

Punycode

Stability: 0 - Deprecated

The version of the punycode module bundled in Node.js is being deprecated. In a future major version of Node.js this module will be removed. Users currently depending on the `punycode` module should switch to using the userland-provided [Punycode.js](#) module instead. For punycode-based URL encoding, see [url.domainToASCII](#) or, more generally, the [WHATWG URL API](#).

The `punycode` module is a bundled version of the [Punycode.js](#) module. It can be accessed using:

```
const punycode = require('punycode');
```

[Punycode](#) is a character encoding scheme defined by RFC 3492 that is primarily intended for use in Internationalized Domain Names. Because host names in URLs are limited to ASCII characters only, Domain Names that contain non-ASCII characters must be converted into ASCII using the Punycode scheme. For instance, the Japanese character that translates into the English word, 'example' is '例'. The Internationalized Domain Name, '例.com' (equivalent to 'example.com') is represented by Punycode as the ASCII string 'xn--fsq.com'.

The `punycode` module provides a simple implementation of the Punycode standard.

The `punycode` module is a third-party dependency used by Node.js and made available to developers as a convenience. Fixes or other modifications to the module must be directed to the [Punycode.js](#) project.

`punycode.decode(string)`

- `string` {string}

The `punycode.decode()` method converts a [Punycode](#) string of ASCII-only characters to the equivalent string of Unicode codepoints.

```
punycode.decode('maana-pta'); // 'mañana'  
punycode.decode('--dqq34k'); // '🇨🇪-🇩🇪'
```

`punycode.encode(string)`

- `string` {string}

The `punycode.encode()` method converts a string of Unicode codepoints to a [Punycode](#) string of ASCII-only characters.

```
punycode.encode('mañana'); // 'maana-pta'  
punycode.encode('🇨🇪-🇩🇪'); // '--dqq34k'
```

`punycode.toASCII(domain)`

- `domain` {string}

The `punycode.toASCII()` method converts a Unicode string representing an Internationalized Domain Name to [Punycode](#). Only the non-ASCII parts of the domain name will be converted. Calling `punycode.toASCII()` on a

string that already only contains ASCII characters will have no effect.

```
// encode domain names
punycode.toASCII('mañana.com'); // 'xn--maana-pta.com'
punycode.toASCII('ø-ø.com'); // 'xn----dgo34k.com'
punycode.toASCII('example.com'); // 'example.com'
```

punycode.toUnicode(domain)

- domain {string}

The `punycode.toUnicode()` method converts a string representing a domain name containing [Punycode](#) encoded characters into Unicode. Only the [Punycode](#) encoded parts of the domain name are be converted.

```
// decode domain names
punycode.toUnicode('xn--maana-pta.com'); // 'mañana.com'
punycode.toUnicode('xn----dgo34k.com'); // 'ø-ø.com'
punycode.toUnicode('example.com'); // 'example.com'
```

punycode.ucs2

punycode.ucs2.decode(string)

- string {string}

The `punycode.ucs2.decode()` method returns an array containing the numeric codepoint values of each Unicode symbol in the string.

```
punycode.ucs2.decode('abc'); // [0x61, 0x62, 0x63]
// surrogate pair for U+1D306 tetragram for centre:
punycode.ucs2.decode('\uD834\uDF06'); // [0x1D306]
```

punycode.ucs2.encode(codePoints)

- codePoints {integer[]}

The `punycode.ucs2.encode()` method returns a string based on an array of numeric code point values.

```
punycode.ucs2.encode([0x61, 0x62, 0x63]); // 'abc'
punycode.ucs2.encode([0x1D306]); // '\uD834\uDF06'
```

punycode.version

- {string}

Returns a string identifying the current [Punycode.js](#) version number.