

Home

Welcome to ISOC-IL API Service Documentation

This documentation provides instructions on how to use and interact with ISOC-IL registry API system.

- Start with the [General](#) section.

General

The ISOC-IL API System introduces an interface between Accredited Registrars (ARs) and the .il ccTLD registry. This interface is available exclusively to ARs, who use it to perform the entire set of functions against the Domain Name Registry (new name registration, modifications and deletion of names, querying the status of names etc.).

This document specifies the technical elements of the interface to the .il API System. The system uses elements of the EPP protocol family, although it is not a standard implementation of EPP, nor does it claim compliance with EPP or RRP.

ARs should use this document (or its subsequent versions) to develop their interface to the system (client side). ISOC-IL will make an effort to make future versions of this specification backwards-compatible, however such compatibility is not guaranteed.

Terminology

Terms used throughout this documentation

- **AR** - Accredited Registrar. A company accredited by ISOC-IL as a Registrar of the .il ccTLD Registry.
- **AR Code** - A two letter identification code of the AR. The code is handed to the AR at the signing of the Accreditation Agreement.
- **AR Main Regular Mail Address** - A mail address of the AR, registered with the Registry, to which regular (i.e., non-XML) mail is sent.
- **Attribute** - As defined in [API Basics - Attribute](#).
- **ccTLD** - country code Top Level Domain. Israel's ccTLD is ".il".
- **Domain Name** - As defined in [domain name](#).
- **Domain Name Record** - The information stored in the Registry about a specific Domain Name, namely its Registrant, Managing Registrar, Administrative Contact, Technical Contact, Zone Contact and DNS Name Servers. See [scheme in API basics](#).
- **Extensions Block** - As defined in [XML - API client request general structure](#).
- **Expiry Date** - The date on which a Domain Name Record is scheduled to expire. Domain Names can be registered, renewed or reassigned in whole Registration Units only, and in every occasion, expiry may not exceed the Maximal Registration Period from the date of the transaction.
- **Handle** - A Contact Object's (as defined in [API Basics - Object](#)) unique identifier within the Registry. A Handle is created automatically by the API Server upon the creation of a Contact Object, and is of the following format: `<AR Code>-<xy><number>-IL` where:
 - `<xy>` - Name initials of the Contact.
 - `<number>` - An integer.
 - The hyphens and the string `IL` are constants.
- **IL Extensions** - The Registry's additions to the Standard EPP. The IL Extensions implementation is guided by RFC 3730.
- **ISOC-IL** - The Israel Internet Association. Administrator and operator of the .il ccTLD Registry.
- **IRI** - The Information Retrieval Interface provided by the Registry to the AR. The IRI is described in detail in the [IRI Documentation Page](#).

Registration Actions - General

In the following sections the various Registration Actions are described in detail, including the various XML files exchanged between the API Client and the API Server during the life- cycle of a Request.

General Legend

- **Occurrence:** Indicates the number of times an element can appear in a Request, where: "1" – exactly 1 instance MUST appear. "0-1" – at most 1 instance MAY appear. "1-*" – as many as needed, but at-least once.
- **String:** Any text, including whitespaces of any form.
- **Token:** A String that contains no space characters.
- **Date:** In the following format: `CCYY-MM-DD`
- **dateTime:** In the following format: `CCYY-MM-DDThh:mm:ss.sss`
- **Request ID:** The time is local Israeli time (IST=GMT+2 or IDT=GMT+3).
- **cITRID:** A unique identifier of a Request in the Registry, of the following format:
`[letter][integer]` , where:
`[letter]` – 'D' for Requests pertaining to Domain Name Records, and 'C' for Requests pertaining to Contact Object Reqeusts.
`[integer]` – a serial number.
- **svTRID:** AR's transaction identifier. The format of the identifier is: "[AR Code]-[integer]". A unique transaction identifier generated by the API Server. The format of the identifier is: `ISOC-IL-[AR Code][integer][Time Stamp]` .

API System Overview

The API System – Overview

The synchronous API System is accessed over HTTPS, XML-based system which gives ISOC-IL's ARs a secured and authenticated tool to manage Registration Actions in the Registry.

The AR's communication (API clients) with the API Server is done by sending HTTPS Request, with TLS Client Authentication to the API Server. The API Server responses are sent synchronously as HTTPS uses TLS as well.

Prior communicating with the API server, the AR must have a valid certificate. Updating the certificate is done through the Registry Management.

API Client is not supplied by the Registry, and needs to be implemented by the AR independently.

The Registry provides a testing environment to allow ARs to test the Clients they build for correct operation.

An Information Retrieval Interface ("IRI") is provided for retrieval of information and follow-up on Objects managed by the AR, the AR's Registration Actions and the AR's available credit.

The API System makes use of principles, objects and terminology from EPP.

However, the API System is **NOT** a Standard EPP implementation, nor does it claim EPP conformance.

For example, a principal difference is that Standard EPP is synchronous, and has a notion of session management; the current implementation of the IL API System, being asynchronous, has no notion of a session.

API System Authentication

Authentication

Identification for the API system is based on mutual TLS (mTLS) where the server (the API service) will verify the connected client (the Registrar client).

To access the API server, the client will need to have a valid certificate signed by one of our [approved Root CA's](#).

The certificate needs to be sent to `registry-mgr AT isoc.org.il` to be approved and attached to the registrar account.

At any given moment, there is only a single valid certificate.



Info

Please note that prior to communicating with the API server, the AR MUST have a valid certificate attached to its registrar profile.

You can see an example for a simple check request at Technical Requirements page under the [Request](#) Section.

Automatic Certificate Renewal

To renew an expired certificate - the AR must generate a new certificate in front of the same CA as before.

Then the AR should connect to the API with the renewed certificate, when its old certificate is still valid. From this point on, the AR should only use the renewed certificate.

The automatic renewal process cannot be performed through Info request or Checkrequest.

Suspicious Certificate Blocking

If there is a suspicion of attempted assault - the AR MUST block his certificate immediately, the AR will no longer be able to use the current certificate, and it will be necessary to contact the Registry Management to insert a new certificate.

API System Basics

Object

An Object is an independent information unit in the Registry. For example, a Contact Person is an Object containing details of a specific contact person.

An Object has a unique identifier in the Registry, and can be updated by submission of a Request pertaining directly to that Object.

An Object can be referred to by different Domain Name Records, and under different Roles in a Domain Name Record.

Objects Management

Only the Managing AR of an Object can update it.

An update made to an Object will be reflected in all Objects pointing to the modified Object. For example, updating the mail address of a Contact Object will be reflected in all Domain Names Records in which this Contact object is listed.

A Contact Object exists only in the realm of the Managing AR. An AR cannot refer (or use in any other way) Contact Objects that are associated with a different AR.

List of API System Object types:

1. Domain Name Record.
2. Contact Object.

Attribute

An Attribute is an information unit which is associated with a specific Object. For example, the Domain Name Registrant is an Attribute of the Domain Name Record.

Managing Attributes

In order to update an Attribute, the Managing AR of the Object containing the Attribute needs to submit a Request pertaining to the Object.

Updating identical Attributes associated with more than one Object, requires the submission of one Request per each Object to which the Attributes associate. For example, updating the physical address of a Registrant who holds 5 Domain Names, will require 5 separate update request, one per each Domain Name.

API System Technical Requirements

The Structure Of A Domain Name under **.il** Registry

A Domain Name in the Registry is a textual string of the form `<3rd level>.<SLD>.il`, where:

- `.il` is the ccTLD (Contry Code Top Level Domain).
- `<SLD>` : is one of the following Second Level Domain Names:
 - (1) `co`, (2) `org`, (3) `net`, (4) `uni`, (5) `ac`
- `<3rd level>` : a string that is comprised only of the following characters:
 - Lowercase Latin letters ('a' to 'z').
 - IDN letters ('א' to 'ת') including final forms (corresponding to Unicode Code - Points U+05D0 to U+05EA).
 - Digits ('0' to '9').
 - Hyphen ('-').
 - At-least 3 characters long.
 - At-most 64 characters long.
 - Does not start or end with a hyphen.
 - Does not contain two consecutive hyphens, except for Hebrew Domain Names in the standard A-Label form (xn--).

For example: in the Domain Name `isoc.org.il`, `isoc` is the 3rd level, `org` is the SLD, and `il` is the ccTLD.

It should be noted that there are additional requirements that may be imposed on the 3rd level string. These requirements are not technical in nature and are primarily derived from the [Domain Name Allocation Rules](#). Being such, they are not relevant for the development of the API Client, but in any case messages sent as a result of such requirements not being met, will be standard API System messages.

The Structure Of A Hebrew Domain Name under **.ישראל** Registry

Around September 2022, the ISOC-IL registry will start supporting the new ccTLD `.for` for Hebrew domain names "ישראל"

- `.ישראל` [ascii: `.xn--4dbrk0ce`] is the ccTLD (Contry Code Top Level Domain).

Create

Purpose

Create a new Domain Name Record.

Authority

Any AR can submit a Create Domain Request.

Result

After passing registry validation, a new domain is created. AR receives a [Success](#) response.

In some cases, when the process cannot be completed immediately, a pending request is created and AR receives a Pending response. The Request will wait in the queue for processing. Reasons can be:

- Domain name under muni.il, ac.il, net.il.
- Domain is limited for immediate approval.
- Domain was registered in the past.
- There is already an open request for this domain.

When processing is complete, a new Domain Name Record is created for the requested Registration Period.

If the request cannot be approved, a reject response is created containing the reject reason.

Response can be requested by `<poll>` request

Within a reasonable period of time after registration, the Registry will perform manual checks on registered domain names.

The process is called "Registrant Verification Process".

For detailed requirements please check [Appendix Registrant Verification Process](#)

Update

Purpose

Updating details in a Domain Name Record.

Authority

Only the Managing AR of a Domain Name Record can submit an Update Domain Request against that name.

Result

According to the update request, registry checks the content and if it is valid, the request is processed immediately. AR receives successful response with details of the changed attributes.

Except special cases when domain is disallowed for approval or domain registrant has changed, request will wait in queue for processing.

When processing is completed, response can be requested by `<poll>` request.

Update Domain Request - content

An Update Domain Request MUST contain only the Domain Name Record information units that are being actually updated.

The data regarding any information unit that is being included in an Update Domain Request MUST include all the unit's mandatory tags. All mandatory tags MUST be non-empty. Information pertaining to specific mandatory tags that is not being updated MUST include the data that is registered in the Domain Name Record prior to the submission of the Request. Empty non-mandatory tags will override the information in the Domain Name Record relating to those tags.

Registration Record units cannot be deleted by inclusion of an empty unit in an Update Domain Request.

There is an option to create new Contact Objects in an Update Domain Request. The implementation is identical to the one used in the Create Domain Request. Please refer to [Creation of new Contact Objects in a Create Domain Request](#) for further information. This option should be used only in cases where the new

Renew

Purpose

Extend the Expiry Date of a Domain Name Record for one Registration Period.

Authority

Only the Managing AR of a Domain Name Record can submit a Renew DomainRequest.

Timing

Domain can be renewed for up to five years. Any renewal request submitted to the registry that goes beyond the five-year maximum time, will be rejected with an error message indicating that the Domain Name renewal exceeds the allowed maximal registration period.

It is possible to renew a domain that have four years and three month to expiration date for one year only.

Result

Upon the completion of the Renew Domain Request, request is processed immediately and the Expiry Date of the Domain Name will be extended by one Registration Period.

Code Example

```
<?xml version="1.0" encoding="UTF-8"?>
<epp>
  <command>
    <renew>
      <domain:renew>
        <domain:name>example.co.il</domain:name>
        <domain:period unit="y">2</domain:period>
      </domain:renew>
    </renew>
    <clTRID>AB-123233</clTRID>
  </command>
</epp>
```


Delete

Purpose

Delete an existing Domain Name Record.

Authority

Only the Managing AR of a Domain Name Record can submit a Delete Domain Request.

Result

Upon the completion of the Delete Domain Request, the Domain Name Record will cease to exist. A Domain Name Record that was deleted, can be re-created by submission of a complete Create Domain Request.

Code Example

```
<?xml version="1.0" encoding="UTF-8"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0"
xmlns:isocil="http://www.isoc.org.il/xml/epp/isocil-1.0"
xmlns:domain="urn:ietf:params:xml:ns:domain-1.0"
xmlns:domainrequest="http://www.isoc.org.il/xml/epp/isocil-domainrequest-1.0"
xmlns:contact="urn:ietf:params:xml:ns:contact-1.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="urn:ietf:params:xml:ns:epp-1.0epp-1.0.xsd
http://www.isoc.org.il/xml/epp/isocil-1.0isocil-1.0.xsd
urn:ietf:params:xml:ns:domain-1.0domain-1.0.xsd
http://www.isoc.org.il/xml/epp/isocil-domainrequest-1.0isocil-
domainrequest-1.0.xsd
urn:ietf:params:xml:ns:contact-1.0contact-1.0.xsd">
  <command>
    <delete>
      <domain:delete>
        <domain:name>example.org.il</domain:name>
      </domain:delete>
    </delete>
    <clTRID>XY-152</clTRID>
  </command>
</epp>
```


Reassign

Purpose

Reassign the Domain Name Record to a new Holder.

Authority

Only the Managing AR of a Domain Name Record can submit a Reassign Domain Request.

A Reassign Domain Request cannot be executed on a domain under clientHold status.

Result

The Reassign process is comprised of two Registration Actions: Delete Domain and Create Domain –executed one after the other as a single atomic Registration Action, ensuring that no other Create Domain Requests for the Domain are processed in between.

Upon the completion of the Reassign Domain Request, a new Domain Name Record is created, with the submitting AR as the Managing AR, for one Registration Period starting the date the reassignment is processed.

IMPORTANT – no information is inherited from the previous Domain Name Record. Where continuity of operation of the Domain Name is required, the DNS Information blocks from the previous Domain Name Record **MUST** be specifically included in the Reassign Request.

Code Example

The Reassign request is very similar to the Create Domain Request. The difference is in the tags identifying the request type (namely, the `<update>` and `<reassign>` tags).

Reassign Domain Request Example

```
<?xml version="1.0" encoding="UTF-8"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0"
  xmlns:isocil="http://www.isoc.org.il/xml/epp/isocil-1.0"
  xmlns:domain="urn:ietf:params:xml:ns:domain-1.0"
```


Transfer

Purpose

Transfer the management of a Domain Name Record from the current Managing AR to another Managing AR.

Authority

A Transfer Domain Request MUST be submitted by the AR acquiring the management of the Domain Name Record ("Acquiring AR").

A Transfer Domain Request cannot be executed on a domain under clientHold status.

Prerequisite Conditions

1. The Transfer Request can only be submitted with regard to Domain Names Records whose Lock Transfer (see [Lock Transfer Option](#)) status is `false`. An attempt to submit a Transfer request against a Domain Name Record with Lock Transfer status "true" will result in an error indicating that the Domain Name is locked for Transfer. The Lock Transfer status of a Domain Name Record can be changed by the Managing AR.
2. The Transfer Domain Request can only refer to existing Contact Objects managed by the Acquiring AR. Where necessary, the Acquiring AR needs to create Contact Object in-advance, prior to submitting a Transfer Domain Request, using the Create Contact (see [Create Contact](#)).
3. As of September 2018, IL Registry implemented the Transfer Authorization Code mechanism, which allows only the domain name registrant to initiate an action of transferring domain name management. **For detailed requirements** please check [Appendix Transfer Authorization Policy](#)
 - Each TRANSFER request MUST include the code
 - The code will be passed on the EPP request "as is"
 - The code is for one-time use, and after the request is performed - whether it is approved or not - the code will no longer be valid
 - A request will be rejected for the following reasons:
 - Code used
 - Code could not be verified

Info

Purpose

Retrieve complete WHOIS information on a Domain Name Record.

Authority

An Info Domain Request can be submitted by any AR.

Result

Valid request is processed immediately.

Code Example

```
<?xml version="1.0" encoding="UTF-8"?>
<epp>
  <command>
    <info>
      <domain:info>
        <domain:name>example.org.il</domain:name>
      </domain:info>
    </info>
    <clTRID>SS-5</clTRID>
  </command>
</epp>
```

Legend

- `<info>` : Defines that the desired Registration Action, is an information retrieval action, where:
 - `<domain:info>` : defines that the requested information is on a Domain Name Record.
 - `<domain:name>` : Defines the queried Domain Name Record.
- `<clTRID>` AR's transaction identifier

Create

Purpose

Create a Contact Object.

Authority

A Create Contact Request can be submitted by any AR.

Result

Valid request is processed immediately.

A Contact Object will be created under the management of the AR submitting the request.

A Handle will be automatically allocated to the Contact Object by the API Server, and will be included in the response sent by the API Server.

Code Example

```
<?xml version="1.0" encoding="UTF-8"?>
<epp>
  <command>
    <create>
      <contact:create>
        <contact:postalInfo type="int">
          <contact:name>Israel Israeli</contact:name>
          <contact:org>Some Company Ltd.</contact:org>
          <contact:addr>
            <contact:street>7 Herzl St.</contact:street>
            <contact:city>Ramat Gan</contact:city>
            <contact:sp></contact:sp>
            <contact:pc></contact:pc>
            <contact:cc>IL</contact:cc>
          </contact:addr>
        </contact:postalInfo>
        <contact:voice>+972 8 1111111</contact:voice>
        <contact:fax>+972 8 2222222</contact:fax>
        <contact:email>registrant@example.co.il</contact:email>
      </contact:create>
    </create>
    <extension>
      <isocil:ext>
        <isocil:create>
          <isocil:contact>
```


Update

Purpose

Update details of an existing Contact Object.

Authority

Only the Managing AR can update Contact Objects under his management.

Scope

The Contact Object's name field CANNOT be updated with Update Contact. If the Contact's name requires modification, a new Object needs to be created, using the [Create Contact](#) Request.

Attempting to update the Contact's name field with an Update Contact Request will result in an error.

Result

Valid request is processed immediately.

The information included in the Update Contact Request will override any existing information.

IMPORTANT: all sub tags of the Contact information block must be included in the Request. Information that needs not be updated, needs to be explicitly included in the Request and contain the Contact's current data.

Code Example

```
<?xml version="1.0" encoding="UTF-8"?>
<epp>
  <command>
    <update>
      <contact:update>
        <contact:id>BL-BL3612-IL</contact:id>
        <contact:chg>
          <contact:postalInfo type="int">
            <contact:name>Israel Israeli</contact:name>
            <contact:addr>
              <contact:street>Moshe Levy 11</contact:street>
              <contact:city>Tel Aviv</contact:city>
```


Info

Purpose

Retrieve information on a Contact Object.

Authority

An Info Contact Request can be submitted only by Managing AR of the Contact Object.

Code Example

```
<?xml version="1.0" encoding="UTF-8"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0"
xmlns:isocil="http://www.isoc.org.il/xml/epp/isocil-1.0"
xmlns:contact="urn:ietf:params:xml:ns:contact-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
http://www.isoc.org.il/xml/epp/isocil-1.0 isocil-1.0.xsd
urn:ietf:params:xml:ns:contact-1.0 contact-1.0.xsd">
  <command>
    <info>
      <contact:info>
        <contact:id>AB-SR5933-IL</contact:id>
      </contact:info>
    </info>
    <clTRID>AB-123002</clTRID>
  </command>
</epp>
```

Legend

- `<info>` : Defines that the desired Registration Action, is an information retrieval action, where:
 - `<contact:info>` : defines that the requested information if on a Contact Object.
 - `<contact:id>` : Defines the queried Contact Object.
- `<clTRID>` : AR's transaction identifier

Delete

Purpose

Delete a Contact Object.

Authority

Only the Managing AR can delete Contact Objects under his management.

Condition for deletion

Only Contact Objects that are not included in any Domain Name Record ("orphans") can be deleted.

Attempting to delete a Contact Object that is included in at least one Domain Name Record will result in an error.

Result

Upon a successful deletion of a Contact Object can not be used in any subsequent request. The deleted Contact Object can not be restored.

Code Example

```
<?xml version="1.0" encoding="UTF-8"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0"
xmlns:isocil="http://www.isoc.org.il/xml/epp/isocil-1.0"
xmlns:contact="urn:ietf:params:xml:ns:contact-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
http://www.isoc.org.il/xml/epp/isocil-1.0 isocil-1.0.xsd
urn:ietf:params:xml:ns:contact-1.0 contact-1.0.xsd">
  <command>
    <delete>
      <contact:delete>
        <contact:id>AB-SR5933-IL</contact:id>
      </contact:delete>
    </delete>
    <clTRID>AB-123002</clTRID>
  </command>
</epp>
```


Cancel Request

Purpose

Withdraw a Request submitted by an AR.

Authority

A Cancel Request can only be submitted by the AR who submitted the Request.

Scope

Request is processed immediately.

Only Requests whose processing has not started can be canceled.

Since the initiation of the processing is not controlled by the AR, the successful cancellation of a Request is not guaranteed, and is done on a BEST EFFORT basis.

If the Cancel Request operation is denied, the AR, in order to undo the change, will be required to submit a correcting Registration Action as necessary.

For example, if the AR tried to cancel an Update Domain Request – another Update Domain Request will need to be submitted in order to reinstate the previous information.

Code Example

```
<?xml version="1.0" encoding="UTF-8"?>
<epp>
  <command>
    <cancel>
      <domainrequest:cancel>
        <domainrequest:id>202107</domainrequest:id>
      </domainrequest:cancel>
    </cancel>
    <clTRID>AB-1971</clTRID>
  </command>
</epp>
```


Check Request

Purpose

Check whether a Domain Name is registered. The request may include up to 10 Domains.

Authority

The request may be submitted by any AR.

Result

Request is processed immediately.

The reply from the API Server will only indicate whether the Domain Name Record is registered without providing additional information.

Code Example

```
<?xml version="1.0" encoding="UTF-8"?>
<epp>
  <command>
    <check>
      <domain:check>
        <domain:name>airforce2.co.il</domain:name>
        <domain:name>rabin.co.il</domain:name>
      </domain:check>
    </check>
    <clTRID>AB-1983</clTRID>
  </command>
</epp>
```

Legend

- `<check>` : Defines that the desired Registration Action, is an information retrieval action, where:
 - `<domain:check>` : Defines that the requested information is on a Domain Name Record.
 - `<domain:name>` : Defines the queried Domain Name Record.
- `<clTRID>` : AR's transaction identifier

Poll Request

Purpose

Allow registrars to discover and retrieve queued messages (EPP responses) from the API Server.

These messages also include [notifications](#) sent by the API Server, such as Low Credit alert.

If the message queue is not empty, a successful response to a `<poll>` command should return the first message from the message queue.

Each response returned from the server includes a server-unique message identifier that should be provided to acknowledge receipt of the message, and a counter that indicates the number of messages in the queue.

After a message has been received by the client, the client should respond to the message with an explicit acknowledgement to confirm that the message has been received.

The server should dequeue the message and decrement the queue counter after receiving acknowledgement from the client, enabling retrieval of the next.

Authority

A Poll Request can be submitted by the AR, concerning its own Requests.

Result

Request is processed immediately.

Attributes

When `'op'` is set to `'req'` – Server will respond the number of queued messages, unique message identifier with its detailed EPP response.

When `'op'` is set to `'ack'` – the unique message identifier must be applied as well. If exists, server dequeue the message and decrement the queue counter.

Code Example

Information Retrieval Interface ("IRI")

The following is a general description of the Information Retrieval Interface ("IRI") provided by the Registry. The information provided by the IRI is real-time information.

Access

The IRI is accessed by HTTP – at the following URL: <http://iri.isoc.org.il/>. Access to the IRI is allowed only from IP addresses pre-registered with the Registry in advance.

Login Page

Upon accessing the above URL, the following screen will appear:

A login form titled 'Registrar Login' in a green header. It contains three input fields: 'Registrar Code' with a pencil icon, 'User Login' with a pencil icon, and 'Password' with a lock icon. A green 'Login' button is located at the bottom right of the form.

Where:

- **"Registrar Code"**: the AR Code.
- **"User Login" and "Password"**: Please refer to the following section on [User Management](#).

The Test Environment

How to connect

The address for sending the request is: `https://api.reg-test.isoc.org.il`

Everything else same as mentioned in [API System - Technical Requirements](#)

Registry behavior simulation in testbed environment

The testbed environment is intended to provide the testers with a tool to technically test the Clients.

In the 'real world' system, the processing of some of the requests on the Registry's side, involves manual handling. In such cases, the testbed system randomly decides whether to approve or reject the request (one request is approved the next one is denied, and so on).

Note that the randomization occurs only in requests that are processed manually. In such cases, an additional "Message Error ID" will be included (please refer to [API System - XML \(the `<msg>` tag\)](#)). The error message id **in the testbed** will always be `400`. The message id refers to denial grounds that are outside the scope of this document (the reasons for denial being non-technical in nature).

EPP Response Codes

API Response Codes

Code	Text between <code><msg></code> Tag
1000	Command completed successfully
1001	Submission completed successfully
1300	Command completed successfully; no messages
1301	Command completed successfully; ack to dequeue
2000	Unknown command
2005	Parameter value syntax error
2105	Object is not eligible for renewal
2201	Authorization error
2202	Invalid authorization information
2203	Client Certificate Has Been Revoked
2302	Object exists
2309	Client Certificate Expired
2400	Command failed
2401	Operation timed out
3001	File contents do not match signature
3002	No pgp signature or file is corrupted

Mail Messages

APPENDIX B – Mail Messages

Notification on certificate block

Subject

[ILREG] <Registrar name> certificate is blcoked

Mail Body

Hello,

The certificate with thumbprint < [Certificate](#) Thumbprint > has been blocked,
for more information contact the registry administrator at < [Contact](#) Mail
Address >

Sincerely,
The .il Registry

Poll Notifications

APPENDIX C – Poll Notifications

Notification on Submitted Transfer by Other Registrar

This notification will be sent to managing registrar

```
<?xml version="1.0" encoding="UTF-8"?>
<epp>
  <response>
    <result>
      <code>1301</code>
      <msg>Command completed successfully; ack to dequeue</msg>
    </result>
    <msgQ count="1" id="19707">
      <qDate>2021-01-10T13:02:07.000Z</qDate>
      <msg code="3189">Transfer request submitted (domain
requestid=D925191)</msg>
    </msgQ>
    <resData>
      <domain:trnData>
        <domain:name>example.co.il</domain:name>
        <domain:requestID>925191</domain:requestID>
        <domain:trStatus>submitted</domain:trStatus>
        <domain:acID>Registrar 1</domain:acID>
        <domain:reID>Registrar 2</domain:reID>
      </domain:trnData>
    </resData>
    <trID>
      <clTRID>XX-777</clTRID>
      <svTRID>ISOCIL-XX1551 20051103161300</svTRID>
    </trID>
  </response>
</epp>
```

Legend:

- `<domain:trnData>` : Information on the transfer of the Domain Name Record, where:
 - `<domain:name>` : Specifies the Domain Name of the submitted transfer.
 - `<domain:requestID>` : Number of the transfer request ID.
 - `<domain:trStatus>` : Specifies the status of the transfer request.
 - `<domain:acID>` : Name of Acquiring Registrar.
 - `<domain:reID>` : Name of Managing Registrar.

Transfer Poll Notifications

APPENDIX D - POLL notifications flow process on TRANSFER request

The acquiring registrar sends a transfer request for a domain that is not locked, and has a valid authorization code.

If the request is valid, the acquiring registrar receives a PENDING response code [1001](#), and the current managing registrar receives a POLL notification code [3189](#) that a transfer request has been submitted.

The request waits in queue for a specified period of time to allow the managing registrar to respond - objection or approval.

If the managing registrar objects, the acquiring registrar will be notified by POLL notification code [3078](#).

If the request is rejected, the acquiring registrar will receive a notification through POLL notification code [3082](#).

The managing registrar will also be notified through POLL code [3087](#).

When the request is approved, the acquiring registrar will receive a notification through POLL code [1000](#).

The former managing registrar will also be notified through POLL with code [3085](#) or [3086](#).

Error Message IDs

Error Message IDs

Code	Description
2	Registrant verification needed due to registrant name
3	Registrant verification needed due to invalid address
4	Registrant verification needed due to wrong organization type
5	Registrant verification needed due to invalid organization number
6	Registrant verification needed due to email address
7	Registrant verification needed due to phone number
101	An attempt to modify the Holder information using the Update Domain Request instead of a Reassign Domain Request.
102	A Domain Name Re-Assign request must be authorized, by the current holder of the Domain Name. We will need a formal, written letter (in case of a company - on company letterhead), quoting the request ID above (important) and the Domain Name being modified. The letter must be signed by a properly authorized official of the current holder of this Domain Name.
103	In this request, there is an attempt to update the name and/or organization of the Domain Name Holder. Usually, such requests are considered to be a "re-assign" rather than "modify", so this request would normally have been rejected. However, the old information and the new information are apparently rather similar. If you can represent that the old and new information both refer to the same holder (i.e. exactly the same legal entity), please send such documentation to ISOC-IL. You can also send it by fax, to +972-3-9700901 or email it to ilreg-docs@isoc.org.il . Please note that we will need formal documentation, on formal letterhead (where applicable) to approve such a modification.

Transfer Authorization Code Policy

APPENDIX E – Transfer Authorization Code Policy in the ISOC-IL Registry

ICANN Definitions:

<https://www.icann.org/resources/pages/auth-2013-05-03-en>

1. In order to initiate a request to transfer the domain name management from one Registrar to another, the Transfer EPP request must include the Transfer Auth Code. This is of course after unlocking the domain. This requirement is **mandatory for all Registrars** working with ISOC-IL.
2. Code Rules:
 - Authorized users who can receive the code are: Managing Registrar, Domain's Registrant and Registry.
 - The registrar must create a unique code for each domain.
 - The code should be treated as a password, known only to those authorized.
 - The code can be created by the Registrar himself or alternatively, by the domain's Registrant through a secure management interface.
 - The code can be provided via the management interface, or by E-mail. Due to Data security, the code should not be transmitted "as is" in the Email. The Registrar can use a mechanism in which the recipient of the Email clicks on a certain link, and after verification, the code would be exposed to him.
 - The Registrant may receive the code from the Registrar at any time he wishes.
 - The Registrar will update ISOC-IL registry, about the code, by submitting an Update EPP request at the same time as it is sent to the holder.
 - The Registrant may update the code at any time.
 - In the Update request, the code is passed as SHA256 HASH that includes exactly 64 characters. On-line tools can be found on the Internet which performs calculation for SHA256, as well as built-in functions in the various programming languages.
 - In the Transfer EPP request, the code is passed "as is" (not in SHA256 HASH format).

Registrant Verification Process

APPENDIX F – Registrant Verification Process

The Registry will perform manual checks on registered domain names within a reasonable period of time after registration.

Please check the [Flow Chart](#) for better understanding.

Registrant Valid

In case the inspection passes successfully, the Registry will change the domain status to `ok`, and will send a [POLL notification](#) to notify the Registrar that the Registrant is valid (`registrantValid`).

Registrant Verification

If the inspection reveals that there is doubt about the validity of the Registrant's information, the domain will start with "Registrant Verification Process".

The Registry will change the domain status to "registrantVerification", and will send a [POLL notification](#) to notify the Registrar that the Registrant should be verified ("registrantVerification"). The notification will specify the [reason](#) if needed.

The domain will remain in `Registrant Verification` status for a minimum of 21 days after receiving the POLL notification.

After at least 21 days, if the Registrar does not update the Registry that the Registrant has been verified, the domain status will change to `Suspended` and eventually it will be deleted.

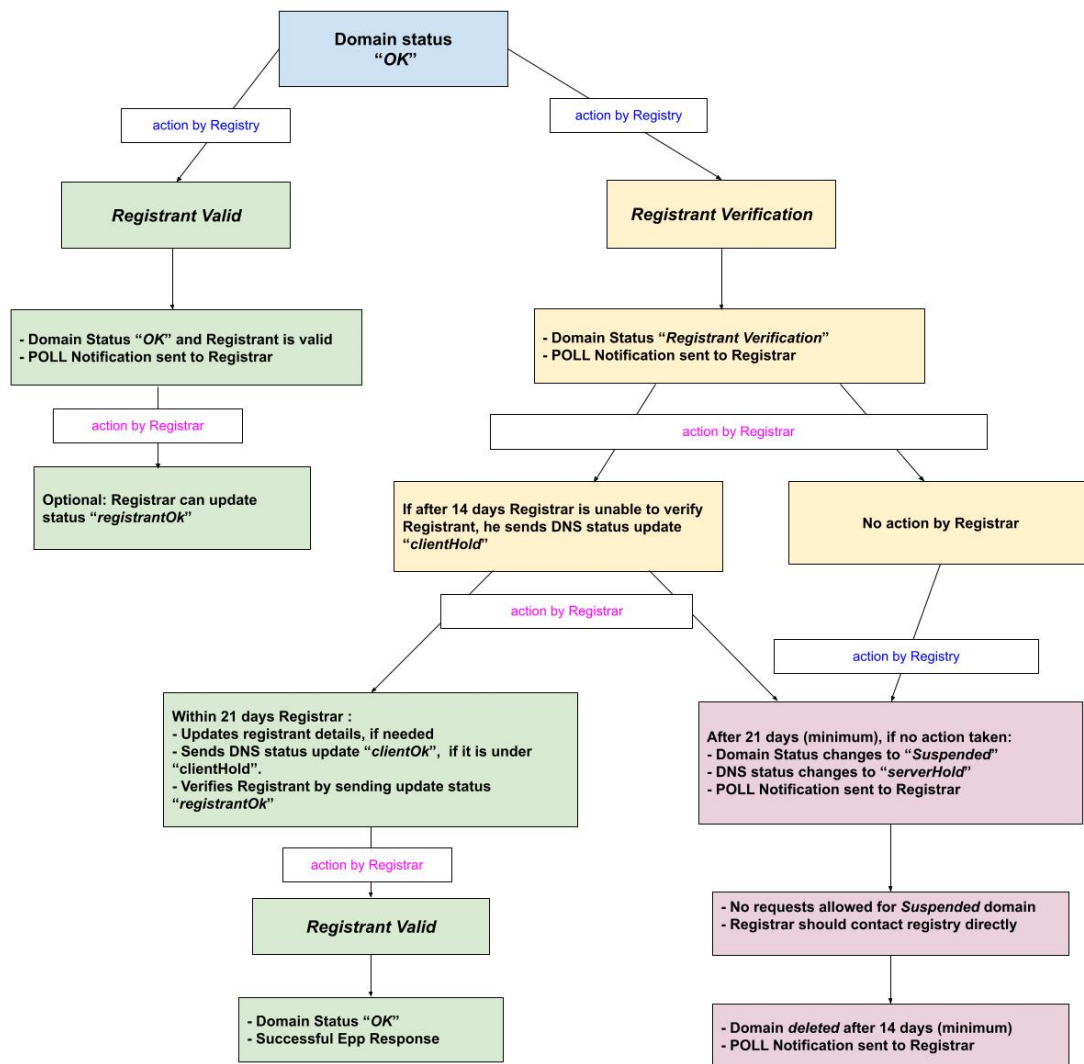
During the first 14 days, the Registrar will have to perform a process of verifying details with the Registrant and if necessary update his details.

After verifying the registrant, the registrar will send an [UPDATE request](#) with the status `registrantOk`. This action will remove the domain name from `Registrant Verification` status and it will change to `ok`.

If after 14 days the Registrar has not been able to verify the Registrant's details, he need to submit an [UPDATE request](#) with `clientHold` status, which will stop referring the domain name to its DNS servers, and prevent the submission of TRANSFER and REASSIGN requests.

Registrant Verification Flow Chart

Registrant Verification Process - Flow Chart



DS Specification

The following table lists all fields of a DS record - tags and allowed values:

Name	Valid Characters
Keytag	Digits only, A number between 1 to 65535
Algorithm	Digits only, at the moment these are the supported algorithms numbers under ISOC-IL registry: 3, 5, 6, 7, 8, 10, 13, 14. For more information see: IANA Domain Name System Security (DNSSEC) Algorithm Numbers
Digest Type	Digits only, At the moment these are the supported digest types under ISOC-IL registry: 1 (SHA1), 2 (SHA256). For more information see: IANA Delegation Signer (DS) Resource Record (RR) Type Digest Algorithms
Digest	Hex digits {0, 1, ... , 9, A, ... , F} only. Length depends on digest type: For SHA1 - the length is 40 hex digits. For SHA256 - the length is 64 hex digits

Information about the DS resource record can be found at [RFC4034 section 5.1](#). All fields are mandatory and must match the key which was used to sign the DNSKEY RRset for the zone.

Optionally KeyData information can be submitted as part of the DS data. KeyData information has the following fields and values:

Name	Valid Characters
Flags	Either 0, 256 or 257
Protocol	Always 3
Algorithm	Digits only, at the moment only the following algorithms numbers are supported under ISOC-IL registry: 3, 5, 6, 7, 8, 10, 13, 14. For more information see: IANA Domain Name System Security (DNSSEC) Algorithm Numbers

Approved Root CA

List of Approved Root CA's

Root CA Name	Certificate Fingerprint
ISOC-IL	SHA256:2380db6c80882fe77b04b585ed8f74d78488f1127af50062a4305b8058af0525
DigiCert	TBD

Convert Certificate Format

Convert Certificate Format

In some cases there is a need to convert the certificate to a specific format to be consumed by the client application, some application expect a cert+key pair while other will support only pfx or will require you to import the cert+key pair into a keystore.

We included few command line option to allow converting from one format to the other.

Convert a PEM file to DER (crt etc)

```
openssl x509 -outform der -in certificate.pem -out certificate.crt
```

Convert cert+key pair to a PFX

```
openssl pkcs12 -export -out certificate.pfx \  
-inkey privateKey.key -in certificate.crt
```

Convert a PKCS#12 file (.pfx .p12) containing a private key and certificates to PEM

```
openssl pkcs12 -in keyStore.pfx -out keyStore.pem -nodes
```

Import cert+key pair to key store

```
keytool -importkeystore -srckeystore mypfxfile.pfx \  
-srcstoretype pkcs12 -destkeystore clientcert.jks -deststoretype JKS
```


Hebrew Domain Name

Syntax rules

Definitions

IDN a set of standards for Domain Names containing characters used in various national languages. IDN defines a translation from a Domain Name comprised of national language characters ("U-Label Form") to a Domain Name comprised exclusively of Latin characters, digits and hyphens ("A-Label Form").

Unicode an international standard for representation of text in computer systems. The standard defines a set of characters which includes, inter alia, all writing systems currently in use in world languages, and maps each character to a numeric code ("Code-Point").

A Code-Point is referenced by a "U+" prefix, followed by a four-hexadecimal-digits number. For example, the code of the Hebrew letter Aleph ("א") is U+05D0.

Domain Name

A "Domain Name" is a sequence of two characters or more, registered in the 2nd or 3rd level, meeting all the following cumulative requirements:

Permitted Characters

A Domain Name may only be comprised of the following characters and must include at least one letter from the Hebrew alphabet characters:

Hebrew alphabet characters – the letters "א" to "ת", including final letters: "ך", "ץ", "ם", "ן" (corresponding to Unicode Code-Points U+05D0 to U+05EA)

Digits – the digits "0" to "9" (corresponding to Unicode Code-Points U+0030 to U+0039).

Hyphen – the character "-" (corresponding to Unicode Code-Point U+002D).

Length

A Domain Name's length will not exceed 63 characters when represented in the standard A-Label Form and will not be shorter than 2 characters in the U-Label Form.

The minimal length of a Domain Name is 2 characters.

Changelog

Changelog

All notable changes to this project will be documented in this file.

- Changes under title **YYYY-MM-DD** are implemented in the operating environment.
- Changes under title **[Release on YYYY-MM-DD]** Implemented in the testbed environment. Will be implemented on the mentioned date, in the operating environment.
- Changes under title **[Unreleased]** are implemented in the testbed environment.

2024-09-16

- Added new EPP response code [3206](#). The code will be returned when trying to register restricted registrant.

2023-07-02

Fixed

- When the domain status is changed by the Registry from [Suspended](#) to [Registrant Verification](#), the DNS changes to [serverOk](#).
Until today, the Registrar was not notified in cases like this. This update fixes it.

2023-05-07

- Please note: the registration rules have been updated, and now [new domains](#) which were registered in the past, will be approve manually.

2022-11-22

- Support domain name registration in Hebrew under the new ccTLD - "ישראל".
- New domains, under the ccTLD - "ישראל", will be registered with DNS status [serverHold](#)
- Implementation of Server Hold mechanism ([serverHold](#) , [serverOk](#)), which has same effect as [clientHold](#) / [clientOk](#) action.