Table of Contents

1. Overview

This manual describes the API commands for domain handling.

1.1. URL

All API access is over HTTP, and accessed from the **api.pananames.com** domain

http://api.pananames.com/api/registrar/

All requests are submitted using the following HTTP interface

BASE_URL*?****VERSION****=1&****SIGNATURE****=secret&****METHOD****=method&****PARAMETER1****=value1&* ***PARAMETER*** *2=value2&* ***PARAMETER*** *3=value3...*

The only way to secure transferred data is using HTTPS, no other security is currently possible.

1.2. Current Version

The current API version is 1.

1.3. Signature

Parameter **SIGNATURE** is merchant encoded constant string generated once when creating merchant.

1.4. Parameter's Types

INTEGER - integer number
TEXT - variable-length array of characters
DOMAIN - TEXT which represents a domain name
IPADDRESS - TEXT which represents an IP address
NAMESERVER - TEXT which represents domain name of the DNS server
TARGET - TEXT which represents domain name or URL to forward

1.5. Response

The server responds with "*Content-type: text/xml*"

1.6. API Roles

API supports one user role - merchant

2. Commands

2.1. Check Domain

Method

check

Description

This command is used to determine if an object can be provisioned within a repository. It provides a hint that allows a client to anticipate the success or failure of provisioning an object using the ***create*** command, as object-provisioning requirements are ultimately a matter of server policy.

Blocking Status

Non-blocking

Input

Parameter	Obligation	Definition	Type
VERSION	Required	API version	INTEGER
SIGNATURE	Required	Merchant encoded string	TEXT
METHOD	Required	Name of command to be executed	TEXT
DOMAIN	Required	Domain name	DOMAIN

Output

When a ***check*** command has been processed successfully, the <resData> element MUST contain a child <domain:chkData> element that identifies the domain namespace. The <domain:chkData> element contains the following child element:

- A <domain:name> element that contains the fully qualified name of the queried domain object. This element MUST contain an "avail" attribute whose value indicates object availability (can it be provisioned or not) at the moment the <check> command was completed. A value of "1" or "true" means that the object can be provisioned. A value of "0" or "false" means that the object can not be provisioned. This element has additional attributes:

  - “claimsExists”. This attribute indicates that domain is in claims period.
  - “claimsKey”. If attribute “claimsExists” = 1, then this attribute will have a non-empty string value.

  - “premium”. If value of "1", then this domain is premium. Prices for operations in element <price:prices>.

- A <price:prices> element contains the following child elements:

  - <price:create fee="1"> - price for register domain; This element MUST contain an "fee" attribute whose value indicates to set param FEE to register domain (METHOD = create)

  - <price:renew fee="1"> - price for renew domain; This element MUST contain an "fee" attribute whose value indicates to set param FEE to renew domain (METHOD = renew)

  - <price:transfer fee="1"> - price for transfer domain; This element MUST contain an "fee" attribute whose value indicates to set param FEE to transfer domain (METHOD = transferdomainsin)

Example

BASE_URL*?****VERSION****=1&****SIGNATURE****= a77d7895-8462-66e0-a1e1-04aafc84a8b6&****METHOD****=check&****DOMAIN****=domain.com*

Response

```
<?xml version="1.0"?>
<epp>
  <response>
    <result code="1000">
      <message>Command completed successfully.</message>
    </result>
    <resData>
      <domain:chkData xmlns:domain="https://serverscom/xmlns/domain.1-0">
        <domain:name avail="0" claimsExists="" claimsKey="" premium="1">SUPER.SITE</domain:name>
      </domain:chkData>
      <price:prices xmlns:price="https://serverscom/xmlns/domain.1-0">
        <price:create fee="1">1000.00</price:create>
        <price:renew fee="1">1000.00</price:renew>
        <price:transfer fee="1">1000.00</price:transfer>
      </price:prices>
    </resData>
    <trID>
      <clTRID>INTRUST-2222</clTRID>
      <svTRID>VS-1111</svTRID>
    </trID>
  </response>
</epp>
```

2.2. Create Domain

Method

create

Description

This command is sending request to register a domain name. In order to complete registration, you have to send SETWHOISINFO command within one hour after CREATE command.

Blocking Status

Blocking 3-5 sec.

Input

Parameter	Obligation	Definition	Type
VERSION	Required	API version	INTEGER
SIGNATURE	Required	Merchant encoded string	TEXT
METHOD	Required	Name of command to be executed	TEXT
DOMAIN	Required	Domain name	DOMAIN
PERIOD	Optional	Registration period (by default = 1)	INTEGER
USECUSTOMCONTACT	Optional	Use own Registrant, Administrative, Technical, Billing contactsIDs (by default = 0)	INTEGER
REGISTRANTCONTACTID	Optional (This is required if USECUSTOMCONTACT = 1)	Registrant contactID	TEXT
ADMINCONTACTID	Optional (This is required if USECUSTOMCONTACT = 1)	Administrative contactID	TEXT
TECHCONTACTID	Optional (This is required if USECUSTOMCONTACT = 1)	Technical contactID	TEXT
BILLCONTACTID	Optional (This is required if USECUSTOMCONTACT = 1)	Billing contactID	TEXT
FEE	Optional (if domain is premium and fee attribute = 1)	Price to register premium domain	FLOAT

Output

When a ***create*** command has been processed successfully, the <resData> element MUST contain a child <domain:creData> element that identifies the domain namespace. The <domain:creData> element contains the following child elements:

- A <domain:name> element that contains the fully qualified name of the domain object.

Example

BASE_URL*?****VERSION****=1&****SIGNATURE****= a77d7895-8462-66e0-a1e1-04aafc84a8b6&****METHOD****= create&****DOMAIN****=domain.com*

Response

```
<?xml version="1.0"?>
<epp>
  <response>
    <result code="1000">
      <msg>Command completed successfully</msg>
    </result>
    <resData>
      <domain:creData xmlns:domain="urn:ietf:params:xml:ns:domain-1.0"
xsi:schemaLocation="urn:ietf:params:xml:ns:domain-1.0 domain-1.0.xsd">
        <domain:name>DOMAIN.COM</domain:name>
      </domain:creData>
    </resData>
    <trID>
      <clTRID>1256830704657</clTRID>
      <svTRID>493443731-1256830704765-47275643727</svTRID>
    </trID>
  </response>
</epp>
```

2.3. Create Virtual Domain

Method

createvirtual

Description

Creates a domain name locally (in the system database). Used to add the domain name in the local system database.

Blocking Status

Blocking 2-3 sec

Input

Parameter	Obligation	Definition	Type
VERSION	Required	API version	INTEGER
SIGNATURE	Required	Merchant encoded string	TEXT
METHOD	Required	Name of command to be executed	TEXT
DOMAIN	Required	Domain name	DOMAIN
PROVIDER	Optional	Registration provider (by default determines automatically by TLD)	INTEGER
PERIOD	Optional	Registration period (by default = 1)	INTEGER

Output

When a ***createvirtual*** command has been processed successfully, the <resData> element MUST contain a child <domain:creData> element that identifies the domain namespace. The <domain:creData> element contains the following child elements:

- A <domain:name> element that contains the fully qualified name of the domain object.

- A <domain:crDate> element that contains the date and time of domain object creation.

- An <domain:exDate> element that contains the date and time identifying the end of the domain object's registration period.

Example

BASE_URL*?****VERSION****=1&****SIGNATURE****= a77d7895-8462-66e0-a1e1-04aafc84a8b6&****METHOD****= createvirtual&****DOMAIN****=domain.com*

Response

```
<?xml version="1.0"?>
<epp>
   <response>
      <result code="1000">
         <msg>Command completed successfully</msg>
      </result>
```

```
    <resData>
      <domain:creData xmlns:domain="urn:ietf:params:xml:ns:domain-1.0"
xsi:schemaLocation="urn:ietf:params:xml:ns:domain-1.0 domain-1.0.xsd">
        <domain:name>DOMAIN.COM</domain:name>
        <domain:crDate>2014-03-28T15:38:24.0000Z</domain:crDate>
        <domain:exDate>2015-03-28T15:38:24.0000Z</domain:exDate>
      </domain:creData>
    </resData>
    <trID>
      <clTRID>1256830704657</clTRID>
      <svTRID>493443731-1256830704765-47275643727</svTRID>
    </trID>
  </response>
</epp>
```

2.4. Info Domain

Method

info

Description

This command is used to retrieve information associated with a domain object.

Blocking Status

Non-blocking

Input

Parameter	Obligation	Definition	Type
VERSION	Required	API version	INTEGER
SIGNATURE	Required	Merchant encoded string	TEXT
METHOD	Required	Name of command to be executed	TEXT
DOMAIN	Required	Domain name	DOMAIN

Output

When an ***info*** command has been processed successfully, the <resData> element MUST contain a child <domain:infData> element that identifies the domain namespace. The <domain:infData> element contains the following child elements:

- A <domain:name> element that contains the fully qualified name of the domain object.

- Zero or more <domain:status> elements that contain the current status descriptors associated with the domain.

- Zero or more <domain:ns> element that contains the fully qualified names of the delegated host objects or host attributes (name servers) associated with the domain object.

- A <domain:clID> element that contains the identifier of the sponsoring client.

- An <domain:crDate> element that contains the date and time of domain object creation.

- An <domain:exDate> element that contains the date and time identifying the end of the domain object's registration period.

- An <domain:upDate> element that contains the date and time of the most recent domain-object modification. This element MUST NOT be present if the domain object has never been modified.

Example

BASE_URL*?****VERSION****=1&****SIGNATURE****= a77d7895-8462-66e0-a1e1-04aafc84a8b6&****METHOD****=info&****DOMAIN****=domain.com*

Response

```
<?xml version="1.0"?>
<epp>
  <response>
    <result code="1000">
      <message>Command completed successfully.</message>
    </result>
    <resData>
      <domain:infData xmlns:domain="https://serverscom/xmlns/domain.1-0">
        <domain:name>DOMAIN.COM</domain:name>
        <domain:status s="ok"> DOMAIN.COM</domain:status>
        <domain:ns>NS1.PANANAMES.COM</domain:ns>
        <domain:ns>NS2.PANANAMES.COM</domain:ns>
        <domain:extStatus s="addPeriod" endDate="2014-02-26T13:36:19.0852Z"/>
        <domain:clID>911</domain:clID>
        <domain:crDate>2014-02-21T13:36:19.0852Z</domain:crDate>
        <domain:exDate>2015-02-21T13:36:19.0852Z</domain:exDate>
        <domain:upDate>2014-02-24T10:31:21.0359Z</domain:upDate>
      </domain:infData>
    </resData>
    <trID>
      <clTRID>INTRUST-2222</clTRID>
      <svTRID>VS-1111</svTRID>
    </trID>
  </response>
```

</epp>

2.5. Renew Domain

Method

renew

Description

This command provides a transform operation that allows a client to extend the validity period of a domain object. Domain may be renewed only for a period 1 to 10 years.

Blocking Status

Non-blocking

Input

Parameter	Obligation	Definition	Type
VERSION	Required	API version	INTEGER
SIGNATURE	Required	Merchant encoded string	TEXT
METHOD	Required	Name of command to be executed	TEXT
DOMAIN	Required	Domain name	DOMAIN
PERIOD	Required	Renew period (min = 1, max = 10)	INTEGER
FEE	Optional (if domain is premium and fee attribute = 1)	Price to renew premium domain	FLOAT

Output

When a ***renew*** command has been processed successfully, the <resData> element MUST contain a child <domain:renData> element that identifies the domain namespace. The <domain:renData> element contains the following child elements:

- A <domain:name> element that contains the fully qualified name of the domain object.

- An <domain:exDate> element that contains the date and time identifying the end of the domain object's registration period.

Example

***BASE_URL**?**VERSION**=1&**SIGNATURE**= a77d7895-8462-66e0-a1e1-04aafc84a8b6&**METHOD**=renew&**DOMAIN**=domain.com&**PERIOD**=1*

Response

```
<?xml version="1.0"?>
<epp>
  <response>
    <result code="1000">
      <message>Command completed successfully.</message>
    </result>
    <resData>
      <domain:renData xmlns:domain=" https://api.pananames.com/xmlns/domain.1-0">
        <domain:name>DOMAIN.COM</domain:name>
        <domain:exDate>2015-01-20T19:04:54.0000Z</domain:exDate>
      </domain:renData>
    </resData>
    <trID>
      <clTRID>SERV-2222</clTRID>
      <svTRID>VS-1111</svTRID>
    </trID>
  </response>
</epp>
```

2.6. Delete Domain

Method

delete

Description

This command provides a transform operation that allows a client to delete a domain object. Domain object can be deleted at any time but only during first 5 (five) days it will be refunded by registry.

Blocking Status

Non-blocking

Input

Parameter	Obligation	Definition	Type
VERSION	Required	API version	INTEGER
SIGNATURE	Required	Merchant encoded string	TEXT
METHOD	Required	Name of command to be executed	TEXT
DOMAIN	Required	Domain name	DOMAIN

Output

When a ***delete*** command has been processed successfully, a server responses with empty <resData> element.

Example

BASE_URL*?****VERSION****=1&****SIGNATURE****= a77d7895-8462-66e0-a1e1-04aafc84a8b6&****METHOD****=delete&****DOMAIN****=domain.com*

Response

```
<?xml version="1.0"?>
<epp>
  <response>
    <result code="1000">
      <message>Command completed successfully.</message>
    </result>
    <resData/>
    <trID>
      <clTRID>SERV-2222</clTRID>
      <svTRID>VS-1111</svTRID>
    </trID>
  </response>
</epp>
```

2.7. Redeem Domain

Method

redeem

Description

Restore domain name during Redemption Grace Period.

Blocking Status

Non-blocking

Input

Parameter	Obligation	Definition	Type
VERSION	Required	API version	INTEGER

SIGNATURE	Required	Merchant encoded string	TEXT
METHOD	Required	Name of command to be executed	TEXT
DOMAIN	Required	Domain name	DOMAIN

Output

When a ***redeem*** command has been processed successfully, the <resData> element MUST contain a child <domain:renData> element that identifies the domain namespace. The <domain:renData> element contains the following child elements:

- A <domain:name> element that contains the fully qualified name of the domain object.

- An <domain:exDate> element that contains the date and time identifying the end of the domain object's registration period.

Example

BASE_URL*?*
VERSION*=1&*
SIGNATURE*= a77d7895-8462-66e0-a1e1-04aafc84a8b6&*
METHOD*=redeem&*
DOMAIN*=domain.com*

Response

```
<?xml version="1.0"?>
<epp>
  <response>
    <result code="1000">
      <message>Command completed successfully.</message>
    </result>
    <resData>
      <domain:renData xmlns:domain=" https://api.pananames.com/xmlns/domain.1-0">
        <domain:name>DOMAIN.COM</domain:name>
        <domain:exDate>2015-01-20T19:04:54.0000Z</domain:exDate>
      </domain:renData>
    </resData>
    <trID>
      <clTRID>SERV-2222</clTRID>
      <svTRID>VS-1111</svTRID>
    </trID>
  </response>
</epp>
```

2.8. Lock Domain

Method

lock

Description

Lock domain name to protect it from theft and accidental transfers. Will be disabled transfer operations with the domain name.

Blocking Status

Non-blocking

Input

Parameter	Obligation	Definition	Type
VERSION	Required	API version	INTEGER
SIGNATURE	Required	Merchant encoded string	TEXT
METHOD	Required	Name of command to be executed	TEXT
DOMAIN	Required	Domain name	DOMAIN

Output

When a ***lock*** command has been processed successfully, a server responses with empty <resData> element.

Example

***BASE_URL**?**VERSION**=1&**SIGNATURE**= a77d7895-8462-66e0-a1e1-04aafc84a8b6&**METHOD**=lock&**DOMAIN**=domain.com*

Response

```
<?xml version="1.0"?>
<epp>
  <response>
    <result code="1000">
      <message>Command completed successfully.</message>
    </result>
    <resData/>
    <trID>
      <clTRID>SERV-2222</clTRID>
      <svTRID>VS-1111</svTRID>
    </trID>
  </response>
</epp>
```

2.9. Unlock Domain

Method

unlock

Description

Unlock domain name to allow update, transfer and delete operations with the domain name.

Blocking Status

Non-blocking

Input

Parameter	Obligation	Definition	Type
VERSION	Required	API version	INTEGER
SIGNATURE	Required	Merchant encoded string	TEXT
METHOD	Required	Name of command to be executed	TEXT
DOMAIN	Required	Domain name	DOMAIN

Output

When a ***unlock*** command has been processed successfully, a server responses with empty <resData> element.

Example

BASE_URL*?****VERSION****=1&****SIGNATURE****= a77d7895-8462-66e0-a1e1-04aafc84a8b6&****METHOD****= unlock&****DOMAIN****=domain.com*

Response

```
<?xml version="1.0"?>
<epp>
  <response>
    <result code="1000">
      <message>Command completed successfully.</message>
    </result>
    <resData/>
    <trID>
      <clTRID>SERV-2222</clTRID>
      <svTRID>VS-1111</svTRID>
    </trID>
  </response>
</epp>
```

2.10.Get Auth Code

Method

getauthcode

Description

Get authorization code of the domain name.

Blocking Status

Non-blocking

Input

Parameter	Obligation	Definition	Type
VERSION	Required	API version	INTEGER
SIGNATURE	Required	Merchant encoded string	TEXT
METHOD	Required	Name of command to be executed	TEXT
DOMAIN	Required	Domain name	DOMAIN

Output

When an ***getauthcode*** command has been processed successfully, the <resData> element MUST contain a child <domain:infData> element that identifies the domain namespace. The <domain:infData> element contains the following child elements:

- A <domain:name> element that contains the fully qualified name of the domain object to be queried.

- A <domain:authInfo> element that contains authorization information associated with the domain object.

Example

BASE_URL*?****VERSION****=1&****SIGNATURE****= a77d7895-8462-66e0-a1e1-04aafc84a8b6&****METHOD****= getauthcode&****DOMAIN****=domain.com*

Response

```
<?xml version="1.0"?>
<epp>
  <response>
    <result code="1000">
      <message>Command completed successfully.</message>
    </result>
    <resData>
      <domain:infData xmlns:domain="https://api.pananames.com/xmlns/domain.1-0">
        <domain:name>DOMAIN.COM</domain:name>
        <domain:authInfo>
          <domain:pw>#$13565152475@48</domain:pw>
        </domain:authInfo>
      </domain:infData>
    </resData>
    <trID>
      <clTRID>SERV-2222</clTRID>
      <svTRID>VS-1111</svTRID>
    </trID>
  </response>
</epp>
```

2.11.Set Auth Code

Method

setauthcode

Description

Set authorization code of the domain name.

Blocking Status

Non-blocking

Input

Parameter	Obligation	Definition	Type
VERSION	Required	API version	INTEGER
SIGNATURE	Required	Merchant encoded string	TEXT

METHOD	Required	Name of command to be executed	TEXT
DOMAIN	Required	Domain name	DOMAIN
AUTHCODE	Required	Authorization code	TEXT

Output

When a ***setauthcode*** command has been processed successfully, a server responses with empty <resData> element.

Example

BASE_URL?***VERSION***=1&***SIGNATURE***= a77d7895-8462-66e0-a1e1-04aafc84a8b6&***METHOD***= setauthcode&***DOMAIN***=domain.com& ***AUTHCODE***=@56134628)5$5682

Response

```
<?xml version="1.0"?>
<epp>
  <response>
    <result code="1000">
      <message>Command completed successfully.</message>
    </result>
    <resData>
      <domain:infData xmlns:domain="https://api.pananames.com/xmlns/domain.1-0">
        <domain:name>DOMAIN.COM</domain:name>
        <domain:authInfo>
          <domain:pw>#$13565152475@48</domain:pw>
        </domain:authInfo>
      </domain:infData>
    </resData>
    <trID>
      <clTRID>SERV-2222</clTRID>
      <svTRID>VS-1111</svTRID>
    </trID>
  </response>
</epp>
```

2.12.Get Domain Host Names

Method

getdomainhostnames

Description

Provides ability to get the list of existing host names.

Blocking Status

Non-blocking

Input

Parameter	Obligation	Definition	Type
VERSION	Required	API version	INTEGER
SIGNATURE	Required	Merchant encoded string	TEXT
METHOD	Required	Name of command to be executed	TEXT
DOMAIN	Required	Domain name	DOMAIN

Output

When a createns command has been processed successfully, <resData> element should containt 2 child elements: <domain:infData> and <hostNames:list>. 1^{St} one (infData) has the following elements:

- A <domain:name>is FQDN

- A <domain:status > has current domain status (ok, pendingDelete, etc)

2nd one (list) has 0 or more <hostNames:name> child elements with host name as value and the following properties as additional params: 'ipv4', 'ipv6', 'crDate', 'upDate' and properties.

Example

BASE_URL*?****VERSION****=1&****SIGNATURE****= a77d7895-8462-66e0-a1e1-04aafc84a8b6&****METHOD****=getdomainhostnames&****DOMAIN****=domain.com*

Response

```
<epp>

   <response>

      <result code="1000">

         <message>Command completed successfully.</message>

      </result>
```

```
    <resData>
      <domain:infData xmlns:domain="https://serverscom/xmlns/domain.1-0">
        <domain:name>DOMAIN.COM</domain:name>
        <domain:status s="ok">DOMAIN.COM</domain:status>
      </domain:infData>
        <hostNames:list xmlns:hostNames="https://serverscom/xmlns/domain/host-names.1-0">
        <hostNames:name ipv4="188.42.236.251" crDate="2015-06-26 13:07:11" upDate="2015-06-26 16:07:11">NS1</hostNames:name>
      </hostNames:list>
    </resData>
    <trID>
      <clTRID>INTRUST-2222</clTRID>
      <svTRID>VS-1111</svTRID>
    </trID>
  </response>
</epp>
```

2.13.Create Name Server

Method

createns

Description

This command provides a transform operation that allows a client to create a host object. Create internal name server for domain.

Blocking Status

Non-blocking

Input

Parameter	Obligation	Definition	Type
VERSION	Required	API version	INTEGER
SIGNATURE	Required	Merchant encoded string	TEXT
METHOD	Required	Name of command to be executed	TEXT
DOMAIN	Required	Domain name	DOMAIN
NS	Required	Name of NS. For example: *ns1*	TEXT
IP	Required *	IP address of name server	IPADDRESS
IPV6	Required *	IPv6 address of name server	IPv6ADDRESS

* must be provided IP or IPV6 or IP+IPV6

Output

When a ***createns*** command has been processed successfully, a server responses with empty <resData> element.

Example

BASE_URL*?****VERSION****=1&****SIGNATURE****= a77d7895-8462-66e0-a1e1-04aafc84a8b6&****METHOD****= createns&****DOMAIN****=domain.com&****NS****=ns1&****IP****=111.222.111.222&****IPV6****=2a00:1450:4001:804:0:0:0:200b*

Response

```
<?xml version="1.0"?>
<epp>
  <response>
    <result code="1000">
      <message>Command completed successfully.</message>
    </result>
    <resData/>
    <trID>
      <clTRID>SERV-2222</clTRID>
      <svTRID>VS-1111</svTRID>
    </trID>
  </response>
</epp >
```

2.14.Delete Host Name

Method

deletehost

Description

Provides ability to delete host name.

Blocking Status

Non-blocking

Input

Parameter	Obligation	Definition	Type
VERSION	Required	API version	INTEGER
SIGNATURE	Required	Merchant encoded string	TEXT
METHOD	Required	Name of command to be executed	TEXT
DOMAIN	Required	Domain name	DOMAIN
NS	Required	Host name to delete	HOST NAME

Output

When a createns command has been processed successfully API server responds with empty <resData>.

Example

BASE_URL*?****VERSION****=1&****SIGNATURE****= a77d7895-8462-66e0-a1e1-04aafc84a8b6&****METHOD****=hostname&****DOMAIN****=domain.com&****NS****=NS1*

Response

```
<epp>

  <response>

    <result code="1000">

      <message>Command completed successfully.</message>

    </result>

    <resData/>

    <trID>

      <clTRID>INTRUST-2222</clTRID>

      <svTRID>VS-1111</svTRID>
    </trID>
  </response>
</epp>
```

2.15.Get Name Servers

Method

getnses

Description

Get current name servers of the domain name. Provide list of delegated host objects or host attributes (name servers) associated with the domain object.

Blocking Status

Non-blocking

Input

Parameter	Obligation	Definition	Type
VERSION	Required	API version	INTEGER
SIGNATURE	Required	Merchant encoded string	TEXT
METHOD	Required	Name of command to be executed	TEXT
DOMAIN	Required	Domain name	DOMAIN

Output

When an ***getnses*** command has been processed successfully, the <resData> element MUST contain a child <domain:infData> element that identifies the domain namespace. The <domain:infData> element contains the following child elements:

- A <domain:name> element that contains the fully qualified name of the domain object to be queried.

- A <domain:ns> element that contains the fully qualified names of the delegated host objects or host attributes (name servers) associated with the domain object to provide resolution services for the domain. A host object MUST be known to the server before the host object can be associated with a domain object.

Example

***BASE_URL**?**VERSION**=1&**SIGNATURE**= a77d7895-8462-66e0-a1e1-04aafc84a8b6&**METHOD**= getnses&**DOMAIN**=domain.com*

Response

```
<?xml version="1.0"?>
<epp>
  <response>
    <result code="1000">
      <message>Command completed successfully.</message>
    </result>
    <resData>
      <domain:infData xmlns:domain="https://api.pananames.com/xmlns/whois.1-0">
        <domain:name>DOMAIN.COM</domain:name>
        <domain:ns>
          <domain:hostObj>NS1.PANANAMES.COM</domain:hostObj>
          <domain:hostObj>NS2.PANANAMES.COM </domain:hostObj>
        </domain:ns>
      </domain:infData>
    </resData>
    <trID>
      <clTRID>SERV-2222</clTRID>
      <svTRID>VS-1111</svTRID>
    </trID>
  </response>
</epp>
```

2.16.Set Name Servers

Method

setnses

Description

Set name servers to the domain name. Associate provided name servers with the domain object (List of currently delegated host objects or host attributes (name servers) associated with the domain object can be retrieved via ***getnses*** command).

Blocking Status

Non-blocking

Input

Parameter	Obligation	Definition	Type
VERSION	Required	API version	INTEGER
SIGNATURE	Required	Merchant encoded string	TEXT
METHOD	Required	Name of command to be executed	TEXT
DOMAIN	Required	Domain name	DOMAIN
NS1	Required	Name server hostname	NAMESERVER
NS2	Required	Name server hostname	NAMESERVER
NS3	Optional	Name server hostname	NAMESERVER
NS4	Optional	Name server hostname	NAMESERVER
NS5	Optional	Name server hostname	NAMESERVER

Output

When a ***setnses*** command has been processed successfully, a server responses with empty <resData> element.

Example

BASE_URL*?****VERSION****=1&****SIGNATURE****= a77d7895-8462-66e0-a1e1-04aafc84a8b6&****METHOD****= setnses&****DOMAIN****=domain.com&****NS1****=ns1.pananames.com&****NS2****=ns2.pananames.com*

Response

```
<?xml version="1.0"?>
<epp>
  <response>
    <result code="1000">
      <message>Command completed successfully.</message>
    </result>
    <resData/>
    <trID>
      <clTRID>SERV-2222</clTRID>
      <svTRID>VS-1111</svTRID>
    </trID>
  </response>
</epp >
```

2.17.Get Host

Method

gethost

Description

Get DNS SOA and Resource Records of the domain name.

Blocking Status

Non-blocking

Input

Parameter	Obligation	Definition	Type
VERSION	Required	API version	INTEGER
SIGNATURE	Required	Merchant encoded string	TEXT
METHOD	Required	Name of command to be executed	TEXT
DOMAIN	Required	Domain name	DOMAIN

Output

When an ***gethost*** command has been processed successfully, the <resData> element MUST contain a child <soa:info> element that identifies the SOA record namespace and <host:hosts> element that contain list of <host:info> elements contained RR namespace.

The <soa:info> element contains the following child elements:

- A <soa:origin> element that contains the fully qualified name of the primary master server of the zone

- A <soa:ns> element that contains FQDN of the name server

- A <soa:mbox> element that contains email address of the person responsible for this zone and to which email may be sent to report errors or problems.

- A <soa:serial> element that contains serial number. Unsigned 32 bit value in range 1 to 4294967295 with a maximum increment of 2147483647.

- A <soa:refresh> element that contains the time when the slave will try to refresh the zone from the master (by reading the master DNS SOA RR). Signed 32 bit time value in seconds.

- A <soa:retry> element that defines the time between retries if the slave (secondary) fails to contact the master when *refresh* (above) has expired. Signed 32 bit value in seconds.

- A <soa:expire> element that indicates when the zone data is no longer authoritative. Used by Slave or (Secondary) servers only. Signed 32 bit value in seconds.

- A <soa:minimum> element that contains the minimum in seconds.

- A <soa:ttl> element that contains the TTL value

- A <soa:own> element that contains a normal owner or label field

The <host:info> element contains the following child elements:

- A <host:id> element that contains the ID of the record in DB

- A <host:name> element that contains the name of the record

- A <host:type> element that contains the type of the record

- A <host:data> element that contains the data of the record

- A <host:aux> element that contains the AUX of the record (specifies the priority for these records)

- A <host:ttl> element that contains the TTL (Time To Live) of the record

Example

_**BASE_URL**?**VERSION**=1&**SIGNATURE**= a77d7895-8462-66e0-a1e1-04aafc84a8b6&**METHOD**= gethost&**DOMAIN**=domain.com_

Response

```
<?xml version="1.0"?>
<epp>
  <response>
    <result code="1000">
      <message>Command completed successfully.</message>
    </result>
    <resData>
      <soa:info xmlns:soa="https://api.pananames.com/xmlns/soa.1-0">
        <soa:origin>DOMAIN.COM.</soa:origin>
        <soa:ns>NS1.PANANAMES.COM.</soa:ns>
        <soa:mbox/>
        <soa:serial>2011053003</soa:serial>
        <soa:refresh>28800</soa:refresh>
        <soa:retry>7200</soa:retry>
        <soa:expire>604800</soa:expire>
        <soa:minimum>86400</soa:minimum>
        <soa:ttl>300</soa:ttl>
        <soa:own/>
      </soa:info>
      <host:hosts xmlns:host="https://api.pananames.com/xmlns/host.1-0">
```

```
        <host:info>
          <host:id>1</host:id>
          <host:name/>
          <host:type>CNAME</host:type>
          <host:data>servers.net.</host:data>
          <host:aux>0</host:aux>
          <host:ttl>300</host:ttl>
        </host:info>
        <host:info>
          <host:id>3</host:id>
          <host:name>*</host:name>
          <host:type>CNAME</host:type>
          <host:data>servers.net.</host:data>
          <host:aux>0</host:aux>
          <host:ttl>300</host:ttl>
        </host:info>
        <host:info>
          <host:id>2</host:id>
          <host:name>www</host:name>
          <host:type>CNAME</host:type>
          <host:data>servers.net.</host:data>
          <host:aux>0</host:aux>
          <host:ttl>300</host:ttl>
        </host:info>
      </host:hosts>
    </resData>
    <trID>
      <clTRID>SERV-2222</clTRID>
      <svTRID>VS-1111</svTRID>
    </trID>
  </response>
</epp>
```

2.18.Set Host

Method

sethost

Description

Set DNS SOA and Resource Records of the domain name.

Blocking Status

Non-blocking

Input

Parameter	Obligation	Definition	Type
VERSION	Required	API version	INTEGER
SIGNATURE	Required	Merchant encoded string	TEXT
METHOD	Required	Name of command to be executed	TEXT
DOMAIN	Required	Domain name	DOMAIN
HOSTS_COUNT	Required	Host records count	INTEGER
NAME[N]	Optional	Name of the host record	TEXT
TYPE[N]	Required	Type of the host record	TEXT
DATA[N]	Required	Data of the host record	TEXT
AUX[N]	Required	AUX of the host record	INTEGER
TTL[N]	Optional	TTL of the host record	INTEGER

Output

When a ***sethost*** command has been processed successfully, a server responses with empty <resData> element.

Example

__BASE_URL__?__VERSION__=1&__SIGNATURE__= a77d7895-8462-66e0-a1e1-04aafc84a8b6&__METHOD__=sethost&__DOMAIN__=domain.com&__HOSTS_COUNT__=1&__NAME1__=bio&__TYPE1__=TXT&__DATA1__=2e7c55685c08bc768f3a6a9a92c199a40a948946&__AUX1__=0&__TTL1__=200

Response

```
<?xml version="1.0"?>
<epp>
  <response>
    <result code="1000">
      <message>Command completed successfully.</message>
    </result>
    <resData/>
    <trID>
      <clTRID>SERV-2222</clTRID>
      <svTRID>VS-1111</svTRID>
    </trID>
  </response>
</epp >
```

2.19.Get Redirects

Method

getredirects

Description

Get domain name forwarding info.

Blocking Status

Non-blocking

Input

Parameter	Obligation	Definition	Type
VERSION	Required	API version	INTEGER
SIGNATURE	Required	Merchant encoded string	TEXT
METHOD	Required	Name of command to be executed	TEXT
DOMAIN	Required	Domain name	DOMAIN

Output

When an ***getredirects*** command has been processed successfully, the <resData> element MUST contain a child <redirects:info> element that identifies the forwarding record namespace. The < redirects:info> element contains the following child elements:

- A <redirects:host> element that contains the fully qualified name of the domain object

- A <redirects:target> element that contains URL or FQDN of the forwarding target

- A <redirects:type> element that contains type of the forwarding

Example

BASE_URL*?****VERSION****=1&****SIGNATURE****= a77d7895-8462-66e0-a1e1-04aafc84a8b6&****METHOD****= getredirects&****DOMAIN****=domain.com*

Response

```
< ?xml version="1.0"?>
<epp>
   <response>
      <result code="1000">
         <message>Command completed successfully.</message>
```

```
    </result>
    <resData>
      <redirects:info xmlns:redirects="https://api.pananames.com/xmlns/redirects.1-0">
        <redirects:host>DOMAIN.COM</redirects:host>
        <redirects:target>OTHERDOMAIN.COM</redirects:target>
        <redirects:type>1</redirects:type>
      </redirects:info>
    </resData>
    <trID>
      <clTRID>SERV-2222</clTRID>
      <svTRID>VS-1111</svTRID>
    </trID>
  </response>
</epp>
```

2.20.Set Redirects

Method

setredirects

Description

Set domain name forwarding info.

Blocking Status

Non-blocking

Input

Parameter	Obligation	Definition	Type
VERSION	Required	API version	INTEGER
SIGNATURE	Required	Merchant encoded string	TEXT
METHOD	Required	Name of command to be executed	TEXT
DOMAIN	Required	Domain name	DOMAIN
ENABLED	Required	Indicates that forwarding enabled	INTEGER (0 or 1)
TARGET	Required	Forwarding target (URL or domain name)	TARGET
TYPE	Required	Forwarding type. Possible values: 1 – Domain Forwarding 2 – URL Forwarding	INTEGER

Output

When a ***setredirects*** command has been processed successfully, a server responses with empty <resData> element.

Example

BASE_URL*?****VERSION****=1&****SIGNATURE****= a77d7895-8462-66e0-a1e1-04aafc84a8b6&****METHOD****= setredirects&****DOMAIN****=domain.com&****ENABLED****=1&****TARGET****=otherdomain.com&****TYPE****=1*

Response

```
<?xml version="1.0"?>
<epp>
   <response>
      <result code="1000">
         <message>Command completed successfully.</message>
      </result>
      <resData/>
      <trID>
         <clTRID>SERV-2222</clTRID>
         <svTRID>VS-1111</svTRID>
      </trID>
   </response>
</epp >
```

2.21.Get WHOIS Info

Method

getwhoisinfo

Description

Get WHOIS information of the domain name.

Blocking Status

Non-blocking

Input

Parameter	Obligation	Definition	Type
VERSION	Required	API version	INTEGER
SIGNATURE	Required	Merchant encoded string	TEXT

METHOD	Required	Name of command to be executed	TEXT
DOMAIN	Required	Domain name	DOMAIN
SHOWPRIVATE	Optional	If privacy protection enabled, for see public data, set SHOWPRIVATE = 1	INTEGER

Output

When an ***getwhoisinfo*** command has been processed successfully, the <resData> element MUST contain a child <whois:info> element that identifies the WHOIS record namespace. The < whois:info> element contains the following child elements:

- A < whois:domain> element that contains information about domain object

- A < whois:adminContact > element that contains information about administrative contact

- A < whois:techContact > element that contains information about tech contact

- A < whois:billContact > element that contains information about billing contact

Example

BASE_URL_?_***VERSION***_=1&_***SIGNATURE***_= a77d7895-8462-66e0-a1e1-04aafc84a8b6&_***METHOD***_=_
getwhoisinfo&***DOMAIN***_=domain.com_

Response

```
<?xml version="1.0"?>
<epp>
  <response>
    <result code="1000">
      <message>Command completed successfully.</message>
    </result>
    <resData>
      <whois:info xmlns:whois="https://serverscom/xmlns/whois.1-0">
        <whois:domain>
          <whois:name>DOMAIN.COM</whois:name>
          <whois:crDate/>
          <whois:exDate/>
          <whois:upDate/>
        </whois:domain>
        <whois:regContact private="true">
          <whois:org>Company Inc</whois:org>
          <whois:name>John Doe</whois:name>
          <whois:email>admin@domain.com</whois:email>
          <whois:address>123 Freedom Str., Suite 456</whois:address>
          <whois:city>Dallas</whois:city>
          <whois:state></whois:state>
          <whois:zip>3105</whois:zip>
          <whois:country> US</whois:country>
          <whois:countryCode>1</whois:countryCode>
          <whois:phone>+1. 1234567890</whois:phone>
        </whois:regContact>
        <whois:adminContact private="true">
          <whois:org>Company Inc</whois:org>
          <whois:name>John Doe</whois:name>
          <whois:email>admin@domain.com</whois:email>
          <whois:address>123 Freedom Str., Suite 456</whois:address>
          <whois:city>Dallas</whois:city>
          <whois:state></whois:state>
          <whois:zip>3105</whois:zip>
          <whois:country> US</whois:country>
          <whois:countryCode>1</whois:countryCode>
          <whois:phone>+1. 1234567890</whois:phone>
        </whois:adminContact>
        <whois:techContact private="true">
          <whois:org> Company Inc </whois:org>
          <whois:name> John Doe </whois:name>
          <whois:email> admin@domain.com </whois:email>
          <whois:address>123 Freedom Str., Suite 456</whois:address>
          <whois:city> Dallas </whois:city>
          <whois:state> TX </whois:state>
          <whois:zip>3105</whois:zip>
          <whois:country> US</whois:country>
          <whois:countryCode>1</whois:countryCode>
          <whois:phone>+1. 1234567890</whois:phone>
        </whois:techContact>
        <whois:billContact private="true">
          <whois:org> Company Inc </whois:org>
          <whois:name> John Doe </whois:name>
          <whois:email> admin@domain.com </whois:email>
          <whois:address>319 28th Oktovriou Str., Suite 201</whois:address>
          <whois:city> Dallas </whois:city>
          <whois:state>TX</whois:state>
          <whois:zip>3105</whois:zip>
          <whois:country>US</whois:country>
          <whois:countryCode>1</whois:countryCode>
          <whois:phone>+1. 1234567890</whois:phone>
        </whois:billContact>
      </whois:info>
```

```
    </resData>
    <trID>
      <clTRID>INTRUST-2222</clTRID>
      <svTRID>VS-1111</svTRID>
    </trID>
  </response>
</epp>
```

2.22.Set WHOIS Info

Method

setwhoisinfo

Description

Set WHOIS information of the domain name.

Blocking Status

Non-blocking

Input

Parameter	Obligation	Definition	Type
VERSION	Required	API version	INTEGER
SIGNATURE	Required	Merchant encoded string	TEXT
METHOD	Required	Name of command to be executed	TEXT
DOMAIN	Required	Domain name	DOMAIN
REGORG	Required	Registrant contact organization	TEXT
REGNAME	Required	Registrant contact name	TEXT
REGEMAIL	Required	Registrant contact email	TEXT
REGADDRESS	Required	Registrant contact address	TEXT
REGCITY	Required	Registrant contact city	TEXT
REGSTATE	Required	Registrant contact state	TEXT
REGZIP	Required	Registrant contact ZIP	TEXT
REGCOUNTRY	Required	Registrant contact country	TEXT
REGCC	Optional	Registrant country code	TEXT
REGPHONE	Required	Registrant contact phone	TEXT
ADMINORG	Required	Admin contact organization	TEXT
ADMINNAME	Required	Admin contact name	TEXT
ADMINEMAIL	Required	Admin contact email	TEXT
ADMINADDRESS	Required	Admin contact address	TEXT
ADMINCITY	Required	Admin contact city	TEXT
ADMINSTATE	Required	Admin contact state	TEXT
ADMINZIP	Required	Admin contact ZIP	TEXT

ADMINCOUNTRY	Required	Admin contact country	TEXT
ADMINCC	Optional	Admin country code	TEXT
ADMINPHONE	Required	Admin contact phone	TEXT
TECHORG	Required	Tech contact organization	TEXT
TECHNAME	Required	Tech contact name	TEXT
TECHEMAIL	Required	Tech contact email	TEXT
TECHADDRESS	Required	Tech contact address	TEXT
TECHCITY	Required	Tech contact city	TEXT
TECHSTATE	Required	Tech contact state	TEXT
TECHZIP	Required	Tech contact ZIP	TEXT
TECHCOUNTRY	Required	Tech contact country	TEXT
TECHCC	Optional	Tech country code	TEXT
TECHPHONE	Required	Tech contact phone	TEXT
BILLORG	Required	Billing contact organization	TEXT
BILLNAME	Required	Billing contact name	TEXT
BILLEMAIL	Required	Billing contact email	TEXT
BILLADDRESS	Required	Billing contact address	TEXT
BILLCITY	Required	Billing contact city	TEXT
BILLSTATE	Required	Billing contact state	TEXT
BILLZIP	Required	Billing contact ZIP	TEXT
BILLCOUNTRY	Required	Billing contact country	TEXT
BILLCC	Optional	Billing country code	TEXT
BILLPHONE	Required	Billing contact phone	TEXT
PRIVATE	Required	Indicates that WHOIS privacy protection enabled	INTEGER (0 or 1)

Output

When a ***setwhoisinfo*** command has been processed successfully, a server responses with empty <resData> element.

Example

***BASE_URL**?**VERSION**=1&**SIGNATURE**= a77d7895-8462-66e0-a1e1-04aafc84a8b6&**METHOD**= setwhoisinfo&**DOMAIN**=domain.com&**ADMINORG**=Company Inc&**ADMINNAME**=John Doe&**ADMINEMAIL**=admin@pananames.com&**ADMINADDRESS**=123 Freedom Str., Suite 456&**ADMINCITY**=Dallas&**ADMINSTATE**= TX&**ADMINZIP**=90210&**ADMINCOUNTRY**=US&**ADMINPHONE**=+1. 1234567890&**TECHORG**= Company Inc&**TECHNAME**= John Doe&**TECHEMAIL**=admin@domain.com&**TECHADDRESS**=123 Freedom Str., Suite 456&**TECHCITY**= Dallas&**TECHSTATE**= TX&**TECHZIP**=90210&**TECHCOUNTRY**=US&**TECHPHONE**=+1.1234567890&**BILLORG**= Company Inc&**BILLNAME**= John Doe&**BILLEMAIL**=admin@domain.com&**BILLADDRESS**=123 Freedom Str., Suite 456&**BILLCITY**= Dallas&**BILLSTATE**= TX&**BILLZIP**=90210&**BILLCOUNTRY**=US&**BILLPHONE**=+1. 1234567890&**PRIVATE**=1*

Response

```
<?xml version="1.0"?>
<epp>
  <response>
    <result code="1000">
      <message>Command completed successfully.</message>
    </result>
    <resData/>
    <trID>
      <clTRID>SERV-2222</clTRID>
      <svTRID>VS-1111</svTRID>
    </trID>
  </response>
</epp >
```

2.23.Get WHOIS Emails

Method

getwhoisemails

Description

Get emails from WHOIS information of the domain name.

Blocking Status

Non-blocking

Input

Parameter	Obligation	Definition	Type
VERSION	Required	API version	INTEGER
SIGNATURE	Required	Merchant encoded string	TEXT
METHOD	Required	Name of command to be executed	TEXT
DOMAIN	Required	Domain name	DOMAIN

Output

When a ***getwhoisemails*** command has been processed successfully, the <resData> element MUST contain a child <domain:infData> element that identifies the domain namespace. The <domain:infData> element contains the following child elements:

- A <domain:name> element that contains fully qualified name of the domain object

- A <domain:emails> element that contains list of WHOIS emails

Example

__BASE_URL__?__VERSION__=1&__SIGNATURE__= a77d7895-8462-66e0-a1e1-04aafc84a8b6&__METHOD__= getwhoisemails&__DOMAIN__=domain.com

Response

```
<?xml version="1.0"?>
<epp>
  <response>
    <result code="1000">
      <message>Command completed successfully.</message>
    </result>
    <resData>
      <domain:infData xmlns:domain="https://serverscom/xmlns/domain.1-0">
        <domain:name>DOMAIN.COM</domain:name>
        <domain:emails>dns-admin@google.com, compliance@markmonitor.com, contact-admin@google.com
        </domain:emails>
      </domain:infData>
    </resData>
    <trID>
      <clTRID>INTRUST-2222</clTRID>
      <svTRID>VS-1111</svTRID>
    </trID>
  </response>
</epp>
```

2.24.Set Pending Transfer

Method

setpendingtransfer

Description

Process domain name transfer out.

Blocking Status

Non-blocking

Input

Parameter	Obligation	Definition	Type
VERSION	Required	API version	INTEGER
SIGNATURE	Required	Merchant encoded string	TEXT
METHOD	Required	Name of command to be executed	TEXT
DOMAIN	Required	Domain name	DOMAIN
OPERATION	Required	Operation to perform. Possible values: 1 - Approve transfer out 4 - Cancel transfer out	INTEGER (1 or 4)

Output

When a ***setpendingtransfer*** command has been processed successfully, a server responses with empty <resData> element.

Example

BASE_URL*?****VERSION****=1&****SIGNATURE****= a77d7895-8462-66e0-a1e1-04aafc84a8b6&****METHOD****= setpendingtransfer&****DOMAIN****=domain.com&****OPERATION****=1*

Response

```
<?xml version="1.0"?>
<epp>
  <response>
    <result code="1000">
      <message>Command completed successfully.</message>
    </result>
    <resData/>
    <trID>
      <clTRID>SERV-2222</clTRID>
      <svTRID>VS-1111</svTRID>
    </trID>
  </response>
</epp>
```

2.25.Transfer Domain In

Method

transferdomainsin

Description

Process domain name transfer in.

Blocking Status

Non-blocking

Input

Parameter	Obligation	Definition	Type
VERSION	Required	API version	INTEGER
SIGNATURE	Required	Merchant encoded string	TEXT
METHOD	Required	Name of command to be executed	TEXT
DOMAIN	Required	Domain name	DOMAIN
AUTHCODE	Required	Authorization code	TEXT
OPERATION	Required	Operation to perform. Possible values: 2 - Cancel transfer in 5 - Initiate transfer in	INTEGER (2 or 5)
FEE	Optional (if domain is premium and fee attribute = 1)	Price to transfer premium domain	FLOAT

Output

When a ***transferdomainsin*** command has been processed successfully, the <resData> element MUST contain a child <domain:infData> element that identifies the domain namespace. The <domain:infData> element contains the following child elements:

- A <domain:name> element that contains fully qualified name of the domain object

- A <domain:status> element that contains domain object transfer status

- A <domain:acID> element that contains the identifier of the client that SHOULD act upon a PENDING transfer request

- A <domain:crDate> element that contains the date and time of domain object transfer creation.

- A <domain:exDate> element that contains the end of the domain object's validity period if the other transfer command caused or causes a change in the validity period

Example

BASE_URL*?***VERSION***=1&***SIGNATURE***= a77d7895-8462-66e0-a1e1-04aafc84a8b6&***METHOD***= transferdomainsin&***DOMAIN***=domain.com&***AUTHCODE***=@56134628&5$5777&***OPERATION***=5*

Response

```
<?xml version="1.0"?>
<epp>
  <response>
    <result code="1000">
      <message>Command completed successfully.</message>
    </result>
    <resData>
      <domain:infData xmlns:domain="https://api.pananames.com/xmlns/domain.1-0">
        <domain:name>DOMAIN.COM</domain:name>
        <domain:status>pending</domain:status>
        <domain:acID>1243</domain:acID>
        <domain:crDate>2010-10-30T20:04:40.0000Z</domain:crDate>
        <domain:exDate>2011-10-15T18:07:13.0000Z</domain:exDate>
      </domain:infData>
    </resData>
    <trID>
      <clTRID>SERV-2222</clTRID>
      <svTRID>VS-1111</svTRID>
    </trID>
  </response>
</epp>
```

2.26.Get Transfer Status

Method

gettransferstatus

Description

Get transfer status of domain.

Blocking Status

Non-blocking

Input

Parameter	Obligation	Definition	Type
VERSION	Required	API version	INTEGER
SIGNATURE	Required	Merchant encoded string	TEXT
METHOD	Required	Name of command to be executed	TEXT
DOMAIN	Required	Domain name	DOMAIN

Output

When an ***gettransferstatus*** command has been processed successfully, the <resData> element MUST contain a child <transfer:infData> element that identifies the transfer namespace. The <transfer:infData> element contains the following child elements:

- A <transfer:name> element that contains the fully qualified name of the domain object to be queried.

- A <transfer:status> element that contains the transfer status of domain. Possible statuses:

0 - unknown
1 - pending
2 - client approved
3 - client rejected
4 - client cancelled
5 - server approved
6 - waiting confirm

Example

__BASE_URL__?__VERSION__=1&__SIGNATURE__= a77d7895-8462-66e0-a1e1-04aafc84a8b6&__METHOD__= gettransferstatus&__DOMAIN__=domain.com

Response

```
<?xml version="1.0"?>
<epp>
  <response>
    <result code="1000">
      <message>Command completed successfully.</message>
    </result>
    <resData>
      <transfer:infData>
        <transfer:name>DOMAIN.COM</transfer:name>
        <transfer:status>1</transfer:status>
      </transfer:infData>
    </resData>
    <trID>
      <clTRID>SERV-2222</clTRID>
      <svTRID>VS-1111</svTRID>
    </trID>
  </response>
</epp>
```

2.27.Get Price List

Method

getpricelist

Description

Get price list for all available tlds.

Blocking Status

Non-blocking

Input

Parameter	Obligation	Definition	Type
VERSION	Required	API version	INTEGER
SIGNATURE	Required	Merchant encoded string	TEXT
METHOD	Required	Name of command to be executed	TEXT

Output

When an ***getpricelist*** command has been processed successfully, the <resData> element MUST contain a child <domain:infData> element that identifies the domain namespace. The <domain:infData> element contains the following child elements:

The <domain:prices> element contains the following child elements:

- A <domain:tld> element that contains domain zone

- A <domain:register> element that contains price for register domain

- A <domain:renew> element that contains price for renew domain

- A <domain:transfer> element that contains price for transfer domain

- A <domain:redeem> element that contains price for redeem domain

Example

BASE_URL*?****VERSION****=1&****SIGNATURE****= a77d7895-8462-66e0-a1e1-04aafc84a8b6&****METHOD****=getpricelist*

Response

```
<?xml version="1.0"?>
<epp>
  <response>
    <result code="1000">
      <message>Command completed successfully.</message>
    </result>
    <resData>
      <domain:infData>
        <domain:prices>
          <domain:tld>COM</domain:tld>
          <domain:register>18.99</domain:register>
          <domain:renew>22.99</domain:renew>
          <domain:transfer>8.99</domain:transfer>
          <domain:redeem>44.99</domain:redeem>
        </domain:prices>
        <domain:prices>
          <domain:tld>NET</domain:tld>
          <domain:register>10.00</domain:register>
          <domain:renew>10.00</domain:renew>
          <domain:transfer>10.00</domain:transfer>
          <domain:redeem>10.00</domain:redeem>
        </domain:prices>
      </domain:infData>
    </resData>
    <trID>
      <clTRID>SERV-2222</clTRID>
      <svTRID>VS-1111</svTRID>
    </trID>
  </response>
</epp>
```

2.28.Update Host Name

Method

updatehost

Description

This command provides a transform operation that allows a client to update a host object.

Blocking Status

Non-blocking

Input

Parameter	Obligation	Definition	Type
VERSION	Required	API version	INTEGER
SIGNATURE	Required	Merchant encoded string	TEXT
METHOD	Required	Name of command to be executed	TEXT
DOMAIN	Required	Domain name	DOMAIN
NS	Required	Name of NS. For example: *ns1*	TEXT
OLDIP	Required *	Remove IP address of host name	IPADDRESS
NEWIP	Required *	Add IP address of host name	IPADDRESS
OLDIPV6	Required *	Remove IPv6 address of host name	IPv6ADDRESS
NEWIPV6	Required *	Add IPv6 address of host name	IPv6ADDRESS

* must be provided OLDIP+NEWIP or OLDIPV6+NEWIPV6

Output

When a ***updatehost*** command has been processed successfully, a server responses with empty <resData> element.

Example

BASE_URL*?****VERSION****=1&****SIGNATURE****= a77d7895-8462-66e0-a1e1-04aafc84a8b6&****METHOD****= updatehost&****DOMAIN****=domain.com&****NS****=ns1&****OLDIP****=111.111.111.111&****NEWIP****=222.222.222.222&****OLDIPV6****= 2a00:1450:4001:804:0:0:0:200b&****NEWIPV6=****2620:0:2d0:2df::7*

Response

```
<?xml version="1.0"?>
<epp>
  <response>
    <result code="1000">
      <message>Command completed successfully.</message>
    </result>
    <resData/>
    <trID>
      <clTRID>SERV-2222</clTRID>
      <svTRID>VS-1111</svTRID>
    </trID>
  </response>
</epp >
```

2.29. Set Array Of Name Servers

Method

setarrnses

Description

Set array of name servers to the domain name. Associate provided name servers with the domain object (List of currently delegated host objects or host attributes (name servers) associated with the domain object can be retrieved via ***getnses*** command).

Blocking Status

Non-blocking

Input

Parameter	Obligation	Definition	Type
VERSION	Required	API version	INTEGER
SIGNATURE	Required	Merchant encoded string	TEXT
METHOD	Required	Name of command to be executed	TEXT
DOMAIN	Required	Domain name	DOMAIN
NS	Required (from 2 to 10 NS)	Name server hostname	NAMESERVER

Output

When a ***setarrnses*** command has been processed successfully, a server responses with empty <resData> element.

Example

BASE_URL*?****VERSION****=1&****SIGNATURE****= a77d7895-8462-66e0-a1e1-04aafc84a8b6&****METHOD****= setarrnses&****DOMAIN****=domain.com&****NS[]****=ns1.pananames.com&****NS[]****=ns2.pananames.com&****NS[]****=ns3.pananames.com&****NS[]****=ns4.pananames.com*

Response

```
<?xml version="1.0"?>
<epp>
  <response>
    <result code="1000">
      <message>Command completed successfully.</message>
    </result>
    <resData/>
    <trID>
      <clTRID>SERV-2222</clTRID>
      <svTRID>VS-1111</svTRID>
    </trID>
  </response>
</epp >
```

2.30.Get Transfer Out Domains

Method

gettransferoutdomains

Description

Get list transfer out domains.

Blocking Status

Non-blocking

Input

Parameter	Obligation	Definition	Type
VERSION	Required	API version	INTEGER
SIGNATURE	Required	Merchant encoded string	TEXT
METHOD	Required	Name of command to be executed	TEXT

Output

When an ***gettransferoutdomains*** command has been processed successfully, the <resData> element MUST contain a child <transfer:infData> element that identifies the transfer namespace. The <transfer:infData> element contains the following child elements:

- A <transfer:domains> element that contains list transfer out domains. This element contains a childs <transfer:domain>

- Each <transfer:domain> element contains the transfer domain name and transfer date.

Example

BASE_URL*?***VERSION***=1&***SIGNATURE****= a77d7895-8462-66e0-a1e1-04aafc84a8b6&***METHOD****= gettransferoutdomains*

Response

```
<?xml version="1.0"?>
<epp>
  <response>
    <result code="1000">
      <message>Command completed successfully.</message>
    </result>
    <resData>
      <transfer:infData>
        <transfer:domains>
          <transfer:domain>
            <transfer:name>EXAMPLE01.COM</transfer:name>
            <transfer:date>2016-05-20 14:05:34</transfer:date>
          </transfer:domain>
          <transfer:domain>
            <transfer:name>EXAMPLE02.COM</transfer:name>
            <transfer:date>2016-05-20 14:41:10</transfer:date>
          </transfer:domain>
        </transfer:domains>
      </transfer:infData>
    </resData>
    <trID>
      <clTRID>SERV-2222</clTRID>
      <svTRID>VS-1111</svTRID>
    </trID>
  </response>
</epp>
```

2.31. Domains List

Method

domainslist

Description

Get domains list.

Blocking Status

Non-blocking

Input

Parameter	Obligation	Definition	Type
VERSION	Required	API version	INTEGER
SIGNATURE	Required	Merchant encoded string	TEXT
METHOD	Required	Name of command to be executed	TEXT
PAGE	Required	Number of page	INTEGER
LIMIT	Required	Count domains per page (min = 1, max = 100)	INTEGER

Output

When an ***domainslist*** command has been processed successfully, the <resData> element MUST contain a child <domain:infData> element that identifies the domain namespace. The <domain:infData> element contains the following child elements:

- A <domain:domains> element that contains domains list. This element has attribute “count” that identifies the count of all domains. This element contains a childs <domain:domain>

- Each <domain:domain> element contains the domain name, created date, expired date and list nses.

Example

BASE_URL*?****VERSION****=1&****SIGNATURE****= a77d7895-804-aaf-a8b6&****METHOD****=domainslist&****PAGE****=1&****LIMIT****=100*

Response

```
<?xml version="1.0"?>
<epp>
  <response>
```

```
    <result code="1000">
      <message>Command completed successfully.</message>
    </result>
    <resData>
      <domain:infData>
        <domain:domains count=”15”>
          <domain:domain>
            <domain:name>EXAMPLE01.COM</domain:name>
            <domain:crDate>2016-05-20 14:05:34</domain:crDate>
            <domain:exDate>2017-05-20 14:05:34</domain:exDate>
            <domain:ns>NS1.PANANAMES.COM</domain:ns>
            <domain:ns>NS2.PANANAMES.COM</domain:ns>
          </domain:domain>
          ………..
          <domain:domain>
            <domain:name>EXAMPLE15.COM</domain:name>
            <domain:crDate>2016-05-20 14:41:10</domain:crDate>
            <domain:exDate>2017-05-20 14:05:34</domain:exDate>
            <domain:ns>NS1.PANASERVERS.COM</domain:ns>
            <domain:ns>NS2.PANASERVERS.COM</domain:ns>
            <domain:ns>NS3.PANASERVERS.COM</domain:ns>
            <domain:ns>NS4.PANASERVERS.COM</domain:ns>
          </domain:domain>
        </domain:domains>
      </domain:infData>
    </resData>
    <trID>
      <clTRID>SERV-2222</clTRID>
      <svTRID>VS-1111</svTRID>
    </trID>
  </response>
</epp>
```

2.32. Email Verification

Method

emailverification

Description

Get email verification info.

Blocking Status

Non-blocking

Input

Parameter	Obligation	Definition	Type
VERSION	Required	API version	INTEGER
SIGNATURE	Required	Merchant encoded string	TEXT
METHOD	Required	Name of command to be executed	TEXT
EMAIL	Required	Email verification	STRING

Output

When an ***emailverification*** command has been processed successfully, the <resData> element MUST contain a child <email:infData> element. The <email:infData> element contains the following child elements:

- A <email:email> element that contains email.
- A <email:createDate> element that contains date, when verification email was sent first time.
- A <email:verifyDate> element that contains date, when email was verified.
- A <email:blockDate> element that contains date, when email will be blocked.
- A <email:status> element that contains status.
- A <email:domains> element that contains domains list linked with email. This element contains a childs <email:domain>
- Each <email:domain> element contains the domain name, status, block date, unblock date. (block and unblock dates only for blocked email).

Example

BASE_URL*?****VERSION****=1&****SIGNATURE****= a77d7895-804-aaf-a8b6&****METHOD****=emailverification&****EMAIL****=example@mail.com*

Response

```
<?xml version="1.0"?>
<epp>
  <response>
    <result code="1000">
```

```
      <message>Command completed successfully.</message>
    </result>
    <resData>
      <email:infData>
        <email:email>example@mail.com</email:email>
        <email:createDate>2017-01-16 14:52:07</email:createDate>
        <email:verifyDate/>
        <email:blockDate>2017-01-31 14:52:07</email:blockDate>
        <email:status>Email 15 sent</email:status>
        <email:domains>
          <email:domain>
            <email:name>EXAMPLE.COM</email:name>
            <email:status>blocked</email:status>
            <email:blockDate>2017-01-31 14:52:07</email:status>
            <email:unblockDate/>
          </email:domain>
          <email:domain>
            <email:name>EXAMPLE.ORG</email:name>
            <email:status>blocked</email:status>
            <email:blockDate>2017-01-31 14:52:07</email:status>
            <email:unblockDate/>
          </email:domain>
        </email:domains>
      </email:infData>
    </resData>
    <trID>
      <clTRID>SERV-2222</clTRID>
      <svTRID>VS-1111</svTRID>
    </trID>
  </response>
</epp>
```

3. Response Codes and Messages

Code	Message	Description
1000	Command completed successfully	Command completed without errors
2000	Unknown command	Called unknown command
2001	Unknown version	Passed invalid version
2002	Unknown signature	Passed invalid signature
2003	Unknown param	Passed unknown parameter
2004	Invalid param	Passed invalid parameter
2100	Missing command	Request missed parameter COMMAND
2101	Missing version	Request missed parameter VERSION
2102	Missing signature	Request missed parameter SIGNATURE
2103	Missing param	Request missed required parameter
2200	Error API	Unhandled error
2201	Permission error	This error occurs in the case of merchant denied access to the API, or if the merchant tries to call some methods for domains that were created by another merchant
2202	Not enough balance	Balance – property of merchant reflecting the amount of money for domain operations that requires payments (purchase, renewal, etc.). This property was introduced as an additional tool for the control merchant amount. Currently balance check is disabled.

4. Additional Information

4.1. Domain Life Cycle

Domain Creating:

1. **Required** Call API command ***check*** to determine if an object can be provisioned within a repository. If domain name available got to step #2
2. **Required** Call API command ***create*** to create a domain object. Will be created domain object with default name servers and default WHOIS information.
3. **Required** Call API command ***createvirtual*** to add the domain name in the local system database.

4. **Optional** Call API command ***setnses*** to associate some name servers with the domain object. If these NS'es doesn't exist then add them using API command ***createns***

5. **Optional** Call API command ***setwhoisinfo*** to set custom WHOIS information for the domain object

6. **Optional** Call API command **sethost** to set DNS SOA and Resource Records of the domain name. For example set A record for pointing domain name to some IP address.

Domain Renewing

Domain object can be renewed at any time until Redemption Grace Period. After expiration domain will be auto-renewed by registry. To avoid domain renewing registrar should drop domain by API command ***delete.***

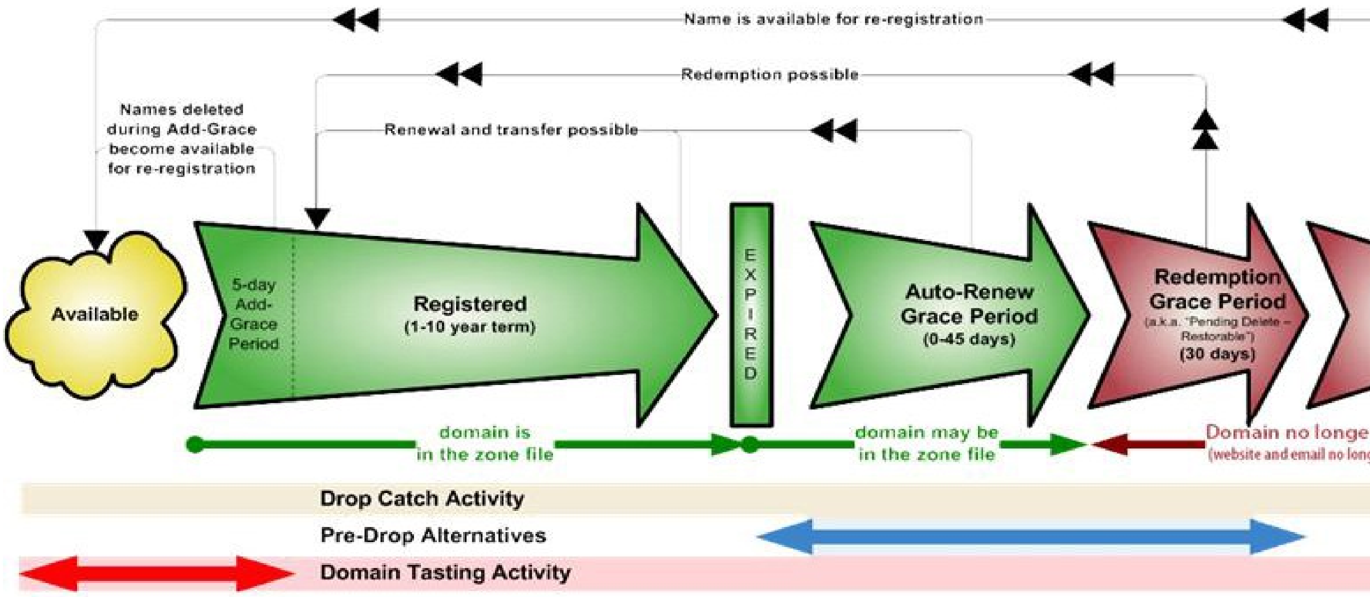
Domain Deleting

Domain object can be deleted at any time but only during first 5 (five) days it will be refunded by registry. After deleting domain moves to Redemption Grace Period.

Domain Restoring

After achieving Redemption Grace Period domain can be restored by API command ***redeem***. If domain object was not renewed and restored it became unrestorable and will be released.

Domain life cycle periods on the image bellow:


Name is available for re-registration
Redemption possible
Names deleted during Add-Grace become available for re-registration
Renewal and transfer possible
Available
5-day Add-Grace Period
Registered (1-10 year term)
EXPIRED
Auto-Renew Grace Period (0-45 days)
Redemption Grace Period (a.k.a. "Pending Delete – Restorable") (30 days)
domain is in the zone file
domain may be in the zone file
Domain no longe
(website and email no long
Drop Catch Activity
Pre-Drop Alternatives
Domain Tasting Activity

4.2. Domain Transfer Workflow

4.2.1. Transfer Domain In

1. Call API command ***transferdomainsin*** with parameter ***OPERATION=5***

2. Current domain name registrar should approve transfer (after this step domain name will be under the control of our registrar)

3. After calling ***transferdomainin*** and successful transfer to our registrar, the domain will have default contact details:

   name: "Private Whois"

   organization: "GLOBAL DOMAIN PRIVACY SERVICES INC"

   email: "support@pananames.com"

   address: "Salduba Bldg, 3rd floor, 53rd East Street, Marbella"

   city: "Panama"

   state: "NA"

   zip: "080807"

   country: "PA"

   country: "507"

   phone: "+507.8365260"

4. Then please set whois data by calling ***setwhoisinfo*** command

New Changes to Transfer Policy effective on 1st of December 2016

ICANN has adopted changes to the Transfer Policy that will become effective on 1st of December 2016. All registrars must comply with these new rules and regulations.

Changes to Transfer Policy includes:

1. FOA (Form of Authorization) that is used to confirm the domain transfer (domain ownership) for gaining registrar will have an expiration date:

- It will expire after 60 days of issue

- It will expire after new FOA is issued for any reasons

- It will expire if domain name(s) is/are locked on the moment of transfer

If your FOA is expired just request new one in order to confirm domain ownership and proceed with a transfer. It can be done by calling one more time transferdomainsin method.

2. Expired domain names can not be transferred in to our registrar or out to another registrar.

3. Significant changes are related to scenario when you change the following fields of Registrant contact for any domain name: name, organization or email address. It is considered as Change of Registrant. In terms of API request this is setwhoisinfo method. This change will now start the following processes:

- The process involves obtaining confirmation via email from current and new registrants before a change can be completed. Unless there was no change in registrant email, then only one email will be send to confirm the changes displayed in the email.

API changes: if your setwhoisinfo request is about to change Registrant name, organization or email address you will get new kind of response from our API: “Changes must be confirmed by clicking link in emails sent to old and new registrants.”

- After a change of registrant has been completed (you clicked on the link provided in email), previous and new registrant and also account owner will receive a confirmation notice informing them that the change has completed. Unless there was no change in registrant email, then only one email will be send to inform about the changes made.

Currently there is no API method that will show if the whois changes were approved or declined by previous and new registrant. You will be able to understand if the changes were applied only by reading the confirmation notice.

- After a change of registrant has been completed, the domain will be locked for transfers out to a new registrar for the following 60 days. We do understand that this kind of lock can be inconvenient. ICANN policy allows us to give Registrants option to opt out of this kind of lock. We will implement opt out feature ASAP.

If you will try to unlock your domain names you will see the following error: Unlock Domain : Domain in 60 days transfer lock.

For additional information the amended Transfer Policy is posted at https://www.icann.org/resources/pages/transfer-policy-2016-06-01-en.

4.2.2. Transfer Domain Out

1. Call API command **setpendingtransfer** with parameter ***OPERATION=1*** for approving transfer request created by other registrar
2. After step #1 domain name will have **PENDING TRANSFER** status

IMPORTANT: Before initiating the transfer, you must call API method ***unlock***. Calling this method removes from the domain status CLIENT TRANSFER PROHIBITED if it was previously installed.

4.3. Domain Register Workflow

1. Call API command ***create*** to create request on register domain
2. Call API command ***setwhoisinfo*** to register domain and set whois data

5. Change log

5.1. Changes on 01.11.2016:

Create Domain (2.2.):

- change params from required to optional: REGISTRANTCONTACTID, ADMINCONTACTID, TECHCONTACTID, BILLCONTACTID

Create Name Server (2.13.):

- added params: IPV6

Get WHOIS Info (2.21.):

- added params: SHOWPRIVATE

Set WHOIS Info (2.22.):

- added params as Optional: REGCC, ADMINCC, TECHCC, BILLCC

Added new API methods:

- 2.27. Get Price List
- 2.28. Update Host Name
- 2.29. Set Array Of Name Servers