LGR Toolset User Guide (Basic Mode)

IDN Program
August 2023
### Availability of LGR Toolset

- **LGR Toolset** is available with the following disclaimer:

  *This software is provided by ICANN and contributors `as is` and any express or implied warranties, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose are disclaimed. In no event shall ICANN or contributors be liable for any direct, indirect, incidental, special, exemplary, or consequential damages (including, but not limited to, procurement of substitute goods or services; loss of use, data, or profits; or business interruption) however caused and on any theory of liability, whether in contract, strict liability, or tort (including negligence or otherwise) arising in any way out of the use of this software, even if advised of the possibility of such damage.*

- Online deployment
  - Visit [https://lgrtool.icann.org/](https://lgrtool.icann.org/)

- Open source package(s) release with BSD license
  - Released at github: lgr-core, lgr-django, munidata, picu

- For queries or feedback
  - Email to IDNProgram@icann.org

- For further details, visit the [LGR Toolset webpage](https://lgrtool.icann.org/) or [www.icann.org/idn](https://www.icann.org/idn)
Agenda

1. Landing Page
   (pp. 5-6)

2. Validate Label(s)
   Against an LGR
   (pp. 7-22)

3. Label Forms
   (pp. 23-30)
Agenda Details

- **Landing Page**
  - The interface with all modes

- **Validate label(s) against an LGR - Basic Mode**
  - Validate label against the selected LGR
  - Validate a single label
    - Check label(s) for collision with existing TLD
    - Include mix-script variant label
  - Validate label(s) by uploading the file

- **Label Forms**
  - Manual input
  - Uploading the list
  - Downloading the result
Landing Page
Landing Page of the LGR tool

- A user guide with more information is published on the [LGR Tool webpage](#).
- Go to link - Use the Label Generation Rules Tool

Welcome to the LGR (Label Generation Ruleset) Tools

Select your mode:

- **Basic Mode**
  - ✓ Validate label(s) against an LGR

- **Advanced Mode**
  - Advanced LGR Tools

- **IDN Table Reviewing**
  - Review IDN table(s)
Validate Label(s) Against an LGR – Basic Mode
Step 1: Click the “Validate label(s) against an LGR” to validate labels.

Welcome to the LGR (Label Generation Ruleset) Tools

Select your mode:

- Validate label(s) against an LGR
- Review IDN table(s)
- Advanced LGR Tools

If you have questions, please contact globalsupport@icann.org

Click to start using the Validate Label(s) Against an LGR - Basic Mode
Validate Label(s) Against an LGR

Step 2: Select the LGR and Input the Label(s)

Label Validation Tool

Validate label(s) against

View RZ-LGR version as XML or HTML

Upload a list of labels

Check for collision with existing TLDs

Include mixed script variants

Validate

2-1. Select required version of RZ-LGR

View RZ-LGR version as XML or HTML

3. Two ways to input label(s)
   3.a. Enter a single label manually
   3.b. Upload a list of labels

To add or remove new label manually
Validate Label(s) Against an LGR

**Step 3:** Select Optional Checks and Click ‘Validate’

Label Validation Tool

- **Validate label(s) against**:
  
  ![Dropdown menu](lgr-5-common-26may22-en.xml)

- **View LGR as** XML | HTML

- **Upload a list of labels**

- **Check for collision with existing TLDs**

- **Include mixed script variants**

4. Check if the label collides with existing TLDs

5. Check if mixed script variants are needed

6. Press “Validate” to get results
Validate a Single Label (Example 1)

Example 1: Validating a label “app” against the Latin Script RZ-LGR without performing collision check or including the mixed script variant calculation.

Add label “app”

When both boxes are unchecked

Select “Latin script LGR”
Validate a Single Label (Example 1 - output)

Example 1: Validating a label “app” against the Latin Script RZ-LGR without performing collision check or including the mixed script variant calculation.

The output shows validity

<table>
<thead>
<tr>
<th>U-label</th>
<th>Disposition</th>
<th>Code point sequence</th>
</tr>
</thead>
<tbody>
<tr>
<td>app</td>
<td>valid</td>
<td>U+0061 (a) U+0070 (p) U+0070 (p)</td>
</tr>
</tbody>
</table>

[Action index: 9] <action disp="valid" comment="catch all (default action) @"/>
The output shows variants, excluded mixed-script variants

<table>
<thead>
<tr>
<th>Variant labels (including original as last)</th>
</tr>
</thead>
<tbody>
<tr>
<td>5 variant label(s) generated.</td>
</tr>
<tr>
<td>By disposition: blocked: 4, valid: 1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Variant label</th>
<th>Disposition</th>
<th>Variant Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>åpp</td>
<td>blocked</td>
<td>U+00E1 (á) U+0070 (p) U+0070 (p)</td>
</tr>
<tr>
<td>xn--pp-lia</td>
<td></td>
<td></td>
</tr>
<tr>
<td>[Action index: 2] &lt;action disp=&quot;blocked&quot; any-variant=&quot;blocked&quot; comment=&quot;any variant label containing blocked variants is blocked&quot;/&gt;</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Variant label</th>
<th>Disposition</th>
<th>Variant Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>åpp</td>
<td>blocked</td>
<td>U+03AC (å) U+03C1 (p) U+03C1 (p)</td>
</tr>
<tr>
<td>xn--hxa8ba</td>
<td></td>
<td></td>
</tr>
<tr>
<td>[Action index: 2] &lt;action disp=&quot;blocked&quot; any-variant=&quot;blocked&quot; comment=&quot;any variant label containing blocked variants is blocked&quot;/&gt;</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Variant label</th>
<th>Disposition</th>
<th>Variant Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>app</td>
<td>blocked</td>
<td>U+03B1 (å) U+03C1 (p) U+03C1 (p)</td>
</tr>
<tr>
<td>xn--rxa8aa</td>
<td></td>
<td></td>
</tr>
<tr>
<td>[Action index: 2] &lt;action disp=&quot;blocked&quot; any-variant=&quot;blocked&quot; comment=&quot;any variant label containing blocked variants is blocked&quot;/&gt;</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Variant label</th>
<th>Disposition</th>
<th>Variant Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>app</td>
<td>blocked</td>
<td>U+0430 (å) U+0440 (p) U+0440 (p)</td>
</tr>
<tr>
<td>xn--80a6aa</td>
<td></td>
<td></td>
</tr>
<tr>
<td>[Action index: 2] &lt;action disp=&quot;blocked&quot; any-variant=&quot;blocked&quot; comment=&quot;any variant label containing blocked variants is blocked&quot;/&gt;</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Variant label</th>
<th>Disposition</th>
<th>Variant Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>app</td>
<td>valid</td>
<td>U+0061 (a) U+0070 (p) U+0070 (p)</td>
</tr>
<tr>
<td>app</td>
<td></td>
<td></td>
</tr>
<tr>
<td>[Action index: 9] &lt;action disp=&quot;valid&quot; comment=&quot;catch all (default action)&quot;/&gt;</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Validate a Single Label (Example 2)

Example 2: Validating a label “app” against the Latin Script RZ-LGR with performing collision check and without including the mixed script variant calculation.

Select “Latin script RZ LGR”

Input label “app”

When Collision Check is selected
Validate a Single Label (Example 2 - output)

Example2: Validating a label “app” against the Latin Script RZ-LGR with performing collision check and without including the mixed script variant calculation.

Result of Checking Collision

❌ The input label or its variant labels collide with an existing TLD or its variant TLDs.

Collison

The label is in the TLDs list

The rest will be the same result as Example 1
Example 3: Validating a label “app” against the Latin Script RZ-LGR with performing both collision check and calculating mixed script variant.

Select “Latin script RZ LGR”

Validate label(s) against

View LGR as XML | HTML

Upload a list of labels

app

Label can be in U-Label or A-Label form or a list of code points.

- Check for collision with existing TLDs
- Include mixed script variants

Input label “app”

When both Collision Check and Include Mixed Script variants are selected
Example 3: Validating a label “app” against the Latin Script RZ-LGR with performing both collision check and calculating mixed script variant.

Result of Checking Collision

- The input label or its variant labels collide with an existing TLD or its variant TLDs.

<table>
<thead>
<tr>
<th>Collision</th>
</tr>
</thead>
<tbody>
<tr>
<td>The label is in the TLDs list</td>
</tr>
</tbody>
</table>
Validate a Single Label (Example 3 - output)

<table>
<thead>
<tr>
<th>Variant labels (including original as last)</th>
</tr>
</thead>
<tbody>
<tr>
<td>45 variant label(s) generated.</td>
</tr>
<tr>
<td>By disposition: blocked: 44, valid: 1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>app</th>
<th>valid</th>
<th>U+0061 (a) U+0070 (p) U+0070 (p)</th>
</tr>
</thead>
</table>

[Action index: 9] <action disp="valid" comment="catch all (default action) ⊗"/>

Too many variant labels to be displayed in the browser. You can download the whole results in .csv.

Download the result in the file, which shows variants, including mixed-script variants.
Validate Labels by Uploading the File (Example 4)

Label Validation Tool

Validate label(s) against

View LGR as XML | HTML

Manually input label(s)

Upload the file with a list of labels

Choose file example4.txt

File should be a text file encoded in UTF-8 and using 0x0A line ending. It must contain one label per line in U-Label or A-Label form or as a list of code points. Comments begin with '#'

- Check for collision with existing TLDs
- Include mixed script variants

Validate
Validate Labels by Uploading the File (Example 4 - output)

Validate label(s) against

Igr-5-arabic-script-26may22-en

View LGR as XML | HTML

Manually input label(s)

Choose file No file chosen

File should be a text file encoded in UTF-8 and using 0x0A line ending. It must contain one label per line in U-Label or A-Label form or as a list of code points. Comments begin with '#'

- Check for collision with existing TLDs
- Include mixed script variants

Your request was sent successfully, the validation task processing can take minutes, even hours. You can follow your task progression on the task status page.

Validate

1. Results and status of list of labels will be present on the Task Status Page
Validate Labels by Uploading the File (Example 4 - output)

<table>
<thead>
<tr>
<th>Input file with labels</th>
<th>Output file of labels validation</th>
</tr>
</thead>
<tbody>
<tr>
<td>( /\text{Downloads/example4.txt} )</td>
<td><img src="https://example.com/labels-validation-output.png" alt="" /></td>
</tr>
</tbody>
</table>

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>شكر</td>
<td></td>
</tr>
<tr>
<td>لذيذ</td>
<td></td>
</tr>
<tr>
<td>نجوم</td>
<td></td>
</tr>
<tr>
<td>جميل</td>
<td></td>
</tr>
<tr>
<td>ميليْه</td>
<td></td>
</tr>
<tr>
<td>حبيبي</td>
<td></td>
</tr>
</tbody>
</table>

```
invalid:
    شكرا
    لذيذ
    نجوم
    جميل
    ميليْه
    حبيبي
```

```
valid:
    شكرا
    لذيذ
    نجوم
    جميل
    حبيبي
```

```
invalid:
    Code point U+FEB7 not in repertoire
    Code point U+FEDC not in repertoire
    Code point U+FEAE not in repertoire
```

```
valid:
    شكرا
    لذيذ
    نجوم
    جميل
    حبيبي
```

```
invalid:
    Code point U+0640 not in repertoire
    Code point U+064E not in repertoire
    Code point U+0640 not in repertoire
    Code point U+064E not in repertoire
```

```
valid:
    شكرا
    لذيذ
    نجوم
    جميل
    حبيبي
```
Validate Labels by Uploading the File  (Example 4 - output)

<table>
<thead>
<tr>
<th>Tasks</th>
<th>Status bar shows the status of task (the latest report on top)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tasks are ordered from newest to latest</td>
<td></td>
</tr>
<tr>
<td>Annotate labels on LGR 5</td>
<td>July 25, 2023, 5:05 p.m.</td>
</tr>
<tr>
<td>Review of 1 IDN table</td>
<td>July 21, 2023, 6:57 a.m.</td>
</tr>
<tr>
<td>Review of 1 IDN table</td>
<td>July 13, 2023, 2:55 p.m.</td>
</tr>
</tbody>
</table>

When you delete a task, the corresponding report is not deleted

File life

Click on the download link on the homepage to get the “Diff labels of two LGRs” results
Label Forms

A label, U label against IDNA2008
Label Forms

- Type in a label or upload a list of labels

This function is available for all modes in the LGR tool. The purpose of this is to provide the U-lable/ A-lable and codepoints of the input label(s).
Display all label forms

Check labels one by one

Label
Label can be in U-Label or A-Label form or a list of code points.

Display forms

Get label forms on a list of labels

Labels
Choose file
No file chosen

File should be a text file encoded in UTF-8 and using 0x0A line ending. It must contain one label per line in U-Label or A-Label form or as a list of code points. Comments begin with '#'

Download forms
Label Forms - Input One Label (Example 1)

1. Input can be: Code point Sequence/ U-label/ A-label

2. Click to see the result

<table>
<thead>
<tr>
<th>Code point sequence</th>
<th>U-label</th>
<th>A-label</th>
</tr>
</thead>
<tbody>
<tr>
<td>U+674E U+5B50</td>
<td>李子</td>
<td>xn--i8sx2z</td>
</tr>
</tbody>
</table>

Code point sequence, U-label, A-label - all 3 forms will be generated based on the input
Label Forms - Input One Label (Example 2)

**Display all label forms**

1. **Input can be:** Code point Sequence/ U-label/ A-label

   **Label**

   U+101D U+1031 U+1006 U+102C

   Label can be in U-Label or A-Label form or a list of code points.

2. **Click to see the result**

   ![Display forms button](image)

   **Code point sequence** | **U-label** | **A-label**
   --- | --- | ---
   U+101D U+1031 U+1006 U+102C | ご覧 | xn--tid9b4bs

**Code point sequence, U-label, A-label - all 3 forms will be generated based on the input**
Label Forms - Input a list of Labels (Example 3)

Get label forms on a list of labels

Labels: Choose file
No file chosen

File should be a text file encoded in UTF-8 and using 0x0A line ending. It must contain one label per line in U-Label or A-Label form or as a list of code points. Comments begin with '#'

Download forms

Upload file to check the list of labels (in txt format)

Download the result file of Label forms (in csv format)
Label Forms - Input a list of Labels (Example 3 - input)

- Input file format is .txt.
- Input can be one label per line.
- Each label can be in:
  - Code point Sequence
  - U-label
  - A-label.
- The lines beginning with ‘#’ are comments and will not be processed.

```
# code points – valid
U+1005 U+102D U+1014 U+103A
# code points – invalid
U+1005 U+102D U+1014 U+103A U+002D
#U-label – valid
မြန်
#U-label – invalid
မြန်--ငျ
#A-label – valid
xn--ridd1ji0d
#A-label – invalid
xn--vi8hua2f
```
### Label Forms - Input a list of Labels (Example 3 - Output)

<table>
<thead>
<tr>
<th>Code point sequence</th>
<th>U-label</th>
<th>A-label</th>
<th>Note</th>
</tr>
</thead>
<tbody>
<tr>
<td>U+1005 U+102D U+1014 U+103A</td>
<td>።</td>
<td>= xn--sid4a9d7b</td>
<td></td>
</tr>
<tr>
<td>U+1005 U+102D U+1014 U+103A U+002D</td>
<td>-</td>
<td>-</td>
<td>። is invalid due to hyphen restrictions in the RFC5891 as it ends with a hyphen-minus.</td>
</tr>
<tr>
<td>U+101B U+1010 U+1014 U+102C</td>
<td>።</td>
<td>= xn--3idit4e</td>
<td></td>
</tr>
<tr>
<td>U+101B U+1010 U+002D U+002D U+1014 U+102C</td>
<td>።</td>
<td>= xn--3idit4e</td>
<td></td>
</tr>
<tr>
<td>U+1006 U+102D U+102F U+1004 U+103A</td>
<td>።</td>
<td>= xn--ridd1ji0d</td>
<td></td>
</tr>
<tr>
<td>U+D83C U+DF55 U+D83C U+DF5F U+D83C U+DF79</td>
<td>-</td>
<td>xn--vi8hua2f</td>
<td>'utf-8' codec can't encode characters in position 42-47: surrogates not allowed</td>
</tr>
</tbody>
</table>

When input is **Code point sequence** and **valid**: U-label and A-label forms will be generated.

When input is **Code point sequence** and **invalid**: U-label and A-label forms will not be generated.

When input is **U-label** and **valid**: Code point sequence and A-label forms will be generated.

When input is **U-label** and **invalid**: Code point sequence will be generated, A-label form will not be generated.

When input is **A-label** and **valid**: Code point sequence and U-label forms will be generated.

When input is **A-label** and **invalid**: Code point sequence will be generated, U-label form will not be generated.
Engage with ICANN and IDN Program

Thank You and Questions
Reach us at: IDNProgram@icann.org
Website: icann.org/idn

twitter.com/icann
gplus.to/icann
facebook.com/icannorg
weibo.com/ICANNorg
linkedin.com/company/icann
flickr.com/photos/icann
youtube.com/user/icannnews
slideshare.net/icannpresentations