.INFO Agreement Appendix 1
Data Escrow Specification

(22 August 2013)

Registry Operator and ICANN agree to engage in good faith negotiations to replace this Appendix with a Data Escrow Specification equivalent to that of new gTLDs within 90 days after the final Data Escrow Specification has been published as an RFC.

Appendix 1 to the Registry Agreement consists of four of the five exhibits to the Registry Data Escrow Agreement that constitutes Appendix 2 to the Registry Agreement:

Exhibit 1-Schedule for Escrow Deposits

Exhibit 2-Escrow Deposit Format Specification

Exhibit 3-Escrow Transfer Process

Exhibit 4-Escrow Verification Procedures

The fifth exhibit (Exhibit 5) to Appendix 2, which sets forth the escrow agent's fees, is subject to negotiation between Registry Operator and the escrow agent.

Exhibit 1-Schedule for Escrow Deposits

Full Deposit Schedule

Full Deposits shall consist of data that reflects the state of the registry as of 00:00 UTC on each Sunday. Pending transactions at that time (i.e. transactions that have not been committed to the Registry Database) shall not be reflected in the Full Deposit.

Full Deposits shall be made, according to the transfer process described in Exhibit C below, within a four-hour window beginning at 04:00 UTC on the same Sunday.

Incremental Deposit Schedule

Incremental Deposits shall reflect database transactions made since the most recent Full or Incremental Deposit. Incremental Deposits for Monday shall include transactions completed through 00:00 UTC on that day that had not been committed to the registry database at the time the last Full Deposit was taken. Incremental Deposits on Tuesday through Saturday shall include transactions completed through 00:00 UTC on the day of the deposit that were not reflected in the immediately prior Incremental Deposit.

Incremental Deposits shall be made, according to the transfer process described in Exhibit C below, within a four-hour window beginning at 04:00 UTC on the day to which the Incremental Deposit relates.

Exhibit 2-Escrow Deposit Format Specification

Each Full and Incremental Deposit consists of a series of reports that are concatenated in the Escrow Process.

Full Deposit Contents. The reports involved in a Full Deposit are:

Domain Object Report-This reports on the contents of all domain objects in the registry database.

Host Object Report-This reports on the contents of all host objects in the registry database.

Contact Object Report-This reports on the contents of all contact objects in the registry database.

Registrar Object Report-This reports on the contents of all registrar objects in the registry database.

Incremental Deposit Contents. The report involved in an Incremental Deposit is:

Transaction Report-This reports on the contents of all transaction records included in the Incremental Deposit.

Format of Reports. All reports are to be formatted in XML format. In compliance with the XML 1.0 specification, certain characters in the data must be escaped, as described in item 1 below. Each Report shall then be prepared according to the general XML format described in items 2 to 7 below. Item 2 describes the report container that is common to all reports. Items 3 to 7 describe the structure of the contents of the report container for each of the specific reports.

1. Escape-Character Requirements. In compliance with the XML 1.0 specification, data escrowed using the XML format the following characters in any data elements must be replaced with the corresponding escape sequences listed here:

Character	Escape Sequence
"	&quot;
&	&amp;
'	&apos;
<	&lt;
>	&gt

2. The Report Container. At its highest level, the XML format consists of an escrow container with header attributes followed by escrow data. The header attributes are required and include the version of escrow (1.0), the Registry TLD ("info"), the report type (domain, host, contact, registrar, or transaction), and database-committed date and time as to which the escrow relates. The date and time of the escrow will be specified in UTC. The general format of the report container is as follows:

```
<?xml version="1.0" encoding='UTF-8' ?>
<!DOCTYPE escrow SYSTEM "whois-export.dtd" >
<escrow version="1.0" tld="info" report="domain" date="26-Aug-2001 3:15:00AM">
```

{Here the report contains the actual data being escrowed. It contains one element for each object of the type (domain, host, contact, registrar, or transaction) covered by the report. The specific format for each report is described in items 3 to 7 below.}

```
</escrow>
```

3. The Domain Element. The domain element has the attribute "fqdn" (the fully qualified name of the domain) and is a container consisting of the following elements:

a. status: The domain status code. Possible values are: NEW, ACTIVE, INACTIVE, HOLD, LOCK, CLIENT-HOLD, CLIENT-LOCK, PENDING-TRANSFER, PENDING-DELETE

b. period: The registration period in years.

c. owned-by: An identification (the "id" attribute of the registrar element) of the sponsoring registrar of the domain.

d. created-code: A reference to the transaction that created the domain object.

e. created-on: The date/time the domain object was originally created.

f. renewed-on: The date/time the domain was last renewed.

g. updated-by: An identification (the "id" attribute of the registrar element) of the registrar that last updated the domain object.

h. updated-on: The date/time the domain object was last updated.

i. transferred-by: An identification (the "id" attribute of the registrar element) of the registrar that last transferred the domain object.

j. transferred-on: The date/time when the domain object was last transferred.

k. transferred-code: A reference to the transaction that last transferred the domain object.

l. host: Up to thirteen (13) host names that are nameservers for the domain to which the domain object relates.

m. contact-id: Up to four (4) contact-ids that reference the contact records for this domain. Contact-id has the attribute "type" to denote the type of contact. "Type" can be one of: Registrant, Administrative, Technical or Billing.

n. ds: DS records that represent the secure entry point keys registered for the domain to which the domain object relates. Records will be in standard DS Presentation Format as shown in the example below.

An example domain container appears below:

```
<domain fqdn="example.info">
   <status>ACTIVE</status>
   <period>1</period>
   <owned-by>42</owned-by>
   <created-code>12345678</created-code>
   <created-on>1-Jul-2001 12:34:56AM</created-on>
   <renewed-on></renewed-on>
   <updated-by>42</updated-by>
   <updated-on>1-Jul-2001 12:34:56AM</updated-on>
   <transferred-by></transferred-by>
   <transferred-on></transferred-on>
   <transferred-code></transferred-code>
   <host>dns1.example.info</host>
   <host>dns2.example.info</host>

<ds>

        <keytag>54135</keytag>
        <algorithm>7</algorithm>
        <digesttype>1</digesttype>        <digest>225F055ACB65C8B60AD18B3640062E8C23A5
FD89</digest>
    </ds>
    <ds>
        <keytag>54135</keytag>
        <algorithm>7</algorithm>
        <digesttype>2</digestype>        <digest>6CDE2DE97F1D07B23134440F19682E7519AD
DAE180E20B1B1EC52E7F58B2831D</digest>
    </ds>
    <ds>
        <keytag>53347</keytag>
        <algorithm>5</algorithm>
```

```
     <digesttype>1</digesttype>       <digest>F4F3248CA668AAA92DB5ABC40EF550F2443
47B4A</digest>
  </ds>
  <contact-id type="Registrant">1</contact-id>
  <contact-id type="Administrative">2</contact-id>
  <contact-id type="Technical">3</contact-id>
  <contact-id type="Billing">4</contact-id>
</domain>
```

4. The Host Element. The host element has the attribute "fqdn" (the fully qualified name of the host) and is a container consisting of the following elements:

a. owned-by: An identification (the "id" attribute of the registrar element) of the sponsoring registrar of the host.

b. created-code: A reference to the transaction that created the host object.

c. created-on: The date/time the host object was originally created.

d. updated-by: An identification (the "id" attribute of the registrar element) of the registrar that last updated the host object.

e. updated-on: The date/time the host object was last updated.

f. ip-address: Any number of IP addresses associated with this host.

An example host container appears below:

```
<host fqdn="dns1.example.info">
  <owned-by>42</owned-by>
  <created-code>12345679</created-code>
  <created-on>1-Jul-2001 12:40:32AM</created-on>
  <updated-by>42</updated-by>
  <updated-on>1-Jul-2001 12:40:32AM</updated-on>
  <ip-address>192.168.1.1</ip-address>
</host>
```

5. The Contact Element. The contact element has the attribute "id" and is a container consisting of the following elements:

a. name: The name of the contact.

b. organization: The organization for the contact.

c. Within the "contact" container is a sub-container named "address" with the following elements:

i. street1: The first part of the street address of the contact.

ii. street2: The second part of the street address of the contact.

iii. city: The name of the city of the contact.

iv. state-province: The name of the state/province of the contact.

v. postal-code: The postal/zip code of the contact.

vi. country: The two-letter ISO 3166 code for the contact's country.

d. voice: The voice phone number of the contact in E164a format.

e. fax: The fax number of the contact in E164a format.

f. email: The e-mail address of the contact.

g. owned-by: An identification (the "id" attribute of the registrar element) of the sponsoring registrar of the contact.

h. created-code: A reference to the transaction that created the contact object.

i. created-on: The date/time the contact object was originally created.

j. updated-by: An identification (the "id" attribute of the registrar element) of the registrar that last updated the contact object.

k. updated-on: The date/time the contact object was last updated.

l. transferred-by: An identification (the "id" attribute of the registrar element) of the registrar that last transferred the contact object.

m. transferred-on: The date/time when the contact object was last transferred.

n. transferred-code: A reference to the transaction that last transferred the contact object.

An example contact container appears below:

```
<contact id="1">
   <name>John Doe</name>
   <organization>aol</organization>
   <address>
      <street1>1234 East 11th Street</street1>
```

```
    <street2></street2>
    <city>New York</city>
    <state-province>NY</state-province>
    <postal-code>12345</postal-code>
    <country>US</country>
  </address>
  <voice>+212.1234567</voice>
  <fax>+212.1234568</fax>
  <email>jdoe@example.info</email>
  <owned-by>42</owned-by>
  <created-code>12345680</created-code>
  <created-on>1-Jul-2001 12:42:22AM</created-on>
  <updated-by>42</updated-by>
  <updated-on>1-Jul-2001 12:42:22AM</updated-on>
  <transferred-by></transferred-by>
  <transferred-on></transferred-on>
  <transferred-code></transferred-code>
</contact>
```

6. The Registrar Element. The registrar element has the attribute "id", which is a unique identifier assigned by the IANA. The registrar element is a container consisting of the following elements:

a. password: The password for the registrar.

b. name: The name of the registrar.

c. status: The registrar status code.

d. contact-id: Any number of contact-id associated with this registrar. Contact-id has the attribute "type" to denote the type of contact. "Type" can be one of: Administrative, Technical or Billing.

An example registrar container appears below:

```
<registrar id="42">
  <password>registrarrus</password>
  <name>Registrar R Us</name>
  <status>ACTIVE</status>
  <contact-id type="Administrative">10</contact-id>
  <contact-id type="Administrative">11</contact-id>
  <contact-id type="Technical">12</contact-id>
  <contact-id type="Technical">13</contact-id>
  <contact-id type="Billing">14</contact-id>
</registrar>
```

7. The Transaction Element. The transaction element has the properties "operation" and "type".

"Operation" can be one of: add, modify or delete. "Type" can be one of: domain, host, contact or registrar. The transaction element is a container consisting of elements from the corresponding "type" element. For example, a transaction element with a "type" of "registrar" will have the same set of elements as a Registrar element. For a "delete" operation, only the object ID is included in the transaction element.

An example transaction container appears below:

```
<transaction operation="modify" type="registrar">
   <password>new password</password>
   <name>Registrar R Us</name>
   <status>ACTIVE</status>
   <contact-id type="Administrative">10</contact-id>
   <contact-id type="Administrative">11</contact-id>
   <contact-id type="Technical">12</contact-id>
   <contact-id type="Technical">13</contact-id>
   <contact-id type="Billing">14</contact-id>
</transaction>
```

Exhibit 3-Escrow Transfer Process

Deposit Transfer Process.

Registry Operator shall prepare and transfer the Deposit file by the following steps, in sequence:

1. The Reports making up the Deposit will first be created according to the format specification. (See Exhibit B above, "Escrow Deposit Format Specification").

2. The Reports making up the Deposit will be concatenated. The resulting file shall be named according to the following format: "infoSEQN", where "SEQN" is a four digit decimal number that is incremented as each report is prepared.

3. Next, the Deposit file will be processed by a program (provided by ICANN) that will verify that it complies with the format specification and contains reports of the same date/time (for a Full Deposit), count the number of objects of the various types in the Deposit, and append to the file a report of the program's results.

4. Registry Operator may optionally split the resulting file using the Unix SPLIT command (or equivalent) to produce files no less than 1 GB each (except the final file). If Deposit files are split, a .MD5 file (produced with MD5SUM or equivalent) must be included with the split files to isolate errors in case of transfer fault.

5. The Deposit file(s) will then be encrypted using Escrow Agent's public key for PGP and signed using Registry Operator's private key for PGP, both version 6.5.1 or above, with a key of DH/DSS type and 2048/1024-byte length. (Note that PGP compresses the Deposit file(s) in

addition to encrypting it (them).)

The formatted, encrypted and signed Deposit file(s) will be sent, by anonymous file transfer, to Escrow Agent's ftp server within the specified time window.

Exhibit 4-Escrow Verification Procedures

Verification Procedures. Escrow Agent will verify the format and completeness of each Deposit by the following steps:

1. At the conclusion of the deposit window, all Deposit files will be moved to a not-publicly-accessible directory and the existence and size of each will be noted.

2. Each Deposit file will be decrypted using Escrow Agent's private key for PGP and authenticated using Registry Operator's public key for PGP. (In this step, PGP will also automatically decompress the escrow file).

3. If there are multiple files, they will be concatenated in sequence.

4. Escrow Agent will run a program on the Deposit file (without report) that will split it in to its constituent reports (including the format report prepared by the Registry Operator and appended to the Deposit) check its format, count the number of objects of each type, and verify that the data set is internally consistent. This program will compare its results with the results of the Registry-generated format report, and will generate a Deposit format and completeness report. The program will encrypt the report using ICANN's public key for PGP and signed using Escrow Agent's private key for PGP, both versions 6.5.1 or above, with a key of DH/DSS type and 2048/1024-byte length. (Note that PGP compresses the Deposit file(s) in addition to encrypting it (them).)

5. The decrypted Deposit file will be destroyed to reduce likelihood of data loss to intruders in case of partial security failure.

Distribution of Public Keys. Each of Registry Operator and Escrow Agent will distribute its public key to the other party (Registry Operator or Escrow Agent, as the case may be) via email to an email address to be specified. Each party will confirm receipt of the other party's public key with a reply email, and the distributing party will subsequently reconfirm the authenticity of the key transmitted. In this way, public key transmission is authenticated to a user able to send and receive mail via a mail server operated by the distributing party. Escrow Agent and ICANN shall exchange keys by the same procedure.