar should update the authInfo to the value that was previously used.

10.3 Response

10.3.1 Status Code

- 202 OK
Domain transfer has been reset
- 404 Bad Request
One of the required parameters accountNumber, domainName or authInfo was missing or invalid.
- 404 Not Found
No pending transfer order found for this domain.
- 403 Forbidden
The domain transfer does not belong to the given customer account.
- 409 Conflict
Domain transfer is pending but is not in a state that can be reset. For example, the transfer was successfully sent to the registry and is awaiting a response from the current registrar. Use the call to get domain transfer status to see if the domain may be reset.
- 500 Internal Server Error
An unknown error occurred.

10.3.2 Response Body

No body.

10.3.3 Parameters

No response parameters.

10.4 Example of getting transfer status.

Request

```
PUT /accounts/acct12345/domains/transfers/test.com
```

```
{  
  'authInfo': 'mySecret'  
}
```

Response

```
204 OK
```

10.5 See also

- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

11 Accounts-accountNumber-domains:GET

12 GET /accounts/{accountNumber}/domains

Search for domains in client's account. Standard paging parameters can be used. The result is in standard paging format: list of list of results and pages count. You can search by name (with wildcard), expired or expiring (within 45 days of expiration) domains, or specify date range for expiration.

12.1 Contents

- 1 GET /accounts/{accountNumber}/domains
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◇ 1.1.4 Request Body
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - ◆ 1.3 Example search for a domain product.
 - ◆ 1.4 Example search for a domain product.
 - ◆ 1.5 See also

12.2 Request

GET /accounts/{accountNumber}/domains

12.2.1 Parameters

filterName
The product name or wildcard pattern (optional).

filterDomainName
The product domain name (optional)

filterExpired
true|false (optional)

filterExpiring
true|false (optional)

filterDays
number of days (optional) to be used in conjunction with expiring or expired. The default value is 45 days, i.e. a domain is considered "expiring" if it expires in 45 days or less.

filterExpiresBefore
yyyy-mm-dd (optional)

filterExpiresAfter
yyyy-mm-dd (optional)

filterStatus
The product status (optional)

filterServiceSpecificationTextKey
Filter by serviceSpecificationTextKey. The filter accepts wildcard search (optional)

page
page number (optional)

pageSize
number of items per page (optional)

12.2.2 URI Parameters

accountNumber
client account number

12.2.3 Request Headers

None.

12.2.4 Request Body

None.

12.3 Response

12.3.1 Status Code

200 OK
Returns list of domains.

12.3.2 Response Body

List of list of domains and pages count.

12.4 Example search for a domain product.

Request

```
GET /accounts/acct839326/domains?page=1&pageSize=10&filterName=ng-test*&filterExpired=false&filterExpiring=false&filterExpiresAfter=2014-04-1
```

Response

```
200 OK
Content-Type: application/json; charset=UTF-8

{
  "total_count": 1,
  "list":
  [
    {
      "status": "open",
      "createDate": "2013-02-02",
      "billingCycle": 12,
      "name": "ng-test-fitness-20130202-0605.com",
      "domainName": "ng-test-fitness-20130202-0605.com",
      "expirationDate": "2018-02-02",
      "serviceSpecification": "http://{host:port}/accounts/acct839326/serviceSpecifications/core.dnr.com",
      "service": "http://{host:port}/accounts/acct839326/products/acct839326-00001asfasgag2abeb7007f00000001000/services/acct839326-000001",
      "serviceSpecificationTextKey": "core.dnr.com"
    }
  ],
  "links":
  [
    {
      "href": "http://{host:port}/accounts/acct839326/domains?page=1&pageSize=10&filterName=ng-test*&filterExpired=false&filterExpiring=false",
      "rel": "self"
    }
  ]
}
```

12.5 Example search for a domain product.

Request

```
GET /accounts/acct839326/domains?filterDomainName=ng-test-fitness-20130202-0605.com
```

Response

```
200 OK
Content-Type: application/json; charset=UTF-8

{
  "total_count": 1,
  "list":
  [
    {
      "status": "open",
      "createDate": "2013-02-02",
      "billingCycle": 12,
      "name": "ng-test-fitness-20130202-0605.com",
      "domainName": "ng-test-fitness-20130202-0605.com",
      "expirationDate": "2018-02-02",
      "serviceSpecification": "http://{host:port}/accounts/acct839326/serviceSpecifications/core.dnr.com",
      "service": "http://{host:port}/accounts/acct839326/products/acct839326-00001asfasgag2abeb7007f00000001000/services/acct839326-000001",
      "serviceSpecificationTextKey": "core.dnr.com"
    }
  ],
  "links":
  [
    {
      "href": "http://{host:port}/accounts/acct839326/domains?filterDomainName=ng-test-fitness-20130202-0605.com",
      "rel": "self"
    }
  ]
}
```

12.6 See also

- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

13 Domain Name Registration

14 Domain Name Registration API

Public API to manage domain name registration records including domain availability check, domain registration, renewal, transfer, whois editing, name server changes, host record creation, locking, additional features, special; registration attributes, domain security, supported TLDs and more.

14.1 Contents

- 1 Domain Name Registration API
 - ◆ 1.1 Top level domains
 - ◇ 1.1.1 Get a list of available tlds
 - ◇ 1.1.2 Get a list of registry agreements
 - ◇ 1.1.3 Get the additional TLD parameters
 - ◆ 1.2 Domain Name Availability Check
 - ◇ 1.2.1 Check availability of a domain
 - ◇ 1.2.2 Name suggestions search (Name Spinner)
 - ◆ 1.3 Domain Name Transfers
 - ◇ 1.3.1 Check transferability of a domain
 - ◇ 1.3.2 Get status of pending transfer
 - ◇ 1.3.3 Reset/Retry domain transfer
 - ◆ 1.4 Ordering domain
 - ◇ 1.4.1 Ordering API
 - ◇ 1.4.2 Order domain shortcut
 - ◆ 1.5 Domain Name Management
 - ◇ 1.5.1 Search for domains in account
 - ◇ 1.5.2 Get domain details
 - ◇ 1.5.3 Lock / Unlock
 - ◇ 1.5.4 Hold / Unhold
 - ◇ 1.5.5 Name Servers
 - ◇ 1.5.6 Whois contacts
 - ◇ 1.5.7 Domain authorization code
 - ◇ 1.5.8 Auto-renewal status
 - ◇ 1.5.9 Domain hosts
 - ◇ 1.5.10 Domain contact verification
 - ◇ 1.5.11 Private registration
 - ◆ 1.6 Renew Domain
 - ◆ 1.7 Domain Pricing
 - ◆ 1.8 Private Registration Availability By TLD
 - ◆ 1.9 See also

[download this selection of articles as a PDF book](#)

14.2 Top level domains

14.2.1 Get a list of available tlds

- [GET /domains/tlds](#) - Get a list of available tlds.

14.2.2 Get a list of registry agreements

- [GET /domains/tlds/registryagreement](#) - Get a list of registry agreements

14.2.3 Get the additional TLD parameters

- [GET /domains/tlds/{tld}/extparams](#) - Get info on the extra parameters (if any) for the tld

14.3 Domain Name Availability Check

14.3.1 Check availability of a domain

- [GET /domains/{domainName}](#) - Check a domain name for availability

14.3.2 Name suggestions search (Name Spinner)

- [GET /domains/{domainName}/namespinner](#) - Get availability results for suggested names

14.4 Domain Name Transfers

14.4.1 Check transferability of a domain

- [GET /domains/{domainName}/transfer](#) - Check a domain name for transferability

14.4.2 Get status of pending transfer

- [GET /accounts/{accountNumber}/domains/transfers/{domainName}](#) - Check transfer status for a domain

14.4.3 Reset/Retry domain transfer

- [PUT /accounts/{accountNumber}/domains/transfers/{domainName}](#) - Reset a domain transfer that is blocked for various reasons

14.5 Ordering domain

14.5.1 Ordering API

- [Ordering Domain](#) - Order domain name using the Ordering API
- [Ordering Transfer Domain](#) - Order transfer domain name using the Ordering API

14.5.2 Order domain shortcut

Register

- [PUT /accounts/{accountId}/domains/{domainName}](#) - Order domain name for an existing account using the "shortcut" command

Transfer

- [PUT /accounts/{accountId}/domains/{domainName}](#) - Order transfer domain name for an existing account using the "shortcut" command

14.6 Domain Name Management

14.6.1 Search for domains in account

- [GET /accounts/{accountNumber}/domains](#) - Search for domains in client's account.

14.6.2 Get domain details

- [GET /domains/{domainName}/details](#) - Get information about the given domain name such as creation date, name servers, lock status and whois info.

14.6.3 Lock / Unlock

- [GET /domains/{domainName}/lock](#) - Get the current lock status of a domain name
- [PUT /domains/{domainName}/lock](#) - Place a lock on a domain name that prevents unauthorized transfer or changes
- [DELETE /domains/{domainName}/lock](#) - Remove a lock on a domain

14.6.4 Hold / Unhold

- [GET /domains/{domainName}/hold](#) - Get the current domain hold status.
- [PUT /domains/{domainName}/hold](#) - Place a hold on a domain name that prevents the domain from resolving
- [DELETE /domains/{domainName}/hold](#) - Remove a hold on a domain

14.6.5 Name Servers

- [GET /domains/{domainName}/nameservers](#) - Get the name servers used to serve the DNS of the domain
- [PUT /domains/{domainName}/nameservers](#) - Change the name servers used to serve the DNS of the domain

14.6.6 Whois contacts

- [GET /domains/{domainName}/{contactType}](#) - Gets the contact info for the specified contact
- [PUT /domains/{domainName}/{contactType}](#) - Changes the contact info of a domain name

14.6.7 Domain authorization code

- [GET /domains/{domainName}/authInfo](#) - Gets the authorization code (authInfo).
- [PUT /domains/{domainName}/authInfo](#) - Sets or generates the authorization info

14.6.8 Auto-renewal status

- [GET /domains/{domainName}/autoRenew](#) - Gets the current auto-renewal status
- [PUT /domains/{domainName}/autoRenew](#) - Sets the auto-renewal status
- [OPTIONS /domains/{domainName}/autoRenew](#) - Check whether the auto-renewal status can be set.

14.6.9 Domain hosts

- [GET /domains/{domainName}/hosts](#) - Gets the current hosts
- [PUT /domains/{domainName}/hosts](#) - Replace the hosts
- [POST /domains/{domainName}/hosts](#) - Add or update hosts
- [DELETE /domains/{domainName}/hosts/{hostName}](#) - Delete a host

14.6.10 Domain contact verification

- [GET /domains/{domainName}/contactverification](#) - Gets the status of the contact verification for {domainName} or Send email contact verification for {domainName}

14.6.11 Private registration

- [GET /domains/{domainName}/privateregistration](#) - Gets the privateregistration info for the domain
- [POST /domains/{domainName}/privateregistration](#) - Update the privateregistration - change forwarding e-mail address, upgrade or downgrade between "full" and "light" version of the privateregistration product.
- [DELETE /domains/{domainName}/privateregistration](#) - Remove the privateregistration

14.7 Renew Domain

- [GET /domains/{domainName}/renewTerms](#) - get available renewal terms and pricing
- [POST /domains/{domainName}/renew](#) - renew a domain for a specific registration period

14.8 Domain Pricing

- [GET /domains/{domainName}/prices](#) - get domain prices
- [GET /domains/{domainName}/prices/register](#) - get domain register prices
- [GET /domains/{domainName}/prices/renew](#) - get domain renew prices
- [GET /domains/{domainName}/prices/transfer](#) - get domain transfer prices

14.9 Private Registration Availability By TLD

- [GET /domains/privateregistration](#) - Is private registration available for tld's

14.10 See also

- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

15 Domains-domainName-authInfo:GET

16 GET /domains/{domainName}/authInfo

Get the current authorization code (authInfo) of a domain. The authorization code is used to transfer a domain.

16.1 Contents

- 1 GET /domains/{domainName}/authInfo
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◇ 1.1.4 Request Body
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - ◆ 1.3 Example of getting the domain authorization code.
 - ◆ 1.4 See also

16.2 Request

GET /domains/{domainName}/authInfo

16.2.1 Parameters

domainName - string

The domain name. IE: **example.com** is a valid domain name, whereas **http://www.example.com** or **www.example.com** are not. If the domain name is invalid, or is otherwise not registerable such as for a non-existent or non-supported TLD, an error will be returned. Domain names are case insensitive. The domain name must be an open or suspended product under the authenticated franchise.

16.2.2 URI Parameters

None.

16.2.3 Request Headers

None.

16.2.4 Request Body

None.

16.3 Response

16.3.1 Status Code

200 OK

Returns the domain authorization code.

400 Bad Request

The domain name is not valid.

403 Forbidden

Domain is not registered in this account and cannot be accessed.

404 Not Found

No domain records found for the given domain name.

16.3.2 Response Body

The authorization code (string)

16.4 Example of getting the domain authorization code.

Request

GET /domains/example.com/authInfo

Response

200 OK

Content-Type: application/json; charset=UTF-8

"6fdcc41f330d7a1-"

16.5 See also

- [Domain Name Registration API](#)
- [Common Features](#)

17 Domains-domainName-authInfo:PUT

18 PUT /domains/{domainName}/authInfo

Set the authorization code (auth info) of a domain.

18.1 Contents

- 1 PUT /domains/{domainName}/authInfo
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◇ 1.1.4 Request Body
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - ◆ 1.3 Example of setting the domain auth info.
 - ◆ 1.4 See also

18.2 Request

PUT /domains/{domainName}/authInfo

18.2.1 Parameters

None.

18.2.2 URI Parameters

domainName - string

The domain name. IE: **example.com** is a valid domain name, whereas **http://www.example.com** or **www.example.com** are not. If the domain name is invalid, or is otherwise not registerable such as for a non-existent or non-supported TLD, an error will be returned. Domain names are case insensitive. The domain name must be an open or suspended product under the authenticated franchise.

18.2.3 Request Headers

Content-Type: application/json

18.2.4 Request Body

The auth info string. If the body is empty and Content-Length is 0, a new random auth info string will be generated.

18.3 Response

18.3.1 Status Code

200 OK

Returned on success

400 Bad Request

The domain name or auth info string is not valid.

403 Forbidden

Domain is not registered in this account and cannot be accessed.

404 Not Found

No domain records found for the given domain name.

18.3.2 Response Body

The new auth info string.

18.4 Example of setting the domain auth info.

Request

```
Content-Length: 0
PUT /domains/example.com/authInfo
```

Response

```
200 OK
"dfouHzxaV=3ytRem"
```

18.5 See also

- [Domain Name Registration API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

19 Domains-domainName-autoRenew:GET

20 GET /domains/{domainName}/autoRenew

Get the current autoRenew status of a domain.

20.1 Contents

- 1 GET /domains/{domainName}/autoRenew
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◇ 1.1.4 Request Body
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - ◆ 1.3 Example of getting the domain autoRenew status.
 - ◆ 1.4 See also

20.2 Request

GET /domains/{domainName}/autoRenew

20.2.1 Parameters

None.

20.2.2 URI Parameters

domainName - string

The domain name. IE: **example.com** is a valid domain name, whereas **http://www.example.com** or **www.example.com** are not. If the domain name is invalid, or is otherwise not registerable such as for a non-existent or non-supported TLD, an error will be returned. Domain names are case insensitive. The domain name must be an open or suspended product under the authenticated franchise.

20.2.3 Request Headers

None.

20.2.4 Request Body

None.

20.3 Response

20.3.1 Status Code

200 OK

Returns the domain autoRenew status.

400 Bad Request

The domain name is not valid.

403 Forbidden

Domain is not registered in this account and cannot be accessed.

404 Not Found

No domain records found for the given domain name.

20.3.2 Response Body

true

Means that domain autoRenew status is set, domain is renewed automatically.

false

Means that domain autoRenew status is not set, domain is not renewed automatically.

20.4 Example of getting the domain autoRenew status.

Request

GET /domains/example.com/autoRenew

Response

200 OK

Content-Type: application/json; charset=UTF-8

true

20.5 See also

- [Domain Name Registration API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

21 Domains-domainName-autoRenew:OPTIONS

22 OPTIONS /domains/{domainName}/autoRenew

Check whether the autoRenew status of a domain can be set.

22.1 Contents

- 1 OPTIONS /domains/{domainName}/autoRenew
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◇ 1.1.4 Request Body
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - ◆ 1.3 Example of setting the domain autoRenew status.
 - ◆ 1.4 See also

22.2 Request

OPTIONS /domains/{domainName}/autoRenew

22.2.1 Parameters

None.

22.2.2 URI Parameters

domainName - string

The domain name. IE: **example.com** is a valid domain name, whereas **http://www.example.com** or **www.example.com** are not. If the domain name is invalid, or is otherwise not registerable such as for a non-existent or non-supported TLD, an error will be returned. Domain names are case insensitive. The domain name must be an open or suspended product under the authenticated franchise.

22.2.3 Request Headers

None.

22.2.4 Request Body

None

22.3 Response

22.3.1 Status Code

204 No content
Returned on success

400 Bad Request
Payment type is not credit card, domain is within 24 hours of expiration or has expired.

22.3.2 Response Body

None.

22.4 Example of setting the domain autoRenew status.

Request

OPTIONS /domains/example.com/autoRenew

Response

204 No content

22.5 See also

- [Domain Name Registration API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

23 Domains-domainName-autoRenew:PUT

24 PUT /domains/{domainName}/autoRenew

Set the autoRenew status of a domain.

24.1 Contents

- 1 PUT /domains/{domainName}/autoRenew
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◇ 1.1.4 Request Body
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - ◆ 1.3 Example of setting the domain autoRenew status.
 - ◆ 1.4 See also

24.2 Request

PUT /domains/{domainName}/autoRenew

24.2.1 Parameters

None.

24.2.2 URI Parameters

domainName - string

The domain name. IE: **example.com** is a valid domain name, whereas **http://www.example.com** or **www.example.com** are not. If the domain name is invalid, or is otherwise not registerable such as for a non-existent or non-supported TLD, an error will be returned. Domain names are case insensitive. The domain name must be an open or suspended product under the authenticated franchise.

24.2.3 Request Headers

None.

24.2.4 Request Body

The autoRenew status value (true or false)

24.3 Response

24.3.1 Status Code

204 No content

Returned on success

400 Bad Request

The domain name is not valid.

403 Forbidden

Domain is not registered in this account and cannot be accessed.

404 Not Found

No domain records found for the given domain name.

24.3.2 Response Body

None.

24.4 Example of setting the domain autoRenew status.

Request

```
PUT /domains/example.com/autoRenew
true
```

Response

204 No content

24.5 See also

- [Domain Name Registration API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

25 Domains-domainName-contactType:GET

26 GET /domains/{domainName}/{contactType}

Get domains contact information for the specified contact type (registrant, administrative, billing or technical). Each domain has all four contacts.

26.1 Contents

- 1 GET /domains/{domainName}/{contactType}
 - ◆ 1.1 Request
 - ◇ 1.1.1 Request Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◇ 1.1.4 Request Body
 - 1.1.4.1 Request Parameters
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - ◇ 1.2.3 Response parameters
 - ◆ 1.3 Example of getting domain contact information.
 - ◆ 1.4 See also

26.2 Request

GET /domains/{domainName}/{contactType}

26.2.1 Request Parameters

None

26.2.2 URI Parameters

domainName - string

The domain name. IE: **example.com** is a valid domain name, whereas **http://www.example.com** or **www.example.com** are not. If the domain name is invalid, or is otherwise not registerable such as for a non-existent or non-supported TLD, an error will be returned. Domain names are case insensitive. The domain name must be an open or suspended product under the authenticated franchise.

contactType - string

The contact type must be one of: registrant, administrative, billing or technical.

26.2.3 Request Headers

Content-Type

Required. Set this header to `application/json; charset=UTF-8`

Cookie

A cookie named **OwnerId** to identify the client (for most browser clients, the browser will do this automatically, depending on the browser configuration). The value of the cookie would come from the response to the create order request in the **Set-Cookie** response header.

26.2.4 Request Body

None

26.2.4.1 Request Parameters

None

26.3 Response

26.3.1 Status Code

200 OK

The account was successfully set.

204 No Content

The account was successfully set.

400 Bad Request

The request body format is invalid or the language or currency is not supported by the franchise.

409 Conflict

Attempt to set currency that is different from currencies of the products or that of the existing account.

26.3.2 Response Body

Response Body for getting contact information

```
{
  "company": "{company}",
  "firstName": "{firstName}",
  "lastName": "{lastName}",
  "street1": "{street1}",
```

```

    "street2": "{street2}",
    "city": "{city}",
    "countryCode": "{countryCode}",
    "stateOrProvince": "{stateOrProvince}",
    "zipOrPostal": "{zipOrPostal}" ,
    "email1": "{email}",
    "phone1": "{phone1}"
}

```

26.3.3 Response parameters

company - *string* (optional)
The company or organization name

firstName - *string*
The first name of the contact person

lastName - *string*
The last name of the contact person

street1 - *string*
The street address

street1 - *string* (optional)
The second line of the street address (if any)

city - *string*
The city

countryCode - *string*
The international 2-letter country code

stateOrProvince - *string*
The state or province. For US states use the 2-letter state code

zipOrPostal - *string*
The postal code

email1 - *string*
The e-mail address of the contact person

phone1 - *string*
The phone number of the contact person in valid [E.164](#) format, e.g. +1.3209270625

26.4 Example of getting domain contact information.

Request

```
GET /domains/ng-test-fitness-20130202-0605.com/registrant
```

Response

```
200 OK
Content-Type: application/json; charset=UTF-8
```

```

{
  "company": "Green Ltd.",
  "firstName": "Barbara",
  "lastName": "Phillips",
  "street1": "Maple Str. 37",
  "city": "San Jose",
  "countryCode": "US",
  "stateOrProvince": "CA",
  "zipOrPostal": "95191" ,
  "email1": "barbara.phillips@test.com",
  "phone1": "+1.3209270625"
}

```

26.5 See also

- [Domain Name Registration API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

27 Domains-domainName-contactType:PUT

28 PUT /domains/{domainName}/{contactType}

Set domains contact information for the specified contact type (registrant, administrative, billing or technical). Each domain has all four contacts.

28.1 Contents

- 1 PUT /domains/{domainName}/{contactType}
 - ◆ 1.1 Request
 - ◇ 1.1.1 Request Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◇ 1.1.4 Request Body
 - 1.1.4.1 Request Parameters
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - ◆ 1.3 Example of setting domain contact information.
 - ◆ 1.4 See also

28.2 Request

PUT /domains/{domainName}/{contactType}

28.2.1 Request Parameters

None

28.2.2 URI Parameters

domainName - string

The domain name. IE: **example.com** is a valid domain name, whereas **http://www.example.com** or **www.example.com** are not. If the domain name is invalid, or is otherwise not registerable such as for a non-existent or non-supported TLD, an error will be returned. Domain names are case insensitive. The domain name must be an open or suspended product under the authenticated franchise.

contactType - string

The contact type must be one of: registrant, administrative, billing or technical.

28.2.3 Request Headers

Content-Type

Required. Set this header to `application/json; charset=UTF-8`

Cookie

A cookie named **OwnerId** to identify the client (for most browser clients, the browser will do this automatically, depending on the browser configuration). The value of the cookie would come from the response to the create order request in the **Set-Cookie** response header.

28.2.4 Request Body

Request Body for setting contact information

```
{
  "company": "{company}",
  "firstName": "{firstName}",
  "lastName": "{lastName}",
  "street1": "{street1}",
  "street2": "{street2}",
  "city": "{city}",
  "countryCode": "{countryCode}",
  "stateOrProvince": "{stateOrProvince}",
  "zipOrPostal": "{zipOrPostal}",
  "email1": "{email}",
  "phone1": "{phone1}"
}
```

28.2.4.1 Request Parameters

company - *string* (optional)

The company or organization name

firstName - *string*

The first name of the contact person

lastName - *string*

The last name of the contact person

street1 - *string*

The street address

street2 - *string* (optional)

The second line of the street address (if any)

city - *string*

The city

countryCode - *string*

The international 2-letter country code

stateOrProvince - *string*

The state or province. For US states use the 2-letter state code

zipOrPostal - *string*
The postal code
email1 - *string*
The e-mail address of the contact person
phone1 - *string*
The phone number of the contact person in valid [E.164](#) format, e.g. +1.3209270625

28.3 Response

28.3.1 Status Code

200 OK
The account was successfully set.
204 No Content
The account was successfully set.
400 Bad Request
The contact information is invalid.
404 Not found
The domain doesn't exist.

28.3.2 Response Body

None.

28.4 Example of setting domain contact information.

Request

```
PUT /domains/ng-test-fitness-20130202-0605.com/registrant
Content-Type: application/json; charset=UTF-8
```

```
{
  "company": "Green Ltd.",
  "firstName": "Barbara",
  "lastName": "Phillips",
  "street1": "Maple Str. 37",
  "city": "San Jose",
  "countryCode": "US",
  "stateOrProvince": "CA",
  "zipOrPostal": "95191",
  "email1": "barbara.phillips@test.com",
  "phone1": "+1.3209270625"
}
```

Response

```
204 (no body)
```

28.5 See also

- [Domain Name Registration API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

29 Domains-domainName-contactverification:GET

30 GET /domains/{domainName}/contactverification

Get the status of the contactverification for {domainName} Or Get will send a verification email for {domainName}

- [This page requires more explanatory detail and examples]*

30.1 Contents

- 1 GET /domains/{domainName}/contactverification
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◇ 1.1.4 Request Body
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - ◇ 1.2.3 Response parameters
 - ◆ 1.3 Example getting contact verification status for a domain.
 - ◆ 1.4 Example sending contact verification for a domain.
 - ◆ 1.5 See also

30.2 Request

GET /domains/{domainName}/contactverification

30.2.1 Parameters

action

The action GET will take.
info | sendVerificationEmail

30.2.2 URI Parameters

domainName - string

The domain name. IE: **example.com** is a valid domain name, whereas **http://www.example.com** or **www.example.com** are not. If the domain name is invalid, or is otherwise not registerable such as for a non-existent or non-supported TLD, an error will be returned. Domain names are case insensitive. The domain name must be an open or suspended product under the authenticated franchise.

30.2.3 Request Headers

None.

30.2.4 Request Body

None.

30.3 Response

30.3.1 Status Code

200 OK

Returns contact verification status when action=info

204 No Content

when action=sendVerificationEmail

30.3.2 Response Body

```
{
  "verificationStatus": "pending"
}
```

30.3.3 Response parameters

verificationStatus - *string*

"Verified", "Cancelled", "Queued, Pending Verification", "Queue Expired", "Pending Suspension", "Suspended"

30.4 Example getting contact verification status for a domain.

Request

GET /domains/example.com/contactverification?action=info

Response

200 OK

Content-Type: application/json; charset=UTF-8

```
{"verificationStatus": "pending"}
```

30.5 Example sending contact verification for a domain.

Request

GET /domains/example.com/contactverification?action=sendVerificationEmail

Response

204 No Content

30.6 See also

- [Domain Name Registration API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

31 Domains-domainName-details:GET

32 GET /domains/{domainName}/details

Returns full details about a domain name such as registrant contact (whois) info, its creation, update and expiration dates, name servers, lock and hold status as well as product URIs.

32.1 Contents

- 1 GET /domains/{domainName}/details
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - 1.2.2.1 Parameters
 - ◆ 1.3 Examples
 - ◇ 1.3.1 Details for a domain name that is locked
 - ◇ 1.3.2 Details for a domain name that is newly registered and has no name servers
 - ◆ 1.4 See also

32.2 Request

GET /domains/{domainName}/details

32.2.1 Parameters

domainName - string

The domain name. IE: **example.com** is a valid domain name, whereas **http://www.example.com** or **www.example.com** are not. If the domain name is invalid, or is otherwise not registerable such as for a non-existent or non-supported TLD, an error will be returned. Domain names are case insensitive. The domain name must be an open or suspended product under the authenticated franchise.

32.2.2 URI Parameters

None.

32.2.3 Request Headers

None.

32.3 Response

32.3.1 Status Code

- 200 OK
Domain info successfully retrieved.
- 202 Accepted
Domain info successfully retrieved, however an action is still pending and the registration or transfer remains incomplete. Domain cannot be updated.
- 412 Precondition Failed
Domain is not registered in this account and cannot be accessed.
- 404 Not Found
No domain records found for the given domain name.
- 400 Bad Request
The domain name is not valid.

32.3.2 Response Body

```
{
  "domainName": "domainpeople.com",
  "authInfo": "0f59fcc1c8badal:",
  "createDate": "2010-03-20T03:23:20-08:00",
  "updateDate": "2013-02-15T05:15:33-08:00",
  "expireDate": "2013-03-20T03:23:20-08:00",
  "nameservers": [
    "a.dns.hostway.net",
    "b.dns.hostway.net"
  ],
  "registrarLock": "true",
  "registrarHold": "false",
  "registryLock": "false",
  "registryHold": "false",
  "udrp": "false",
  "immutable": "false",
  "contactInfo": {
    "registrant": {
      "type": "registrant",
      "company": "DomainPeople, Inc.",
      "lastName": "Domain",
      "firstName": "Admin",
      "city": "Vancouver",
      "countryCode": "CA",
      "street1": "Suite 200",
      "street2": "550 Burrard Street",
      "zipOrPostal": "V6C 2B5",
      "stateOrProvince": "BC",
      "phone1": "+1.6046888946",
```

```

    "email1":"support@domainpeople.com"
  },
  "administrative": {
    "type":"administrative",
    "company":"DomainPeople, Inc.",
    "lastName":"Domain",
    "firstName":"Admin",
    "city":"Vancouver",
    "countryCode":"CA",
    "street1":"Suite 200",
    "street2":"550 Burrard Street",
    "zipOrPostal":"V6C 2B5",
    "stateOrProvince":"BC",
    "phone1":"+1.6046888946",
    "email1":"support@domainpeople.com"
  },
  "billing": {
    "type":"billing",
    "company":"DomainPeople, Inc.",
    "lastName":"Domain",
    "firstName":"Admin",
    "city":"Vancouver",
    "countryCode":"CA",
    "street1":"Suite 200",
    "street2":"550 Burrard Street",
    "zipOrPostal":"V6C 2B5",
    "stateOrProvince":"BC",
    "phone1":"+1.6046888946",
    "email1":"support@domainpeople.com"
  },
  "technical": {
    "type":"technical",
    "company":"DomainPeople, Inc.",
    "lastName":"Domain",
    "firstName":"Admin",
    "city":"Vancouver",
    "countryCode":"CA",
    "street1":"Suite 200",
    "street2":"550 Burrard Street",
    "zipOrPostal":"V6C 2B5",
    "stateOrProvince":"BC",
    "phone2":"",
    "phone1":"+1.6046888946",
    "email1":"support@domainpeople.com"
  }
}
}
}
}

```

32.3.2.1 Parameters

domainName - *string*

The domain name in lowercase.

authInfo - *string*

the authInfo for the domain. This is the password that is used to transfer the domain from one registrar to another. authInfo is a 6-16 character string that may contain characters a-zA-Z0-9`~=-_+!@#%&*'()[]{}|\,.; For example:

- aD3glp-@bQl-7

createDate - *datetime*

Creation date of the domain at the registry.

Datetime formats follow the standard 'yyyy-mm-ddTHH:MM:ss+0:00'

updateDate - *datetime*

Last update date of the domain at the registry.

Datetime formats follow the standard 'yyyy-mm-ddTHH:MM:ss+0:00'

expireDate - *datetime*

The actual expiration date of the domain at the registry. Note that in some cases the expiration date of a domain that is displayed in the 'whois' record may indicate the date following a registry autorenewal. This whois expiration date may be inaccurate. The actual expiration date must be extended with an explicit renewal or the domain may be deleted. Do not rely on the whois expiration date.

Datetime formats follow the standard 'yyyy-mm-ddTHH:MM:ss+0:00'

nameservers - *list*

An array of 0 to 13 unique nameservers that are assigned to the domain in listed order of assignment.

registrarLock - *boolean*

Indicates if the domain is locked at the registry. The registrarLock corresponds to a whois status of 'clientTransferProhibited'. If a domain is locked, it cannot be transferred to another registrar.

A domain may be renewed in this state

registrarHold - *boolean*

Indicates if the domain is on hold at the registry. If a domain is on hold then it will not resolve as the name servers are withheld from the registry DNS zone files. A hold status corresponds to a 'clientHold' whois status. A domain may be placed on hold if it has expired, or the customer has explicitly requested the product to be closed, or by court order.

A domain may be renewed in this state

registryLock - *boolean*

Indicates if the domain is locked at the registry. A registryLock status corresponds to 'registryTransferProhibited' whois status. A domain that is locked by the registry cannot be transferred to another registrar. A registryLock cannot be removed automatically. A domain may be on registryLock if that has been requested by the client for security reasons, or by court order to the registry.

A domain may be renewed in this state

registryHold - *boolean*

A rareley occurring state where the registry has placed the domain on hold by court order. The domain will not resolve while in this status. The court proceedings will determine the outcome of the domain.

A domain **cannot be renewed** in this state

udrp - *boolean*

If a domain has UDRP status set it cannot be unlocked, or modified in any way. A domain may be placed in this state in the case of a Uniform Dispute Resolution Proceeding (UDRP) filing.
A domain may be renewed in this state

immutable - *boolean*

A domain is immutable when it cannot be unlocked, or modified in any way. A domain may be placed in this state by client request for security purposes, or by court order in the case of a Uniform Dispute Resolution Proceeding (UDRP) filing.
A domain may be renewed in this state

contactInfo - *set*

an set of contact objects with a name and a **type** of 'registrant', 'administrative', 'billing', or 'technical'

- *company - string*
Contact's company name (optional). In no company name is supplied, the contact is considered to be an individual.
- *firstName - string*
Contact's first name.
- *lastName - string*
Contact's last name.
- *street1 - string*
Contact's street address.
- *street2 - string*
Contact's street address second line (optional).
- *city - string*
Contact's city.
- *stateOrProvince - string*
Contact's state or province string or two character code for US and CA.
- *zipOrPostal - string*
Contact's postal or ZIP code.
- *countryCode - string*
Contact's 2 character country code per ISO 3166-1 alpha-2[1].
- *phone1 - string*
Contact's phone number in ITU-T E.164[2] format.
- *fax - string*
Contact's fax number in ITU-T E.164[3] format (optional).
- *email1 - string*
Contact's primary e-mail address.

links - *list*

[Hypermedia](#) for this resource.

32.4 Examples

32.4.1 Details for a domain name that is locked

Request

```
GET /domains/example.org/details
```

Response

```
{
  "domainName": "example.org",
  "authInfo": "0F59fcc1c8badal:",
  "createDate": "1993-03-20T03:23:20-08:00",
  "updateDate": "2013-02-15T05:15:33-08:00",
  "expireDate": "2023-03-20T03:23:20-08:00",
  "nameservers": [
    "a.dns.hostway.net",
    "b.dns.hostway.net"
  ],
  "registrarLock": "true",
  "registrarHold": "false",
  "registryLock": "false",
  "registryHold": "false",
  "udrp": "false",
  "immutable": "false",
  "contactInfo": {
    "registrant": {
      "type": "registrant",
      "company": "DomainPeople, Inc.",
      "lastName": "Domain",
      "firstName": "Admin",
      "city": "Vancouver",
      "countryCode": "CA",
      "street1": "Suite 200",
      "street2": "550 Burrard Street",
      "zipOrPostal": "V6C 2B5",
      "stateOrProvince": "BC",
      "phone1": "+1.6046888946",
      "email1": "support@domainpeople.com"
    },
    "administrative": {
```



```

    "type": "administrative",
    "company": "DomainPeople, Inc.",
    "lastName": "Domain",
    "firstName": "Admin",
    "city": "Vancouver",
    "countryCode": "CA",
    "street1": "Suite 200",
    "street2": "550 Burrard Street",
    "zipOrPostal": "V6C 2B5",
    "stateOrProvince": "BC",
    "phone1": "+1.6046888946",
    "email1": "support@domainpeople.com"
  },
  "billing": {
    "type": "billing",
    "company": "DomainPeople, Inc.",
    "lastName": "Domain",
    "firstName": "Admin",
    "city": "Vancouver",
    "countryCode": "CA",
    "street1": "Suite 200",
    "street2": "550 Burrard Street",
    "zipOrPostal": "V6C 2B5",
    "stateOrProvince": "BC",
    "phone1": "+1.6046888946",
    "email1": "support@domainpeople.com"
  },
  "technical": {
    "type": "technical",
    "company": "DomainPeople, Inc.",
    "lastName": "Domain",
    "firstName": "Admin",
    "city": "Vancouver",
    "countryCode": "CA",
    "street1": "Suite 200",
    "street2": "550 Burrard Street",
    "zipOrPostal": "V6C 2B5",
    "stateOrProvince": "BC",
    "phone2": "",
    "phone1": "+1.6046888946",
    "email1": "support@domainpeople.com"
  }
}
}
}

```

32.4.2 Details for a domain name that is newly registered and has no name servers

Request

GET /domains/example.com/details

Response

```

{
  "domainName": "example.com",
  "authInfo": "0F59FCC1C8BAD1:",
  "createDate": "2013-03-20T23:39:20-08:00",
  "updateDate": "2013-03-21T05:15:33-08:00",
  "expireDate": "2014-03-20T23:39:20-08:00",
  "nameservers": [],
  "registrarLock": "false",
  "registrarHold": "false",
  "registryLock": "false",
  "registryHold": "false",
  "udrp": "false",
  "immutable": "false",
  "contactInfo": {
    "registrant": {
      "type": "registrant",
      "company": "DomainPeople, Inc.",
      "lastName": "Domain",
      "firstName": "Admin",
      "city": "Vancouver",
      "countryCode": "CA",
      "street1": "Suite 200",
      "street2": "550 Burrard Street",
      "zipOrPostal": "V6C 2B5",
      "stateOrProvince": "BC",
      "phone1": "+1.6046888946",
      "email1": "support@domainpeople.com"
    },
    "administrative": {
      "type": "administrative",
      "company": "DomainPeople, Inc.",
      "lastName": "Domain",
      "firstName": "Admin",
      "city": "Vancouver",
      "countryCode": "CA",
      "street1": "Suite 200",
      "street2": "550 Burrard Street",
      "zipOrPostal": "V6C 2B5",
      "stateOrProvince": "BC",
      "phone1": "+1.6046888946",
      "email1": "support@domainpeople.com"
    },
    "billing": {
      "type": "billing",
      "company": "DomainPeople, Inc.",
      "lastName": "Domain",
      "firstName": "Admin",
      "city": "Vancouver",
      "countryCode": "CA",
      "street1": "Suite 200",

```

```
    "street2": "550 Burrard Street",
    "zipOrPostal": "V6C 2B5",
    "stateOrProvince": "BC",
    "phone1": "+1.6046888946",
    "email1": "support@domainpeople.com"
  },
  "technical": {
    "type": "technical",
    "company": "DomainPeople, Inc.",
    "lastName": "Domain",
    "firstName": "Admin",
    "city": "Vancouver",
    "countryCode": "CA",
    "street1": "Suite 200",
    "street2": "550 Burrard Street",
    "zipOrPostal": "V6C 2B5",
    "stateOrProvince": "BC",
    "phone2": "",
    "phone1": "+1.6046888946",
    "email1": "support@domainpeople.com"
  }
}
}
```

32.5 See also

- [Domain Name Registration API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

33 Domains-domainName-hold:DELETE

34 DELETE /domains/{domainName}/hold

Delete a domain name hold status. This corresponds to a status of 'clientHold' at the registry. Deleting a hold status removes the clientHold status from a domain. This means the DNS records for the domain will be re-published in the registry name servers and the domain should start resolving. Removal of a hold status may fail if there is another status that prevents the operation such as 'pendingTransfer', 'clientUpdateProhibited' or 'serverUpdateProhibited'.

34.1 Contents

- 1 DELETE /domains/{domainName}/hold
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◇ 1.1.4 Request Body
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - ◆ 1.3 Example remove a hold from a domain.
 - ◆ 1.4 See also

34.2 Request

DELETE /domains/{domainName}/hold

34.2.1 Parameters

domainName - string

The valid domain name to unhold including only the labels that are appropriate for a valid domain name. IE: **example.com** is a valid domain name, whereas **http://www.example.com** or **www.example.com** are not. If the domain name is invalid, or is otherwise not registered, an error will be returned. Domain names are case insensitive. The domain name must be an open or suspended product under the authenticated franchise.

34.2.2 URI Parameters

None.

34.2.3 Request Headers

None.

34.2.4 Request Body

None

34.3 Response

34.3.1 Status Code

204 OK (No Content)

Domain lock hold successfully removed.

400 Bad Request

The domain name is not valid.

403 Forbidden

Domain is not registered in this account and cannot be accessed.

404 Not Found

No domain records found for the given domain name.

34.3.2 Response Body

None

34.4 Example remove a hold from a domain.

Request

DELETE /domains/example.com/hold

Response

204 No Content

34.5 See also

- [Domain Name Registration API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

35 Domains-domainName-hold:GET

36 GET /domains/{domainName}/hold

Get the current hold status of a domain. This corresponds to a status of 'clientHold' at the registry. A domain name with this status is removed from the registry zone files and will cease resolving but may be transferred to another registrar, may be renewed and it may be modified; such as changes to the name servers or contact info.

36.1 Contents

- 1 GET /domains/{domainName}/hold
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◇ 1.1.4 Request Body
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - ◆ 1.3 Example of getting the domain hold status.
 - ◆ 1.4 See also

36.2 Request

GET /domains/{domainName}/hold

36.2.1 Parameters

domainName - string

The valid domain name. IE: **example.com** is a valid domain name , whereas **http://www.example.com** or **www.example.com** are not. If the domain name is invalid, or is otherwise not registerable such as for a non-existent or non-supported TLD, an error will be returned. Domain names are case insensitive. The domain name must be an open or suspended product under the authenticated franchise.

36.2.2 URI Parameters

None.

36.2.3 Request Headers

None.

36.2.4 Request Body

None.

36.3 Response

36.3.1 Status Code

200 OK

Returns the domain hold status.

400 Bad Request

The domain name is not valid.

403 Forbidden

Domain is not registered in this account and cannot be accessed.

404 Not Found

No domain records found for the given domain name.

36.3.2 Response Body

true

Means that domain hold status is set, domain is on-hold.

false

Means that domain hold status is not set, domain is not on-hold.

36.4 Example of getting the domain hold status.

Request

GET /domains/example.com/hold

Response

200 OK

Content-Type: application/json; charset=UTF-8

true

36.5 See also

- [Domain Name Registration API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

37 Domains-domainName-hold:PUT

38 PUT /domains/{domainName}/hold

Put a domain name on hold status. This corresponds to a status of 'clientHold' at the registry. A domain name with this status is removed from the registry zone files and will cease resolving but may be transferred to another registrar, may be renewed and it may be modified; such as changes to the name servers or contact info.

38.1 Contents

- 1 PUT
/domains/{domainName}/hold
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◇ 1.1.4 Request Body
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - ◆ 1.3 Example remove hold on a domain.
 - ◆ 1.4 See also

38.2 Request

PUT /domains/{domainName}/hold

38.2.1 Parameters

domainName - string

The valid domain name to put on hold including only the labels that are appropriate for a valid domain name. IE: **example.com** is a valid domain name , whereas **http://www.example.com** or **www.example.com** are not. If the domain name is invalid, or is otherwise not registered, an error will be returned. Domain names are case insensitive. The domain name must be an open or suspended product under the authenticated franchise.

38.2.2 URI Parameters

None.

38.2.3 Request Headers

None.

38.2.4 Request Body

true

Set the body to 'true' in order to add the hold.

false

Set the body to 'false' in order to have the hold removed.

38.3 Response

38.3.1 Status Code

204 OK (No Content)

Domain hold status applied successfully.

400 Bad Request

The domain name is not valid.

403 Forbidden

Domain is not registered in this account and cannot be accessed.

404 Not Found

No domain records found for the given domain name.

38.3.2 Response Body

None

38.4 Example remove hold on a domain.

Request

```
PUT /domains/example.com/hold
```

```
false
```

Response

```
204 No Content
```

38.5 See also

- [Domain Name Registration API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

39 Domains-domainName-hosts:DELETE

40 DELETE /domains/{domainName}/hosts/{hostName}

Delete a single host from a domain. To delete multiple hosts you can use the "replace hosts" feature.

40.1 Contents

- 1 DELETE /domains/{domainName}/hosts/{hostName}
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◇ 1.1.4 Request Body
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - ◆ 1.3 Example of deleting a host
 - ◆ 1.4 See also

40.2 Request

DELETE /domains/{domainName}/hosts/{hostName}

40.2.1 Parameters

None.

40.2.2 URI Parameters

domainName - string

The domain name. IE: **example.com** is a valid domain name, whereas **http://www.example.com** or **www.example.com** are not. If the domain name is invalid, or is otherwise not registerable such as for a non-existent or non-supported TLD, an error will be returned. Domain names are case insensitive. The domain name must be an open or suspended product under the authenticated franchise.

hostName - string

The name of the host to delete.

40.2.3 Request Headers

Content-Type: application/json

40.2.4 Request Body

None.

40.3 Response

40.3.1 Status Code

204 No Content

Returned on success

404 Not Found

No domain records found for the given domain name.

500 Internal Server Error

An error has occurred with the registry,

40.3.2 Response Body

40.4 Example of deleting a host

Request

```
DELETE /domains/example.com/hosts/ns1.example.com
```

Response

```
204 No Content
```

40.5 See also

- [Domain Name Registration API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

41 Domains-domainName-hosts:GET

42 GET /domains/{domainName}/hosts

Get the current hosts in a domain.

42.1 Contents

- 1 GET /domains/{domainName}/hosts
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◇ 1.1.4 Request Body
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - ◆ 1.3 Example of getting the domain authorization code.
 - ◆ 1.4 See also

42.2 Request

GET /domains/{domainName}/hosts

42.2.1 Parameters

None.

42.2.2 URI Parameters

domainName - string

The domain name. IE: **example.com** is a valid domain name, whereas **http://www.example.com** or **www.example.com** are not. If the domain name is invalid, or is otherwise not registerable such as for a non-existent or non-supported TLD, an error will be returned. Domain names are case insensitive. The domain name must be an open or suspended product under the authenticated franchise.

42.2.3 Request Headers

None.

42.2.4 Request Body

None.

42.3 Response

42.3.1 Status Code

200 OK
Returns the hosts

404 Not Found
No domain records found for the given domain name.

500 Internal Server Error
An error has occurred with the registry.

42.3.2 Response Body

The hosts in a domain.

42.4 Example of getting the domain authorization code.

Request

GET /domains/example.com/hosts

Response

200 OK
Content-Type: application/json; charset=UTF-8

```
{ "hosts": [ {  
  "name": "ns1.ng-test-fitness-20130202-0605.com",  
  "address": "133.45.67.1"}, {  
  "name": "ns2.ng-test-fitness-20130202-0605.com",  
  "address": "133.45.67.2"} ] }
```

42.5 See also

- [Domain Name Registration API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

43 Domains-domainName-hosts:POST

44 POST /domains/{domainName}/hosts

Add or update the hosts list of a domain. If a host doesn't exist it will be created, otherwise its IP address will be updated.

44.1 Contents

- 1 POST /domains/{domainName}/hosts
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◇ 1.1.4 Request Body
 - 1.1.4.1 Parameters
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - ◆ 1.3 Example of updating hosts
 - ◆ 1.4 See also

44.2 Request

POST /domains/{domainName}/hosts

44.2.1 Parameters

None.

44.2.2 URI Parameters

domainName - string

The domain name. IE: **example.com** is a valid domain name, whereas **http://www.example.com** or **www.example.com** are not. If the domain name is invalid, or is otherwise not registerable such as for a non-existent or non-supported TLD, an error will be returned. Domain names are case insensitive. The domain name must be an open or suspended product under the authenticated franchise.

44.2.3 Request Headers

Content-Type: application/json

44.2.4 Request Body

The hosts for the domain.

44.2.4.1 Parameters

hosts - *list*
The list of hosts

name - *string*
The host name

address - *string*
The IP address of the host.

44.3 Response

44.3.1 Status Code

204 No Content
Returned on success

404 Not Found
No domain records found for the given domain name.

500 Internal Server Error
An error has occurred with the registry,

44.3.2 Response Body

The hosts to set for the domain.

44.4 Example of updating hosts

Request

```
Content-Length: 0
POST /domains/example.com/hosts
```

```
{"hosts": [{
  "name": "ns2.ng-test-fitness-20130202-0605.com",
  "address": "133.45.67.2"}]}
```

Response

```
204 No Content
```

44.5 See also

- [Domain Name Registration API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

45 Domains-domainName-hosts:PUT

46 PUT /domains/{domainName}/hosts

Replaces the hosts list of a domain. Host names must be valid names for the domain, e.g. ns1.example.com is valid host for the domain example.com. Creating hosts will add the proper addresses to the root DNS servers of the registry,

46.1 Contents

- 1 PUT /domains/{domainName}/hosts
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◇ 1.1.4 Request Body
 - 1.1.4.1 Parameters
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - ◆ 1.3 Example of replacing the hosts
 - ◆ 1.4 See also

46.2 Request

PUT /domains/{domainName}/hosts

46.2.1 Parameters

None.

46.2.2 URI Parameters

domainName - string

The domain name. IE: **example.com** is a valid domain name, whereas **http://www.example.com** or **www.example.com** are not. If the domain name is invalid, or is otherwise not registerable such as for a non-existent or non-supported TLD, an error will be returned. Domain names are case insensitive. The domain name must be an open or suspended product under the authenticated franchise.

46.2.3 Request Headers

Content-Type: application/json

46.2.4 Request Body

The hosts for the domain.

46.2.4.1 Parameters

hosts - *list*
The list of hosts

name - *string*
The host name

address - *string*
The IP address of the host.

46.3 Response

46.3.1 Status Code

204 No Content
Returned on success

404 Not Found
No domain records found for the given domain name.

500 Internal Server Error
An error has occurred with the registry,

46.3.2 Response Body

The hosts to set for the domain.

46.4 Example of replacing the hosts

Request

```
Content-Length: 0
PUT /domains/example.com/hosts

{"hosts": [{
  "name": "ns1.ng-test-fitness-20130202-0605.com",
  "address": "133.45.67.1"}, {
  "name": "ns2.ng-test-fitness-20130202-0605.com",
  "address": "133.45.67.2"}]}
```

Response

```
204 No Content
```

46.5 See also

- [Domain Name Registration API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

47 Domains-domainName-lock:DELETE

48 DELETE /domains/{domainName}/lock

Delete a domain name lock status. This corresponds to a status of 'clientTransferProhibited' at the registry. Deleting a lock status removes the clientTransferProhibited status from a domain. This means the domain name may be transferred to another registrar, providing no other status is assigned to the domain that prevents a transfer such as serverTransferProhibited, pendingTransfer, redemptionPeriod, or pendingDelete.

48.1 Contents

- 1 DELETE
/domains/{domainName}/lock
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◇ 1.1.4 Request Body
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - ◆ 1.3 Example unlock domain.
 - ◆ 1.4 See also

48.2 Request

DELETE /domains/{domainName}/lock

48.2.1 Parameters

domainName - string

The valid domain name to unlock including only the labels that are appropriate for a valid domain name. IE: **example.com** is a valid domain name , whereas **http://www.example.com** or **www.example.com** are not. If the domain name is invalid, or is otherwise not registered, an error will be returned. Domain names are case insensitive. The domain name must be an open or suspended product under the authenticated franchise.

48.2.2 URI Parameters

None.

48.2.3 Request Headers

None.

48.2.4 Request Body

None

48.3 Response

48.3.1 Status Code

204 OK (No Content)

Domain lock status applied successfully.

400 Bad Request

The domain name is not valid.

403 Forbidden

Domain is not registered in this account and cannot be accessed.

404 Not Found

No domain records found for the given domain name.

48.3.2 Response Body

None

48.4 Example unlock domain.

Request

DELETE /domains/example.com/lock

Response

48.5 See also

- [Domain Name Registration API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

49 Domains-domainName-lock:GET

50 GET /domains/{domainName}/lock

Get the current lock status of a domain. This corresponds to a status of 'clientTransferProhibited' at the registry. A domain name with this status cannot be transferred to another registrar but may be renewed and it may be modified; such as changes to the name servers or contact info.

50.1 Contents

- 1 GET /domains/{domainName}/lock
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◇ 1.1.4 Request Body
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - ◆ 1.3 Example of getting the domain lock status.
 - ◆ 1.4 See also

50.2 Request

GET /domains/{domainName}/lock

50.2.1 Parameters

domainName - string

The domain name. IE: **example.com** is a valid domain name, whereas **http://www.example.com** or **www.example.com** are not. If the domain name is invalid, or is otherwise not registerable such as for a non-existent or non-supported TLD, an error will be returned. Domain names are case insensitive. The domain name must be an open or suspended product under the authenticated franchise.

50.2.2 URI Parameters

None.

50.2.3 Request Headers

None.

50.2.4 Request Body

None.

50.3 Response

50.3.1 Status Code

200 OK

Returns the domain lock status.

400 Bad Request

The domain name is not valid.

403 Forbidden

Domain is not registered in this account and cannot be accessed.

404 Not Found

No domain records found for the given domain name.

50.3.2 Response Body

true

Means that domain lock status is set, domain is locked.

false

Means that domain lock status is not set, domain is unlocked.

50.4 Example of getting the domain lock status.

Request

GET /domains/example.com/lock

Response

200 OK

Content-Type: application/json; charset=UTF-8

true

50.5 See also

- [Domain Name Registration API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

51 Domains-domainName-lock:PUT

52 PUT /domains/{domainName}/lock

Put a domain name on lock status. This corresponds to a status of 'clientTransferProhibited' at the registry. A domain name with this status cannot be transferred to another registrar but may be renewed and it may be modified; such as changes to the name servers or contact info.

52.1 Contents

- 1 PUT /domains/{domainName}/lock
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 URI
 - ◇ 1.1.3 Parameters
 - ◇ 1.1.3 Request Headers
 - ◇ 1.1.4 Request Body
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - ◆ 1.3 Example lock domain.
 - ◆ 1.4 See also

52.2 Request

PUT /domains/{domainName}/lock

52.2.1 Parameters

domainName - string

The valid domain name to retrieve including only the labels that are appropriate for a valid domain name. IE: **example.com** is a valid domain name , whereas **http://www.example.com** or **www.example.com** are not. If the domain name is invalid, or is otherwise not registerable such as for a non-existent or non-supported TLD, an error will be returned. Domain names are case insensitive. The domain name must be an open or suspended product under the authenticated franchise.

52.2.2 URI Parameters

None.

52.2.3 Request Headers

None.

52.2.4 Request Body

true

Set the body to 'true' in order to add the lock.

false

Set the body to false in order to have the lock removed.

52.3 Response

52.3.1 Status Code

204 OK (No Content)

Domain lock status applied successfully.

400 Bad Request

The domain name is not valid.

403 Forbidden

Domain is not registered in this account and cannot be accessed.

404 Not Found

No domain records found for the given domain name.

52.3.2 Response Body

None

52.4 Example lock domain.

Request

```
PUT /domains/example.com/lock
true
```

Response

```
204 No Content
```

52.5 See also

- [Domain Name Registration API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

53 Domains-domainName-nameservers:GET

54 GET /domains/{domainName}/nameservers

Get the current name servers used to serve the Domain Name System (DNS) records for the domain. Name servers are a set of 0 to 13 distinct hostnames that may be assigned to a domain. Once the name servers are modified using this command, the changes are sent immediately to the registry. However these changes may take several hours to propagate depending on the individual zone files and possible registry zone file publication delays.

54.1 Contents

- 1 GET /domains/{domainName}/nameservers
 - ◆ 1.1 Request
 - ◇ 1.1.1 URI Parameters
 - ◇ 1.1.2 Request Headers
 - ◇ 1.1.3 Request Body
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - ◆ 1.3 Example getting nameservers on a domain.
 - ◆ 1.4 See also

54.2 Request

PUT /domains/{domainName}/nameservers

54.2.1 URI Parameters

domainName - string

The domain name. IE: **example.com** is a valid domain name, whereas **http://www.example.com** or **www.example.com** are not. If the domain name is invalid, or is otherwise not registerable such as for a non-existent or non-supported TLD, an error will be returned. Domain names are case insensitive. The domain name must be an open or suspended product under the authenticated franchise.

54.2.2 Request Headers

None.

54.2.3 Request Body

None

54.3 Response

54.3.1 Status Code

200 OK
Domain nameservers successfully updated

400 Bad Request
The domain name, or one of the given hostnames is not valid or is not a registered name server at the registry. The request was not completed even if one or more of the name servers is considered valid.

403 Forbidden
Domain is not registered in this account and cannot be accessed.

404 Not Found
No domain records found for the given domain name.

54.3.2 Response Body

Name server array

An array (list) of name server host names. Name servers are assigned in the order listed.
Example: ["a.dns.hostway.net", "b.dns.hostway.net"]

54.4 Example getting nameservers on a domain.

Request

GET /domains/example.com/nameservers

Response

200 OK
Content-Type: application/json; charset=UTF-8
["ns0.example.com", "ns1.example.com"]

54.5 See also

- [Domain Name Registration API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

55 Domains-domainName-nameservers:PUT

56 PUT /domains/{domainName}/nameservers

Set (overwrite) the name servers used to serve the Domain Name System (DNS) records for the domain. Name servers are a set of 0 to 13 distinct hostnames that may be assigned to a domain. Once the name servers are modified using this command, the changes are sent immediately to the registry. However these changes may take several hours to propagate depending on the individual zone files and possible registry zone file publication delays.

56.1 Contents

- 1 PUT /domains/{domainName}/nameservers
 - ◆ 1.1 Request
 - ◇ 1.1.1 URI Parameters
 - ◇ 1.1.2 Request Headers
 - ◇ 1.1.3 Request Body
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - ◆ 1.3 Example updating nameservers on a domain.
 - ◆ 1.4 Example updating nameservers using one or more name server that is not registered.
 - ◆ 1.5 See also

56.2 Request

```
PUT /domains/{domainName}/nameservers
```

56.2.1 URI Parameters

domainName - string

The domain name. IE: **example.com** is a valid domain name, whereas **http://www.example.com** or **www.example.com** are not. If the domain name is invalid, or is otherwise not registerable such as for a non-existent or non-supported TLD, an error will be returned. Domain names are case insensitive. The domain name must be an open or suspended product under the authenticated franchise.

56.2.2 Request Headers

None.

56.2.3 Request Body

Name server array

An array (list) of name server host names. Name servers are assigned in the order listed.
Example: ["a.dns.hostway.net", "b.dns.hostway.net"]

56.3 Response

56.3.1 Status Code

204 OK (No Content)

Domain nameservers successfully updated

400 Bad Request

The domain name, or one of the given hostnames is not valid or is not a registered name server at the registry.
The request was not completed even if one or more of the name servers is considered valid.

403 Forbidden

Domain is not registered in this account and cannot be accessed.

404 Not Found

No domain records found for the given domain name.

56.3.2 Response Body

None

56.4 Example updating nameservers on a domain.

Request

```
PUT /domains/example.com/nameservers
```

```
["ns0.example.com", "ns1.example.com", "ns2.example.com", "ns3.example.com", "ns4.example.com", "ns5.example.com", "ns6.example.com"]
```

Response

```
204 No Content
```

56.5 Example updating nameservers using one or more name server that is not registered.

Request

```
PUT /domains/example.com/nameservers
```

```
["not-registered.example.com", "ns1.example.com", "ns2.example.com", "ns3.example.com", "ns4.example.com", "ns5.example.com", "ns6.example.com"]
```

Response

400 Bad Request

"not-registered.example.com" is not a registered name server and cannot be used to update the domain.

56.6 See also

- [Domain Name Registration API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

57 Domains-domainName-namespinner:GET

58 GET /domains/{domainName}/namespinner

The NameSpinner command generates domain name suggestions that are similar to the given domain. The results include information on the availability of each suggested domain in different Top Level Domains (TLD). Results can be adjusted by TLD, the number of results to return, the inclusion of hyphens to separate words, and the use of related terms or phrases. The name spinner can be used to suggest alternative choices for domains when the customers preferred choices are all taken. It is therefore useful to use in conjunction with the standard domain availability check command.

58.1 Contents

- 1 GET /domains/{domainName}/namespinner
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - ◇ 1.2.3 Response parameters
 - ◆ 1.3 Example Name Spinner
 - ◇ 1.3.1 Request
 - ◇ 1.3.2 Response
 - ◆ 1.4 NameSpinner Sample Code
 - ◆ 1.5 See also

58.2 Request

GET /domains/{domainName}/namespinner

58.2.1 Parameters

domainName

Required, the domain name, word or search terms for which to get suggestions. The input must include only alphanumeric characters, the 'dot' character, hyphens and spaces. For example the following are valid domainName values "Check my domain", "checkmydomain.com", "check-my-domain". The following are not valid "<http://www.checkmydomain.com>", "www.checkmydomain.com"

58.2.2 URI Parameters

maxResults - *int*

Optional, if specified controls the maximum number of suggestions per TLD. The value must be between 1 and 99. The default value is 20. NOTE: the number of results returned may be less than maxResults if there are not enough suggestions to provide based on the search terms.

useHyphens - *boolean*

Optional, if specified controls whether to use hyphens when generating suggestions. Depending on the search terms and the number of results, the useHyphens toggle may or may not have an impact on results. The default is 'false'.

addRelated - *boolean*

Optional, if specified controls whether to use related terms or phrases when generating suggestions. For example if the input includes the words 'HorseTrading' the addRelated parameter might cause a result such as SportHorseTrading, since sport and horse are related. The default is false.

TLDList - *string*

Optional, if specified the list of TLDs may include COM, NET, ORG, INFO, BIZ, CO, PRO, US, CA, ASIA & MOBI. The list is a comma separated list of the desired TLDs to include in the results, and must not include the 'dot' ('.') separator. The default is to include all supported TLDs.

58.2.3 Request Headers

Accept

Allows the client to specify the preferred response content type. Currently supported values:
application/json

58.3 Response

58.3.1 Status Code

200 OK

Success.

58.3.2 Response Body

```
{ "suggestions":  
  [  
    { "available": "{available}", "name": "{suggestion}", "tld": "{tld}"  
  }  
]
```

58.3.3 Response parameters

available - *boolean*

Indicates true or false if the suggested domain name is available for registration for this TLD.

suggestion - *string*

The second level domain name label.

tld - *string*

The TLD associated with this suggestion. The 'name' and 'tld' should be combined to form the complete domain name. For example 'name'.tld'

58.4 Example Name Spinner

58.4.1 Request

GET /domains/greenenergy/namespinner?maxResults=10&addRelated=true&useHyphens=true&TLDList=com,net,org,us

58.4.2 Response

```
{  
  "suggestions": [  
    {  
      "available": "true",  
      "name": "Green-Energy-Home",  
      "tld": "Com"  
    },  
    {  
      "available": "false",  
      "name": "Green-Energy-Home",  
      "tld": "Net"  
    },  
    {  
      "available": "true",  
      "name": "Green-Energy-Home",  
      "tld": "Org"  
    },  
    {  
      "available": "true",  
      "name": "Green-Energy-Home",  
      "tld": "Us"  
    },  
    {  
      "available": "true",  
      "name": "Best-Green-Energy",  
      "tld": "Com"  
    },  
    {  
      "available": "true",  
      "name": "Best-Green-Energy",  
      "tld": "Net"  
    },  
    {  
      "available": "true",  
      "name": "Best-Green-Energy",  
      "tld": "Org"  
    },  
    {  
      "available": "true",  
      "name": "Best-Green-Energy",  
      "tld": "Us"  
    },  
    {  
      "available": "true",  
      "name": "Free-Green-Energy",  
      "tld": "Com"  
    },  
    {  
      "available": "true",  
      "name": "Free-Green-Energy",  
      "tld": "Net"  
    },  
    {  
      "available": "true",  
      "name": "Free-Green-Energy",  
      "tld": "Org"  
    },  
    {  
      "available": "true",  
      "name": "Free-Green-Energy",  
      "tld": "Us"  
    },  
    {  
      "available": "true",  
      "name": "Green-Energy-Savings",  
      "tld": "Com"  
    },  
    {  
      "available": "true",  
      "name": "Green-Energy-Savings",  
      "tld": "Com"  
    }  
  ]  
}
```



```

    "tld": "Net"
  },
  {
    "available": "true",
    "name": "Green-Energy-Savings",
    "tld": "Org"
  },
  {
    "available": "true",
    "name": "Green-Energy-Savings",
    "tld": "Us"
  },
  {
    "available": "true",
    "name": "Green-Energy-Technologies",
    "tld": "Com"
  },
  {
    "available": "true",
    "name": "Green-Energy-Technologies",
    "tld": "Net"
  },
  {
    "available": "true",
    "name": "Green-Energy-Technologies",
    "tld": "Org"
  },
  {
    "available": "true",
    "name": "Green-Energy-Technologies",
    "tld": "Us"
  },
  {
    "available": "true",
    "name": "Green-Clean-Energy",
    "tld": "Com"
  },
  {
    "available": "true",
    "name": "Green-Clean-Energy",
    "tld": "Net"
  },
  {
    "available": "true",
    "name": "Green-Clean-Energy",
    "tld": "Org"
  },
  {
    "available": "true",
    "name": "Green-Clean-Energy",
    "tld": "Us"
  },
  {
    "available": "true",
    "name": "Green-Energy-Future",
    "tld": "Com"
  },
  {
    "available": "true",
    "name": "Green-Energy-Future",
    "tld": "Net"
  },
  {
    "available": "false",
    "name": "Green-Energy-Future",
    "tld": "Org"
  },
  {
    "available": "true",
    "name": "Green-Energy-Future",
    "tld": "Us"
  },
  {
    "available": "true",
    "name": "Green-Wind-Energy",
    "tld": "Com"
  },
  {
    "available": "true",
    "name": "Green-Wind-Energy",
    "tld": "Net"
  },
  {
    "available": "true",
    "name": "Green-Wind-Energy",
    "tld": "Org"
  },
  {
    "available": "false",
    "name": "Green-Wind-Energy",
    "tld": "Us"
  },
  {
    "available": "true",
    "name": "Green-Energy-Barrier",
    "tld": "Com"
  },
  {
    "available": "true",
    "name": "Green-Energy-Barrier",
    "tld": "Net"
  },
  {
    "available": "true",
    "name": "Green-Energy-Barrier",
    "tld": "Org"
  },

```

```
{
  "available":"true",
  "name":"Green-Energy-Barrier",
  "tld":"Us"
},
{
  "available":"true",
  "name":"Austin-Green-Energy",
  "tld":"Com"
},
{
  "available":"true",
  "name":"Austin-Green-Energy",
  "tld":"Net"
},
{
  "available":"true",
  "name":"Austin-Green-Energy",
  "tld":"Org"
},
{
  "available":"true",
  "name":"Austin-Green-Energy",
  "tld":"Us"
}
]
}
```

58.5 NameSpinner Sample Code

HTML and JavaScript sample code is available for download [here](#).

58.6 See also

- [Domain Name Registration API](#)
- [Check domain availability](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

59 Domains-domainName-namespinner:SAMPLE

60 SAMPLE /domains/{domainName}/namespinner

This page contains JavaScript and HTML sources used to perform domain name availability searches for suggested domain names.

60.1 Contents

- 1 SAMPLE /domains/{domainName}/namespinner
 - ◆ 1.1 JavaScript
 - ◆ 1.2 HTML
 - ◆ 1.3 See also

60.2 JavaScript

```
/*  
DISCLAIMER
```

The code examples, API, information, data, and other materials (collectively "Materials") made available on this website are provided on an "as is", "as available" basis, without any warranty as to the completeness, accuracy, reliability, suitability, availability, merchantability or fitness for a particular purpose. Hostway expressly makes no warranty, representation, or guaranty as to the content, sequence, timeliness, or completeness of the Materials. Hostway further makes no warranty, representation, or guaranty that the Materials will be uninterrupted or error free or bug free. Hostway will not provide any support for the Materials, and Hostway is not obligated correct any issues may exist in the Materials whether or not Hostway has knowledge of such issues. However, Hostway reserves the right to amend, modify, or delete all or any portion of the Materials at any time at Hostway's sole discretion, and such amendment, modification, or deletion may affect, alter, or terminate the functionality of any previous versions of the Materials (such as the code examples) without notice.

Hostway will have absolutely no liability in connection with or arising from the Materials, your use of the Materials, or the availability or unavailability of any of the Materials. Furthermore, Hostway expressly disclaims all liability for, without limitation:

- Any and all damage to your computer, hardware, data, information, materials, and business arising from the Materials or your use, modification, or distribution of the Materials;
- Any and all indirect or consequential loss or damage, loss of data or loss of profits arising out of, or in connection with, your use, modification, or distribution of the Materials;
- Any and all functions of any modifications that you make to the Materials, including any loss, damage, or other liability resulting from your modification of the Materials;
- Any and all losses or damages caused, in whole or in part, by Hostway's actions, omissions, or negligence in procuring, compiling, or delivering the Materials;
- Any and all errors, omissions, or inaccuracies in the Materials regardless of how caused, or delays or interruptions in delivery of the Materials; or
- Any decision you or any third party may make, or action taken or not taken, in reliance upon the Materials.

For the purposes of this disclaimer, "Hostway" includes, without limitation, Hostway's divisions, affiliates, subsidiaries, successors, parent companies, and their (including Hostway's) executives, directors, officers, managers, employees, consultants, contractors, agents, affiliates, third-party providers, merchants, licensors or the like.

```
****/
```

```
var publicApiUrl = 'http://coreapi01.ote.chicago.hostway:8092';
```

```
/**  
 * suggestNames :  
 *  
 * Make an ajax request to get the availability of suggested domain names.  
 * Assembles querystring via queryStringMap  
 * Assemble url with  
 * 1. publicApiUrl  
 * 2. nameSuggest  
 * 3. querystring  
 *  
 * @constructor  
 * @this {suggestNames}  
 * @param {string} nameSuggest, the second level domain name text to spin.  
 * @param {Object.<string, string>} queryStringMap, a map of all the additional attributes to send as querystring.  
 * @param {function( { status: number, name: string, suggestions[{name:string, tld:string, available:boolean}] } )}  
 *      resultHandler, function object holding the results of a spin call.  
 *  
 */  
  
function suggestNames(nameSuggest, queryStringMap, resultHandler) {  
  
    var qs = "";  
  
    //  
    // build query string with additional parameters to be sent with name spinner request  
    //  
    for(var key in queryStringMap){  
  
        if (queryStringMap.hasOwnProperty(key)) {  
  
            if(key=='maxResults' && queryStringMap[key] >= 0 && new Number(queryStringMap[key]) > 0 ){
```

```

        qs = qs + "maxResults="+queryStringMap[key]+'&';
    }

    if(key=='addRelated' && queryStringMap[key]){
        qs = qs + "AddRelated="+queryStringMap[key]+'&';
    }

    if(key=='useHyphens'){
        qs = qs + "useHyphens="+queryStringMap[key]+'&';
    }

    if(key=='tldList' && (queryStringMap[key] != null && queryStringMap[key].length > 0)){
        qs = qs + "TLDList="+queryStringMap[key]+'&';
    }
}

// Remove invalid chars. Input should be a set of keywords like 'Smart Investor'
// or a second level domain name like 'smartinvestor.com'
//
nameSuggest = nameSuggest.replace("\\", "");
nameSuggest = nameSuggest.replace("http://", "");
nameSuggest = nameSuggest.replace("/", "");
nameSuggest = nameSuggest.replace("www\\.", "");

var uri = '/domains/' + nameSuggest + '/namespinner';

//
// exec spinner request
//
$.ajax({
    type: "GET",
    url: publicApiUrl + uri +'?' +qs,
    dataType: "json",
    success: function(data, textStatus, jqXHR) {
        console.log('SUCCESS ' + textStatus);
        result = { status: jqXHR.status, name: nameSuggest, suggestions:data.suggestions };
        resultHandler(result);
    },
    error: function(jqXHR, textStatus, errorThrown) {
        console.log('Error status=' + jqXHR.status);
        result = { status: jqXHR.status, name: nameSuggest};
        resultHandler(result);
    }
});
}

```

60.3 HTML

```

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml2/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
<head>

```

```

    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />

```

```

    <title>Domain Name Suggestions Availability Check</title>
    <script src="http://ajax.googleapis.com/ajax/libs/jquery/1.8.0/jquery.min.js"></script>
    <script type="text/javascript" src="namespinner.js"></script>
    <script type="text/javascript">

```

```

// show or hide advance attributes, default to false
var advacedAttr=false;

// show or hide help content, default to false
var help=false;

// Used only to show how long each name suggestion search takes.
var curTime = (new Date()).getTime();

```

```

/*
 * UI help content hover action.
 */
(function($){
    $(document).ready(function() {

        //
        // onclick show/remove advanced settings
        //
        $('#advancedSettings').click(function() {
            if(!advacedAttr) {
                $('[name=advancedSettings]').css("visibility", "visible");
                $('#content').css("height", "290");
                advacedAttr=true;
            }else{
                $('[name=advancedSettings]').css("visibility", "hidden");
                $('#content').css("height", "50");
                advacedAttr=false;

                $('[name=help]').empty();
                help=false;
            }
        });

        //
        // onclick show/remove help content
        //
        $('#help').click(function() {
            if(!help){

```

```

        $('[name=help]').append(
        '<ul>'+
        '<li>Add Related : Return domain names that are related to the SLD</li>'+
        '<li>Use Hyphens : Return domain names that include hyphens</li>'+
        '<li>Max Results : Maximum number of suggested names to return in <br/>'+
        'addition to your input (between 0 and 99)'+
        '</ul>'
        );
        help=true;
        $('#content').css("height","490");
        $('#content').css("width","400");
    }else{
        $('[name=help]').empty();
        help=false;
        $('#content').css("height","290");
        $('#content').css("width","400");
    }
});

//
// formatting of suggestion text area field
//
$('#[name=nameSuggest]').click(function(){
    if ($(this).val() == 'Enter search term') {
        $(this).val('');
    }
    $(this).css('color','#333');
});

});
})(window.jQuery || window.$);

/****
 *
 * Callback function to display results returned in the suggestNames
 * call. This function reads the name suggestion results returned by
 * the Hostway Public API Server as a JSON structure. It keeps track
 * of all the TLDs returned in the result and builds a list of the
 * suggested names for each TLD and whether that name is available or
 * not. It then builds an HTML table with the TLDs in the top row.
 *
 */
function displayResult(result) {
/****
-----
An example of the results returned for the search term
'DerivativesMarket' and Max Results was set to 3. The three results
are 'DerivativesSell', 'DerivativesStore' and 'DerivativesShop'.
Note that useHyphens and addRelated do not have an impact on
the results in this case as the first 3 results do not make use of
hyphens or related words.

{
    "suggestions": [
        {"available": "true", "name": "DerivativesSell", "tld": "Com"},
        {"available": "true", "name": "DerivativesSell", "tld": "Net"},
        {"available": "true", "name": "DerivativesSell", "tld": "Org"},
        {"available": "true", "name": "DerivativesSell", "tld": "Info"},
        {"available": "true", "name": "DerivativesSell", "tld": "Biz"},
        {"available": "true", "name": "DerivativesSell", "tld": "Pro"},
        {"available": "true", "name": "DerivativesSell", "tld": "Mobi"},
        {"available": "true", "name": "DerivativesSell", "tld": "Asia"},
        {"available": "true", "name": "DerivativesSell", "tld": "Ca"},
        {"available": "true", "name": "DerivativesSell", "tld": "Co"},
        {"available": "true", "name": "DerivativesSell", "tld": "Us"},
        {"available": "true", "name": "DerivativesStore", "tld": "Com"},
        {"available": "true", "name": "DerivativesStore", "tld": "Net"},
        {"available": "true", "name": "DerivativesStore", "tld": "Org"},
        {"available": "true", "name": "DerivativesStore", "tld": "Info"},
        {"available": "true", "name": "DerivativesStore", "tld": "Biz"},
        {"available": "true", "name": "DerivativesStore", "tld": "Pro"},
        {"available": "true", "name": "DerivativesStore", "tld": "Mobi"},
        {"available": "true", "name": "DerivativesStore", "tld": "Asia"},
        {"available": "true", "name": "DerivativesStore", "tld": "Ca"},
        {"available": "true", "name": "DerivativesStore", "tld": "Co"},
        {"available": "true", "name": "DerivativesStore", "tld": "Us"},
        {"available": "false", "name": "DerivativesShop", "tld": "Com"},
        {"available": "false", "name": "DerivativesShop", "tld": "Net"},
        {"available": "true", "name": "DerivativesShop", "tld": "Org"},
        {"available": "true", "name": "DerivativesShop", "tld": "Info"},
        {"available": "true", "name": "DerivativesShop", "tld": "Biz"},
        {"available": "false", "name": "DerivativesShop", "tld": "Pro"},
        {"available": "true", "name": "DerivativesShop", "tld": "Mobi"},
        {"available": "true", "name": "DerivativesShop", "tld": "Asia"},
        {"available": "true", "name": "DerivativesShop", "tld": "Ca"},
        {"available": "true", "name": "DerivativesShop", "tld": "Co"},
        {"available": "true", "name": "DerivativesShop", "tld": "Us"}
    ]
}
}
-----
*****/
$('#resultMsg').css('text-decoration', 'none');

if (result.status == 200) {
    var headerArr = [];
    var suggestions = {};
    var content = "";

```

```

$.each(result.suggestions, function(key, suggestion) {

    // Add each TLD to a header array for display.
    if ( $.inArray( suggestion.tld, headerArr ) == -1 ) {
        headerArr.push(suggestion.tld);
    }

    // And build a dictionary of results that looks like this:
    // {
    //   "suggested1": {"com": {"available" : "true"}, "net": {"available" : "false"}},
    //   "suggested2": {"com": {"available" : "false"}, "net": {"available" : "false"}}
    // }
    if (!(suggestion.name in suggestions)) {
        suggestions[suggestion.name] = {};
    }
    suggestions[suggestion.name][suggestion.tld] = {};
    suggestions[suggestion.name][suggestion.tld].available = suggestion.available;

});

// Show the results, including TLDs header in a table
content += "<table><tr><td><strong>Name</strong></td></tr>";
for(name in headerArr) {
    content += "<td cellpadding=\"25px\"><strong>."+headerArr[name].toLowerCase()+"</strong></td>";
}
content += "</tr>";

for(var name in suggestions) {
    var str = "<tr><td>"+name+"</td>";
    for (var tld in suggestions[name]) {
        if (suggestions[name][tld].available == 'true') {
            // include a checkbox for a domain because the domain is available and could
            // be selected for registration
            str += "<td align=\"center\"><input type=\"checkbox\" name=\"domain\" value=\""+name+"."+tld+"\"/></td>";
        }
        else {
            // suggested domain name is not available for this TLD.
            str += "<td align=\"center\">X</td>";
        }
    }
    str += "</tr>";
    content += str;
}
content += "</table>";
alert
var deltaTime = (new Date()).getTime() - curTime ;

$("#namespinnerResult").html(content);
$("#resultMsg").css('text-decoration', 'none');
$("#resultMsg").text("      (results in "+deltaTime + " ms)");

} else {
    $("#resultMsg").text("Error (" + result.status + ")");
}
}

/*
 * Submit the query.
 */
function nameSearch() {

    //
    // nameSuggest - the second level domain string used to search for additional suggestions
    //
    var nameSuggest = $('[name=nameSuggest]').val();

    // used to show the time taken to perform the lookup.
    curTime = (new Date()).getTime();

    // gather form submission fields
    var tldList="";
    var i = 0;
    $(':checkbox').each(function() {
        if(this.name == 'tld' && this.checked)
            // append the TLD to the query string
            tldList = tldList + $(this).attr('value') + ',';
    });

    if(tldList.charAt( tldList.length-1 ) == ",") {
        tldList = tldList.slice(0, -1);
    }

    var addRelated = $('[name=AddRelated]').attr('checked')?true:false;
    var useHyphens = $('[name=UseHyphens]').attr('checked')?true:false;
    var maxResults = $('[name=MaxResults]').val();
    $("#resultMsg").css('text-decoration', 'blink');
    $("#namespinnerResult").empty();
    $("#resultMsg").text("Checking ...");

    // Form field validation
    var errorMessage;
    if (isNaN(maxResults / 1) != false) {
        errorMessage = "maxResults needs to be between 0 and 99";
    }
}

```

```

} else if (nameSuggest.indexOf("\\") != -1) {
    errorMessage = "Please remove \\ from the name suggestion string";
} else if (nameSuggest.indexOf("http://") == 0) {
    errorMessage = "Please remove http:// from the name suggestion string";
} else if (nameSuggest.indexOf("/") != -1) {
    errorMessage = "Please remove / from the name suggestion string";
} else if (nameSuggest.indexOf("www.") == 0) {
    errorMessage = "Please remove www. from the name suggestion string";
}

// Upon error, display message and stop execution
if(errorMessage != null){
    $("#resultMsg").text(errorMessage);
    $("#resultMsg").css('color','red');
    return;
}

// querystring map information in order to submit additional attribute (not required).
// empty querystring will results in default behaviour
var queryString = {
    'tldList': tldList,
    'addRelated': addRelated,
    'useHyphens': useHyphens,
    'maxResults': maxResults
};

// call namespinner
//
suggestNames(nameSuggest, queryString, displayResult);
}

</script>
</head>

<style type="text/css">
.div1{ display:inline; color:green;}
.div2{ display:inline;visibility:hidden;}
</style>

<body>

<div id="info" style="width: 400px;">

<b>Domain name suggestion tool </b> -
This page provides a basic example of the domain name suggestion tool
which generates search results for domains that are similar to the one
provided as input. Search results can be adjusted by allowing the use
of hyphens separating words and by allowing the use of related words.
The number of results can set from 1 to 99 with the default being 20.
The set of To Level Domains (TLDs) can also be controlled.

</div>
<br/>
<div id="content" style="background-color: #fafafa; border: solid 1px #333333; padding: 20px; padding-top: 0px; width: 400px; height:

<form method="post" id="domainSearchForm" onsubmit="nameSearch(); return false;">
  <div id="formContent">
    <br>
    <input type="text" name="nameSuggest" size="30" maxlength="60" value="Enter search term"/><input type="submit" value="Search"/>
    <br/>
    <a id="advancedSettings" href="#">Advanced setting</a>
    <br>
    <div name="advancedSettings" class="div2">
      <table border="0" width="400">
        <tr>
          <td><input type="checkbox" name="UseHyphens" value="1"/></td><td>Use Hyphens</td>
        </tr>
        <tr>
          <td><input type="checkbox" name="AddRelated" value="1"/></td><td>Add Related</td>
        </tr>
        <tr>
          <td><input type="text" name="MaxResults" size="2" maxlength="2"/></td><td>Max Results</td>
        </tr>
        <tr>
          <td colspan="2"><td valign="top">Tld list<div class="div1" name="tldListHelp"></div></td>
          <td>
            <table>
              <tr>
                <td><input type="checkbox" name="tld" value="com" id="com"/><label for="com">com</label>
                <td><input type="checkbox" name="tld" value="net" id="net"/><label for="net">net</label>
                <td><input type="checkbox" name="tld" value="org" id="org"/><label for="org">org</label>
                <td><input type="checkbox" name="tld" value="info" id="info"/><label for="info">info</label>
            </tr>
            <tr>
                <td><input type="checkbox" name="tld" value="co" id="co"/><label for="co">co</label>
                <td><input type="checkbox" name="tld" value="biz" id="biz"/><label for="biz">biz</label>
                <td><input type="checkbox" name="tld" value="pro" id="pro"/><label for="pro">pro</label>
                <td><input type="checkbox" name="tld" value="mobi" id="mobi"/><label for="mobi">mobi</label>
            </tr>
            <tr>
                <td><input type="checkbox" name="tld" value="us" id="us"/><label for="us">us</label>
                <td><input type="checkbox" name="tld" value="ca" id="ca"/><label for="ca">ca</label>
                <td><input type="checkbox" name="tld" value="asia" id="asia"/><label for="asia">asia</label>
            </tr>
            <tr>
                <td><input type="checkbox" name="tld" value="uk" id="uk"/><label for="uk">uk</label>
                <td><input type="checkbox" name="tld" value="de" id="de"/><label for="de">de</label>
            </tr>
          </td>
        </tr>
      </table>
    </div>
  </div>
</form>

```



```

        </td>
    </tr>
    <tr>
        <td colspan="2" align="left">
            <a id="help" href="#">Help</a>
            <br/>
            <div name="help"></div>
        </td>
    </tr>
</table>
</div>
</form>
</div>
<div style="text-align:left" id="resultMsg"></div>
<div class="namespinnerResult"></div>
</body>
</html>

```

60.4 See also

- [Domain Name Registration API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

61 register:GET

62 GET /domains/{domainName}/prices/register

Get register prices.

62.1 Contents

- 1 GET /domains/{domainName}/prices/register
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◇ 1.1.4 Request Body
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - ◆ 1.3 Example of getting register prices.
 - ◆ 1.4 See also

62.2 Request

GET /domains/{domainName}/prices/register

62.2.1 Parameters

domainName - string

The domain name. IE: **example.com** is a valid domain name, whereas **http://www.example.com** or **www.example.com** are not. If the domain name is invalid, or is otherwise not registerable such as for a non-existent or non-supported TLD, an error will be returned. Domain names are case insensitive. The domain name must be an open or suspended product under the authenticated franchise.

62.2.2 URI Parameters

filterCurrency- *string*
(OPTIONAL)

Currency for which prices are displayed. If not provided will default to USD

62.2.3 Request Headers

None.

62.2.4 Request Body

None.

62.3 Response

62.3.1 Status Code

200 OK

Returns register domain prices.

404 Not Found

No domain register prices found.

62.3.2 Response Body

List of the available register prices.

62.4 Example of getting register prices.

Request

GET /domains/example.com/prices/register

Response

200 OK

Content-Type: application/json; charset=UTF-8

```
[
{"currency": "USD", "amount": 16.0, "type": "domain_registration_fee", "monthlyCycle": 12, "deferred": 0},
{"currency": "USD", "amount": 17.899999999999999, "type": "domain_registration_fee", "monthlyCycle": 24, "deferred": 0},
{"currency": "USD", "amount": 26.850000000000001, "type": "domain_registration_fee", "monthlyCycle": 36, "deferred": 0},
{"currency": "USD", "amount": 35.799999999999997, "type": "domain_registration_fee", "monthlyCycle": 48, "deferred": 0},
{"currency": "USD", "amount": 44.75, "type": "domain_registration_fee", "monthlyCycle": 60, "deferred": 0}
]
```

Note: Deferred flag is showing whether there is delayed billing configured.

62.5 See also

- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

63 renew:GET

64 GET /domains/{domainName}/prices/renew

Get renewal prices.

64.1 Contents

- 1 GET /domains/{domainName}/prices/renew
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◇ 1.1.4 Request Body
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - ◆ 1.3 Example of getting renew prices.
 - ◆ 1.4 See also

64.2 Request

GET /domains/{domainName}/prices/renew

64.2.1 Parameters

domainName - string

The domain name. IE: **example.com** is a valid domain name, whereas **http://www.example.com** or **www.example.com** are not. If the domain name is invalid, or is otherwise not registerable such as for a non-existent or non-supported TLD, an error will be returned. Domain names are case insensitive. The domain name must be an open or suspended product under the authenticated franchise.

64.2.2 URI Parameters

filterCurrency- *string*
(OPTIONAL)

Currency for which prices are displayed. If not provided will default to USD

64.2.3 Request Headers

None.

64.2.4 Request Body

None.

64.3 Response

64.3.1 Status Code

200 OK
Returns the renewal prices.

404 Not Found
No domain prices found.

64.3.2 Response Body

List of the available renew prices.

64.4 Example of getting renew prices.

Request

GET /domains/example.com/prices/renew

Response

200 OK
Content-Type: application/json; charset=UTF-8

```
[
{"currency": "USD", "amount": 8.949999999999999, "type": "domain_renewal_fee", "monthlyCycle": 12, "deferred": 0},
{"currency": "USD", "amount": 17.899999999999999, "type": "domain_renewal_fee", "monthlyCycle": 24, "deferred": 0},
{"currency": "USD", "amount": 26.850000000000001, "type": "domain_renewal_fee", "monthlyCycle": 36, "deferred": 0},
{"currency": "USD", "amount": 35.799999999999997, "type": "domain_renewal_fee", "monthlyCycle": 48, "deferred": 0},
{"currency": "USD", "amount": 44.75, "type": "domain_renewal_fee", "monthlyCycle": 60, "deferred": 0},
{"currency": "USD", "amount": 89.5, "type": "domain_renewal_fee", "monthlyCycle": 120, "deferred": 0}
]
```

Note: Deferred flag is showing whether there is delayed billing configured.

64.5 See also

- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

65 transfer:GET

66 GET /domains/{domainName}/prices/transfer

Get renewal prices.

66.1 Contents

- 1 GET /domains/{domainName}/prices/transfer
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◇ 1.1.4 Request Body
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - ◆ 1.3 Example of getting transfer prices.
 - ◆ 1.4 See also

66.2 Request

GET /domains/{domainName}/prices/transfer

66.2.1 Parameters

domainName - string

The domain name. IE: **example.com** is a valid domain name, whereas **http://www.example.com** or **www.example.com** are not. If the domain name is invalid, or is otherwise not registerable such as for a non-existent or non-supported TLD, an error will be returned. Domain names are case insensitive. The domain name must be an open or suspended product under the authenticated franchise.

66.2.2 URI Parameters

filterCurrency- *string*
(OPTIONAL)

Currency for which prices are displayed. If not provided will default to USD

66.2.3 Request Headers

None.

66.2.4 Request Body

None.

66.3 Response

66.3.1 Status Code

200 OK

Returns domain transfer prices.

404 Not Found

No domain transfer prices found.

66.3.2 Response Body

List of the available transfer prices.

66.4 Example of getting transfer prices.

Request

GET /domains/example.com/prices/transfer

Response

200 OK

Content-Type: application/json; charset=UTF-8

```
[
{"currency": "USD", "amount": 10.5, "type": "domain_transfer_fee", "monthlyCycle": 12, "deferred": 1}
]
```

Note: Deferred flag is showing whether there is delayed billing configured.

66.5 See also

- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

67 Domains-domainName-prices:GET

68 GET /domains/{domainName}/prices

Get prices for register, renew and transfer.

68.1 Contents

- 1 GET /domains/{domainName}/prices
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◇ 1.1.4 Request Body
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - ◆ 1.3 Example of getting prices.
 - ◆ 1.4 See also

68.2 Request

GET /domains/{domainName}/prices

68.2.1 Parameters

domainName - string

The domain name. IE: **example.com** is a valid domain name, whereas **http://www.example.com** or **www.example.com** are not. If the domain name is invalid, or is otherwise not registerable such as for a non-existent or non-supported TLD, an error will be returned. Domain names are case insensitive. The domain name must be an open or suspended product under the authenticated franchise.

68.2.2 URI Parameters

filterCurrency- *string*
(OPTIONAL)

Currency for which prices are displayed. If not provided will default to USD

68.2.3 Request Headers

None.

68.2.4 Request Body

None.

68.3 Response

68.3.1 Status Code

200 OK

Returns domain prices.

404 Not Found

No domain prices found.

68.3.2 Response Body

List of the available prices.

68.4 Example of getting prices.

Request

GET /domains/example.com/prices

Response

200 OK

Content-Type: application/json; charset=UTF-8

```
[{"currency": "USD", "amount": 10.5, "type": "domain_transfer_fee", "monthlyCycle": 12, "deferred": 0}, {"currency": "USD", "amount": 16.0, "type": "domain_registration_fee", "monthlyCycle": 12, "deferred": 0}, {"currency": "USD", "amount": 8.949999999999999, "type": "domain_renewal_fee", "monthlyCycle": 12, "deferred": 0}, {"currency": "USD", "amount": 17.899999999999999, "type": "domain_renewal_fee", "monthlyCycle": 24, "deferred": 0}, {"currency": "USD", "amount": 26.850000000000001, "type": "domain_renewal_fee", "monthlyCycle": 36, "deferred": 0}, {"currency": "USD", "amount": 35.799999999999997, "type": "domain_renewal_fee", "monthlyCycle": 48, "deferred": 0}, {"currency": "USD", "amount": 44.75, "type": "domain_renewal_fee", "monthlyCycle": 60, "deferred": 0},
```

```
{ "currency": "USD", "amount": 89.5, "type": "domain_renewal_fee", "monthlyCycle": 120, "deferred": 0 },
{ "currency": "USD", "amount": 17.899999999999999, "type": "domain_registration_fee", "monthlyCycle": 24, "deferred": 0 },
{ "currency": "USD", "amount": 26.850000000000001, "type": "domain_registration_fee", "monthlyCycle": 36, "deferred": 0 },
{ "currency": "USD", "amount": 35.799999999999997, "type": "domain_registration_fee", "monthlyCycle": 48, "deferred": 0 },
{ "currency": "USD", "amount": 44.75, "type": "domain_registration_fee", "monthlyCycle": 60, "deferred": 0 }
]
```

68.5 See also

- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

69 Domains-domainName-privateregistration-:GET

70 GET /domains/privateregistration

Show if private registration is available for the supplied comma delimited tld list

70.1 Contents

- 1 GET /domains/privateregistration
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 Request Headers
 - ◇ 1.1.3 Request Body
 - 1.1.3.1 Request Parameters
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - ◆ 1.3 Example search for a domain product.
 - ◆ 1.4 See also

70.2 Request

GET /domains/privateregistration

70.2.1 Parameters

tld
one or more valid tld's separated by comma

70.2.2 Request Headers

None.

70.2.3 Request Body

None.

70.2.3.1 Request Parameters

tld - *list of strings*
comma separated list of tlds
for example, tld=com,net.org

70.3 Response

70.3.1 Status Code

200 OK
Returns private registration availability.

400 Bad Request
Invalid request

70.3.2 Response Body

List of private registration availability.

70.4 Example search for a domain product.

Request

GET /domains/privateregistration?tld=pro,ca,asia,eu,be,nl,co.uk,us,com,net,info,org

Response

200 OK
Content-Type: application/json; charset=UTF-8

```
[
  {"privateRegistration": false, "tld": "pro"},
  {"privateRegistration": false, "tld": "ca"},
  {"privateRegistration": false, "tld": "asia"},
  {"privateRegistration": false, "tld": "eu"},
  {"privateRegistration": false, "tld": "be"},
  {"privateRegistration": false, "tld": "nl"},
  {"privateRegistration": false, "tld": "co.uk"},
  {"privateRegistration": false, "tld": "us"}
]
```

```
{ "privateRegistration": true, "tld": "com"},  
{ "privateRegistration": true, "tld": "net"},  
{ "privateRegistration": true, "tld": "info"},  
{ "privateRegistration": true, "tld": "org"}  
]
```

70.5 See also

- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

71 Domains-domainName-privateregistration:DELETE

72 DELETE /domains/{domainName}/privateregistration

Remove the privateregistration for a domain.

72.1 Contents

- 1 DELETE /domains/{domainName}/privateregistration
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◇ 1.1.4 Request Body
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - ◆ 1.3 Example of deleting a host
 - ◆ 1.4 See also

72.2 Request

```
DELETE /domains/{domainName}/privateregistration
```

72.2.1 Parameters

None.

72.2.2 URI Parameters

domainName - string

The domain name. IE: **example.com** is a valid domain name, whereas **http://www.example.com** or **www.example.com** are not. If the domain name is invalid, or is otherwise not registerable such as for a non-existent or non-supported TLD, an error will be returned. Domain names are case insensitive. The domain name must be an open or suspended product under the authenticated franchise.

72.2.3 Request Headers

Content-Type: application/json

72.2.4 Request Body

None.

72.3 Response

72.3.1 Status Code

204 No Content
Returned on success

404 Not Found
No domain records found for the given domain name.

500 Internal Server Error
An error has occurred with the registry,

72.3.2 Response Body

72.4 Example of deleting a host

Request

```
DELETE /domains/example.com/privateregistration
```

Response

```
204 No Content
```

72.5 See also

- [Domain Name Registration API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

73 Domains-domainName-privateregistration:GET

74 GET /domains/{domainName}/privateregistration

Get the privateregistration info for a domain.

74.1 Contents

- 1 GET /domains/{domainName}/privateregistration
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◇ 1.1.4 Request Body
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - 1.2.2.1 Parameters
 - ◆ 1.3 Example of getting the domain authorization code.
 - ◆ 1.4 See also

74.2 Request

GET /domains/{domainName}/privateregistration

74.2.1 Parameters

None.

74.2.2 URI Parameters

domainName - string

The domain name. IE: **example.com** is a valid domain name, whereas **http://www.example.com** or **www.example.com** are not. If the domain name is invalid, or is otherwise not registerable such as for a non-existent or non-supported TLD, an error will be returned. Domain names are case insensitive. The domain name must be an open or suspended product under the authenticated franchise.

74.2.3 Request Headers

None.

74.2.4 Request Body

None.

74.3 Response

74.3.1 Status Code

200 OK

Returns the privateregistration info

404 Not Found

No domain records found for the given domain name.

500 Internal Server Error

An error has occurred with the back end or the domain registry.

74.3.2 Response Body

The privateregistration info.

74.3.2.1 Parameters

type - *string*

The privateregistration product type, either "full" or "light"

productTypeConfiguration - *string*

The product name, e.g. "core.privateregistration"

email - *string*

(Optional) The forwarding e-mail address for "full" type privateregistration.

74.4 Example of getting the domain authorization code.

Request

GET /domains/example.com/privateregistration

Response

200 OK

Content-Type: application/json; charset=UTF-8

```
{  
  "type": "full",  
  "productTypeConfiguration": "core.privateregistration",  
  "email": "test@test.com"  
}
```

74.5 See also

- [Domain Name Registration API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

75 Domains-domainName-privateregistration:POST

76 POST /domains/{domainName}/privateregistration

Update the privateregistration for a domain - change forwarding e-mail address, upgrade or downgrade between "full" and "light" version of the privateregistration product.

76.1 Contents

- 1 POST /domains/{domainName}/privateregistration
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◇ 1.1.4 Request Body
 - 1.1.4.1 Parameters
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - ◆ 1.3 Example of updating e-mail address for "full" type of privateregistration.
 - ◆ 1.4 Example of downgrading "full" type of privateregistration to "light".
 - ◆ 1.5 Example of upgrading "light" type of privateregistration to "full".
 - ◆ 1.6 See also

76.2 Request

POST /domains/{domainName}/privateregistration

76.2.1 Parameters

None.

76.2.2 URI Parameters

domainName - string

The domain name. IE: **example.com** is a valid domain name, whereas **http://www.example.com** or **www.example.com** are not. If the domain name is invalid, or is otherwise not registerable such as for a non-existent or non-supported TLD, an error will be returned. Domain names are case insensitive. The domain name must be an open or suspended product under the authenticated franchise.

76.2.3 Request Headers

Content-Type: application/json

76.2.4 Request Body

The hosts for the domain.

76.2.4.1 Parameters

type - *string*

(Optional) The privateregistration product type, either "full" or "light". If it's different than the current type, product will be upgraded or downgraded to match the new type.

email - *string*

(Optional) The forwarding e-mail address for "full" type privateregistration.

76.3 Response

76.3.1 Status Code

204 No Content

Returned on success

404 Not Found

No domain records found for the given domain name.

500 Internal Server Error

An error has occurred with the registry,

76.3.2 Response Body

None.

76.4 Example of updating e-mail address for "full" type of privateregistration.

Request

POST /domains/example.com/privateregistration

```
{ "email": "test-changed@test.com" }
```

Response

204 No Content

76.5 Example of downgrading "full" type of privateregistration to "light".

Request

```
POST /domains/example.com/privateregistration
{ "type": "light" }
```

Response

```
204 No Content
```

76.6 Example of upgrading "light" type of privateregistration to "full".

Request

```
POST /domains/example.com/privateregistration
{ "type": "full", "email": "test-forward@test.com" }
```

Response

```
204 No Content
```

76.7 See also

- [Domain Name Registration API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

77 Domains-domainName-renew:POST

78 POST /domains/{domainName}/renew

Renew a domain for a specified registration period.

78.1 Contents

- 1 POST /domains/{domainName}/renew
 - ◆ 1.1 Request
 - ◇ 1.1.1 URI Parameters
 - ◇ 1.1.2 Request Headers
 - ◇ 1.1.3 Request Body
 - 1.1.3.1 Request Parameters
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - ◆ 1.3 Example of ordering domain renewal.
 - ◆ 1.4 See also

78.2 Request

POST /domains/{domainName}/renew

78.2.1 URI Parameters

domainName - string
The domain name to renew. Domain name is case insensitive.

78.2.2 Request Headers

Content-Type
Required. Set this header to `application/json; charset=UTF-8`
Authorization
Required.

78.2.3 Request Body

Request Body for renewing a domain

```
{
  "term": "{term}",
  "currentExpireDate": "{expireDate}"
}
```

78.2.3.1 Request Parameters

term - integer
Required. The number of years to add to the registration term. This amount must match one of the available cycles for renewing a domain. For example, if the [pricing](#) call shows that a domain has available cycles of 1 year, 5 and 10 this means a domain may not be renewed for 3 years, since that is not an available term for renewal. A domain cannot be renewed beyond its maximum registration term. Therefore if the max registration term is 10 years and a domain expires in 5 days, it cannot be renewed for 10 years, since this would mean the new term would be 10 years and 5 days into the future. Only a domain that is already expired can be renewed for its max registration term.

expireDate - date
The current expiration date of the domain is required to prove that the renewal request is unique and is not a duplicate renewal that has already been processed. If the given expireDate does not match the current expiration date of the domain, the renewal request is rejected. Use the [domain info command](#) to get the expireDate of the domains. The format for the date is yyyy-mm-dd

78.3 Response

78.3.1 Status Code

201 Accepted
The domain renewal order has been submitted.

400 Bad Request
If the given data is not acceptable. The domain name may be invalid if it includes parts of the URL like `'http://'`. If the registration term is not not a supported registration term for the domain, or if the expire date is in an invalid format. This error is also invoked if any of the required parameters are missing from the request.

403 - Not authorized

The domain name does not belong in the account of the customer who is attempting to renew the domain. Only the account holder may renew the domain.

404 - Not found

The domain name does not exist or is in a closed or suspended state and cannot be renewed.

409 - Conflict

When the registration term exceeds the maximum registration period, or the current expiration date supplied in the request does not match the expire date of the domain.

78.3.2 Response Body

Response Body for renewing a domain None

78.4 Example of ordering domain renewal.

Request

```
POST /domains/ng-test-fitness-shortcut-20130412-0940.com/renew
Content-Type: application/json; charset=UTF-8
Authorization: Basic ...
```

```
{
  "term": "1",
  "currentExpireDate": "2013-09-21"
}
```

Response

```
201 Accepted
```

78.5 See also

- [Domain Name Registration API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

79 Domains-domainName-renewTerms:GET

80 GET /domains/{domainName}/renewTerms

Get the available renewal terms and pricing for a domain.

80.1 Contents

- 1 GET /domains/{domainName}/renewTerms
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 URI Parameters
 - ◇ 1.1.3 Request Headers
 - ◇ 1.1.4 Request Body
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - ◆ 1.3 Example of getting the available renewal terms and pricing.
 - ◆ 1.4 See also

80.2 Request

GET /domains/{domainName}/renewTerms

80.2.1 Parameters

domainName - string

The domain name. IE: **example.com** is a valid domain name, whereas **http://www.example.com** or **www.example.com** are not. If the domain name is invalid, or is otherwise not registerable such as for a non-existent or non-supported TLD, an error will be returned. Domain names are case insensitive. The domain name must be an open or suspended product under the authenticated franchise.

80.2.2 URI Parameters

promotionCode

String (optional)

80.2.3 Request Headers

None.

80.2.4 Request Body

None.

80.3 Response

80.3.1 Status Code

200 OK

Returns the available renewal terms and pricing.

404 Not Found

No domain records found for the given domain name.

500 Internal Server Error

If the back end returns no renewal terms and/or pricing.

80.3.2 Response Body

List of the available renewal terms and pricing.

80.4 Example of getting the available renewal terms and pricing.

Request

GET /domains/example.com/renewTerms

Response

200 OK

Content-Type: application/json; charset=UTF-8

```
[{"currency": "USD", "amount": 0.0, "cycleName": "Yearly", "monthlyCycle": 12}, {"currency": "USD", "amount": 0.0, "cycleName": "Two Yearly", "monthlyCycle": 24}, {"currency": "USD", "amount": 0.0, "cycleName": "Three Yearly", "monthlyCycle": 36}, {"currency": "USD", "amount": 0.0, "cycleName": "Four Yearly", "monthlyCycle": 48}, {"currency": "USD", "amount": 0.0, "cycleName": "Five Yearly", "monthlyCycle": 60}, {"currency": "USD", "amount": 0.0, "cycleName": "Ten Yearly", "monthlyCycle": 120}]
```

80.5 See also

- [Domain Name Registration API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

81 Domains-domainName-transfer:GET

82 GET /domains/{domainName}/transfer

Performs domain transferability check.

82.1 Contents

- 1 GET /domains/{domainName}/transfer
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - 1.2.2.1 Parameters
 - ◆ 1.3 See also

82.2 Request

GET /domains/{domainName}/transfer

82.2.1 Parameters

domainName - string

The valid domain name to check including only the labels that are appropriate for the valid registration of a domain name at the registry. IE: **example.com** is a valid domain name that could be or is already registered at the registry. Whereas <http://www.example.com> or **www.example.com** are not. If the domain name is invalid, or is otherwise not registerable such as for a non-existent or non-supported TLD, an error will be returned. Domain names are case insensitive.

82.3 Response

82.3.1 Status Code

An HTTP response code value indicating ancillary information about the domain availability. See notes below for the meaning of the response code. Check the 'available' value to determine if the given domain name is available at the registry.

200 OK
transferability check ran ok

82.3.2 Response Body

```
{
  "transferrable": "{transferrable}",
  "name": "{domainName}",
  "code": 2106,
  "description": "Domain is not yet eligible for transfer, same franchise"
  "adminEmail": "{adminEmail}",
}
```

82.3.2.1 Parameters

transferrable - *boolean*

domainName - *string*

code - *string*

code holds the exact value of the transferability check
2300 = Domain transfer already in progress
2302 = Domain is already registered with this registrar
2303 = Domain is not registered
2304 = Domain status prohibits operation
2202 = Invalid authorization information
2106 = Domain is not yet eligible for transfer
2102 = Transfer not supported for this TLD
3100 = whois error

description - *string*

refer to the above code for the description

adminEmail - *string*

Whois admin email, if whois is available

82.4 See also

- [Domain Name Registration API](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

83 Domains-domainName:GET

84 GET /domains/{domainName}

Performs domain availability check.

84.1 Contents

- 1 GET /domains/{domainName}
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - 1.2.2.1 Parameters
 - ◆ 1.3 See also

84.2 Request

GET /domains/{domainName}

84.2.1 Parameters

domainName - string

The valid domain name to check including only the labels that are appropriate for the valid registration of a domain name at the registry. IE: **example.com** is a valid domain name that could be or is already registered at the registry. Whereas **http://www.example.com** or **www.example.com** are not. If the domain name is invalid, or is otherwise not registerable such as for a non-existent or non-supported TLD, an error will be returned. Domain names are case insensitive.

84.3 Response

84.3.1 Status Code

An HTTP response code value indicating ancillary information about the domain availability. See notes below for the meaning of the response code. Check the 'available' value to determine if the given domain name is available at the registry.

200 OK

Domain is either registered and does not exist in ServicePlex, or available. The 'available' result will be 'false' if the domain is registered (in this case and the domain can be presented as already taken and perhaps transferable to ServicePlex) and 'true' if the name is available.

203 Non Authoritative

Domain is already registered in ServicePlex. Check the 'available' result to see if the domain exists at the registry already. For example, a domain may be pending provisioning and therefore it already exists in ServicePlex and so is not available for registration even if the 'available' value is 'true'

400 Bad Request

The domain name is not valid.

412 Unauthorized

Domain is not permitted for registration in ServicePlex. Check the 'available' value to see if the domain is registered at the registry. A domain might not be permitted if it is in a restricted list of domain names that are prevented from being registered in ServicePlex. For example **buy-olympic-tickets.com** might be available but it is restricted. Special privileges are required to register restricted domains.

84.3.2 Response Body

```
{
  "available": "{available}",
  "name": "{domainName}"
}
```

84.3.2.1 Parameters

available - *boolean*

Indicates if the domain is *available* for registration **true**, or if the domain is already registered and subsequently not available for registration **false**. The value of 'available' is determined by checking the registry to see if a record of the domain exists or not, independent of the status or rules for registering the domain in ServicePlex.

domainName - *string*

The domain name that was supplied in the GET request.

84.4 See also

- [Domain Name Registration API](#)
- [Name suggestion search](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

85 Domains-tlds-extparams:GET

Some TLDs require additional parameters for registration. This request returns the extra parameters (if any) for the specified TLD, e.g. for .it (Italy) it would return the following data structure:

```
{
  "attributes":[
    {
      "description":"Tax ID. If the requester is ...",
      "userDefined":true,
      "required":1,
      "isChild":0,
      "application":2,
      "options":{
        "option":[
          {
            "description":"If the requester is ...",
            "requires":null,
            "id":972,
            "value":"1",
            "title":"Codice Fiscale"
          }
        ]
      },
      "id":151,
      "name":"it_pin"
    },
    ...
  ]
}
```

This data can be used by the client application to display a form with the required fields, along with descriptions and values. It is also used by the Public API to validate domain orders. For example, from the above data you can see that the "it_pin" parameter is required when registering .it domains, so you need to include it in your order as "extraParams" (either as a service attribute if using the standard ordering process - see [Ordering Domain](#), or in the domain shortcut registration request - see [PUT /accounts/{accountId}/domains/{domainName}](#)):

```
{
  "domainName": "testname.it",
  "extraParams": [{"name": "it_pin", "value": "1"}, ...]
  ...
}
```

As a common example, the .us domain registration requires the "us_nexus" and "us_purpose" parameters:

```
{
  "domainName": "ng-test-20141020.us",
  "extraParams": [{"name": "us_nexus", "value": "C11"}, {"name": "us_purpose", "value": "P1"}],
  ...
}
```

Another example would be the .pro registration, which requires "pro_profession" parameter:

```
{
  "domainName": "ng-test-20141020.pro",
  "extraParams": [{"name": "pro_profession", "value": "engineer"}],
  ...
}
```

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

86 Domains-tlds-registryagreements:GET

87 GET /domains/tlds/registryagreement

Get a list of registry agreements

87.1 Contents

- 1 GET /domains/tlds/registryagreement
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◇ 1.1.2 Request Headers
 - ◇ 1.1.3 Request Body
 - 1.1.3.1 Request Parameters
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - ◆ 1.3 Example get registry agreements for com,net,ca
 - ◆ 1.4 Example get registry agreements (supplying invalid tld) for com,net,ca,invalidTld
 - ◆ 1.5 See also

87.2 Request

GET /domains/tlds/registryagreement

87.2.1 Parameters

tld
one or more valid tld's separated by comma

87.2.2 Request Headers

None.

87.2.3 Request Body

None.

87.2.3.1 Request Parameters

tld - *list of strings*
comma separated list of tlds
for example, tlds=com,net.org

87.3 Response

87.3.1 Status Code

200 OK
Returns list of registry agreements

400 Bad Request
Invalid request, invalid parameter supplied

87.3.2 Response Body

List of registry agreements

87.4 Example get registry agreements for com,net,ca

Request

GET /domains/tlds/registryagreement?tlds=com,net,ca

Response

200 OK
Content-Type: application/json; charset=UTF-8

```
{ "agreements":  
  [  
    { "com": "" },  
    { "net": "" },  
    { "ca": "https://services.cira.ca/agree/agreement/agreementVersion2.0.jsp" },  
  ]  
}
```

87.5 Example get registry agreements (supplying invalid tld) for com,net,ca,invalidTld

Request

GET /domains/tlds/registryagreement?tlds=com,net,ca,invalidTld

Response

200 OK

Content-Type: application/json; charset=UTF-8

```
{ "agreements":  
  [  
    { "com": "" },  
    { "net": "" },  
    { "ca": "https://services.cira.ca/agree/agreement/agreementVersion2.0.jsp" },  
    { "invalidtld": "Tld not supported" }  
  ]  
}
```

87.6 See also

- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

88 Domains-tlds:GET

89 GET /domains/tlds

Get a list of available TLDs.

89.1 Contents

- 1 GET /domains/tlds
 - ◆ 1.1 Request
 - ◇ 1.1.1 Parameters
 - ◆ 1.2 Response
 - ◇ 1.2.1 Status Code
 - ◇ 1.2.2 Response Body
 - 1.2.2.1 Parameters
 - ◆ 1.3 See also

89.2 Request

GET /domains/tlds

89.2.1 Parameters

89.3 Response

89.3.1 Status Code

200 OK

A list of the TLDs available is returned.

500 Internal Server Error

There was an error getting the domain products from back end.

89.3.2 Response Body

```
[ "biz", "com", "info", "name", "net", "org", "us", "asia", "ae", "pro", "am", "ag", "bz", ... ]
```

89.3.2.1 Parameters

89.4 See also

- [Domain Name Registration API](#)
- [Name suggestion search](#)
- [Common Features](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

90 OrderingDomain

You can order a domain name using the [Ordering API](#). The ordering process should have the following steps. You need to create an order, create an account or set an existing account to use with the order, add a product, add service to the product, add attributes to the service and finally you need to validate the order, get a quote and submit the order. The general process is documented in the [Ordering API](#).

When creating a product, you need to use the product offering (specification) associated with the domain product, e.g.

```
{ "offering": "http://coreapi01.ote.chicago.hostway:8100/productSpecifications/core.plan.dnr.only/offerings/USD-12" }
```

When adding a service to the product you need to use service specification associated with domain name registration:

```
{  "name": "ng-test-fitness-20130412-0840.com",  "serviceSpecification": "http://coreapi01.ote.chicago.hostway:8100/serviceSpecifications/core.dnr.com",  "billingCycle": 12}
```

Finally, the service needs to have attributes such as domain contacts, nameservers, registration term, etc.:

Register .com

```
{  "domainName": "ng-test-fitness-20130412-0840.com",  "operation": "register",  "term": 1,  "authInfo": "mysecret",  "nameservers": [ "ns1.test.com", "ns2.test.com" ],  "contactInfo": {    "registrant": {      "type": "registrant",      "firstName": "John",      "lastName": "Doe",      "company": "ACME Inc.",      "street1": "123 First Str.",      "city": "Chicago",      "zipOrPostal": "34567",      "stateOrProvince": "IL",      "countryCode": "US",      "email1": "john.doe@test.com",      "phone1": "+1.312.456789098" },      "administrative": {        "type": "administrative",        ...      }    },    "billing": {      "type": "billing",      ...    },    "technical": {      "type": "technical",      ...    }  } }
```

Register .us

```
{  "domainName": "ng-test-fitness-20130412-0840.us",  "operation": "register",  "term": 1,  "authInfo": "mysecret",  "nameservers": [ "ns1.test.com", "ns2.test.com" ],  "extraParams": [{"name": "us_nexus", "value": "C11"}, {"name": "us_purpose", "value": "P1"}],  "contactInfo": {    "registrant": {      "type": "registrant",      "firstName": "John",      "lastName": "Doe",      "company": "ACME Inc.",      "street1": "123 First Str.",      "city": "Chicago",      "zipOrPostal": "34567",      "stateOrProvince": "IL",      "countryCode": "US",      "email1": "john.doe@test.com",      "phone1": "+1.312.456789098" },      "administrative": {        "type": "administrative",        ...      }    },    "billing": {      "type": "billing",      ...    },    "technical": {      "type": "technical",      ...    }  } }
```

90.1 Service Attributes

domainName - *string*
The domain name
operation - *string*

"register" to register a domain

term - *integer*
The registration term (period) in years

authInfo - *string*
Authentication info, currently not supported

nameservers - *list of strings*
Domain name servers (DNS) for the domain

extraParams - *list of extra params* (optional)
List of additional parameters required for some TLDs. Each parameter is an object with "name" and "value" fields. For example, .us registration requires the fields "us_nexus" which can be one of the following [C11, C12, C21, C31, C32] - see [What are Domain Name Application Purpose Codes?](#) and "us_purpose" which can be one of [P1, P2, P3, P4, P5] - see [What do the Nexus Codes mean?](#) You can find more information on extra parameters in the [GET /domains/tlds/{tld}/extparams](#) documentation.

company - *string* (optional)
The company or organization name

firstName - *string*
The first name of the contact person

lastName - *string*
The last name of the contact person

street1 - *string*
The street address

street2 - *string* (optional)
The second line of the street address (if any)

city - *string*
The city

countryCode - *string*
The international 2-letter country code

stateOrProvince - *string*
The state or province. For US states use the 2-letter state code

zipOrPostal - *string*
The postal code

email1 - *string*
The e-mail address of the contact person

phone1 - *string*
The phone number of the contact person in valid [E.164](#) format, e.g. +1.3209270625

90.2 See also

- [Ordering API](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)

91 OrderingTransferDomain

You can order a transfer domain name using the [Ordering API](#). The ordering process should have the following steps. You need to create an order, create an account or set an existing account to use with the order, add a product, add service to the product, add attributes to the service and finally you need to validate the order, get a quote and submit the order. The general process is documented in the [Ordering API](#).

When creating a product, you need to use the product offering (specification) associated with the domain product, e.g.

```
{ "offering": "http://coreapi01.ote.chicago.hostway:8100/productSpecifications/core.plan.dnr.only/offerings/USD-12" }
```

When adding a service to the product you need to use service specification associated with domain name registration:

```
{  "name": "ng-test-fitness-20130412-0840.com",  "serviceSpecification": "http://coreapi01.ote.chicago.hostway:8100/serviceSpecifications/core.dnr.com",  "billingCycle": 12}
```

Finally, the service needs to have attributes such as domain contacts, nameservers, registration term, etc.:

Transfer .com

```
{  "domainName": "ng-test-fitness-20130412-0840.com",  "operation": "transfer",  "term": 1,  "authInfo": "mysecret",  "nameservers": [ "ns1.test.com", "ns2.test.com" ],  "contactInfo": {    "registrant": {      "type": "registrant",      "firstName": "John",      "lastName": "Doe",      "company": "ACME Inc.",      "street1": "123 First Str.",      "city": "Chicago",      "zipOrPostal": "34567",      "stateOrProvince": "IL",      "countryCode": "US",      "email1": "john.doe@test.com",      "phone1": "+1.312.456789098" },      "administrative": {        "type": "administrative",        ...      }    },    "billing": {      "type": "billing",      ...    },    "technical": {      "type": "technical",      ...    }  } }
```

Transfer .us

```
{  "domainName": "ng-test-fitness-20130412-0840.us",  "operation": "transfer",  "term": 1,  "authInfo": "mysecret",  "nameservers": [ "ns1.test.com", "ns2.test.com" ],  "additionalParams": {    "usNexusCode": "{usNexusCode}",    "usNexusCategory": "{usNexusCategory}"  },  "contactInfo": {    "registrant": {      "type": "registrant",      "firstName": "John",      "lastName": "Doe",      "company": "ACME Inc.",      "street1": "123 First Str.",      "city": "Chicago",      "zipOrPostal": "34567",      "stateOrProvince": "IL",      "countryCode": "US",      "email1": "john.doe@test.com",      "phone1": "+1.312.456789098" },      "administrative": {        "type": "administrative",        ...      }    },    "billing": {      "type": "billing",      ...    },    "technical": {      "type": "technical",      ...    }  } }
```

91.1 Service Attributes

domainName - *string*
The domain name

operation - *string*
"register" to register a domain

term - *integer*
the registration term (period) in years, term must equal 1

authInfo - *string*
authentication info, currently not supported

nameservers - *list of strings*
domain name servers (DNS) for the domain

usNexusCode - *string* (optional) (required for .us)
one of the following [P1, P2, P3, P4, P5]

usNexusCategory - *string* (optional) (required for .us)
one of the following [C11, C12, C21, C31, C32]

company - *string* (optional)
The company or organization name

firstName - *string*
The first name of the contact person

lastName - *string*
The last name of the contact person

street1 - *string*
The street address

street2 - *string* (optional)
The second line of the street address (if any)

city - *string*
The city

countryCode - *string*
The international 2-letter country code

stateOrProvince - *string*
The state or province. For US states use the 2-letter state code

zipOrPostal - *string*
The postal code

email1 - *string*
The e-mail address of the contact person

phone1 - *string*
The phone number of the contact person in valid E.164 format, e.g. +1.3209270625

91.2 See also

- [Ordering API](#)

Please enable JavaScript to view the [comments powered by Disqus](#). [blog comments powered by Disqus](#)