

The interaction with the system is carried out through the exchange of parameters (command-response) in XML format.

The connection to the server is carried out via the TCP / TLS protocol using the certificate created during the registration of the client (registrar).

When creating a certificate, the login of the client (registrar) must be specified in the CN block.

Test connection parameters:

- Server – epp.it.com
- Port – 700
- User – epp@it.com
- Password – ryn5p8KDauWC
- SSL certificate:



epp_it_com.zip

greeting

После установки TCP/TLS соединения, сервер отправляет элемент greeting согласно RFC 5730:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
  <greeting>
    <svID>IT.Com EPP server</svID>
    <svDate>2022-11-09T09:53:50.987450Z</svDate>
    <svcMenu>
      <version>1.0</version>
      <lang>en</lang>
      <objURI>urn:ietf:params:xml:ns:domain-1.0</objURI>
      <objURI>urn:ietf:params:xml:ns:host-1.0</objURI>
      <objURI>urn:ietf:params:xml:ns:contact-1.0</objURI>
      <svcExtension>
        <extURI>urn:ietf:params:xml:ns:epp:fee-1.0</extURI>
      </svcExtension>
    </svcMenu>
    <dcP>
      <access><all/></access>
      <statement>
        <purpose><admin/><prov/></purpose>
        <recipient><ours/><public/></recipient>
        <retention><stated/></retention>
      </statement>
    </dcP>
  </greeting>
</epp>
```

login - start a session

Authenticates the user and starts the session.

The <clID> element must contain the login specified during registration.

The <pw> element must contain a password. The password is case sensitive.

It also checks the validity of the user certificate and the presence of the IP address from which the connection was made in the user's White List, if it is filled in the registrar's account.

The <clTRID> element must contain the session identifier. The identifier must be present in other commands, and must not change within the session.

The remaining elements of the command must match those specified in the example.

Example:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
  <command>
    <login>
      <clID>epp@it.com</clID>
      <pw>XXXXXXXXXX</pw>
      <options>
        <version>1.0</version>
        <lang>en</lang>
      </options>
      <svcs>
        <objURI>urn:ietf:params:xml:ns:domain-1.0</objURI>
        <objURI>urn:ietf:params:xml:ns:contact-1.0</objURI>
        <objURI>urn:ietf:params:xml:ns:host-1.0</objURI>
        <svcExtension>
          <extURI>urn:ietf:params:xml:ns:epp:fee-1.0</extURI>
        </svcExtension>
      </svcs>
    </login>
    <clTRID>IT.COM-login-1656487504012</clTRID>
  </command>
</epp>
```

The response returns the code and description of the result of the command execution and the client session ID.

Possible results:

- 1000 - Command completed successfully
- 2200 - User not found
- 2200 - User not registrar
- 2200 - User is blocked
- 2200 - Bad password
- 2200 - Bad certificate
- 2200 - IP is not in white list for user

Example

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
```

```
<response>
  <result code="1000">
    <msg>Command completed successfully</msg>
  </result>
  <trID>
    <clTRID>IT.COM-login-1656487504012</clTRID>
    <svTRID/>
  </trID>
</response>
</epp>
```

logout - ending the session

Ends the session with the ID specified in the <clTRID> element.

The remaining elements of the command must match those specified in the example.

Example:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
  <command>
    <logout/>
    <clTRID>IT.COM-logout-1656496559772</clTRID>
  </command>
</epp>
```

The response returns the code and description of the result of the command execution and the client session ID.

Possible results:

- 1500 - Command completed successfully; ending session

Example

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
  <response>
    <result code="1500">
      <msg>Command completed successfully; ending session</msg>
    </result>
    <trID>
      <clTRID>IT.COM-logout-1656496559772</clTRID>
      <svTRID/>
    </trID>
  </response>
</epp>
```

domain:check - checks if the domain exists in the registry

Checks if the registry contains a domain with the name specified in the <domain:name> element. Multiple <domain:name> elements are allowed if multiple domain names need to be validated.

The <clTRID> element must contain the session identifier.

If you need to get information about the cost of registering a domain, you can use the <extension> element with the <fee:check> element. The <fee:check> element must contain one or more <fee:command> elements with a name parameter that can take the following values:

- create - domain lease
- renew – domain lease extension

The <fee:command> element must contain one or more <fee:period unit="y"> elements (unit can only take the value "y"), with a period value in years for which the price is requested. Value must be integers from 1 to 5

The remaining elements of the command must match those specified in the example.

Example:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
  <command>
    <check>
      <domain:check>
        <domain:name>test-epp3.it.com</domain:name>
      </domain:check>
      <extension>
        <fee:check xmlns:fee="urn:ietf:params:xml:ns:epp:fee-1.0">
          <fee:command name="create">
            <fee:period unit="y">1</fee:period>
          </fee:command>
        </fee:check>
      </extension>
    </check>
    <clTRID>TEST-domain-check-1656507674719</clTRID>
  </command>
</epp>
```

The response returns the code and description of the result of the command execution and the client session ID.

Possible results :

- 1000 - Command completed successfully
- 2004 - Parameter value range error
- 2004 - Command failed
- 2201 - Authorization error

In the case of a result code of 1000, the domain name and the avail parameter are returned in the <domain:name> element , which can be equal to:

- "1" - the domain is available for registration
- "0" - the domain is not available for registration

If the value is "0", a <domain:reason> element is also returned, containing a description of the reason for the domain name being unavailable.

If the command had a <fee:check> element, the response includes an <extension> element with a <fee:chkData> element. It contains information about the domain class (standard , premium) and the price of registration and (or) renewal for terms in accordance with the parameters passed in the command.

Example of a response if the domain is **available**

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
  <response>
    <result code="1000">
      <msg>Command completed successfully</msg>
    </result>
    <resData>
      <domain:chkData xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
        <domain:cd>
          <domain:name avail="1">test-epp3.it.com</domain:name>
        </domain:cd>
      </domain:chkData>
    </resData>
    <extension>
      <fee:chkData xmlns:fee="urn:ietf:params:xml:ns:epp:fee-1.0">
        <fee:currency>USD</fee:currency>
        <fee:cd avail="1">
          <fee:objID>test-epp3.it.com</fee:objID>
          <fee:class>standard</fee:class>
          <fee:command name="create">
            <fee:period unit="y">1</fee:period>
            <fee:fee>25.00</fee:fee>
          </fee:command>
        </fee:cd>
      </fee:chkData>
    </extension>
    <trID>
      <clTRID>TEST-domain-check-1656507674719</clTRID>
      <svTRID/>
    </trID>
  </response>
</epp>
```

Example of a response if the domain is **not available**

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
  <response>
    <result code="1000">
      <msg>Command completed successfully</msg>
    </result>
    <resData>
      <domain:chkData xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
        <domain:cd>
```

```
        <domain:name avail="0">test-epp3.it.com</domain:name>
        <domain:reason>In use</domain:reason>
      </domain:cd>
    </domain:chkData>
  </resData>
  <extension>
    <fee:chkData xmlns:fee="urn:ietf:params:xml:ns:epp:fee-1.0">
      <fee:currency>USD</fee:currency>
      <fee:cd avail="0">
        <fee:objID>test-epp3.it.com</fee:objID>
        <fee:reason>In use</fee:reason>
      </fee:cd>
    </fee:chkData>
  </extension>
  <trID>
    <clTRID>TEST-domain-check-1656507674719</clTRID>
    <svTRID/>
  </trID>
</response>
</epp>
```

Note: the *fee* extension is only implemented for the `<domain:check>` command

domain:info – Retrieval of information about the domain

Returns information about the domain whose name is specified in the <domain:name> element.

The <clTRID> element must contain the session ID.

The remaining elements of the command must match those specified in the example.

Example command:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
  <command>
    <info>
      <domain:info>
        <domain:name>test-epp2.it.com</domain:name>
      </domain:info>
    </info>
    <clTRID>IT.COM-domain-info-1656505411798</clTRID>
  </command>
</epp>
```

The response returns the code and description of the result of the command execution and the client session ID.

Possible results :

- 1000 - Command completed successfully
- 2303 - Object does not exist

In the case of a result code of 1000, the <domain:infData> element contains information about the domain, as shown in the examples. If the domain was registered by the registrar on whose behalf the request was made, the full domain information is returned in the response. In other cases - abbreviated information.

An example of a response if the command is executed by the registrar who registered the domain.

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
  <response>
    <result code="1000">
      <msg>Command completed successfully</msg>
    </result>
    <resData>
      <domain:infData xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
        <domain:name>test-epp2.it.com</domain:name>
        <domain:roid>1110</domain:roid>
        <domain:clID>epp@it.com</domain:clID>
        <domain:status s="ok"/>
        <domain:registrant>IT.COM-REGISTRANT132</domain:registrant>
        <domain:contact type="admin">IT.COM-ADMIN132</domain:contact>
        <domain:contact type="tech">IT.COM-TECH132</domain:contact>
        <domain:contact type="billing">IT.COM-BILLING132</domain:contact>
      </domain:infData>
    </resData>
  </response>
</epp>
```

```

        <domain:ns>
            <domain:hostObj>ns1.dn.it.com</domain:hostObj>
            <domain:hostObj>ns2.dn.it.com</domain:hostObj>
        </domain:ns>
        <domain:crDate>2022-06-29T12:22:53.528469Z</domain:crDate>
        <domain:exDate>2023-06-30T00:00:00Z</domain:exDate>
        <domain:authInfo>
            <domain:pw>XXXXXXXXXX</domain:pw>
        </domain:authInfo>
    </domain:infData>
</resData>
<trID>
    <clTRID>IT.COM-domain-info-1656505411798</clTRID>
    <svTRID/>
</trID>
</response>
</epp>

```

An example of the answer in other cases.

```

<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
    <response>
        <result code="1000">
            <msg>Command completed successfully</msg>
        </result>
        <resData>
            <domain:infData xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
                <domain:name>test-epp2.it.com</domain:name>
                <domain:roid>l110</domain:roid>
                <domain:clID>epp@it.com</domain:clID>
            </domain:infData>
        </resData>
        <trID>
            <clTRID>IT.COM-domain-info-1656505411798</clTRID>
            <svTRID/>
        </trID>
    </response>
</epp>

```

contact:check - check if the contact exists in the registry

Checks if the registry contains a contact with the ID specified in the <contact:id> element.

The <clTRID> element must contain the session ID.

The remaining elements of the command must match those specified in the example.

Example command :

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
  <command>
    <check>
      <contact:check>
        <contact:id>IT.COM-REGISTRANT132</contact:id>
      </contact:check>
    </check>
    <clTRID>IT.COM-contact-check-165650537333</clTRID>
  </command>
</epp>
```

The response returns the code and description of the result of the command execution and the client session ID.

Possible results :

- 1000 - Command completed successfully

The <contact:id> element returns the contact ID and the avail parameter , which can be equal to:

- "1" - the contact is available for registration
- "0" - the contact is not available for registration

If the value is "0", a <contact:reason> element is also returned, containing a description of the reason for the contact being unavailable.

Sample response if contact is available

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
  <response>
    <result code="1000">
      <msg>Command completed successfully</msg>
    </result>
    <resData>
      <contact:chkData xmlns:contact="urn:ietf:params:xml:ns:contact-1.0">
        <contact:cd>
          <contact:id avail="1">IT.COM-REGISTRANT132</contact:id>
        </contact:cd>
      </contact:chkData>
    </resData>
    <trID>
      <clTRID>IT.COM-contact-check-165650537333</clTRID>
      <svTRID/>
    </trID>
  </response>
</epp>
```

```
</response>  
</epp>
```

Example of a response if the contact is not available

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>  
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">  
  <response>  
    <result code="1000">  
      <msg>Command completed successfully</msg>  
    </result>  
    <resData>  
      <contact:chkData xmlns:contact="urn:ietf:params:xml:ns:contact-1.0">  
        <contact:cd>  
          <contact:id avail="0">IT.COM-REGISTRANT132</contact:id>  
          <contact:reason>In use</contact:reason>  
        </contact:cd>  
      </contact:chkData>  
    </resData>  
    <trID>  
      <clTRID>IT.COM-contact-check-165650537333</clTRID>  
      <svTRID/>  
    </trID>  
  </response>  
</epp>
```

host:check - checks if the host exists the registry

Checks if the registry contains a host with the name specified in the <host:name> element.

The <clTRID> element must contain the session identifier.

The remaining elements of the command must match those specified in the example.

Example command :

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
  <command>
    <check>
      <host:check>
        <host:name>ns1.dn.it.com</host:name>
      </host:check>
    </check>
    <clTRID>IT.COM-host-check-1656505373469</clTRID>
  </command>
</epp>
```

The response returns the code and description of the result of the command execution and the client session ID.

Possible results :

- 1000 - Command completed successfully

host:id > element returns the host name and the avail parameter , which can be equal to:

- "1" - the host is available for registration
- "0" - the host is not available for registration

If the value is '0', a <host:reason> element is also returned, containing a description of the reason for the host being unavailable.

Example response if the host is **available for creation**

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
  <response>
    <result code="1000">
      <msg>Command completed successfully</msg>
    </result>
    <resData>
      <host:chkData xmlns:host="urn:ietf:params:xml:ns:host-1.0">
        <host:cd>
          <host:id avail="1">ns1.dn.it.com</host:id>
        </host:cd>
      </host:chkData>
    </resData>
    <trID>
      <clTRID>IT.COM-host-check-1656505373469</clTRID>
      <svTRID/>
    </trID>
  </response>
</epp>
```

```
</trID>
</response>
</epp>
```

Sample response if the host is **not available/exists**

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
  <response>
    <result code="1000">
      <msg>Command completed successfully</msg>
    </result>
    <resData>
      <host:chkData xmlns:host="urn:ietf:params:xml:ns:host-1.0">
        <host:cd>
          <host:id avail="0">ns1.dn.it.com</host:id>
          <host:reason>In use</host:reason>
        </host:cd>
      </host:chkData>
    </resData>
    <trID>
      <clTRID>IT.COM-host-check-1656505373469</clTRID>
      <svTRID/>
    </trID>
  </response>
</epp>
```

contact:info – Retrieval of the information about a contact

Checks for the presence in the registry system of a contact with the name specified in the <host:name> element.

Example command :

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
  <command>
    <info>
      <contact:info>
        <contact:id>IT.COM-BILLING132</contact:id>
      </contact:info>
    </info>
    <clTRID>IT.COM-contact-info-1656512737831</clTRID>
  </command>
</epp>
```

Example response :

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
  <response>
    <result code="1000">
      <msg>Command completed successfully</msg>
    </result>
    <resData>
      <contact:infData xmlns:contact="urn:ietf:params:xml:ns:contact-1.0">
        <contact:id>IT.COM-BILLING132</contact:id>
        <contact:status s="linked"/>
        <contact:postalInfo type="int">
          <contact:name>Pavel Kuzovkin</contact:name>
          <contact:org>Company</contact:org>
          <contact:addr>
            <contact:street>Street Street</contact:street>
            <contact:city>City</contact:city>
            <contact:pc>kuzovkin</contact:pc>
            <contact:cc>GB</contact:cc>
          </contact:addr>
        </contact:postalInfo>
        <contact:voice>+44.123453445665</contact:voice>
        <contact:email>pavel_kuzovkin@mail.ru</contact:email>
      </contact:infData>
    </resData>
    <trID>
      <clTRID>IT.COM-contact-info-1656512737831</clTRID>
      <svTRID/>
    </trID>
  </response>
</epp>
```

contact:create - create a contact in the registry

Creates a contact in the registry. The command must contain a <contact:create> element with the following elements:

- <contact:id> - unique identifier for the created contact
- <contact:postalInfo> - postal information of the contact. Допустим только тип элемента int (параметр type="int"). Если параметр не указан, по умолчанию принимается тип int. Если будет указан тип loc, команда завершится с ошибкой. There must be one such element. Contains elements:
 - <contact:name> - contact name
 - <contact:org> - the name of the organization the contact is associated with
 - <contact:addr> - адресная информация, связанная с контактом. Содержит следующие элементы:
 - <contact:street> - street. There may be several such elements.
 - <contact:city> - city
 - <contact:sp> - state, region, region
 - <contact:pc> - zip code/po box
 - <contact:cc> - country
 - <contact:voice> - phone number
 - <contact:email> - email address
 - <contact:fax> - fax phone number
 - <contact:authInfo> This must contain an element which will contain <contact:pw> - the authorization code for the contact

The <clTRID> element must contain the session ID.

Example command :

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
  <command>
    <create>
      <contact:create xmlns:contact="urn:ietf:params:xml:ns:contact-1.0">
        <contact:id>IT.COM-BILLING132</contact:id>
        <contact:postalInfo>
          <contact:name>User</contact:name>
          <contact:org>Company</contact:org>
          <contact:addr>
            <contact:street>Street</contact:street>
            <contact:city>City</contact:city>
            <contact:sp>Bedfordshire</contact:sp>
            <contact:pc>12345</contact:pc>
            <contact:cc>GB</contact:cc>
          </contact:addr>
        </contact:postalInfo>
        <contact:voice>+44.123453445665</contact:voice>
        <contact:email>user@mail.org</contact:email>
        <contact:authInfo>
          <contact:pw>XXXXXXXXXXXX</contact:pw>
        </contact:authInfo>
      </contact:create>
    </create>
    <clTRID>IT.COM-contact-create-1656505373454</clTRID>
  </command>
</epp>
```


The response returns the code and description of the result of the command execution and the client session ID.

Possible results :

- 1000 - Command completed successfully
- 2201 - Authorization error

In the case of a result code of 1000, the following elements are returned in the response:

- <contact:id> - contact ID
- <contact:crDate> - date/time the contact was created

Example response :

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
  <response>
    <result code="1000">
      <msg>Command completed successfully</msg>
    </result>
    <resData>
      <contact:creData xmlns:contact="urn:ietf:params:xml:ns:contact-1.0">
        <contact:id>IT.COM-BILLING132</contact:id>
        <contact:crDate>2022-06-29T12:22:53.465712Z</contact:crDate>
      </contact:creData>
    </resData>
    <trID>
      <clTRID>IT.COM-contact-create-1656505373454</clTRID>
      <svTRID/>
    </trID>
  </response>
</epp>
```

domain:create - Creating a domain name

Creates a domain in the registry system. The command must contain a <domain:create> element with the following elements:

- <contact:id> - unique identifier for the created contact
- <domain:name> - domain name
- <domain:period> with unit parameter - domain registration period. The parameter can take the value " y " - the value is specified in years. Element value must be between 1 and 5
- <domain:ns> - can contain up to 4 <domain:hostObj> elements, which contains the NS name of the domain server. Hosts for the specified names are created automatically
- <domain:registrant> - registrar contact ID
- <domain:contact> - identifier contact types admin, tech, billing. The contact type is indicated by the type parameter
- <contact:authInfo> This must contain the element <contact:pw> - the authorization code for the domain

The <clTRID> element must contain the session ID.

Example command :

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="urn:ietf:params:xml:ns:epp-1.0 epp-1.0.xsd">
  <command>
    <create>
      <domain:create
        xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
        <domain:name>test-epp2.it.com</domain:name>
        <domain:period unit="y">1</domain:period>
        <domain:ns>
          <domain:hostObj>ns1.dn.it.com</domain:hostObj>
          <domain:hostObj>ns2.dn.it.com</domain:hostObj>
        </domain:ns>
        <domain:registrant>IT.COM-REGISTRANT132</domain:registrant>
        <domain:contact type="admin">IT.COM-ADMIN132</domain:contact>
        <domain:contact type="tech">IT.COM-TECH132</domain:contact>
        <domain:contact type="billing">IT.COM-BILLING132</domain:contact>
        <domain:authInfo>
          <domain:pw>XXXXXXXXXX</domain:pw>
        </domain:authInfo>
      </domain:create>
    </create>
    <clTRID>IT.COM-domain-create-1656505373493</clTRID>
  </command>
</epp>
```

The response returns the code and description of the result of the command execution and the client session ID.

Possible results :

- 1000 - Command completed successfully
- 2303 - object does not exist (when trying to create a domain in a zone not supported by the registry)
- 2302 - Object exists
- 2004 - Parameter value range error (when specifying a lease term outside the range from 1 to 5 years)

In the case of a result code of 1000, the following elements are returned in the response:

- <domain:name> - the name of the created domain name
- <domain:crDate> - date/time of domain name creation
- <domain:ex Date> - date/time

Example response :

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
  <response>
    <result code="1000">
      <msg>Command completed successfully</msg>
    </result>
    <resData>
      <domain:creData xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
        <domain:name>test-epp2.it.com</domain:name>
        <domain:crDate>2022-06-29T12:22:53.528469Z</domain:crDate>
        <domain:exDate>2023-06-30T00:00:00Z</domain:exDate>
      </domain:creData>
    </resData>
    <trID>
      <clTRID>IT.COM-domain-create-1656505373493</clTRID>
      <svTRID/>
    </trID>
  </response>
</epp>
```

contact:update – Contact Modification

Modifies the contact with the specified ID in the registry. The command must contain a <contact:update> element with elements similar to those described for the contact:create command, except for <contact:authInfo>.

The <clTRID> element must contain the session identifier.

Example command :

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
  <command>
    <update>
      <contact:update xmlns:contact="urn:ietf:params:xml:ns:contact-1.0">
        <contact:id>IT.COM-REGISTRANT132</contact:id>
        <contact:chg>
          <contact:postalInfo type="int">
            <contact:name>User</contact:name>
            <contact:org>New Company</contact:org>
            <contact:addr>
              <contact:street>Street Street</contact:street>
              <contact:city>City</contact:city>
              <contact:sp></contact:sp>
              <contact:pc> </contact:pc>
              <contact:cc>GB</contact:cc>
            </contact:addr>
          </contact:postalInfo>
          <contact:voice>+44.123453445665</contact:voice>
          <contact:fax></contact:fax>
          <contact:email>user@mail.org</contact:email>
        </contact:chg>
      </contact:update>
    </update>
    <clTRID>IT.COM-contact-chg-1656513842717</clTRID>
  </command>
</epp>
```

The response returns the code and description of the result of the command execution and the client session ID.

Possible results :

- 1000 - Command completed successfully
- 2201 - Authorization error
- 2303 - Object does not exist

Example response :

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
  <response>
    <result code="1000">
      <msg>Command completed successfully</msg>
    </result>
    <trID>
      <clTRID>IT.COM-contact-chg-1656513842717</clTRID>
      <svTRID/>
    </trID>
  </response>
</epp>
```

domain:update - domain name update

Updates the DNS nameservers for the domain in the registry. The command must contain the <domain:update> element with the following elements:

- <domain:name> is the name of the domain for which the changes are being made
- НЕОБЯЗАТЕЛЬНЫЙ элемент <domain:add>, содержащий значения атрибутов, которые будут добавлены к объекту
- НЕОБЯЗАТЕЛЬНЫЙ элемент <domain:rem>, содержащий значения атрибутов, которые должны быть удалены из объекта

Элементы <domain:add> и <domain:rem> содержат дочерний элемент <domain:ns>, который can contain up to 4 <domain:hostObj> elements, which contains the name servers. Hosts for new names are created automatically.

The <clTRID> element must contain the session ID.

Example command :

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
  <command>
    <update>
      <domain:update xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
        <domain:name>api-test.it.com</domain:name>
        <domain:add><domain:ns>
          <domain:hostObj>ns5.it.com</domain:hostObj>
          <domain:hostObj>ns6.it.com</domain:hostObj>
        </domain:ns></domain:add>
        <domain:rem><domain:ns>
          <domain:hostObj>ns1.it.com</domain:hostObj>
          <domain:hostObj>ns2.it.com</domain:hostObj>
          <domain:hostObj>ns3.it.com</domain:hostObj>
        </domain:ns></domain:rem>
      </domain:update>
    </update>
    <clTRID>IT.COM-domain-update-1667979341872</clTRID>
  </command>
</epp>
```

The response returns the code and description of the result of the command execution and the client session ID.

Possible result:

- 1000 - Command completed successfully
- 2201 - Authorization error
- 2303 - Object does not exist

Example response :

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
```

```
<response>
  <result code="1000">
    <msg>Command completed successfully</msg>
  </result>
  <trID>
    <clTRID>IT.COM-domain-update-1667979341872</clTRID>
    <svTRID/>
  </trID>
</response>
</epp>
```

domain:renew – продление домена

Продлевает срок действия домена на указанное количество лет. Команда должна содержать элемент `<domain:renew>`, который должен содержать элементы:

- `<domain:name>` - имя домена
- `<domain:curExpDate>` - текущая дата окончания действия домена. Для исключения ошибочного повторного продления домена при повторном получении команды на продление
- `<domain:period>` with unit parameter - период продления домена. The parameter can take the value " y " - the value is specified in years. Element value must be between 1 and 5

The `<clTRID>` element must contain the session ID.

Example command:

```
<?xml version="1.0" encoding="UTF-8"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
  <command>
    <renew>
      <domain:renew>
        <domain:name>api-test.it.com</domain:name>
        <domain:curExpDate>2024-11-02</domain:curExpDate>
        <domain:period unit="y">1</domain:period>
      </domain:renew>
    </renew>
    <clTRID>IT.COM-domain-renew-1667988957382</clTRID>
  </command>
</epp>
```

The response returns the code and description of the result of the command execution and the client session ID.

Possible result:

- 1000 - Command completed successfully
- 2201 - Authorization error
- 2303 - Object does not exist
- 2304 - Object status prohibits operation
- 2005 - Parameter value syntax error
- 2004 - Parameter value range error
- 2104 - Billing failure

Example response:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
  <response>
    <result code="1000">
      <msg>Command completed successfully</msg>
    </result>
    <resData>
```

```
<domain:renData xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
  <domain:name> api-test.it.com </domain:name>
  <domain:exDate>2025-11-02T00:00:00Z</domain:exDate>
</domain:renData>
</resData>
<trID>
  <clTRID>IT.COM-domain-renew-1667988957382</clTRID>
  <svTRID/>
</trID>
</response>
</epp>
```