

# **International Domain Name Software Development Kit Registrar's Guide**

Document Version 2.0  
October 29<sup>th</sup>, 2010

**COPYRIGHT NOTIFICATION**

Copyright © 2010 VeriSign, Inc., as an unpublished work. All rights reserved.

Copyright laws and international treaties protect this document, and any VeriSign product to which it relates.

**VERISIGN PROPRIETARY INFORMATION**

This document is the property of VeriSign, Inc. It may be used by recipient only for the purpose for which it was transmitted and shall be returned upon request or when no longer needed by recipient. It may not be copied or communicated without the prior written consent of VeriSign.

**DISCLAIMER AND LIMITATION OF LIABILITY**

VeriSign, Inc. has made every effort to ensure the accuracy and completeness of all information in this document. However, VeriSign, Inc. assumes no liability to any party for any loss or damage caused by errors or omissions or by statements of any kind in this document, its updates, supplements, or special editions, whether such errors, omissions, or statements result from negligence, accident, or any other cause. VeriSign, Inc. assumes no liability arising out of applying or using the product and no liability for incidental or consequential damages arising from using this document. VeriSign, Inc. disclaims all warranties regarding the information contained herein (whether expressed, implied, or statutory) including implied warranties of merchantability or fitness for a particular purpose.

VeriSign, Inc. makes no representation that interconnecting products in the manner described herein will not infringe upon existing or future patent rights nor do the descriptions contained herein imply granting any license to make, use, or sell equipment or products constructed in accordance with this description. VeriSign, Inc. reserves the right to make changes to any information herein without further notice.

**NOTICE AND CAUTION**

Concerning U.S. Patent or Trademark Rights

The inclusion in this document, the associated online file, or the associated software of any information covered by any patent, trademark, or service mark rights shall not constitute nor imply a grant of, or authority to exercise, any right or privilege protected by such patent, trademark, or service mark. All such rights and privileges are vested in the patent, trademark, or service mark owner, and no other person may exercise such rights without express permission, authority, or license secured from the patent, trademark, or service mark owner.

10/29/2010

**VERISIGN® NAMING SERVICES**

Email [Info@verisign-grs.com](mailto:Info@verisign-grs.com)

Internet <http://www.verisign.com>

**CONTENTS**

Introduction .....	4
Overview .....	4
Unicode .....	4
Punycode .....	5
IDNA Rules .....	5
Installing the Verisign IDN SDK .....	6
Verisign IDN Conversion tool .....	6
Options Menu .....	7
Input Type Menu .....	8
Output Type Menu .....	9
Troubleshooting A Failed IDN Registration .....	11
Example: Never-Allowed Code Points .....	12
Example: Hyphens in the Wrong Position .....	13
Example: Combining Marks (Disallowed Characters in Beginning of Name) .....	14
Example: Sometimes-Allowed Code Points .....	14
Appendices .....	14
Disallowed Code Points .....	15
Verisign-Specific Disallowed Code Points .....	17
Combining Marks .....	17
Sometimes-Allowed Code Points .....	18
Contextual Rules .....	18
Bidi Properties .....	19
Supported Scripts .....	20
Example: Verisign IDN SDK Usage .....	20
Definitions and Acronyms .....	22
References .....	23

## INTRODUCTION

This guide provides background on the protocols that allow users to register a domain name with international characters. It also includes instructions on how to use the VeriSign IDN SDK and related tools to troubleshoot some common issues that may arise.

## OVERVIEW

Internationalized Domain Names in Applications (IDNA) is a mechanism for handling Internationalized Domain Names that contain non-[ASCII](#) characters. The [Domain Name System](#), which performs a lookup service to translate user-friendly names into network addresses for locating Internet resources, is restricted to the use of ASCII characters. This technical limitation initially set the standard for acceptable domain names. The internationalization of domain names is a technical solution to translate names written in language-native scripts into ASCII text representations that are compatible with the Domain Name System.

## UNICODE

“Unicode provides a unique number for every character, no matter what the platform, no matter what the program, no matter what the language.”<sup>1</sup>

Fundamentally, computers deal with numbers. They store letters and other characters by assigning a number to each one. Before Unicode was invented, there were hundreds of different encoding systems for assigning these numbers. Unicode introduced a single standard and a single encoding system. The number assigned to each character in Unicode is called a code point. How the code point is represented in binary form depends on the encoding (e.g., UTF-8 or UTF-16) used.

For example, the symbol © represents the notion of copyright.

The Unicode point for this symbol is 00A9.

In UTF-8, the binary representation of © is 11000010 10101001.

In UTF-16, the binary representation of © is 00000000 10101001.

<sup>1</sup> Unicode Web site, <http://www.unicode.org/standard/WhatIsUnicode.html>, October 2010

## PUNYCODE

IDNA uses Punycode to squeeze the richness of Unicode into DNS. Punycode is an algorithm by which a string of Unicode characters is transformed into a smaller, restricted ASCII character set. Punycode labels have a special prefix: **xn--**

The following four characters precede all IDN labels: 0x58, 0x4E, 0x2D, 0x2D. ASCII characters from the label remain unchanged, and they directly follow the xn-- prefix. Unicode characters become letters and digits that follow the ASCII characters.

Unicode	Punycode
©opy©at.com	xn--opyat-bhad.com

In the Punycode column above, the ASCII characters from ©opy©at.com immediately follow the xn-- prefix. Then, a hyphen separates the encoding of Unicode characters into ASCII. That is, the two copyright symbols © are encoded as ASCII “bhad.” The Unicode points that make up ©opy©at.com are as follows:

00A9 0x6F 0x70 0x79 00A9 0x61 0x74 0x2E 0x63 0x6F 0x6D

## IDNA RULES

Of the 1.1 million possible code points encompassed by Unicode's mapping system, Unicode has assigned roughly 107,000 code points to characters. However, not all these code points are allowed as part of an IDN. These code points can be classified broadly as follows:

- **Always** – The code points in this category are always allowed to be part of an IDN.
- **Never** – The code points in this category are never allowed in an IDN. Please refer to section 8.1 for a list of disallowed code points.
- **Sometimes** – The code points in this category are allowed in an IDN only in limited contexts and only if they meet certain restrictions. Please refer to section 8.4 for a list of code points that are allowed in limited contexts.

To ensure the validity of IDNs being registered within the Domain Name System, IDNA specifies that registries implement the following high-level rules:

- Reject domain names that have characters that are never allowed.
- Reject domain names that have hyphens in certain positions.
- Selectively reject characters that are sometimes allowed.
- Reject certain characters that cannot appear at the beginning of an IDN domain name.
- Reject characters that cannot appear in any situation.

- Apply rules for characters that are written right-to-left (e.g., Arabic).

The VeriSign Internationalized Domain Names Software Development Kit (IDN SDK) implements all these rules.

## INSTALLING THE VERISIGN IDN SDK

The VeriSign IDN SDK is distributed in a compressed zip file format. This is a common format, and conventional operating systems provide tools to inspect and decompress these files. To install the IDN SDK, simply decompress the zip file to a convenient location on your hard drive.

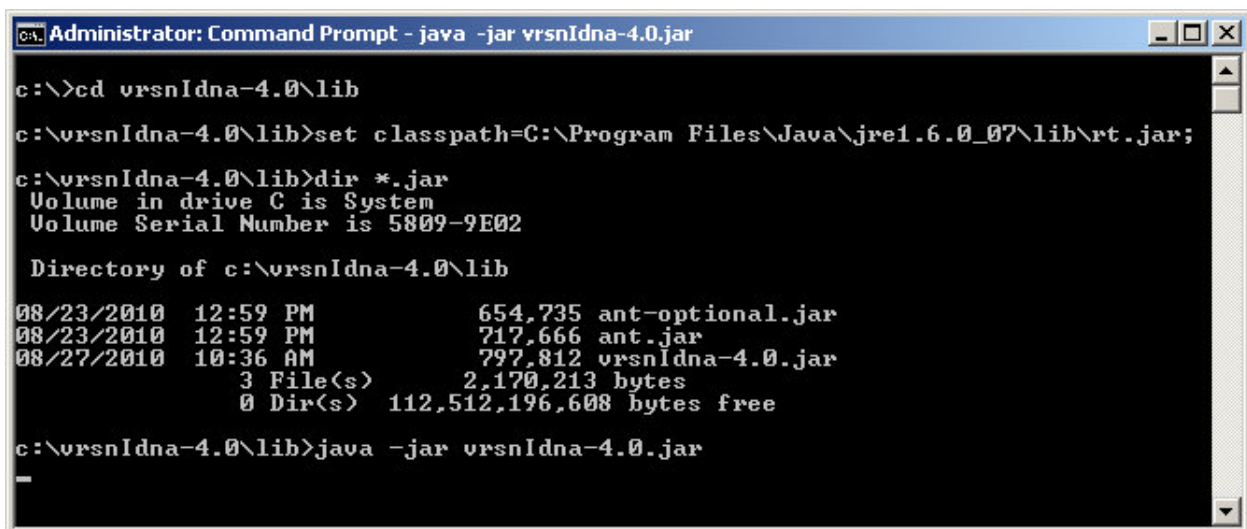
## VERISIGN IDN CONVERSION TOOL

The VeriSign IDN SDK provides a Java GUI-based IDN Conversion Tool to help users assess whether a particular IDN is valid as per IDNA rules. To run this program, Java must be installed on the host machine. Users can obtain the most recent version of Java from [www.java.com](http://www.java.com). Install Java Runtime Environment (JRE) 1.6 on the machine on which you intend to run the IDN Conversion Tool. Note the location of the Java JRE installation folder. You will use this path when setting the CLASSPATH environment variable.

Assuming the following folders:

- IDN SDK installation: **C:\vrsnIdna-4.0**
- Java JRE installation: **C:\Program Files\jre1.6.0\_07**

To start the VeriSign IDN Conversion Tool, execute the commands shown in the following screen shot.



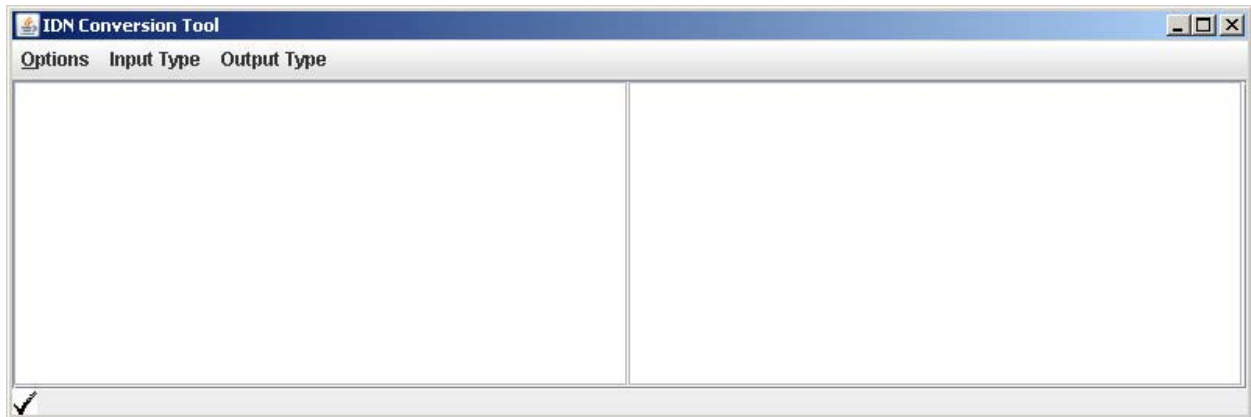
```
C:\>cd vrsnIdna-4.0\lib
c:\vrsnIdna-4.0\lib>set classpath=C:\Program Files\Java\jre1.6.0_07\lib\rt.jar;
c:\vrsnIdna-4.0\lib>dir *.jar
Volume in drive C is System
Volume Serial Number is 5809-9E02

Directory of c:\vrsnIdna-4.0\lib
08/23/2010  12:59 PM                654,735 ant-optional.jar
08/23/2010  12:59 PM                717,666 ant.jar
08/27/2010  10:36 AM                797,812 vrsnIdna-4.0.jar
           3 File(s)                2,170,213 bytes
           0 Dir(s)  112,512,196,608 bytes free

c:\vrsnIdna-4.0\lib>java -jar vrsnIdna-4.0.jar
-
```

The VeriSign IDN Conversion Tool should appear as shown in the following figure.

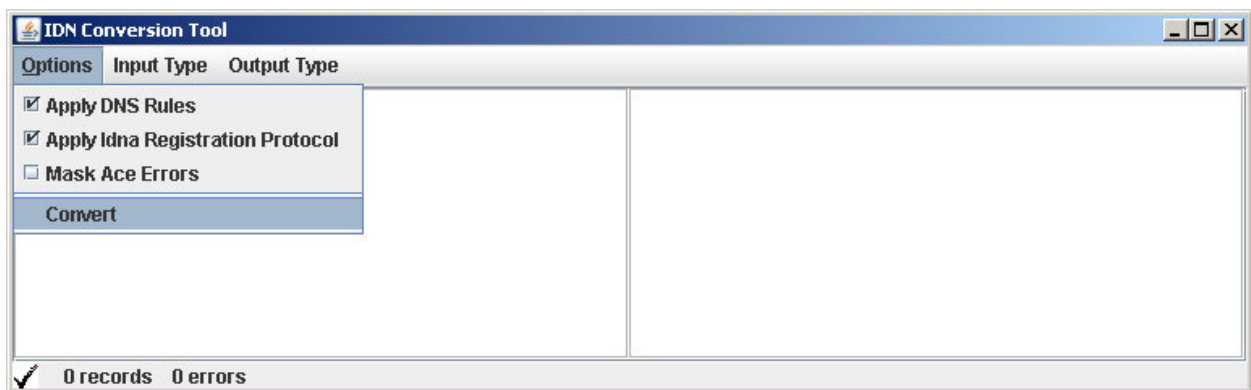
Alternatively, if the CLASSPATH environment variable is set correctly, then double-clicking the **vrnsIdna-4.0.jar** file in Windows Explorer will also start the IDN Conversion Tool.



The IDN Conversion Tool offers many features. For example, it can convert Unicode points to Punycode and vice-versa. Input is entered in the left pane. The resulting conversion is displayed in the right pane.

#### OPTIONS MENU

The Options menu includes three check boxes and the Convert tool.



**Apply DNS Rules:** Use this flag to specify whether the validation algorithm will use the STD 3 ASCII rules. These rules assert that an IDN may contain only letters, digits, and hyphens. They also assert that IDNs must not begin or end with a hyphen character. This option is selected by default.

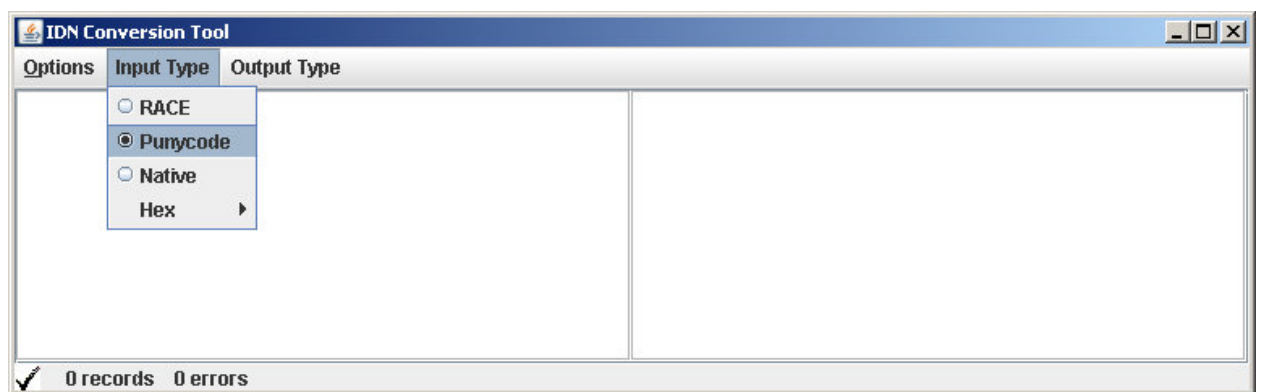
**Apply Idna Registration Protocol:** Use this option to specify whether IDNA rules should be applied. This option is selected by default.

**Mask Ace Errors:** Select this option if you want the IDN Conversion Tool to ignore any errors encountered during ASCII-Compatible Encoding (ACE) encoding. If an error is encountered, the input will be returned as output with alteration. Selecting this option disguises errors because an output is returned. The IDN Conversion Tool includes this option only to be compatible with the specification. This option is not selected by default.

**Convert:** Select this option to convert the data in the input pane (left pane) to the corresponding data type, which is specified in the Output Type menu. The conversion results will appear in the output pane (right pane).

#### INPUT TYPE MENU

The Input Type menu and the Output Type menu have the same options.



**RACE:** RACE stands for Row-based ASCII-Compatible Encoding. Like Punycode, this algorithm converts Unicode characters into an ASCII character set. IDNs that have a RACE prefix (bq--) are not allowed for registration. The IDN Conversion Tool includes this option only for backwards compatibility purposes.

**Punycode:** Choose this option when the input is a Punycode IDN.

**Native:** Choose this option when the input has Native characters. For example, you can input non-ASCII characters when this option is selected.

**Hex:** When selected, this option expands to offer a number of character set options, as shown in the following section.

**Warning** Punycode should be entered in lowercase. Latin capital letters like "A" are prohibited by the IDNA standard. Punycode input with Latin capital letters embedded as ASCII will result in an error.

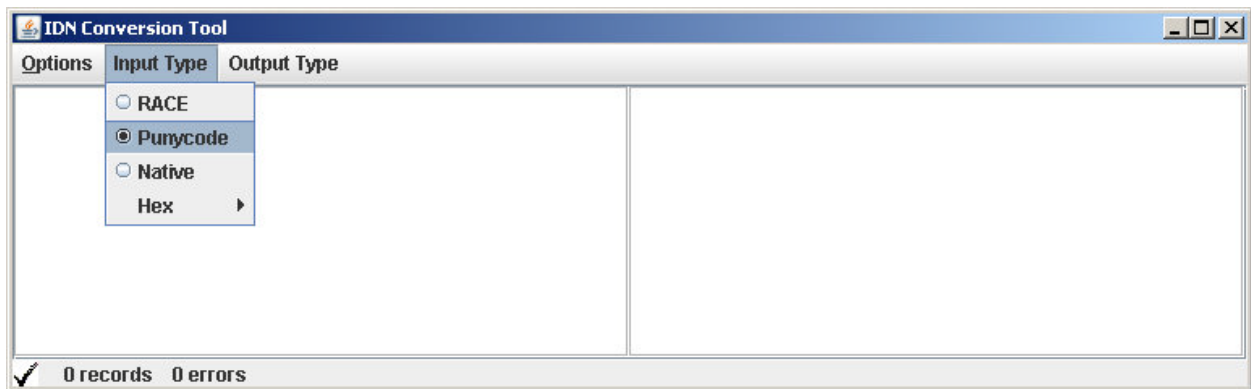


## OUTPUT TYPE MENU

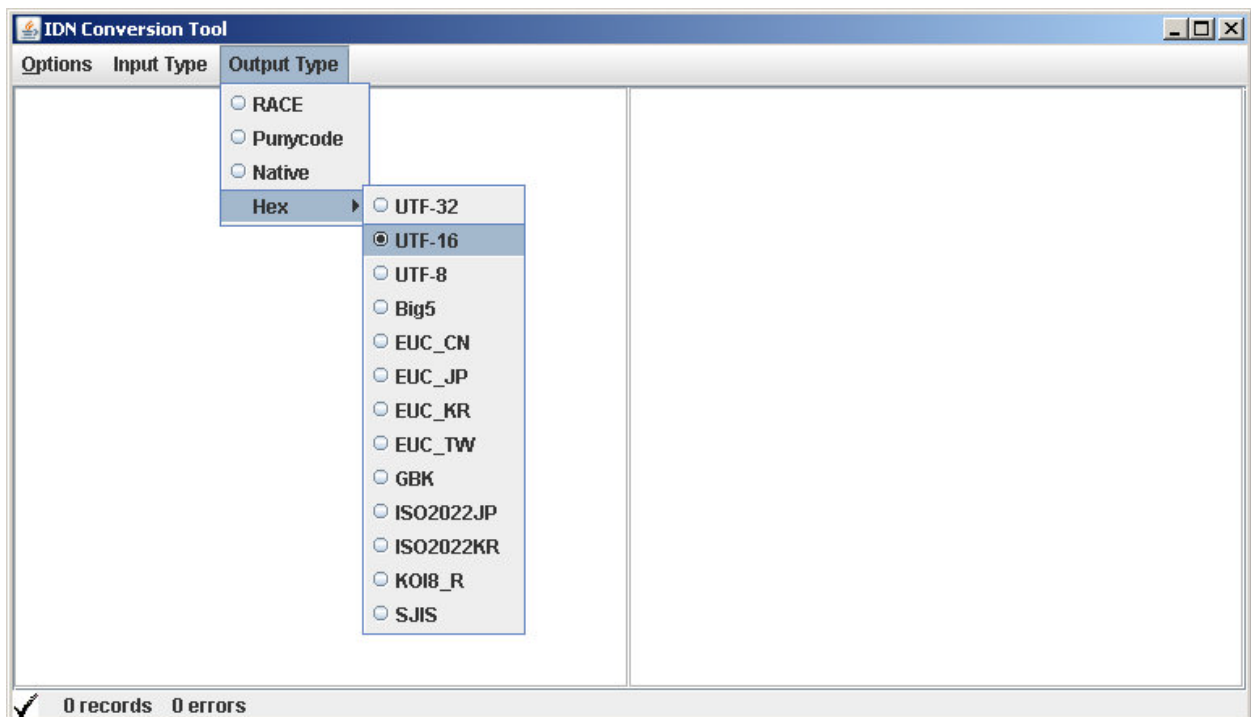
As mentioned in the preceding section, the Input Type and Output Type menus have the same options. Use the Output Type tool to convert IDNs from one type to another. For example, if you have a domain name in Punycode form, use the Output Type tool to see the Unicode points contained in that Punycode IDN.

To convert a Punycode IDN to Unicode points, follow these steps:

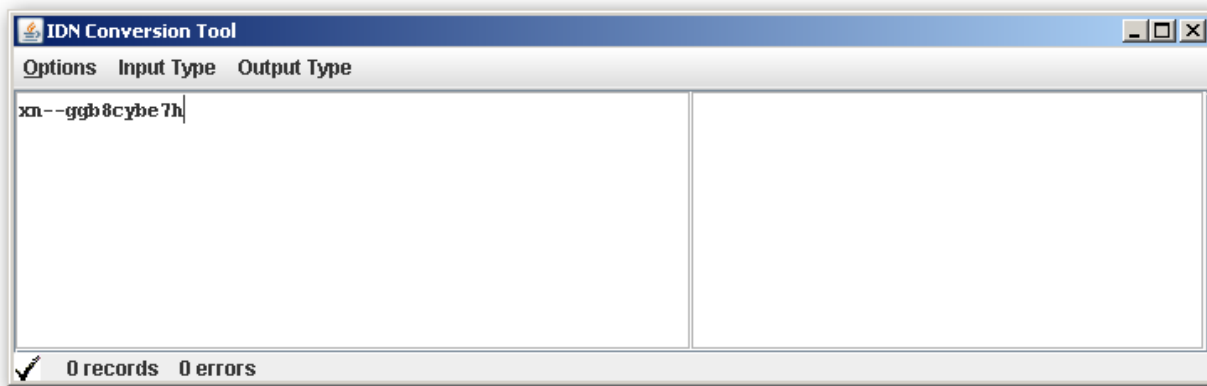
In the Input Type menu, select Punycode.



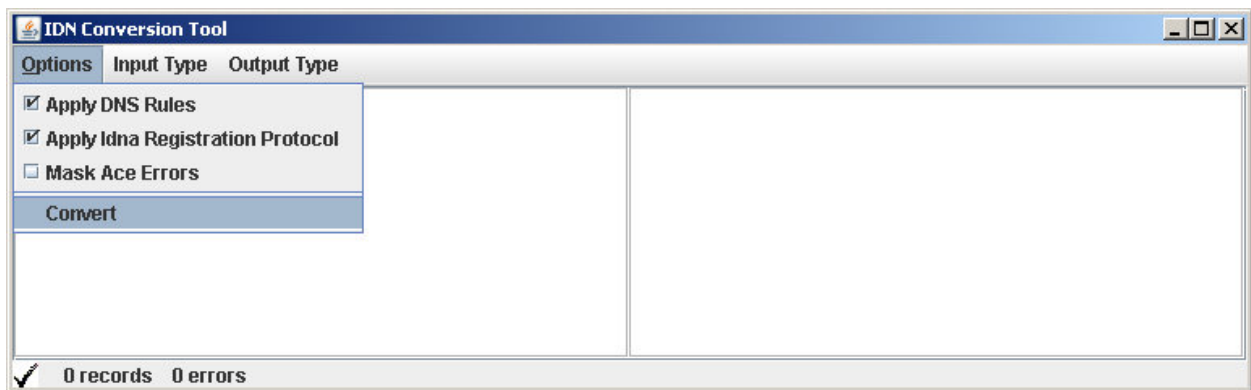
In the Output Type menu, select Hex → UTF-16, as shown in the following figure.



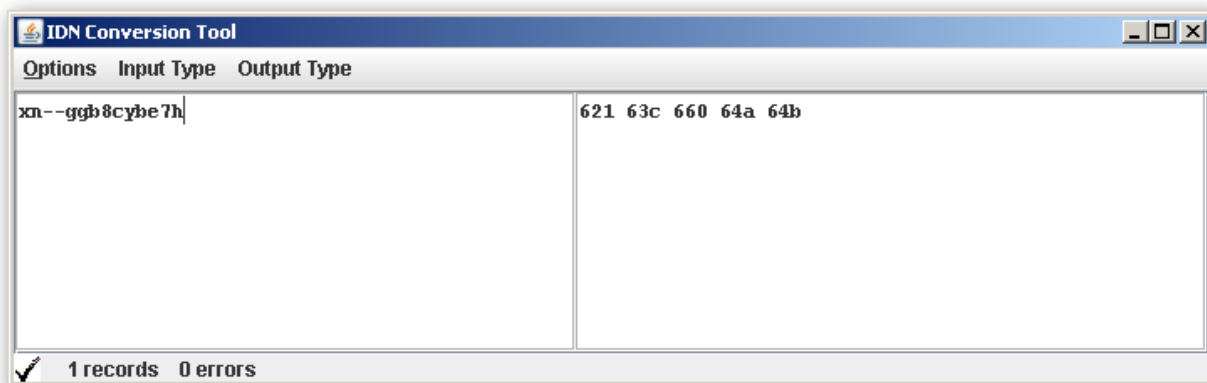
In the left pane, enter the IDN in Punycode form.



In the Options menu, select Convert.



If the IDN is valid, the Unicode points of the IDN will appear in the right pane, as shown in the following figure. Any errors encountered during conversion will also appear in the right pane.



## TROUBLESHOOTING A FAILED IDN REGISTRATION

This section explains how to troubleshoot why an IDN cannot be registered. In most cases, an IDN registration failure occurs when a customer has sent an IDN domain name in a Punycode form that does not adhere to the rules and restrictions regarding valid domain names.

To determine why an IDN registration has failed, follow the procedure outlined here.

1. In the Options menu, ensure that the Apply DNS Rules and Apply Idna Registration Protocol options are selected.
2. In the Input Type menu, select Punycode.
3. In the Output Type menu, select Hex → UTF-16.
4. In the Options menu, click Convert.

If the domain name is valid, the Unicode points of the domain name will appear in the right pane. If the domain name is not valid as per the rules and restrictions, the cause of the error will appear in the right pane.

You can display a list of the code points in the IDN domain name without applying the IDNA registration rules. To do so, uncheck the Apply Idna Registration Protocol option in the Options menu, as shown in the following figure.



In some cases, VeriSign “fails” to register the IDN domain name, yet the right pane does not display an error. This condition results when violations of VeriSign-specific restrictions occur.

VeriSign imposes the following VeriSign-specific restrictions on top of the IDNA2008 rules and restrictions on IDN domain names:

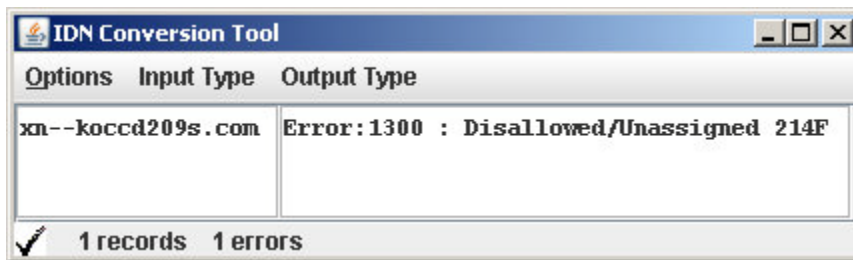
**Disallowed code points:** VeriSign disallows certain code points in an IDN, even though they are allowed in the IDNA RFC. Section 8.2 includes a list of the VeriSign-specific disallowed code points.

**More than one script:** VeriSign also mandates that an IDN domain name cannot contain code points from more than one script. For example, VeriSign will not register IDN domain names that contain code points from both French and Greek.

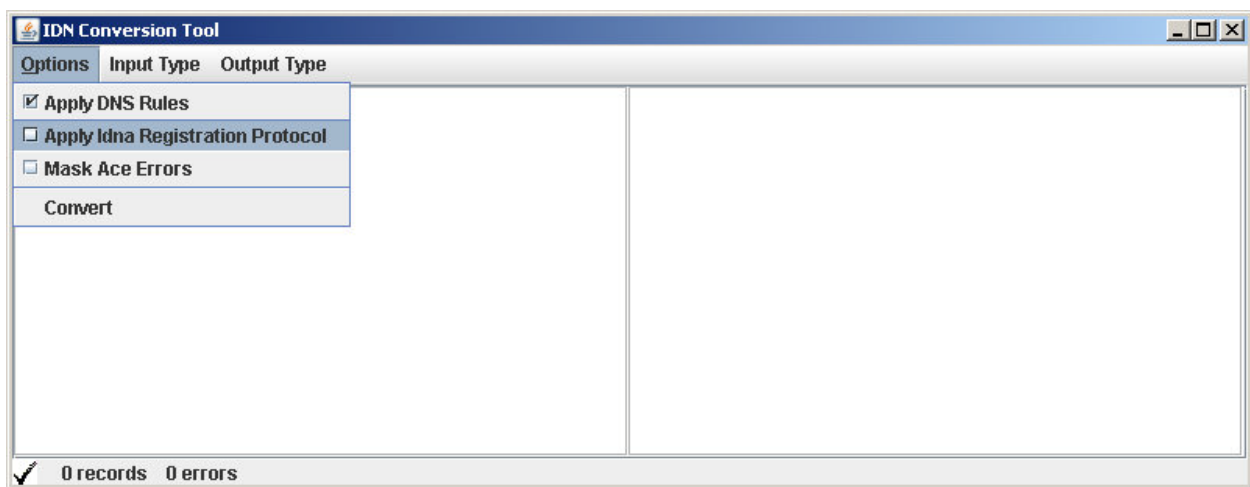
Once the IDN Conversion Tool outputs a list of the code points in the IDN, users can check for VeriSign-specific disallowed code points or code points that belong to more than one script. The code points belonging to each known script in the world are available from Unicode in the file [Scripts.txt](#).

#### EXAMPLE: NEVER-ALLOWED CODE POINTS

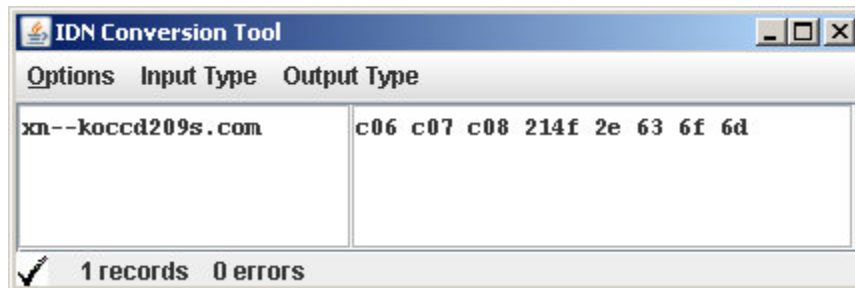
In the following example, the IDN contains code point 0x214F, which is never allowed. The error appears in the right pane.



To see the Unicode points that make up the **xn--koccd209s.com** domain name, the IDNA validation rules must be disabled. To disable the IDNA validation rules, clear the Apply Idna Registration Protocol check box in the Options menu.

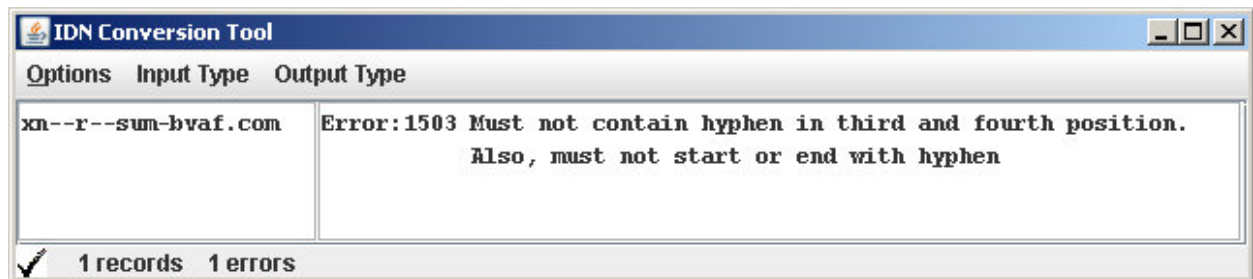


Once the check box is cleared, the IDN Conversion Tool decodes the domain name and displays the constituent Unicode points in the right pane, as shown in the following figure. As seen in the right pane, the 0x214F code point is present in the IDN domain name. Because it is a never-allowed code point, it has been disallowed for registration purposes.

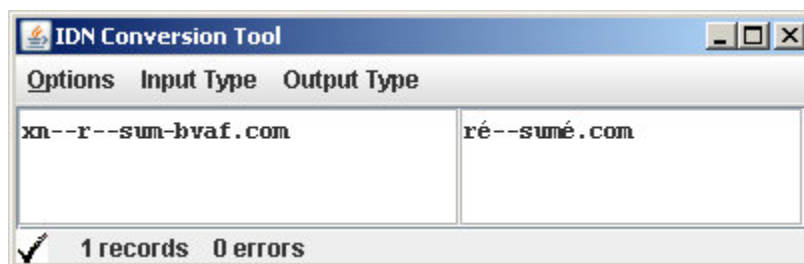


#### EXAMPLE: HYPHENS IN THE WRONG POSITION

IDNA rules specify at what positions the hyphen is disallowed in an IDN domain name. In the following figure, the hyphens are in the third and fourth positions, causing an error.

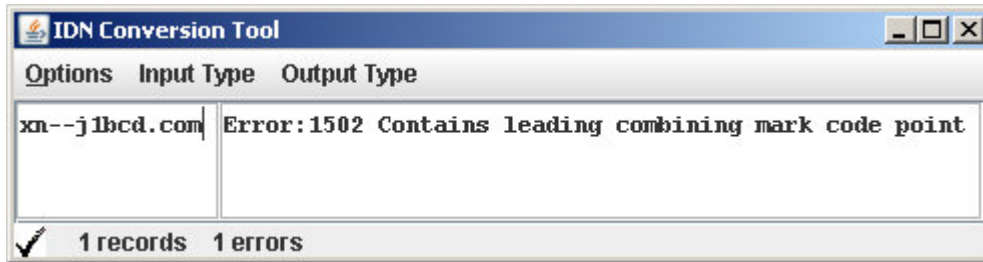


By clearing, the Apply Idna Registration Protocol check box and setting the Output Type to Native, users can see the hyphens in the third and fourth positions of the actual domain name, as shown in the following figure. Here the conversion from Punycode to Native characters has been done without applying the IDNA rules.

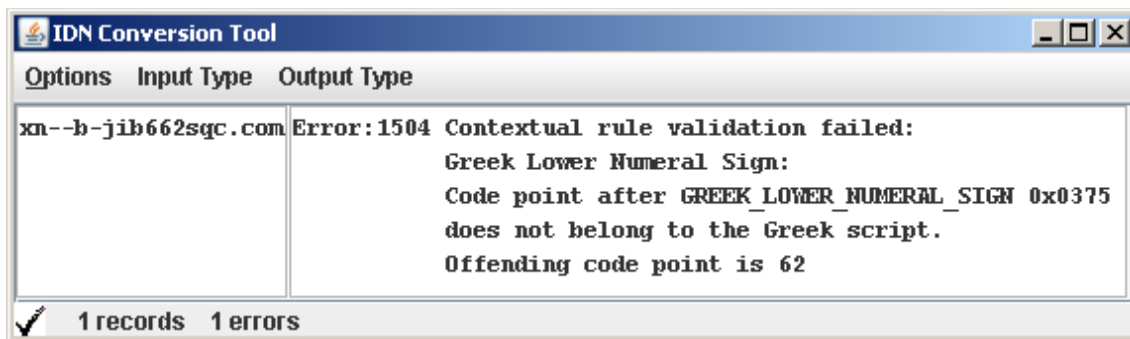


**EXAMPLE: COMBINING MARKS (DISALLOWED CHARACTERS IN BEGINNING OF NAME)**

IDNA rules prohibit certain types of characters, for example the grave accent (`), from being the first code point in an IDN domain name. These types of characters are called combining marks. They must always be combined with other characters. The following example shows the error when such a character is present in the first position of a label.

**EXAMPLE: SOMETIMES-ALLOWED CODE POINTS**

IDNA allows some code points only in limited contexts. That is, IDNA rules govern how these code points may be used in an IDN domain name. For example, the code point U+0375 (Greek Lower Numeral Sign) can be used only when the next code point also belongs to the Greek script. In the following example, the code point that follows the Greek Lower Numeral Sign is the lowercase letter 'b,' which causes the IDN to be invalid, as shown in the error message.

**APPENDICES**

This section includes the following appendices:

- Disallowed Code Points
- VeriSign-Specific Disallowed Code Points
- Combining Marks
- Sometimes-Allowed Code Points
- Contextual Rules

- Bidi Properties
- Supported Scripts
- Example – VeriSign IDN SDK Usage
- Definitions and Acronyms
- References

## DISALLOWED CODE POINTS

0000-002C	014E	01B7-01B8	0224	03EC	04AE	0504	0956-095F
002E-002F	0150	01BC	0226	03EE	04B0	0506	0964-0965
003A-0060	0152	01C4-01CD	0228	03F0-03F2	04B2	0508	0970
007B-00B6	0154	01CF	022A	03F4-03F7	04B4	050A	0973-0978
00B8-00DE	0156	01D1	022C	03F9-03FA	04B6	050C	0980
00F7	0158	01D3	022E	03FD-042F	04B8	050E	0984
0100	015A	01D5	0230	0460	04BA	0510	098D-098E
0102	015C	01D7	0232	0462	04BC	0512	0991-0992
0104	015E	01D9	023A-023B	0464	04BE	0514	09A9
0106	0160	01DB	023D-023E	0466	04C0-04C1	0516	09B1
0108	0162	01DE	0241	0468	04C3	0518	09B3-09B5
010A	0164	01E0	0243-0246	046A	04C5	051A	09BA-09BB
010C	0166	01E2	0248	046C	04C7	051C	09C5-09C6
010E	0168	01E4	024A	046E	04C9	051E	09C9-09CA
0110	016A	01E6	024C	0470	04CB	0520	09CF-09D6
0112	016C	01E8	024E	0472	04CD	0522	09D8-09DF
0114	016E	01EA	02B0-02B8	0474	04D0	0524	09E4-09E5
0116	0170	01EC	02C2-02C5	0476	04D2	0526-0558	09F2-0A00
0118	0172	01EE	02D2-02EB	0478	04D4	055A-0560	0A04
011A	0174	01F1-01F4	02ED	047A	04D6	0587-0590	0A0B-0A0E
011C	0176	01F6-01F8	02EF-02FF	047C	04D8	05BE	0A11-0A12
011E	0178-0179	01FA	0340-0341	047E	04DA	05C0	0A29
0120	017B	01FC	0343-0345	0480	04DC	05C3	0A31
0122	017D	01FE	034F	0482	04DE	05C6	0A33-0A34
0124	017F	0200	0370	0488-048A	04E0	05C8-05CF	0A36-0A37
0126	0181-0182	0202	0372	048C	04E2	05EB-05EF	0A3A-0A3B
0128	0184	0204	0374	048E	04E4	05F5-060F	0A3D
012A	0186-0187	0206	0376	0490	04E6	061B-0620	0A43-0A46
012C	0189-018B	0208	0378-037A	0492	04E8	0640	0A49-0A4A
012E	018E-0191	020A	037E-038F	0494	04EA	065F	0A4E-0A50
0130	0193-0194	020C	0391-03AB	0496	04EC	066A-066D	0A52-0A5B
0132-0134	0196-0198	020E	03CF-03D6	0498	04EE	0675-0678	0A5D-0A65
0136	019C-019D	0210	03D8	049A	04F0	06D4	0A76-0A80
0139	019F-01A0	0212	03DA	049C	04F2	06DD-06DE	0A84
013B	01A2	0214	03DC	049E	04F4	06E9	0A8E
013D	01A4	0216	03DE	04A0	04F6	0700-070F	0A92
013F-0141	01A6-01A7	0218	03E0	04A2	04F8	074B-074C	0AA9
0143	01A9	021A	03E2	04A4	04FA	07B2-07BF	0AB1
0145	01AC	021C	03E4	04A6	04FC	07F6-07FF	0AB4
0147	01AE-01AF	021E	03E6	04A8	04FE	082E-08FF	0ABA-0ABB
0149-014A	01B1-01B3	0220	03E8	04AA	0500	093A-093B	0AC6
014C	01B5	0222	03EA	04AC	0502	094F	0ACA

OACE-OACF	0CD7-0CDD	0F43	17D4-17D6	1E22	1E92	1F46-1F4F	2CB6
OAD1-OADF	0CDF	0F48	17D8-17DB	1E24	1E94	1F58-1F5F	2CB8
OAE4-OAE5	0CE4-0CE5	0F4D	17DE-17DF	1E26	1E9A-1E9B	1F68-1F6F	2CBA
OAF0-0B00	0CF0-0D01	0F52	17EA-180F	1E28	1E9E	1F71	2CBC
OB04	0D04	0F57	181A-181F	1E2A	1EA0	1F73	2CBE
OB0D-0B0E	0D0D	0F5C	1878-187F	1E2C	1EA2	1F75	2CC0
OB11-0B12	0D11	0F69	18AB-18AF	1E2E	1EA4	1F77	2CC2
OB29	0D29	0F6D-0F70	18F6-18FF	1E30	1EA6	1F79	2CC4
OB31	0D3A-0D3C	0F73	191D-191F	1E32	1EA8	1F7B	2CC6
OB34	0D45	0F75-0F79	192C-192F	1E34	1EAA	1F7D-1FAF	2CC8
OB3A-0B3B	0D49	0F81	193C-1945	1E36	1EAC	1FB2-1FB5	2CCA
OB45-0B46	0D4E-0D56	0F85	196E-196F	1E38	1EAE	1FB7-1FC5	2CCC
OB49-0B4A	0D58-0D5F	0F8C-0F8F	1975-197F	1E3A	1EB0	1FC7-1FCF	2CCE
OB4E-0B55	0D64-0D65	0F93	19AC-19AF	1E3C	1EB2	1FD3-1FD5	2CD0
OB58-0B5E	0D70-0D79	0F98	19CA-19CF	1E3E	1EB4	1FD8-1FDF	2CD2
OB64-0B65	0D80-0D81	0F9D	19DB-19FF	1E40	1EB6	1FE3	2CD4
OB70	0D84	0FA2	1A1C-1A1F	1E42	1EB8	1FE8-1FF5	2CD6
OB72-0B81	0D97-0D99	0FA7	1A5F	1E44	1EBA	1FF7-200B	2CD8
OB84	0DB2	0FAC	1A7D-1A7E	1E46	1EBC	200E-214D	2CDA
OB8B-0B8D	0DBC	0FB9	1A8A-1A8F	1E48	1EBE	214F-2183	2CDC
OB91	0DBE-0DBF	0FBD-0FC5	1A9A-1AA6	1E4A	1EC0	2185-2C2F	2CDE
OB96-0B98	0DC7-0DC9	0FC7-0FFF	1AA8-1AFF	1E4C	1EC2	2C5F-2C60	2CE0
OB9B	0DCB-0DCE	104A-104F	1B4C-1B4F	1E4E	1EC4	2C62-2C64	2CE2
OB9D	0DD5	109E-10CF	1B5A-1B6A	1E50	1EC6	2C67	2CE5-2CEB
OBAA-0BA2	0DD7	10FB-11FF	1B74-1B7F	1E52	1EC8	2C69	2CED
OBAS-0BA7	0DE0-0DF1	1249	1BAB-1BAD	1E54	1ECA	2C6B	2CF2-2CFF
OBAB-0BAD	0DF4-0E00	124E-124F	1BBA-1BFF	1E56	1ECC	2C6D-2C70	2D26-2D2F
OBBA-0BBD	0E33	1257	1C38-1C3F	1E58	1ECE	2C72	2D66-2D7F
OBC3-0BC5	0E3B-0E3F	1259	1C4A-1C4C	1E5A	1ED0	2C75	2D97-2D9F
OBC9	0E4F	125E-125F	1C7E-1CCF	1E5C	1ED2	2C7C-2C80	2DA7
OBCE-0BCF	0E5A-0E80	1289	1CD3	1E5E	1ED4	2C82	2DAF
OBDD-0BD6	0E83	128E-128F	1CF3-1CFF	1E60	1ED6	2C84	2DB7
OBDD-0BE5	0E85-0E86	12B1	1D2C-1D2E	1E62	1ED8	2C86	2DBF
OBFO-0C00	0E89	12B6-12B7	1D30-1D3A	1E64	1EDA	2C88	2DC7
OC04	0E8B-0E8C	12BF	1D3C-1D4D	1E66	1EDC	2C8A	2DCF
OC0D	0E8E-0E93	12C1	1D4F-1D6A	1E68	1EDE	2C8C	2DD7
OC11	0E98	12C6-12C7	1D78	1E6A	1EE0	2C8E	2DDF
OC29	0EA0	12D7	1D9B-1DBF	1E6C	1EE2	2C90	2E00-2E2E
OC34	0EA4	1311	1DE7-1DFC	1E6E	1EE4	2C92	2E30-3004
OC3A-0C3C	0EA6	1316-1317	1E00	1E70	1EE6	2C94	3008-3029
OC45	0EA8-0EA9	135B-135E	1E02	1E72	1EE8	2C96	302E-303B
OC49	0EAC	1360-137F	1E04	1E74	1EEA	2C98	303D-3040
OC4E-0C54	0EB3	1390-139F	1E06	1E76	1EEC	2C9A	3097-3098
OC57	0EBA	13F5-1400	1E08	1E78	1EEE	2C9C	309B-309C
OC5A-0C5F	0EBE-0EBF	166D-166E	1E0A	1E7A	1EF0	2C9E	309F-30A0
OC64-0C65	0EC5	1680	1E0C	1E7C	1EF2	2CA0	30FF-3104
OC70-0C81	0EC7	169B-169F	1E0E	1E7E	1EF4	2CA2	312E-319F
OC84	0ECE-0ECF	16EB-16FF	1E10	1E80	1EF6	2CA4	31B8-31EF
OC8D	0EDA-0EFF	170D	1E12	1E82	1EF8	2CA6	3200-33FF
OC91	0F01-0F0A	1715-171F	1E14	1E84	1EFA	2CA8	4DB6-4DFF
OCA9	0F0C-0F17	1735-173F	1E16	1E86	1EFC	2CAA	9FCC-9FFF
OCB4	0F1A-0F1F	1754-175F	1E18	1E88	1EFE	2CAC	A48D-A4CF
OCBA-0CBB	0F2A-0F34	176D	1E1A	1E8A	1F08-1F0F	2CAE	A4FE-A4FF
OCC5	0F36	1771	1E1C	1E8C	1F16-1F1F	2CB0	A60D-A60F
OCC9	0F38	1774-177F	1E1E	1E8E	1F28-1F2F	2CB2	A62C-A640
OCCE-0CD4	0F3A-0F3D	17B4-17B5	1E20	1E90	1F38-1F3F	2CB4	A642



A644	A670-A67B	A728	A752	A782	AAC3-AADA	1004E-1004F	10916-1091F
A646	A67E	A72A	A754	A784	AADE-ABBF	1005E-1007F	1093A-109FF
A648	A680	A72C	A756	A786	ABEB	100FB-101FC	10A04
A64A	A682	A72E	A758	A789-A78B	ABEE-ABEF	101FE-1027F	10A07-10A0B
A64C	A684	A732	A75A	A78D-A7FA	ABFA-ABFF	1029D-1029F	10A14
A64E	A686	A734	A75C	A828-A83F	D7A4-FA0D	102D1-102FF	10A18
A650	A688	A736	A75E	A874-A87F	FA10	1031F-1032F	10A34-10A37
A652	A68A	A738	A760	A8C5-A8CF	FA12	10341	10A3B-10A3E
A654	A68C	A73A	A762	A8DA-A8DF	FA15-FA1E	1034A-1037F	10A40-10A5F
A656	A68E	A73C	A764	A8F8-A8FA	FA20	1039E-1039F	10A7D-10AFF
A658	A690	A73E	A766	A8FC-A8FF	FA22	103C4-103C7	10B36-10B3F
A65A	A692	A740	A768	A92E-A92F	FA25-FA26	103D0-10427	10B56-10B5F
A65C	A694	A742	A76A	A954-A97F	FA2A-FB1D	1049E-1049F	10B73-10BFF
A65E	A696	A744	A76C	A9C1-A9CE	FB1F-FE1F	104AA-107FF	10C49-1107F
A660-A662	A698-A69F	A746	A76E	A9DA-A9FF	FE27-FE72	10806-10807	110BB-11FFF
A664	A6E6-A6EF	A748	A770	AA37-AA3F	FE74-FFFF	10809	1236F-12FFF
A666	A6F2-A716	A74A	A779	AA4E-AA4F	1000C	10836	1342F-1FFFF
A668	A720-A722	A74C	A77B	AA5A-AA5F	10027	10839-1083B	2A6D7-2A6FF
A66A	A724	A74E	A77D-A77E	AA77-AA79	1003B	1083D-1083E	2B735-10FFFF
A66C	A726	A750	A780	AA7C-AA7F	1003E	10856-108FF	

## VERISIGN-SPECIFIC DISALLOWED CODE POINTS

00DF	1380-139F	2190-245F	27F0-2E7F	4DC0-4DFF	FE30-FE6F	E0080-EFFFF
003C2	18B0-1DFF	2500-25D9	27F0-2E7F	A4D0-ABFF	FFF0-1FFFF	
0750-077F	2000-206F	2600-27BF	2FE0-2FEF	D7B0-D7FF	2A6E0-2F7FF	
07C0-08FF	20A0-20FF	27D0-27EB	3000-303F	FE00-FE1F	2FA20-DFFFF	

## COMBINING MARKS

0300-036F	0829-082D	0A81-0A83	0C4A-0C4D	0DF2-0DF3	1062-1064	1A60-1A7C	A6F0-A6F1
0483-0489	0900-0903	0ABC	0C55-0C56	0E31	1067-106D	1A7F	A802
0591-05BD	093C	0ABE-0AC5	0C62-0C63	0E34-0E3A	1071-1074	1B00-1B04	A806
05BF	093E-094E	0AC7-0AC9	0C82-0C83	0E47-0E4E	1082-108D	1B34-1B44	A80B
05C1-05C2	0951-0955	0ACB-0ACD	0CBC	0EB1	108F	1B6B-1B73	A823-A827
05C4-05C5	0962-0963	0AE2-0AE3	0CBE-0CC4	0EB4-0EB9	109A-109D	1B80-1B82	A880-A881
05C7	0981-0983	0B01-0B03	0CC6-0CC8	0EBB-0EBC	135F	1BA1-1BAA	A8B4-A8C4
0610-061A	09BC	0B3C	0CCA-0CCD	0EC8-0ECD	1712-1714	1C24-1C37	A8E0-A8F1
064B-065E	09BE-09C4	0B3E-0B44	0CD5-0CD6	0F18-0F19	1732-1734	1CD0-1CD2	A926-A92D
0670	09C7-09C8	0B47-0B48	0CE2-0CE3	0F35	1752-1753	1CD4-1CE8	A947-A953
06D6-06DC	09CB-09CD	0B4B-0B4D	0D02-0D03	0F37	1772-1773	1CED	A980-A983
06DE-06E4	09D7	0B56-0B57	0D3E-0D44	0F39	17B6-17D3	1CF2	A9B3-A9C0
06E7-06E8	09E2-09E3	0B62-0B63	0D46-0D48	0F3E-0F3F	17DD	1DC0-1DE6	AA29-AA36
06EA-06ED	0A01-0A03	0B82	0D4A-0D4D	0F71-0F84	180B-180D	1DFD-1DFF	AA43
0711	0A3C	0BBE-0BC2	0D57	0F86-0F87	18A9	20D0-20F0	AA4C-AA4D
0730-074A	0A3E-0A42	0BC6-0BC8	0D62-0D63	0F90-0F97	1920-192B	2CEF-2CF1	AA7B
07A6-07B0	0A47-0A48	0BCA-0BCD	0D82-0D83	0F99-0FBC	1930-193B	2DE0-2DFF	AAB0
07EB-07F3	0A4B-0A4D	0BD7	0DCA	0FC6	19B0-19C0	302A-302F	AAB2-AAB4
0816-0819	0A51	0C01-0C03	0DCF-0DD4	102B-103E	19C8-19C9	3099-309A	AAB7-AAB8
081B-0823	0A70-0A71	0C3E-0C44	0DD6	1056-1059	1A17-1A1B	A66F-A672	AABE-AABF
0825-0827	0A75	0C46-0C48	0DD8-0DDF	105E-1060	1A55-1A5E	A67C-A67D	AAC1

ABE3-ABEA	FE00-FE0F	10A01-10A03	10A38-10A3A	110B0-110BA	1D17B-1D182	1D242-1D244
ABEC-ABED	FE20-FE26	10A05-10A06	10A3F	1D165-1D169	1D185-1D18B	E0100-E01EF
FB1E	101FD	10A0C-10A0F	11080-11082	1D16D-1D172	1D1AA-1D1AD	

#### SOMETIMES-ALLOWED CODE POINTS

00B7	0375	05F3-05F4	0660-0669	06F0-06F9	200C-200D	30FB
------	------	-----------	-----------	-----------	-----------	------

#### CONTEXTUAL RULES

Please use the following references, as needed, when reviewing the contextual rules in the table below:

- To determine the Canonical Combining Class property of a Unicode point, please refer to the Unicode [UnicodeData.txt](#) document.
- To determine the Joining Type property of a Unicode point, please refer to the Unicode [DerivedJoiningType.txt](#) document.
- To determine the Script Name property of a Unicode point, please refer to the Unicode [Scripts.txt](#) document.

Contextual Rule	Description
Zero Width Non-Joiner	The Unicode point U+200C is allowed only if the Canonical Combining Class of the preceding character is Virama and the Joining Class of the surrounding characters matches the following pattern: (L or D) (Zero or more T) U+200C (Zero or more T) (R or D)
Zero Width Joiner	The Unicode point U+200D is allowed only if the Canonical Combining Class of the preceding character is Virama.
Middle Dot	The Unicode point U+00B7 is allowed only if the preceding and following Unicode point is a lowercase Latin "l" (U+006C).
Greek Lower Numeral Sign	The Unicode point U+0375 is allowed only if the Script Name of the following Unicode point is Greek.
Hebrew Punctuation Geresh	The Unicode point U+05F3 is allowed only if the Script Name of the preceding Unicode point is Hebrew.
Hebrew Punctuation Gershayim	The Unicode point U+05F4 is allowed only if the Script Name of the preceding Unicode point is Hebrew.
Katakana Middle Dot	The Unicode point U+30FB is allowed if and only if the Script Name of at least one Unicode point in the label is one of ("Hiragana", "Katakana", or "Han")
Arabic Indic Digits	The Unicode points U+0660 – U+0669 cannot appear in the same label as the Unicode points U+06F0 – U+06F9.
Extended Arabic Indic Digits	The Unicode points U+06F0 – U+06F9 cannot appear in the same label as the Unicode points U+0660 – U+0669 .

## BIDI PROPERTIES

Labels in Bidi domain names must meet all six conditions in the following list in order to comply with IDNA rules.

To determine the R, AL, or L Bidi property of a Unicode point, please refer to the UnicodeData.txt document.

Section 2 of RFC 5893<sup>2</sup> stipulates the following six rules.

- 1. The first character must be a character with Bidi property L, R, or AL. If it has the R or AL property, it is an RTL label; if it has the L property, it is an LTR label.*
- 2. In an RTL label, only characters with the Bidi properties R, AL, AN, EN, ES, CS, ET, ON, BN, or NSM are allowed.*
- 3. In an RTL label, the end of the label must be a character with Bidi property R, AL, EN, or AN, followed by zero or more characters with Bidi property NSM.*

4. *In an RTL label, if an EN is present, no AN may be present, and vice versa.*
5. *In an LTR label, only characters with the Bidi properties L, EN, ES, CS, ET, ON, BN, or NSM are allowed.*
6. *In an LTR label, the end of the label must be a character with Bidi property L or EN, followed by zero or more characters with Bidi property NSM.*

<sup>2</sup> Alvestrand and Karp, RFC 5893: Right-to-Left Scripts for IDNA, August 2010, <http://tools.ietf.org/html/rfc5893>

## SUPPORTED SCRIPTS

Arabic	Cham	Glagolitic	Inscriptional_Pa	Lepcha	Ol_Chiki	Saurashtra	Thaana
Armenian	Cherokee	Gothic	hlavi	Limbu	Old_Italic	Shavian	Thai
Avestan	Common	Greek	Inscriptional_Pa	Linear_B	Old_Persian	Sinhala	Tibetan
Balinese	Coptic	Gujarati	rthian	Lisu	Old_South_Ara	Sundanese	Tifinagh
Bamum	Cuneiform	Gurmukhi	Javanese	Lycian	bian	Syloti_Nagri	Ugaritic
Bengali	Cypriot	Han	Kaith	Lydian	Old_Turkic	Syriac	Vai
Bopomofo	Cyrillic	Hangul	Kannada	Malayalam	Oriya	Tagalog	Yi
Braille	Deseret	Hanunoo	Katakana	Meetei_Mayek	Osmanya	Tagbanwa	
Buginese	Devanagari	Hebrew	Kayah_Li	Mongolian	Phags_Pa	Tai_Le	
Buhid	Egyptian_Hiero	Hiragana	Kharoshthi	Myanmar	Phoenician	Tai_Tham	
Canadian_Aborigi	glyphs	Imperial_Aram	Khmer	New_Tai_Lue	Rejang	Tai_Viet	
nal	Ethiopic	aic	Lao	Nko	Runic	Tamil	
Carian	Georgian	Inherited	Latin	Ogham	Samaritan	Telugu	

## EXAMPLE: VERISIGN IDN SDK USAGE

The following code shows how to use the VeriSign IDN SDK programmatically to enforce the IDNA2008 Protocol rules. “ “

```
public class DemoCmdLine {
    static private boolean useStd3AsciiRules = true;
    static private boolean throwExceptions = true;
    static private boolean idnaRegistrationProtocol = true;

    static public void main ( String[] args ) throws XcodeException {
        final Ace ace = new Punycode( useStd3AsciiRules );
        final Idna idna = new Idna(ace, throwExceptions,
            idnaRegistrationProtocol);
        final String utf16 = "résumé";
        final int[] unicode = Unicode.encode( utf16.toCharArray() );
    }
}
```

```
final String punycode = new String(idna.domainToAscii(unicode) );
final int[] roundTripCheck = idna.domainToUnicode(punycode.toCharArray());
System.out.println("utf16: " + utf16 );
System.out.println("unicode: " + Hex.encode( unicode ) );
System.out.println("punycode: " + punycode );
System.out.println("roundTripCheck: "+Hex.encode(roundtripCheck));
}
}
```

In the preceding code sample, the `useStd3AsciiRules` boolean variable indicates that the standard DNS rules must be enforced. These rules state that the domain name may contain only letters, digits, and hyphens. This variable must be set to true to enforce the IDNA2008 Protocol rules.

The **`throwExceptions`** boolean variable indicates that errors encountered while decoding Unicode points from an ASCII label should be allowed to be thrown. This variable must be set to true to enforce the IDNA2008 Protocol rules.

The **`idnaRegistrationProtocol`** boolean variable indicates that the IDNA2008 Protocol rules must be enforced. By setting this variable to false, the SDK will only ACE encode/decode the domain name.

The IDNA2008 Protocol uses only the Punycode encoding. For this reason, the code sample instantiates a Punycode object. ACE objects like Punycode can be configured to apply a special set of rules called STD 3 ASCII rules. These rules enforce certain character restrictions for domain names. In this example, we pass a true flag to the constructor, indicating that we want to apply these domain name rules.

The `Idna` object is instantiated, passing in the Punycode object instantiated above it. The constructor also takes two boolean variables, indicating that errors during the decode process must be thrown and that the IDNA2008 Registration protocol rules must be enforced.

The Unicode points present in the “résumé” IDN are printed.

The **`Idna.domainToAscii()`** method is used to convert the IDN “résumé” into the Punycode domain name “xn--rsum-bpad.”

The **Idna.domainToUnicode()** method is used to print the code points present in the encoded Punycode domain name “xn--rsum-bpad.” If there are any errors from the conversion to/from Unicode/Punycode, they are printed to the console.

Here is the result of executing the preceding code.

```
utf16: résumé
unicode: 72 e9 73 75 6d e9
punycode: xn--rsum-bpad
check: 72 e9 73 75 6d e9
```

## DEFINITIONS AND ACRONYMS

Term	Description
URI	Uniform/Universal Resource Identifier (US-ASCII only)
IRI	Internationalized Resource Identifier (Unicode)
domain name	A name in the DNS
DNS	Domain Name System
IDN	Internationalized Domain Names
IDNA	Internationalized Domain Names in Applications
Unicode	A computing industry standard for the consistent representation and handling of text expressed in most of the world's writing systems
Punycode	A simple and efficient ASCII-Compatible Encoding (ACE) algorithm designed for use with Internationalized Domain Names. It transforms a Unicode string into a string of characters (i.e., ASCII letters, digits, and hyphens) allowed in hostname labels and back again.
RFC	Request for Comment
IETF	Internet Engineering Task Force

Unicode point	An integer value between 0 and 0x10FFFF inclusive. Each such integer value represents exactly one Unicode character such as the Latin capital letter 'A' (0x0041) or the Greek small letter $\pi$ (0x03C0).
ACE	ASCII-Compatible Encoding
ACE label	An ACE label is an internationalized label that can be rendered in ASCII and is equivalent to an internationalized label that cannot be rendered in ASCII.

## REFERENCES

Topic	Description
IDNA2008 Definitions	<a href="#">rfc5890-ietf-idnabis-defs</a>
IDNA2008 Rationale	<a href="#">rfc5894-ietf-idnabis-rationale</a>
IDNA2008 Protocol	<a href="#">rfc5891-ietf-idnabis-protocol</a>
IDNA2008 Bidi	<a href="#">rfc5893-ietf-idnabis-bidi</a>
IDNA2008 Tables	<a href="#">rfc5892-ietf-idnabis-tables</a>
Unicode	<a href="http://www.unicode.org">http://www.unicode.org</a>