RST Input Data Instructions

Instructions for the data to be prepared and submitted to the RST process.
Version: 2.15
2026-08-17

0. Introduction

This document contains information about what the Registry Operator (RO), or its Registry Service Provider (RSP), needs to prepare for the Registry System Testing (RST tests). All ROs are urged to carefully read this document.

0.1. Methodology

The tests are documented according to the standard IEEE 829-2008, as described in the Master Test Plan. The standard allows for different types of tests, e.g. unit, system, and acceptance tests. This test environment will focus on acceptance testing. Multiple areas have been identified within the system requirements:

- DNS
- RDAP
- Searchable WHOIS (tested only if required by the Registry Agreement)
- EPP
- IDN
- Data Escrow
- Documentation

Each test area is further described in its own Level Test Plan and one or more Level Test Case documents, which will be published by ICANN.

0.2. Data entry into the Naming Services portal (NSp)

All input data to the tests shall be provided by the RO in NSp. The defined contact for the RST process will receive an email with the instructions on how to provide the input files once the RST is scheduled.

We strongly recommend to encrypt the input files before including them as attachments. Please use the following PGP public key in the case you decide to encrypt the files.

0.3. Document changes

Major updates of this document compared to version 2.8 (2017-07-28):

- Updated RST Test Nodes.
- Updated instructions to provide RST input files.

1. RST Test Nodes

The RST Service Provider will verify the TLDs infrastructure (RDAP, EPP and DNS) from the RST Test Nodes. The RO must make sure that firewalls and access lists are open for the following IP addresses (both IPv4 and IPv6). The TLD's infrastructure must be available for testing during the whole test period.

Please make sure that you look at the latest version of this document for the most current set of IP addresses of the RST Test Nodes.

1.1. General RST Test Nodes

Tested systems must allways accept queries from the IP addresses below.

RST Test Node	IPv4	IPv6	Hostname	Date of change of IP address or addition of node
North America	104.225.11.140	2607:fc50:3000:2::7d	rstnode-us-na1.rst.icann.org	2021-06-10
Latin America and Caribbean	148.163.220.146	2607:f740:1::a6e	rstnode-br-lac1.rst.icann.org	2021-06-10
Asia/ Australia/ Pacific	103.84.155.222	2403:2500:400:20::e72	rstnode-jp-ap1.rst.icann.org	2021-06-10
Africa	102.67.165.239	2c0f:edb0:0:10::ce6	rstnode-za-af1.rst.icann.org	2021-06-10
Europe	176.58.88.90	2a00:dd80:3a::81c	rstnode-uk-eur1.rst.icann.org	2021-06-10

1.2. RST Test Nodes for SRS Gateway Test

For RST Type SRS GW Testing, the SRS Gateway must accept queries from relevant test node. Currently only one such test node has been defined.

RST Test Node	IPv4	IPv6	Hostname	Date of change or addition
Mainland China	8.140.185.5	N/A	rstnode-cn-ap1.rst.icann.org	2021-06-10

2. DNS test

2.1. Introduction

The RST Service Provider will verify TLD's DNS infrastructure over both UDP and TCP, and that DNSSEC is supported including life cycle management of Zone and Key signing keys. All tests will be carried out for both IPv4 and IPv6.

- For the DNS tests, RO must supply the following information to the RST Service Provider: FQDN of all authoritative name servers
- IPv4 and IPv6 addresses for same
- Delegation Signer (DS) information
- A domain with NS and DS records published in the TLD. This domain must be delegated to a name server different from the TLD name servers, but it does not have to exist on this name server.

2.2. Expected input

The RO shall provide the input data for the DNS test in a single XML file according to the XML schemas provided by the RST Service Provider:

- Mandatory file format: XML
- File name on submitted file: pdtdns.xml

Your XML file will be validated against the schema after uploading and you will be notified if there are any errors. However, for better error reporting you are encouraged to validate your XML yourself before uploading.

2.2.1. XML schemas

There are two different XML schemas of which the RO can choose from. These are the W3C XML Schema (XSD) and the RELAX NG (RNG). The submitted XML file must match the schema.

2.2.2. Example of input

For your convenience a sample XML file containing input data for the DNS test is provided.

2.2.3. Schema files and sample XML file

Schema files and sample XML files are found in the Input Data Templates zip file found on https://www.icann.org/resources/registry-system-testing. Always use the latest version.

- pdtdns.rng (DNS RELAX NG Schema)
- pdtdns.rnc (DNS RELAX NG Compact Schema)
- pdtdns.xsd (DNS XML Schema Definition)
- pdtdns.xml (DNS Sample XML)

3. RDAP test

The RST Service Provider verifies that the Registry's RDAP service is reachable over IPv4 and IPv6 via HTTPS on port 443 at the declared **RDAP Base URL**, and that responses for domains, entities and nameservers conform to the IETF RDAP RFCs and gTLD RDAP Profile. Conformance is verified **by following the RST** 2.0 Test Specifications section 5.3 **(Registration Data Access Protocol)** and is **executed manually** using the RDAP Conformance Tool.

RO must supply the following information to the RST Service Provider in a single JSON file (pdtrdap.json):
[REQUIRED]

- rdap.baseURL, HTTPS only; URL must end with a trailing "/"; if a port is present it MUST be 443; hostnames per RFC 1123, IDN labels per IDNA2008.
- rdap.profileVersion, RDAP profile version, it must be "february-2024".
- rdap.testDomains, Existing domain name(s) used to validate domain RDAP responses, and MUST include at least one domain name.
- rdap.testEntities, an array of entity objects to be queried to validate RDAP entity responses, and one entity MUST be provided. If the TLD uses the thick (maximum) registry model, the RST contact MUST provide at least one contact-entity handle (e.g., registrant) in addition to a registrar entity handle.
- rdap.testNameservers, The nameservers(s) that will be queried to validate nameserver responses. This input parameter is an array of objects. At least one nameserver MUST be provided if the TLD supports the host object model (RFC 5731), but it is optional if the TLD uses host -attributes.

3.1. Expected input

The RO shall provide the input data for the RDAP test in a single JSON file according to the JSON schema provided by the RST Service Provider.
Your JSON file will be validated against the schema after uploading and you will be notified if there are any errors. However, for better error reporting you are encouraged to validate the JSON yourself before uploading.

- Mandatory file format: JSON
- File name on submitted file: pdtrdap.json

3.1.1. JSON schema

```
{
  "type": "object",
  "required": [
    "rdap.baseURL",
    "rdap.testDomains",
    "rdap.testEntities",
    "rdap.testNameservers"
  ],
  "properties": {
    "rdap.baseURL": {
      "type": "string",
      "format": "url"
    },
    "rdap.testDomains": {
      "type": "array",
      "minItems": 1,
      "items": {
```

```
        "type": "string",
        "format": "hostname"
      }
    },
    "rdap.testEntities": {
      "type": "array",
      "minItems": 1,
      "items": {
        "type": "string"
      }
    },
    "rdap.testNameservers": {
      "type": "array",
      "minItems": 1,
      "items": {
        "type": "string",
        "format": "hostname"
      }
    }
  }
}
```

3.1.2. Example of input

For your convenience, a sample JSON file containing input data for the RDAP test is provided.

```
{
  "rdap.baseURL": "https://rdap.nic.example/rdap/v1/",
  "rdap.testDomains": [
    "test1.example"
  ],
  "rdap.testEntities": [
    9999
  ],
  "rdap.testNameservers": [
    "ns1.test1.example"
  ]
}
```

3.1.3. sample JSON file

A sample JSON file is found in the Input Data Templates zip file found on https://www.icann.org/resources/registry-system-testing. Always use the latest version.

- pdtrdap.json (RDAP Sample XML)

4. Searchable WHOIS test

The Searchable WHOIS test cases are only run if Exhibit A of the Registry Agreement states that the TLD has support for searchable Whois.
For the Searchable Whois tests, RO must supply the following information to the RST Service Provider:

[REQUIRED]

- An existing domain name for this TLD, which has Whois data for Searchable WHOIS test.
- The domain name of an existing name server, which has Whois data Searchable WHOIS test.

[OPTIONAL]

- Credentials, e.g. username and password, if required for accessing the Whois search service as a logged in user

4.1. Expected input

The RO shall provide the input data for the Searchable WHOIS test in a single json XML file according to the XML schemas provided by the RST Service Provider.
Your XML file will be validated against the schema after uploading and you will be notified if there are any errors. However, for better error reporting you are encouraged to validate the XML yourself before uploading.

- Mandatory file format: XML
- File name on submitted file: pdtSearchablewhois.xml

4.2. XML schemas

A W3C XML Schema (XSD) is available for use by the RO.

4.3. Example of input

For your convenience, a sample XML file containing input data for the Searchable Whois test is provided.

4.4. Schema file and sample XML file

Schema file and sample XML file is found in the Input Data Templates zip file found on https://www.icann.org/resources/registry-system-testing. Always use the latest version.

- pdtSearchablewhois.xsd (Whois XML Schema Definition)
- pdtSearchablewhois.xml (Whois Sample XML)

5. EPP test

The RST Service Provider will verify that TLD's EPP Service conforms to appropriate RFCs, including EPP extensions for DNSSEC. For Create, the new object has to be visible in zone file and RDPA within 24 hours. For changes, the zone data must be visible in TLD's zone file and RDAP service within 60 minutes. If RO states support for EPP over IPv6, the RST Service Provider will verify this too.
If your EPP server requires a client certificate for a client to be able to login, make sure to include the pkcs12 file and password in the Client/KeyPair part of the XML.
The address to your EPP server can be given as an IP address or as a hostname, once for IPv4 and once for IPv6, if supported. If a hostname is given, the test script client will use that name as SNI parameter (Server Name Indication, RFC 6066) in the TLS handshake. The hostname should either be resolvable in public DNS or be a name under the TLD in question and resolvable using the DNS delegation information provided in the RST DNS XML file.
For the EPP tests, RO must supply the following information to the RST Service Provider:
[REQUIRED SETUP INFORMATION]

- Information about the EPP server: IPv4 address+port, server certificate and optionally IPv6 address+port. Instead of an IP address a hostname can be provided in one or both of the IP address fields.
- Valid login ID and password required to access the EPP service
-

- URI and Schema Location for Domain, Contact and Host objects
- URI and Schema Location for the SecDns extension (DNSSEC)
- URI and Schema Location for other extensions, if applicable

[FOR THE TESTS]

- FQDN for at least two name servers
- Three non-registered domain names, ready to be created
- One registered domain name ready to be renewed
- Two registered domain names ready for transfer
- One registered domain name ready to be updated
- One domain name that can be deleted
- A new Contact object, to be created
- Contact and Host objects to be updated
- Contact and Host objects to be deleted

5.1. Reserved registrars

There are two reserved registrar ID's. It is recommended to use those for the RST. The data for the reserved registrar ID's are below.

- NEW GURID, Account Add #1 (for use by new gTLD registries)
  - GURID: 9995
  - "Registrar Name": Pre-Delegation Testing Registrar #1
- NEW GURID, Account Add #2 (for use by new gTLD registries)
  - GURID: 9996
  - "Registrar Name": Pre-Delegation Testing Registrar #2

5.2. Expected input

The RO shall provide the input data for the EPP test in a single XML file according to the XML schemas provided by the RST Service Provider.
Your XML file will be validated against the schema after uploading and you will be notified if there are any errors. However, for better error reporting you are encouraged to validate your XML yourself before uploading.
While EPP allows for a large number of extensions to each object, the RO should only submit values for those extensions that are mandatory for their registration system. Please refer to the XML schema below for a precise description of the various objects and their attributes.

- Mandatory file format: XML
- File name on submitted file: pdtepp.xml

5.2.1. XML schemas

There are two different XML schemas of which the RO can choose from. These are the W3C XML Schema (XSD) and the RELAX NG (RNG).

5.2.2. Example of input

For your convenience, a sample XML file containing input data for the EPP test is provided.

5.2.3. Schema files and sample XML file

Schema files and sample XML files are found in the Input Data Templates zip file found on https://www.icann.org/resources/registry-system-testing. Always use the latest version.

- pdtepp.rng (EPP RELAX NG Schema)
- pdtepp.rnc (EPP RELAX NG Compact Schema)
- pdtepp.xsd (EPP XML Schema Definition)
- pdtepp.xml (EPP Sample XML)

5.3. Submitting EPP extensions

If TLD's EPP server needs extra data not found in the standard object definitions, corresponding EPP extensions have to be specified as part of the input data.
In Appendix 3 you will find working XML sample code to input three different EPP extensions and their associated data.

6. SRS GW Test

This section about SRS GW Testing is only relevant for TLDs that will undergo RST Type SRSGW testing.
SRS GW Tests consists of EPP and RDAP tests where most of the information above apply. The Test Levels are called

- Test Sequence One EPP
- Test Sequence Two EPP

6.1. Summary of data files

Data files for SRS GW Testing are submitted in the same way as other data files through Naming Services portal . The table below describes the files. It assumes that the reader has studied the sections above about EPP tests and RDAP tests.

File name	Used in Test Sequence…	Description
pdtepp.xml	One, Two	Contains • IP address to TLD SRS EPP server • Log in information to TLD SRS EPP system • Data for creation, deletion and update of EPP data through the TLD SRS EPP system All information is used in Sequence One. IP address and log in information is used in Sequence Two. This file is just as the pdtepp.xml file used for other RST Type, see EPP section above.
srsgwepp.xml	Two	Contains • IP address to SRS GW EPP server • Log in information to SRS GW EPP system • Data for creation, deletion and update of EPP data through the SRS GW EPP system This file is just as the pdtepp.xml file, but the data for creation, deletion and update must not be the same.
pdtrdap.srsgw.json	Three	Contains • The IP address of SRS RDAP service.

6.2. Schema files

Schema files and sample XML and json files are found in the Input Data Templates zip file found on https://www.icann.org/resources/registry-system-testing. Always use the latest version.

File name	Used in Test Sequence…	Schema
pdtepp.xml	One, Two	pdtepp.xsd
srsgwepp.xml	Two	pdtepp.xsd
pdtrdap.srsgw.json	Three	

6.3. Datafiles for Sequence One

In SRS Gateway Testing Sequence One a number EPP transformations are run against the TLD SRS EPP system. They are described in detail in the EPP Test Area document.
The datafile for SRS Gateway Testing Sequence One has the same format as the datafile for other RST EPP testing. It must contain data to be used in the same way as in other RST EPP testing. It must have the file name given below.

- Filename: pdtepp.xml
- Schema: pdtepp.xsd

6.3.1. Limitation

In the XML file, the EPP system should be given as an IP address (or addresses). If it is given as a hostname, that hostname must be resolvable on public DNS. The EPP system should only be given as a hostname if it is necessary, i.e. if TLS/SNI is required.

6.4. Datafiles for Sequence Two

In SRS Gateway Testing Sequence Two a number of EPP transformations are run against the SRS Gateway EPP system plus EPP INFO commands against both the TLD SRS EPP system and the SRS Gateway EPP system. They are described in detail in the EPP Test Area document.
The Test Cases in Sequence Two will log in to both the TLD SRS EPP system and the SRS GW EPP system. The Test Cases will reuse the certificates and log-in information submitted for Sequence One when accessing the TLD SRS EPP system.
The datafile for SRS Gateway Testing Sequence Two has the same format as the datafile for Sequence One (and other RST EPP testing). It must contain data to be used in the same way. The two files must not have the same content since most of the data will be consumed during the test. The exception is the certificates and log-in information. The file must have the file name given below.

- Filename: srsgwepp.xml
- Schema: pdtepp.xsd

6.4.1. Limitation

In the XML file, the EPP system should be given as an IP address (or addresses). If it is given as a hostname, that hostname MUST be resolvable on public DNS. The EPP system should only be given as a hostname if it is necessary, i.e. if TLS/SNI is required.

6.5. Datafiles for Sequence Three

In SRS Gateway Testing Sequence Three a number of RDAP lookups are run both against the SRS Gateway RDAP service that is provided in pdtrdap.srsgw.json file. and against the TLD SRS RDAP service that RO provided in pdtrdap.json. The use of both RDAP services will be as described in RST2.0 specification sections 7.118. srsgw-13 - Domain RDAP synchronization, srsgw-14 - Nameserver RDAP synchronization, and srsgw-15 - Registrar RDAP synchronization.

Two datafiles are needed for Sequence Three. It must have the following file name:

-

The second datafile must contain the IP addresses to the TLD RDAP service, at least one IPv4 address and one IPv6 address. It must the have the following file name:

- Filename: pdtrdap.json
- Schema: See section 3.1.1.

The third datafile must contain the IP addresses to the SRS GW RDAP service system, at least one IPv4 address and one IPv6 address. It must the have the following file name:

- Filename: pdtrdap.srsgw.json

  A sample JSON XML file containing input data for the SRS GW Sequence Three is provided.

```
{
  "rdap.baseURL": "https://rdap.nic.example/rdap/v1/"
}
```

- Schema:

```
{
  "type": "object",
  "required": [
    "rdap.baseURL",
  ],
  "properties": {
    "rdap.baseURL": {
      "type": "string",
      "format": "url"
    }
  }
}
```

7. IDN test

7.1. Introduction

The IDN testing has two basic components. The first is a review of the submitted IDN table(s) and associated policy statements, together with the EPP documentation described below. This is both to verify conformance with the reference specifications and to enable a determination of the expected response of the back-end registry to a request for the registration of a given label. The second component of the IDN testing verifies that the registry responds in the anticipated manner, rejecting labels that are not permitted and accepting those that are.
The conformance tests are performed on all IDN tables listed in Exhibit A of the Registry Agreement for the TLD. Testing the EPP response of the back-end registry (test cases IDNvalid10, IDNvalid11 and part of IDNvalid07) is restricted to the IDN tables that are also listed in Section 5 of the Self-Certification Document.
IDN test cases are described in detail in the Test Case document for IDN found on the ICANN Micro-Site for RST, https://www.icann.org/resources/registry-system-testing. ROs should study that document in detail when preparing for the IDN Test Area of the RST.

7.2. Expected input

The following material must be submitted in order for the IDN testing to be conducted:

- Every IDN table that is listed in Exhibit A of the Registry Agreement for the TLD, in IDN tables formatted according to RFC 7940, RFC 4290 or RFC 3743.
- A complete statement of the policies that apply to IDN registration, i.e. a summary of responses to questions asked in the Applicant Guidebook necessary for the understanding of a submitted IDN table. If variant relationships between codepoints exist, the policy document(s) should describe how these variants are managed. If contextual rules are applicable to any code point in an IDN table, those rules should be explicitly stated.
- If the IDN tables are in RFC 7940 format (LGR) then everything that can be encoded in the IDN table must be encoded there, i.e. external information only if strictly necessary.
- A complete statement of which languages and scripts that are to be supported at General Registration.
- A complete example of an EPP Domain Create command for a valid, unregistered, IDN label, applicable to all IDN tables that are listed both in Exhibit A of the Registry Agreement and in the IDN Self-Certification Document.
- A list of all EPP extensions, such as language tags that may be needed for the registration of IDN labels, covering all IDN tables that are listed both in Exhibit A of the Registry Agreement and in the IDN Self-Certification Document.

7.3. IDN Self-certification template

It is recommended that the IDN Self-certification template is used for the information below. It also contains useful information and instructions.

- Policy statement
- List of languages and scripts
- EPP example
- EPP extensions

7.4. Policy statement and EPP information

If no IDN self-certification document is provided, then a separate policy statement must be submitted together with a separate file with the EPP example.

7.5. IDN tables

The IDN tables are expected to be submitted as separate files, one IDN table per file. The submission of material that is not referenced in Exhibit A of the Registry Agreement will delay the testing. ROs should take care to ensure a one-to-one correspondence between the listed IDN tables and those that are forwarded for testing.

7.6. File format

The IDN tables and associated documents are subject to automated processing and must be submitted as LGR/XML files (RFC 7940) or TXT files (RFC 4290 or RFC 3743). Any IDN table or document including non-ASCII characters must be encoded in Unicode UTF-8.
The IDN Self-certification document is expected to be submitted as a PDF/A file.

7.7. TXT IDN table structure and processing

The IDN tables are parsed algorithmically using a script that is publicly available at https://github.com/dotse/idn-properties. It is based on the requirements of the reference specifications for IDN table format, RFC 4290 and RFC 3743. The script provides a degree of additional latitude in order to deal with submissions which, for documented local reasons, are not strictly conformant to those RFCs.
The basic requirements are that every row in a IDN table starts with a code point indicated in the form "U+nnnn", or with a hash mark "#" indicating a remark, or is blank. The notation of a continuous sequence of code points in the form "nnnn..mmmm" is not permitted. If a row starting with a code point contains any additional data, this must either be initiated with a hash mark, or use a notational syntax described in one of the reference RFCs. Every submitted IDN table must be parsable by this script.
More details are found in the IDN Self-certification template.

7.8. LGR IDN tables

ICANN provides a web based tool to inspect LGR IDN tables. It is found via
https://www.icann.org/resources/pages/lgr-toolset-2015-06-21-en

7.9. Special consideration on code points

Pay special attention on the code points in the IDN table. Investigate the need and requirement of contextual rules for them and the mixing of those with other code points.

- Code points with Unicode script property value COMMON or INHERITED.
- Code points with IDN property CONTEXTJ or CONTEXTO.
- Right-to-left code points.
- Code points with general Unicode category Modifier Letter or Mark.
- Mixing of code points from different explicit Unicode scripts.
- The use of listed code points in the specific language (if language IDN table).

7.10. General considerations

The documentation needed for the IDN testing is not intended to provide background detail about the nature and purpose of IDN nor does it need to restate the terms of reference for the testing. The submitted documentation need provide only the detail necessary to conduct the individual IDN tests. Noting again that the Test Case document cited above should be studied carefully when preparing for the IDN testing.

8. Data Escrow test

8.1. Introduction

The RST Service Provider will verify that the Data Escrow deposit conforms to the relevant IETF draft or RFC document. This includes, among other things, file formats, file names and methods for encryption and signing of the escrow file(s).

8.2. Expected input

Both the full deposit and the differential deposit may be split over several files. In this case each data file shall be individually signed and accompanied by a corresponding signature file.
The deposit shall be supplied in files formatted according to the rules in http://tools.ietf.org/html/draft-arias-noguchi-registry-data-escrow (refer to latest version) or corresponding RFC as well as RFC 4880.
Encryption shall be done using the RST Service Provider's public test key, pdtescrow.pub, which is found in appendix 2.
For the Data Escrow tests, the RO should supply the following information to the RST Service Provider. The file name patterns below must be followed.

- Filename: {gTLD}_{YYYY-MM-DD}_full_S{#}_R{rev}.ryde
- Filename: {gTLD}_{YYYY-MM-DD}_full_S{#}_R{rev}.sig
- Filename: {gTLD}.pub
- Filename: {gTLD}_{YYYY-MM-DD}_diff_S{#}_R{rev}.ryde
- Filename: {gTLD}_{YYYY-MM-DD}_diff_S{#}_R{rev}.sig

In the file name patterns above:

1. {gTLD} is equal to the TLD string. If it is an IDN TLD, then this must be the A label.
2. {YYYY-MM-DD} is equal to year, month, and day. The file must be maximum 30 days old.
3. {#} is a number greater than or equal to 1. Leading zeroes are not allowed.
4. {rev} is a number greater than or equal to 0. Leading zeroes are not allowed.

8.3. Sample file names

File names for sample TLD "example":

```
example_2013-11-13_full_S1_R0.ryde
example_2013-11-13_full_S1_R0.ryde
example.pub
example_2013-11-13_diff_S1_R0.ryde
example_2013-11-13_diff_S1_R0.ryde
```

File names for sample TLD "xn--example":

```
xn--example_2013-11-13_full_S1_R0.ryde
xn--example_2013-11-13_full_S1_R0.ryde
xn--example.pub
xn--example_2013-11-13_diff_S1_R0.ryde
xn--example_2013-11-13_diff_S1_R0.ryde
```

9. Self-certification documentation test

9.1. Introduction

This section contains the specification for the self-certification data that the RO shall provide as part of the pre-delegation test.
The documentation tests are divided into the following groups:

- Self-certification of DNS, RDAP and EPP
- DNSSEC Practice Statement
- Data Escrow

9.2. Expected input

The RO shall provide the following input data for the different groups of self-certification documentation tests:

Documentation Test	Input data
Self-certification	A single PDF/A file adhering to the chapter structure of the template provided by the RST Service Provider
DNSSEC Practice Statement	A single PDF/A file containing the DNSSEC Practice Statement
Data Escrow	A single PDF/A file containing the escrow provider agreement with all appendices

9.2.1. File names

Documentation Test	Name of file to be submitted	Sample file names (TLD is “example” and “xn--example”, respectively)
Self-certification	Self-certification_{gTLD}.pdf	Self-certification_example.pdf Self-certification_xn--example.pdf
DNSSEC Practice Statement	DPS_{gTLD}.pdf	DPS_example.pdf DPS_xn--example.pdf
Data Escrow	Data-Escrow_{gTLD}.pdf	Data-Escrow_example.pdf Data-Escrow_xn--example.pdf

{gTLD} is equal to the TLD string. If it is an IDN TLD, then this must be the A label. The string must be in lower case.

9.2.2. Template

Use that for the self-certification document to be submitted. Always preserve chapter structure and headlines.

Documentation Test	Files (same file in different formats)
Self-certification	Self-certification_document.docx Self-certification_document.pdf Self-certification_document.rtf

The template is found in the Input Data Templates zip file found on https://www.icann.org/resources/registry-system-testing. Always use the latest version.

10. Appendix 1: Attachments

The following files are attachments to this document and are found in the Input Data Templates zip file found on https://www.icann.org/resources/registry-system-testing. Always use the latest version.

- Self-certification_document.docx
- Self-certification_document.pdf
- Self-certification_document.rtf
- RST_IDN_Self-certification_document.docx
- RST_IDN_Self-certification_document.pdf
- RST_IDN_Self-certification_document.rtf
- pdtdns.rnc
- pdtdns.rng
- pdtdns.xml
- pdtdns.xsd
- pdtepp.rnc
- pdtepp.rng
- pdtepp.xml
- pdtepp.xsd
- pdtrdap.json
- pdtrdap.srsgw.json
- pdtrdap.schema.json
- pdtSearchablewhois.rnc
- pdtSearchablewhois.rng
- pdtSearchablewhois.xml
- pdtSearchablewhois.xsd
- lgr.rng

11. Appendix 2: pdtescrow.pub

The Public key for encryption of the .ryde files for the Data Escrow tests is available in the “RST” section located at https://www.icann.org/resources/pages/pgp-keys-2012-02-25-en#RST

Please, copy everything from and including “-----BEGIN PGP PUBLIC KEY BLOCK-----“ up to and including “-----END PGP PUBLIC KEY BLOCK-----“ to a file called “pdtescrow.pub”. Preserve all line breaks.

12. Appendix 3: Example of EPP extensions

Below are working XML sample code to input three different EPP extensions and their associated data. Together with each example, the Extensions part of the resulting EPP command that will be sent to TLD's server during testing is shown.

12.1.1. A normal extension with two Name:Value fields

```
<Extension>
   <URI>urn:se:iis:xml:epp:iis-1.2</URI>
   <SL>urn:se:iis:xml:epp:iis-1.2 iis-1.2-xsd</SL>
   <Field>
      <Name>orgno</Name>
      <Value>[SE]551112-3282</Value>
   </Field>
   <Field>
      <Name>vatno</Name>
      <Value>SE551112328201</Value>
   </Field>
</Extension>
```

Resulting EPP fragment

```
<extension>
   <ex01:create xmlns:ex01="urn:se:iis:xml:epp:iis-1.2"
    xsi:schemaLocation="urn:se:iis:xml:epp:iis-1.2 iis-1.2-xsd">
      <ex01:orgno>[SE]551112-3282</ex01:orgno>
      <ex01:vatno>SE551112328201</ex01:vatno>
   </ex01:create>
</extension>
```

12.1.2. An IDN extension with a single text node

```
<Extension>
   <ExtName>language</ExtName>
   <URI>urn:ietf:params:xml:ns:idn-1.0</URI>
   <SL>urn:ietf:params:xml:ns:idn-1.0 idn-1.0.xsd</SL>
   <ExtValue>ger</ExtValue>
</Extension>
```

Resulting EPP fragment

```
<extension>
   <ex01:language xmlns:ex01="urn:ietf:params:xml:ns:idn-1.0"
    xsi:schemaLocation="urn:ietf:params:xml:ns:idn-1.0 idn-1.0.xsd">ger</ex01:language>
</extension>
```

12.1.3. An IDN extension with a single Name:Value field

```
<Extension>
   <URI>urn:ar:params:xml:ns:idn-1.0</URI>
   <SL></SL>
   <Field>
      <Name>languageTag</Name>
      <Value>ar</Value>
   </Field>
</Extension>
```

Resulting EPP fragment

```
<extension>
   <ex01:create xmlns:ex01="urn:ar:params:xml:ns:idn-1.0" xsi:schemaLocation="">
      <ex01:languageTag>ar</ex01:languageTag>
   </ex01:create>
</extension>
```

12.1.4. An extension with fields within field

```
<Extension>
   <URI>http://www.tcinet.ru/epp/tci-contact-ext-1.0</URI>
   <SL>http://www.tcinet.ru/epp/tci-contact-ext-1.0 tci-contact-ext-1.0.xsd</SL>
   <Field>
      <Name>person</Name>
      <Field>
         <Name>birthday</Name>
         <Value>1970-11-11</Value>
      </Field>
```

```
      <Field>
         <Name>passport</Name>
         <Value>passport string</Value>
      </Field>
      <Field>
         <Name>TIN</Name>
         <Value>444444444444444</Value>
      </Field>
   </Field>
</Extension>
```

Resulting EPP fragment

```
<extension>
   <ex01:create xmlns:ex01="http://www.tcinet.ru/epp/tci-contact-ext-1.0"
xsi:schemaLocation="http://www.tcinet.ru/epp/tci-contact-ext-1.0 tci-contact-ext-1.0.xsd">
      <ex01:person>
         <ex01:birthday>1970-11-11</ex01:birthday>
         <ex01:passport>passport string</ex01:passport>
         <ex01:TIN>444444444444444</ex01:TIN>
      </ex01:person>
   </ex01:create>
</extension>
```