

Introducción a la Aceptación Universal (UA)

Grupo Directivo sobre Aceptación Universal (UASG)

23/09/2019



ÍNDICE

Acerca de este documento	4
Destinatarios	4
Conceptos de referencia	5
Nombres de dominio	5
Dominios de Alto Nivel con Código de País (ccTLD)	5
Dominios Genéricos de Alto Nivel (gTLD)	5
Internacionalización de nombres de dominio	6
La necesidad de la Aceptación Universal (UA)	6
Etiquetas U y etiquetas A	6
Internacionalización de Direcciones de Correo Electrónico (EAI)	7
Generación dinámica de vínculos (Vinculación)	8
La naturaleza dinámica del registro de la zona raíz	9
La Aceptación Universal en acción	9
Cinco criterios de Aceptación Universal	9
Escenarios de usuarios	12
No conformidad con las prácticas universales	13
Requisitos técnicos para la preparación de la UA	14
Requisitos de alto nivel	14
Consideraciones del desarrollador	15
Diseño de software para compatibilidad y flexibilidad	15
Prácticas recomendadas para desarrollar y actualizar el software para lograr la preparación para la Aceptación Universal	16
Fuentes autoritativas para nombres de dominio: Listas de la IANA y la zona raíz del DNS	23
Correo electrónico con IDN y por qué no es lo mismo que EAI	23
La vinculación y sus desafíos	24
Recomendaciones de buenas prácticas	24
Unicode: antecedentes y atributos de puntos de código	25
UTF8, UTF16 y otros métodos de codificación	26
IDNA – Breve reseña histórica y su estado actual	27
Casos de uso para pruebas	27
Actualización de software para EAI	27
Temas avanzados	28
Códigos de escritura complejos	28
Idiomas de derecha a izquierda y conformidad con Unicode	28
El algoritmo Bidi	28
La regla Bidi para nombres de dominio	30
Uniones	30
Homóglifos y caracteres similares	31



Normalización, conversión de mayúsculas y minúsculas y preparación de cadenas de caracteres	32
Asignación y conversión de mayúsculas y minúsculas	33
Glosario y otros recursos	34
Glosario	34
RFC y normas clave	38
Normas clave	41
Recursos en línea	42



Acerca de este documento

Las tecnologías de Internet, incluidos sus componentes de nombres, evolucionan y cambian constantemente. En los últimos años, la Corporación para la Asignación de Nombres y Números en Internet (ICANN) ha aprobado nuevos dominios de alto nivel (TLD), algunos con caracteres ASCII tradicionales y otros con caracteres no ASCII (Nombres de Dominio Internacionalizados). Algunos ejemplos incluyen .рус, .संगठन, .есо y .католик. Sin embargo, muchas aplicaciones y servicios no se han actualizado para gestionar esta gama ampliada de TLD. Además, los estándares de correo electrónico de Internet ahora permiten caracteres no ASCII en las direcciones de correo electrónico, por lo que hasta que no se actualice el software, no gestionará correctamente estos dominios y direcciones. Esto afecta la experiencia del usuario de varias maneras:

- No se reconocen o aceptan direcciones de correo electrónico válidas.
- Los nombres de dominio se tratan erróneamente como términos de búsqueda en la barra de direcciones del navegador.

A menos que el software reconozca y procese todos los nombres de dominio y direcciones de correo electrónico, un estado conocido como Aceptación Universal (UA), no será posible proporcionar una experiencia coherente y positiva para todos los usuarios de Internet. El presente documento proporciona una extensa introducción a la Aceptación Universal y los esfuerzos que se llevan a cabo para ayudar en el desarrollo de software listo para Aceptación Universal.

Destinatarios

Este documento tiene por objeto presentar la Aceptación Universal a un público técnico (desarrolladores, administradores y operadores) que puede estar familiarizado con algunos aspectos de la tecnología de Internet, pero no necesariamente con los detalles de cómo los nuevos IDN, nombres de dominio y direcciones de correo electrónico afectan a la forma en que deberían ser aceptados, validados, almacenados, procesados y visualizados. Representa un punto de partida para que las personas con diversas formaciones técnicas comiencen a explorar la Aceptación Universal.



Conceptos de referencia

Nombres de dominio

Un nombre de dominio es un identificador técnico amigable con el usuario para las computadoras y las redes en Internet. Se representa habitualmente como una secuencia de etiquetas de texto separadas por "puntos" (el punto o signo de puntuación de punto); por ejemplo, `www.ejemplo.tld`. Cada etiqueta representa un nivel en la jerarquía del Sistema de Nombres de Dominio (DNS).

En el nivel más alto o "raíz" de la jerarquía se encuentran etiquetas de dominio de alto nivel (TLD) como `com`, `jp`, y `বাংলা`, que aparecen al final de un nombre de dominio. Debido a que aparecen al final, los TLD en ocasiones se denominan "sufijos".

Siguiendo la jerarquía del DNS desde la raíz, la siguiente etiqueta identifica un subdominio del TLD, comúnmente denominado dominio de segundo nivel; la etiqueta que le sigue identifica un subdominio del dominio de segundo nivel, comúnmente denominado dominio de tercer nivel; y así sucesivamente, con cada una de las etiquetas separadas de su vecina por un punto. Por ejemplo, un nombre de dominio con tres niveles podría tener este aspecto:



Dominios de Alto Nivel con Código de País (ccTLD)

Algunos TLD se delegan a países o territorios específicos. Estos se denominan dominios de alto nivel con código de país (ccTLD). En el pasado, todos los ccTLD eran dos letras que coincidían con el código de dos letras ISO 3166 asignado al país o territorio por la Organización Internacional de Normalización. Desde 2010, también ha habido ccTLD internacionalizados que representan el nombre de un país o territorio en el propio código de escritura de ese país o territorio.

Dominios Genéricos de Alto Nivel (gTLD)

La mayoría de los TLD que no son ccTLD se denominan dominios genéricos de alto nivel (gTLD), que están abiertos a cualquiera o restringidos a los miembros de una comunidad definida. Estos incluyen los conocidos `.com`, `.net` y `.org`, así como otras adiciones más recientes.



A través del [Programa de Nuevos gTLD](#), una iniciativa coordinada por la ICANN, el Sistema de Nombres de Dominio (DNS) se ha expandido exponencialmente a través de la introducción de nuevos dominios genéricos de alto nivel. Estos nuevos gTLD pueden representar marcas, comunidades de interés, áreas geográficas (ciudades, regiones) y otros conceptos.

Internacionalización de nombres de dominio

Los nombres de dominio se limitaban originalmente a un subconjunto de caracteres ASCII (letras a-z, dígitos 0-9 y el guión "-"). Desde la registración .com más antigua, symbolics.com, en 1985, el número y las características de los nombres de dominio se han expandido para reflejar las necesidades del uso global cada vez mayor de Internet como recurso comunitario. En la actualidad, la mayoría de los usuarios de Internet no hablan inglés; sin embargo, el idioma dominante en Internet es el inglés. Para ayudar con la internacionalización de Internet, en 2003, el Grupo de Trabajo en Ingeniería de Internet (IETF) comenzó a publicar normas que proporcionaban directrices técnicas para el despliegue de Nombres de Dominio Internacionalizados (IDN) a través de un mecanismo de traducción para admitir representaciones de nombres de dominio que no son ASCII en cualquier código de escritura compatible con Unicode (*por ejemplo*, 普遍接受-测试.世界, ua-test.كاثوليك, etc.).

La Junta Directiva de la ICANN aprobó el proceso para introducir nuevos ccTLD de IDN en octubre de 2009, y los primeros ccTLD de IDN se agregaron a la zona raíz en mayo de 2010. En junio de 2011, la Junta Directiva aprobó y autorizó el lanzamiento del Programa de Nuevos gTLD, que incluía nuevos IDN TLD y ASCII. El primer lote de TLD de este programa se agregó a la zona raíz en 2013.

La necesidad de la Aceptación Universal (UA)

Una década después de que el IETF publicara sus directrices relacionadas con IDN, y a través del Programa de Nuevos TLD de la ICANN, ahora ya hay activos más de 1000 TLD nuevos. Sin embargo, algunos programas y aplicaciones siguen estando desactualizados y no pueden manejar estos nuevos TLD. Esto genera problemas a los usuarios de Internet, incluidos los que utilizan caracteres y códigos de escritura que no son ASCII.

La Aceptación Universal asegura que todos los nombres de dominio y direcciones de correo electrónico válidos son aceptados, validados, almacenados, procesados y mostrados correcta y uniformemente por todos los sistemas, dispositivos y aplicaciones habilitados para Internet. Por ejemplo, cada dirección web válida se resuelve para el recurso previsto en el sitio web correcto y cada dirección de correo electrónico válida da lugar a la entrega de correo al destinatario previsto.

El Grupo Directivo sobre Aceptación Universal (UASG) es una iniciativa de la comunidad de Internet, con el apoyo de la ICANN, que tiene encargada la tarea de llevar a cabo actividades que promuevan eficazmente la Aceptación Universal y ayuden a garantizar una experiencia coherente y positiva para los usuarios de Internet a nivel global.

Etiquetas U y etiquetas A

Los nombres de dominio que utilizan caracteres que no son ASCII se denominan Nombres de Dominio Internacionalizados (IDN). La parte internacionalizada de un nombre de dominio puede estar en cualquier etiqueta, no solo en el TLD.



Como el propio DNS antes solo utilizaba ASCII, era necesario crear una codificación adicional para permitir que los puntos de código Unicode que no eran ASCII se representaran como cadenas de caracteres ASCII. El algoritmo que implementa esta codificación de Unicode a ASCII se denomina Punycode; las cadenas de caracteres de salida se denominan Etiquetas A. Las Etiquetas A se pueden distinguir de una etiqueta ASCII ordinaria porque siempre comienzan con los siguientes cuatro caracteres:

xn--

Estos caracteres se denominan prefijo ACE.¹

La transformación de Punycode es reversible: puede transformarse de Unicode a una Etiqueta A y desde una Etiqueta A se puede volver a una cadena de caracteres de Unicode (conocida como Etiqueta U).

El único uso estándar del algoritmo Punycode es para expresar dominios internacionalizados. Si bien uno podría hipotéticamente codificar otras cadenas de caracteres UTF-8 utilizando Punycode, no sería algo estándar y no interoperaría con otros sistemas.

Ejemplos de IDN (imaginarios)

Versión de etiqueta U	Versión de etiqueta A
ejemplo.みんな	ejemplo.xn--q9jyb4c
大坂.info	xn--uesx7b.info
みんな.大坂	xn--q9jyb4c.xn--uesx7b

Internacionalización de Direcciones de Correo Electrónico (EAI)

Las direcciones de correo electrónico contienen dos partes:

- una parte local (antes del carácter “@”)
- una parte del dominio (después del carácter “@”)

Dado que tanto los códigos de escritura de izquierda a derecha (LTR) como los de derecha a izquierda (RTL) pueden utilizarse en direcciones de correo electrónico y nombres de dominio, los términos "antes" y "después" deben entenderse con respecto a la direccionalidad del código de escritura.

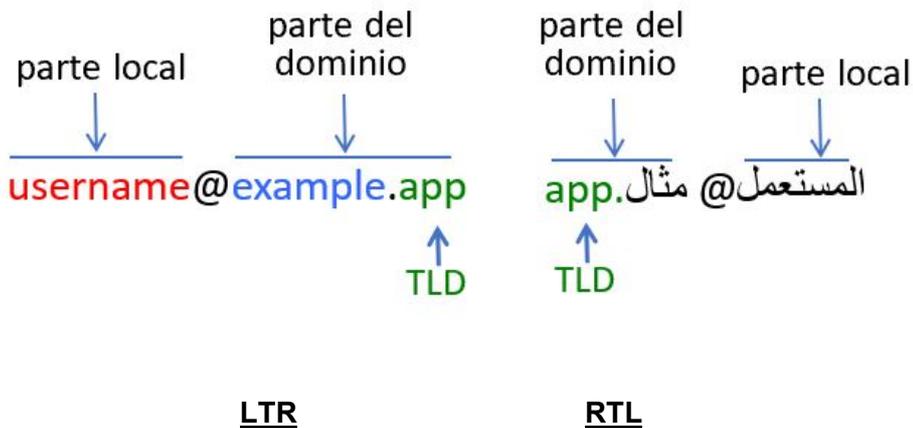
Ejemplos de direcciones de EAI (imaginarias)

¹ El prefijo de Codificación Compatible con ASCII (ACE), distingue las etiquetas codificadas de Punycode de las otras etiquetas ASCII.



usuario@ejemplo.みんな	Utiliza IDN TLD
usuario@大坂.info	Utiliza dominio de segundo nivel de IDN
用戶@ejemplo.abogado	Utiliza parte local de Unicode y nuevo gTLD

Ejemplo de texto de derecha a izquierda en una dirección de EAI



En una dirección de correo electrónico internacionalizada, la parte del dominio puede contener cualquier nombre de dominio, incluidos aquellos con nuevos TLD, y puede contener etiquetas U de Unicode. La parte local no es un nombre de dominio y en principio puede contener casi cualquier carácter Unicode, aunque en la práctica los sistemas de correo limitarán los caracteres utilizados en sus nombres de buzón.

El término Internacionalización de Direcciones de Correo Electrónico (EAI) se utiliza a menudo para describir el uso de direcciones internacionalizadas en el correo electrónico.

Generación dinámica de vínculos (Vinculación)

El software moderno, como las aplicaciones populares de procesadores de textos u hojas de cálculo, a veces permite a un usuario crear un hipervínculo con tan solo escribir una cadena de caracteres que se parece a una dirección web, dirección de correo electrónico o ruta de red. Por ejemplo, si se escribe "www.icann.org" en un mensaje de correo electrónico puede hacer que se cree automáticamente un vínculo a <http://www.icann.org> si la aplicación reconoce a "www." como un prefijo especial o a ".org" como un sufijo especial.

Cuando se utiliza, la vinculación debería funcionar de forma coherente para todas las direcciones web, direcciones de correo electrónico o rutas de red que se formen correctamente, y no solo para algunas. La vinculación precisa es difícil y depende del



contexto del texto (por ejemplo, en algunos idiomas "www" no indica una dirección web), por lo que no se aborda aquí más a fondo.

La naturaleza dinámica del registro de la zona raíz

El DNS es una gran base de datos distribuida que se divide en secciones llamadas zonas. La sección que contiene todos los TLD se llama zona raíz porque conceptualmente se encuentra en la raíz del árbol de nombres del DNS. Todas las zonas del DNS, incluida la zona raíz, se actualizan según sea necesario. A medida que se agregan nuevos TLD o se retiran los antiguos, sus nombres se agregan o se eliminan de la zona raíz.

Esto significa que cualquier lista fija de TLD, como una lista almacenada en una aplicación o en un archivo, quedará inevitablemente desactualizada. Para validar de forma confiable el TLD de un nombre de dominio, el software puede verificarlo en línea con una consulta al DNS o, si utiliza un archivo, actualizarlo de forma periódica. Ambos métodos se describen con más detalle más adelante.

La Aceptación Universal en acción

Cinco criterios de Aceptación Universal

Aceptación Universal es el estado en el cual todos los nombres de dominio y direcciones de correo electrónico válidos son aceptados, validados, almacenados, procesados y mostrados correcta y uniformemente por todos los sistemas, dispositivos y aplicaciones habilitados para Internet. A continuación, se describen los cinco criterios de la Aceptación Universal.

1. Aceptar²	<p><i>Aceptar es el proceso que ocurre cuando una dirección de correo electrónico o un nombre de dominio es recibido como una cadena de caracteres de una interfaz del usuario, un archivo o una API que utiliza una aplicación de software o un servicio en línea.</i></p> <p>Las aplicaciones y los servicios permiten que los nombres de dominio y las direcciones de correo electrónico:</p> <ul style="list-style-type: none">▪ se introduzcan a través de interfaces de usuario, o bien▪ se reciban de otras aplicaciones y servicios a través de las API.
-------------------------------	--

² La aceptación se trata de forma distinta a la validación en este documento. En la práctica, las acciones pueden superponerse.



2. Validar³	<p>La <i>Validación</i> puede ocurrir en muchos lugares siempre que una aplicación o servicio en línea reciba o emita una dirección de correo electrónico o un nombre de dominio como una cadena de caracteres.</p> <p>La validación está destinada a garantizar que la información especificada sea válida o al menos que concretamente no sea información no válida. La validación asegura que la información tenga la sintaxis correcta y puede hacer otras verificaciones. Para los nombres de dominio y las direcciones de correo electrónico, muchos programadores han confiado tradicionalmente en métodos de validación <i>ad hoc</i> como la verificación de que un TLD está dentro de los límites de longitud o que los caracteres son del conjunto de caracteres ASCII. Sin embargo, estos métodos se basan en supuestos que ya no son aplicables porque Internet está cambiando:</p> <ul style="list-style-type: none">▪ Los nombres de dominio y las direcciones de correo electrónico ahora pueden incluir caracteres Unicode no ASCII.▪ La lista de TLD está cambiando.▪ Cualquier etiqueta en un nombre de dominio, incluida la etiqueta de TLD, puede tener hasta 63 caracteres de longitud.⁴ <p>Sigue siendo posible validar los TLD utilizando otras técnicas, como se describe más adelante.</p>
3. Almacenar	<p>El <i>Almacenamiento</i> ocurre cuando una dirección de correo electrónico o un nombre de dominio se almacenan como una cadena de caracteres en una base de datos o archivo utilizado por una aplicación de software o servicio en línea y que luego es recuperado por la misma u otras aplicaciones de software.</p> <p>Las aplicaciones y los servicios pueden requerir almacenamiento a largo plazo o transitorio de nombres de dominio y direcciones de correo electrónico. Independientemente de la duración de los datos, deben almacenarse en:</p> <ul style="list-style-type: none">▪ Formatos definidos por una Solicitud de Comentarios (RFC) estándar de Internet, o (menos conveniente)▪ Formatos alternativos que se pueden traducir desde y hacia formatos definidos por RFC.

³ La aceptación y el procesamiento se tratan de forma distinta a la validación en este documento. En la práctica, las acciones pueden superponerse.

⁴ El límite de longitud de 63 caracteres se aplica a la propia etiqueta si es una etiqueta ASCII, o a la forma de etiqueta A de la etiqueta si es un IDN.



	<p>Aunque Unicode en los nombres y direcciones de correo electrónico del DNS se almacena como UTF-8, es posible que haya otros formatos en el código legado. Consulte la sección Prácticas recomendadas a continuación.</p>
4. Procesar⁵	<p><i>El Procesamiento</i> tiene lugar cuando una aplicación o un servicio utiliza una dirección de correo electrónico o un nombre de dominio para realizar una actividad, como realizar una búsqueda o clasificar una lista, o se transforma a un formato alternativo (como convertir etiquetas U en etiquetas A).</p> <p>Puede realizarse una validación adicional durante el procesamiento. Las formas en que se pueden procesar las direcciones de correo electrónico y los nombres de dominio están limitadas únicamente por la imaginación de los desarrolladores de aplicaciones, pero es importante no hacer suposiciones (por ejemplo, que el correo electrónico dirigido a <code>pākehā@tetaurawhiri.govt.nz</code> está destinado a una persona en Nueva Zelanda) que dependan de políticas ajenas al DNS.</p>
5. Visualizar	<p>La acción de <i>Visualizar</i> tiene lugar siempre que una dirección de correo electrónico o un nombre de dominio se presenten dentro de una interfaz del usuario.</p> <p>La visualización de nombres de dominio y direcciones de correo electrónico suele ser sencilla, aunque no siempre, cuando las cadenas de caracteres utilizadas en el nombre o la dirección son compatibles con el sistema operativo subyacente y cuando las cadenas de caracteres se almacenan en Unicode.⁶ If these conditions are not met, application-specific transformations may be required. Además, incluso si el sistema operativo subyacente admite las cadenas de caracteres, la visualización puede ser complicada si, por ejemplo, los códigos de escritura RTL y LTR son mixtos, o si la direccionalidad general del texto no es clara.</p>

⁵ El procesamiento se trata de forma distinta a la validación en este documento. En la práctica, las acciones pueden superponerse.

⁶ Es importante reconocer que la visualización no es sencilla, incluso cuando se cumplen estas condiciones para algunos códigos de escritura complejos.



Escenarios de usuarios

Los ejemplos y definiciones anteriores pueden dar la impresión de que la Aceptación universal solo se trata de sistemas informáticos y servicios en línea. La realidad, sin embargo, es que también se trata de las personas que utilizan esos sistemas y servicios.

A continuación, se incluyen algunos ejemplos de actividades que requieren la Aceptación Universal:

Registrar un nuevo TLD	<p>Una organización adopta un TLD de "marca" para ofrecer a sus clientes una experiencia de cliente diferenciada al proporcionar direcciones de correo electrónico en el formato: nombredelcliente @ejemplo.marca.</p> <p>La Aceptación Universal significa:</p> <ul style="list-style-type: none">▪ Los sitios web y las aplicaciones aceptan estas direcciones de correo electrónico "@ejemplo.marca" como lo harían con los antiguos TLD como .com, .net y .org.
Acceso a un gTLD	<p>Un usuario accede a un sitio web, cuyo nombre de dominio contiene un nuevo TLD, cuando escribe una dirección en un navegador o haciendo clic en un enlace en un documento.</p> <p>La Aceptación Universal significa:</p> <ul style="list-style-type: none">▪ Aunque el TLD sea nuevo, el navegador del usuario muestra la dirección web en su formato original y accede al sitio como lo prevé el usuario. El navegador no muestra nombres de dominio como etiquetas A al usuario a menos que beneficie al usuario de alguna manera.
Uso de una dirección de correo electrónico que contenga un nuevo gTLD como identidad en línea	<p>Un usuario adquiere una dirección de correo electrónico con la parte de dominio utilizando un nuevo gTLD y usa esta dirección de correo electrónico como su identidad para acceder a sus cuentas de fidelidad al banco y a la línea aérea.</p> <p>La Aceptación Universal significa:</p> <ul style="list-style-type: none">▪ Aunque el dominio utilizado en la dirección de correo electrónico es nuevo, el sitio del banco o de la aerolínea acepta la dirección de la misma forma en la cual acepta una dirección en un TLD antiguo, como .biz o .eu.



Acceso a un IDN	<p>Un usuario accede a una URL de IDN, si escribe el URL en un navegador o si hace clic en un enlace en un documento.</p> <p>La Aceptación Universal significa:</p> <ul style="list-style-type: none">▪ Incluso si el nombre de dominio contiene caracteres diferentes a la configuración de idioma en la computadora del usuario, cualquier navegador que el usuario desee utilizar mostrará la dirección web como se esperaba y accederá al sitio con éxito.
Uso de una dirección de correo electrónico internacionalizada para el correo electrónico	<p>Un usuario ha adquirido una nueva dirección de correo electrónico que incluye caracteres que no son ASCII en el nombre de dominio (<i>por ejemplo</i>, <code>Info@普遍接受-测试.世界</code>).</p> <p>La Aceptación Universal significa:</p> <ul style="list-style-type: none">▪ El usuario puede enviar hacia y recibir desde cualquier dirección de correo electrónico, con cualquier cliente de correo electrónico.
Uso de una dirección de correo electrónico internacionalizada como identidad en línea	<p>Un usuario adquiere una dirección de correo electrónico que no es ASCII y utiliza esta dirección de correo electrónico como su identidad para acceder a sus cuentas de fidelidad del banco y de la aerolínea.</p> <p>La Aceptación Universal significa:</p> <ul style="list-style-type: none">▪ El sitio del banco o de la aerolínea acepta la nueva identidad exactamente como si fuera cualquier otra identidad de correo electrónico.
Creación dinámica de un hipervínculo en una aplicación	<p>Un usuario escribe una dirección web en un documento o mensaje de correo electrónico.</p> <p>La Aceptación Universal significa:</p> <ul style="list-style-type: none">▪ Las reglas utilizadas por la aplicación para generar automáticamente un hipervínculo son las mismas si la dirección no es ASCII o contiene un nuevo TLD.
Desarrollo de una aplicación	<p>Un desarrollador escribe una aplicación que accede a los recursos web.</p> <p>La Aceptación Universal significa:</p> <ul style="list-style-type: none">▪ Las herramientas utilizadas por los desarrolladores incluyen bibliotecas que permiten la Aceptación Universal mediante la compatibilidad con nuevos TLD e IDN.

No conformidad con las prácticas universales

Se considera que las siguientes son prácticas deficientes:



✘	Mostrar etiquetas A al usuario sin el correspondiente beneficio para el usuario, por ejemplo, para mostrar la asignación entre una etiqueta U y una etiqueta A.
✘	Exigir que un usuario especifique etiquetas A al registrarse para obtener una nueva dirección de correo electrónico o requerir que un usuario ingrese etiquetas A al registrarse para un nuevo dominio alojado.
✘	Validar la sintaxis del nombre de dominio o la dirección de correo electrónico utilizando criterios desactualizados o recursos de nombres de dominio en línea no autorizados.
✘	Utilizar una lista desactualizada de TLD aunque se agreguen y se eliminen regularmente nuevos TLD.
✘	Exponer a los usuarios el uso interno de etiquetas A. Por ejemplo, la conversión de dominios en direcciones de EAI a etiquetas A al responder a un usuario de EAI.
✘	Tratar algunos nombres de dominio como términos de búsqueda en lugar de como nombres de dominio porque la aplicación no los reconoce como tales.
✘	Configurar bloqueadores de spam para que bloqueen automáticamente TLD enteros (nuevos) sin evidencia de abuso.

Requisitos técnicos para la preparación de la UA

Para que una aplicación o sitio web esté preparado para la UA, debe cumplir una variedad de requisitos.

Requisitos de alto nivel

Una aplicación o servicio que admite la Aceptación Universal (UA):

1. Admite todos los nombres de dominio independientemente de la longitud o el conjunto de caracteres.

Véase [RFC 5892](#).

2. Permite todos los conjuntos de caracteres que son válidos para nombres de dominio y direcciones de correo electrónico.

Acepta puntos de código Unicode así como ASCII.

3. Puede representar correctamente todos los puntos de código en cadenas de caracteres Unicode.

Véase [RFC 3490](#). Tenga en cuenta que Unicode agrega con frecuencia nuevos puntos de código, por lo que se trata de un objetivo en movimiento.

4. Puede reproducir correctamente las cadenas de derecha a izquierda (RTL), como las de árabe y hebreo.



Para obtener información acerca de códigos de escritura RTL, consulte [RFC 5893](#).

5. Puede comunicar datos entre aplicaciones y servicios en UTF-8 y, cuando sea necesario, otras codificaciones que pueden convertirse a y desde UTF-8.

Para obtener información acerca de UTF-8, consulte [RFC 3629](#).

6. Ofrece API públicas y privadas compatibles con UTF-8.

Las API privadas se aplican solo a las llamadas entre servicios del mismo proveedor.

7. Trata a las direcciones de EAI correctamente.

En particular, no convierte IDN en direcciones a etiquetas A.

8. Puede enviar y recibir correos electrónicos de los destinatarios, independientemente del nombre de dominio o conjunto de caracteres.

Véase [RFC 6530](#).

9. Almacena datos de usuario en formatos que admiten Unicode y es convertible a y desde UTF-8.

Dichas conversiones solo serían visibles para el operador del producto o servicio.

10. Admite todos los nombres de dominio de alto nivel de la lista autoritativa de TLD de la ICANN, independientemente de la longitud o el conjunto de caracteres.

Consulte la lista autoritativa en <https://data.iana.org/TLD/>.

Consideraciones del desarrollador

Dado que muchos sistemas de software existentes contienen suposiciones codificadas sobre dominios y direcciones de correo electrónico, es posible que se requieran cambios de código para reconocer a los IDN, los nuevos TLD y las direcciones de correo electrónico de EAI. Esta sección explica cómo los desarrolladores pueden incorporar cambios de código que permitirán la Aceptación Universal.

Diseño de software para compatibilidad y flexibilidad

El Principio de Robustez, tal y como fue articulado por Jon Postel en el documento [RFC 793](#), es una guía general de diseño para el desarrollo de software:

"Ser conservador en lo que se hace, ser liberal en lo que se acepta de los demás".

Es decir, sea conservador en lo que envía: en cualquier área en la que una especificación pueda ser ambigua o poco clara, evite cualquier cosa que pueda sorprender a otros. Por otro lado, al recibir, acepte cualquier cosa que sea plausiblemente válida. Esto *no* significa cambiar el código para evitar errores claros en otras implementaciones, dado que esto conduce a un lío indocumentado e "indepurable".



Prácticas recomendadas para desarrollar y actualizar el software para lograr la preparación para la Aceptación Universal

Aceptar

✓	<p>Muestre los nombres como Unicode siempre que sea posible.</p> <p>Se debería permitir, pero no exigir, que los usuarios introduzcan nombres de dominio como etiquetas A en lugar de etiquetas U. Sin embargo, se debería mostrar las etiquetas U de manera predeterminada y que las etiquetas A se muestren al usuario solo cuando aporte un beneficio.</p>
!	<p>No genere direcciones de correo electrónico de EAI con etiquetas A, pero sí tenga la capacidad de manejarlas si las presenta el software de otra persona.</p>
✓	<p>Todo elemento de interfaz del usuario que requiere que el usuario escriba un nombre de dominio o dirección de correo electrónico debe aceptar nombres largos. Los nombres de dominio ASCII pueden tener hasta 63 caracteres en cada etiqueta y pueden tener un total de hasta 253 bytes. Las etiquetas UTF-8 pueden ser mucho más largas que las etiquetas ASCII y la longitud total puede ser de hasta 670 bytes. Recuerde que el código UTF-8 para la mayoría de los puntos de código Unicode requiere más de un byte.</p> <ul style="list-style-type: none">▪ Véase RFC 1035.

Validar

✓	<p>Valide únicamente cuando sea apropiado.</p> <p>Validar únicamente si es necesario para el funcionamiento de la aplicación o del servicio. Ésta es la forma más confiable de garantizar que todos los nombres de dominio válidos sean aceptados en sus sistemas.</p>
✓	<p>Reconozca que las entradas sintácticamente correctas pueden representar nombres de dominio o direcciones de correo electrónico que están actualmente en uso en Internet. Eso puede o no ser válido en función de la aplicación.</p>



Cuando valide, considere lo siguiente:

- Compruebe la parte del TLD de un nombre de dominio con una tabla autoritativa. La IANA publica la lista de dominios de alto nivel en:
 - * <https://data.iana.org/TLD/tlds-alpha-by-domain.txt>
 - * Véase también: <https://www.icann.org/en/system/files/files/sac-070-en.pdf>
- Consulte el nombre de dominio con el DNS.
 - * La API getdns (<http://getdnsapi.net/>) es una forma muy portátil de consultar el DNS.
 - * La mayoría de los sistemas operativos también tienen una API de consulta del DNS nativa.
- Solicite la entrada repetida de una dirección de correo electrónico para detectar errores tipográficos.
- Valide los caracteres de las etiquetas verificando que cada etiqueta siga las reglas de Internacionalización de Nombres de Dominio en Aplicaciones (IDNA 2008).
 - * Véase [RFC 5892](#)
- Limite la validación de etiquetas en sí a un pequeño número de reglas de etiquetas completas definidas en las RFC.
 - * Véase [RFC 5894](#)
- Asegúrese de que el producto o la función manejen los números de forma correcta.
 - * Por ejemplo: Los caracteres de dígitos arábigos-índicos deberían tratarse como números en los campos de entrada numéricos, así como los dígitos ASCII.
 - * Tenga en cuenta que los dígitos arábigos-índicos son válidos en las etiquetas U, pero no se consideran equivalentes a los dígitos ASCII en ese contexto.

Almacenar



Las aplicaciones y los servicios deberían ser compatibles con la mayoría de los estándares Unicode actuales.



La información debería almacenarse en formato UTF-8 siempre que sea posible. Algunos sistemas pueden requerir compatibilidad legada con UTF-16 también pero, por lo general, se recomienda UTF-8. UTF-7 es obsoleto y UTF-32 es demasiado voluminoso para el almacenamiento de archivos.



	<p>Las cadenas de caracteres deberían normalizarse cuando sea adecuado (alguna normalización puede, en algún contexto, provocar la pérdida de información).</p>
!	<p>Considere todos los escenarios completos antes realizar una conversión entre Etiquetas A y Etiquetas U al momento de almacenar datos.</p> <p>En aplicaciones nuevas, es mejor mantener solo Etiquetas U en un archivo o base de datos, porque simplifica la búsqueda, el almacenamiento y la presentación. Sin embargo, la conversión puede tener implicancias al interoperar con aplicaciones y servicios más antiguos no habilitados con Unicode.</p>
✓	<p>Etiquete las direcciones de correo electrónico y los nombres de dominio como tales en el almacenamiento para facilitar el acceso.</p> <p>El archivado de direcciones de correo electrónico y nombres de dominio en el campo “autor” de un documento o “información de contacto” en un archivo de registro ha generado la pérdida de la dirección original.</p>
✓	<p>Independientemente de la forma en que se almacenen las direcciones y los nombres de dominio, debe poder hacer coincidir las cadenas de caracteres en varios formatos.</p> <p>Por ejemplo, una búsqueda de ejemplo.みんな también debería encontrar ejemplo.xn--q9jyb4c.</p>

Procesar

✓	<p>Asegúrese de que todas las respuestas del servidor web y del correo MIME tengan UTF-8 especificado en el tipo de contenido.</p>
✓	<p>Especifique UTF-8 en el encabezado http del servidor web.</p> <ul style="list-style-type: none">▪ Es importante asegurarse de que la codificación se especifique en cada respuesta.
!	<p>Considere el contexto antes de convertir Etiquetas A a Etiquetas U y viceversa durante el procesamiento.</p> <p>Es conveniente mantener solo Etiquetas U en un archivo o base de datos, porque simplifica la búsqueda y la clasificación. Sin embargo, la conversión puede tener implicancias al interoperar con aplicaciones y servicios más antiguos no habilitados con Unicode.</p>
✓	<p>Asegúrese de que el producto o la función maneje el orden de clasificación, las búsquedas y la clasificación de acuerdo con las especificaciones de idioma y configuración regional, y que aborde la búsqueda y clasificación en varios idiomas.</p>



✗	<p>No utilice la codificación con porcentajes para nombres de dominio:</p> <ul style="list-style-type: none">▪ ejemplo.みんな es correcto▪ ejemplo.%E3%81%BF%E3%82%93%E3%81%AA no es correcto.
✓	<p>Debido a que el estándar Unicode se expande de manera continua, los puntos de código no definidos al momento en que se creó el servicio o la aplicación deberían ser controlados a fin de garantizar que no generen un resultado erróneo o confuso.</p> <p>Tipos de letras faltantes en el sistema operativo subyacente pueden ocasionar caracteres que no pueden mostrarse (con frecuencia, se utiliza un cuadro pequeño para representar los mismos), pero esta situación no debería generar un fallo irrecuperable ni un mensaje de error.</p>
✓	<p>Utilice API habilitadas con Unicode compatibles.</p>
✓	<p>Utilice la última versión del Protocolo de Nombres de Dominio Internacionalizados en Aplicaciones (IDNA 2008) y los documentos de tablas para IDN:</p> <ul style="list-style-type: none">▪ RFC 5891▪ RFC 5892
✓	<p>Realice el procesamiento de texto en formato UTF-8 siempre que sea posible.</p>
✓	<p>Coordine las actualizaciones de las aplicaciones y los servicios de los que dependen.</p> <p>Si el servidor es Unicode y el cliente no es Unicode, o viceversa, los datos tendrán que convertirse en cada transacción, lo cual es propenso a errores y puede ser lento.</p>
✓	<p>Al realizar la transformación de caracteres, las cadenas de caracteres de texto pueden crecer o reducirse considerablemente. Cada punto de código UTF-8 puede ser de 1 a 4 bytes y, en algunos casos, un solo carácter en otra codificación puede corresponder a varios puntos de código UTF-8 o viceversa.</p>

Visualizar

✓	<p>Visualice todos los puntos de código Unicode compatibles con el sistema operativo subyacente.</p> <p>Todos los sistemas operativos modernos son compatibles con Unicode, pero sus motores de representación no siempre son correctos para todos los códigos de escritura e idiomas. Proporcione representación de caracteres en aplicaciones solo cuando la representación correcta no esté disponible en los sistemas operativos de destino.</p>
✓	<p>Al desarrollar una aplicación o un servicio, considere los lenguajes compatibles y asegúrese de que los sistemas operativos y las aplicaciones incluyan dichos lenguajes.</p>



✓	<p>Convierta las etiquetas A en etiquetas U antes de la visualización.</p> <p>Por ejemplo, el usuario final debería ver “ejemplo.みんな” en lugar de “ejemplo.xn--q9jyb4c”. (Esta conversión es un ejemplo de procesamiento preparado para UA).</p>
✓	<p>Muestre los nombres de dominio como etiquetas U de forma predeterminada.</p> <p>Visualice etiquetas A para el usuario solo cuando aporte un beneficio.</p>
!	<p>Tenga en cuenta que los nombres de dominio con combinación de códigos de escritura son posibles.</p> <ul style="list-style-type: none">▪ Algunos caracteres Unicode pueden parecer iguales para el ojo humano, pero diferentes para las computadoras; por ejemplo, el latín O, el cirílico O y el griego ómicron O.▪ Las cadenas de caracteres con combinación de códigos de escritura son comunes en cadenas de caracteres estrechamente relacionados (por ejemplo, Kanji, Katakana, Hiragana y Romaji del japonés). De lo contrario, las cadenas de caracteres con combinación de códigos de escritura pueden estar destinados a fines maliciosos, como phishing. Consultar la Norma Técnica 39 de Unicode, Mecanismos de seguridad de Unicode,⁷ para verificar que los códigos de escritura en una secuencia de Unicode sigan las prácticas recomendadas.▪ Si la interfaz del usuario hace que las cadenas de caracteres llamen la atención del usuario, asegúrese que lo haga de manera que no resulte perjudicial a los usuarios de códigos de escritura no latinos. <p>Obtenga más información sobre las Consideraciones de seguridad de Unicode en: http://unicode.org/reports/tr36.</p>
✓	<p>Tenga presente los caracteres no asignados y no permitidos para nombres de dominio.</p> <ul style="list-style-type: none">▪ Véase RFC 5892

Unicode

✓	<p>Utilice API habilitadas con Unicode compatibles.</p>
✗	<p>Utilice API estándar bien depuradas para:</p> <ul style="list-style-type: none">▪ conversiones de formato de cadenas de caracteres▪ determinar qué código de escritura comprende una cadena de caracteres▪ determinar si una cadena de caracteres contiene una combinación de códigos de escritura▪ normalización/descomposición Unicode

⁷ Véase https://www.unicode.org/reports/tr39/#Restriction_Level_Detection



✗	<p>No utilice UTF-7 y restrinja el uso de UTF-32.</p> <ul style="list-style-type: none">▪ La codificación UTF-7 está obsoleta.▪ UTF-32 utiliza cuatro bytes por cada punto de código. Dado que cada punto de código ocupa la misma cantidad de espacio y puede indexarse directamente en matrices, es conveniente utilizarla dentro del código del programa, pero puede ser demasiado voluminosa para almacenarla en archivos y bases de datos.
✗	<p>No utilice UTF-16 excepto cuando se requiera explícitamente (como en ciertas API de Windows y aplicaciones de Javascript).</p> <p>En UTF-16, 16 bits solo pueden representar caracteres de 0x0 a 0xFFFF. Los valores por encima de este rango (0x10000 a 0x10FFFF) utilizan pares de pseudocaracteres conocidos como sustitutos. Si el manejo de pares sustitutos no se prueba exhaustivamente, puede generar errores complicados y posibles brechas de seguridad.</p>
✓	<p>Utilice UTF-8 en las cookies para que las aplicaciones puedan leerlas correctamente.</p>
✓	<p>Utilice los documentos de tablas y el Protocolo 2008 de IDNA:</p> <ul style="list-style-type: none">▪ RFC 5891▪ RFC 5892
✗	<p>No utilice IDNA 2003, que ha sido reemplazado por IDNA 2008.</p>
!	<p>Mantenga las tablas de IDNA y Unicode que sean compatibles en lo que respecta a las versiones.</p> <p>Por ejemplo, a menos que la aplicación realmente ejecute las reglas de clasificación en el documento de Tablas para interpretar puntos de código según se introduzcan (RFC 5892), sus tablas de IDNA deben derivarse de la versión de Unicode que se admita en el sistema. Las tablas no necesitan reflejar la última versión de Unicode, pero deben ser compatibles.</p>
✓	<p>Valide las etiquetas utilizando las reglas de etiquetas completas de IDNA 2008.</p> <ul style="list-style-type: none">▪ En algunos contextos, una validación adicional puede ser apropiada; por ejemplo, si la aplicación sabe qué códigos de escritura están permitidos en los nombres de dominio que utiliza.

General

✓	<p>Utilice recursos autorizados para validar nombres de dominio. No haga suposiciones <i>ad hoc</i> obsoletas, como "todos los TLD tienen 6 caracteres o menos".</p>
---	--



✓	<p>Asegúrese de que el producto o la función manejen los números de forma correcta.</p> <p>Por ejemplo, los numerales de ASCII y las representaciones numéricas ideográficas asiáticas deberían tratarse todos como números en contextos numéricos.</p>
!	<p>Busque direcciones de correo que puedan ser direcciones de EAI en lugares inesperados:</p> <ul style="list-style-type: none">▪ Artista/autor/fotógrafo/metadatos de derechos de autor.▪ Metadatos de fuente.▪ Registros de contacto del DNS.▪ Información de versión binaria.▪ Información de apoyo.▪ Información de contacto de OEM.▪ Registración, comentarios y otros formularios.
!	<p>Restrinja los puntos de código permitidos al generar nuevos nombres de dominio y direcciones de correo electrónico:</p> <p>Todos los productos que utilizan direcciones de correo electrónico deben aceptar direcciones de correo electrónico internacionalizadas y permitir la mayoría de los caracteres imprimibles de UTF-8 en la parte local. Sin embargo, una aplicación o servicio no necesita permitir todos estos caracteres cuando un usuario crea un nuevo IDN o dirección de EAI.</p> <p>Evitar que se creen ciertos IDN o direcciones de correo electrónico en primer lugar puede mitigar algunas posibles preocupaciones de seguridad y accesibilidad. (NOTA: la práctica recomendada aún requeriría que el software acepte dichas cadenas de caracteres si se presentan).</p>
!	<p>Tenga en cuenta que la Aceptación Universal no siempre se puede medir a través de casos de prueba automatizados solamente.</p> <p>Por ejemplo, no siempre es posible probar cómo una aplicación o protocolo gestiona los recursos de la red y, en ocasiones, es mejor verificar el cumplimiento a través de la revisión de la especificación funcional y del diseño.</p>
!	<p>No suponga que porque un componente no invoca directamente a las API de resolución de nombres, ni utiliza directamente direcciones de correo electrónico, no lo afectan.</p> <p>Comprenda cómo el componente obtiene los nombres de dominio. No siempre es a través de la interacción del usuario. Los siguientes son algunos ejemplos sobre cómo el componente puede obtener un nombre de dominio:</p> <ul style="list-style-type: none">▪ Política de grupo.▪ Consulta LDAP.▪ Archivos de configuración.▪ Registro de Windows.▪ Transferido a o desde otro componente o función.
✓	<p>Realice revisiones de código para evitar ataques de desbordamiento de búfer.</p> <ul style="list-style-type: none">▪ En Unicode, las cadenas de caracteres pueden expandirse o reducirse cuando hay normalización o conversión de mayúsculas y minúsculas.



- Al realizar la conversión de caracteres, el texto puede crecer o reducirse considerablemente.

Otros desafíos

Mecanismo para detectar y convertir conjuntos de caracteres

Algunas aplicaciones de correo electrónico más antiguas utilizaban codificaciones de caracteres y no tenían una forma de detectar y convertir texto hacia y desde UTF-8 según fuera necesario. Esto ocurría especialmente en los encabezados de correo electrónico (PARA, CC, CCO, Asunto).

Gestionar múltiples direcciones de correo electrónico en una sola identidad de usuario

Cuando un usuario tiene varias direcciones de correo electrónico, puede resultar complicado gestionar estas direcciones como una única identidad de usuario.

Los programas de correo electrónico pueden dirigir el tráfico que va dirigido hacia dichos seudónimos al mismo buzón, pero las aplicaciones aún tratan a las direcciones como identidades diferentes.

Fuentes autoritativas para nombres de dominio: Listas de la IANA y la zona raíz del DNS

Existen dos fuentes para la lista autoritativa de TLD. La primera fuente es la zona raíz del DNS en sí. Está firmada con Extensiones de Seguridad del Sistema de Nombres de Dominio (DNSSEC), por lo tanto, su contenido se puede autenticar mediante un servidor de nombres compatible con las DNSSEC, aunque su contenido es bastante difícil de analizar como archivo de texto. Otra fuente es el archivo de texto de los TLD que publica la IANA (un TLD por línea en orden alfabético). Estos archivos se encuentran en servidores web https, por lo que resulta una práctica recomendada verificar que el certificado de Seguridad de la Capa de Transporte (TLS) del sitio sea válido cuando se realice la descarga para asegurarse de obtener el archivo correcto.

Puede obtener la lista de TLD desde cualquiera de los siguientes enlaces:

- <https://www.internic.net/domain/root.zone> (archivo de la zona raíz)
- <https://data.iana.org/TLD/tlds-alpha-by-domain.txt> (archivo de texto del TLD)

Correo electrónico con IDN y por qué no es lo mismo que EAI

El correo de Internacionalización de Direcciones de Correo Electrónico (EAI) prefiere los nombres de dominio con UTF-8; se desaconseja el uso de etiquetas A con código ASCII. Algunos sistemas de correo han establecido disposiciones parciales para direcciones de correo electrónico que incluyen IDN en lugar de proporcionar compatibilidad completa de EAI. Debido a que los IDN se pueden representar como etiquetas A de ASCII, algunos programas existentes permiten representar los IDN de una dirección de correo electrónico en ASCII o Unicode. Por ejemplo, algunos programas tratarán estas dos direcciones de IDN de manera equivalente para todos los fines (envío, recepción y búsqueda):



usuario@ejemplo.みんな = usuario@ejemplo.xn--q9jyb4c

Sin embargo, algunos tipos de software no tratarán a estas direcciones como equivalentes, incluso si ambas son válidas, porque no convierten una etiqueta A ("xn--q9jyb4c") en su equivalente Etiqueta U ("みんな") antes de comparar. Esto puede generar una experiencia de usuario impredecible. La experiencia del usuario puede resultar especialmente confusa si algún software convierte las etiquetas U en etiquetas A para que sean "compatibles". A medida que los mensajes se respondan o reenvíen, pueden aumentar las direcciones que son visiblemente diferentes para un usuario o que no se pueden buscar y ordenar como se espera.

Como en el ejemplo a continuación, algunos programas pueden intentar convertir la parte local de la dirección de correo electrónico usando Punycode, el algoritmo que se utiliza para convertir etiquetas A en etiquetas U (y viceversa). Este tipo de conversión no es válida y creará direcciones no válidas e imposibles de entregar.

Nunca intente convertir la parte local de una dirección de correo electrónico en una forma diferente.

- ✓ 用戶@ejemplo.みんな
- ✗ xn--youq53b@ejemplo.xn--q9jyb4c

El software y los servicios robustos preparados para UA deberían ser capaces de manejar y tratar todos estos formatos correctamente y deberían ser capaces de manejar tanto las partes locales de UTF-8 como las etiquetas U de UTF-8 en las direcciones, y al mismo tiempo también aceptar las etiquetas A en las direcciones para la compatibilidad con versiones previas.

La vinculación y sus desafíos

El software moderno a veces permite a un usuario crear automáticamente un hipervínculo con tan solo escribir una cadena de caracteres que se parece a una dirección web, nombre de correo electrónico o ruta de red. Por ejemplo, si se escribe "www.icann.org" en un mensaje de correo electrónico puede hacer que se cree automáticamente un vínculo a <http://www.icann.org> si la aplicación reconoce a "www." como un una etiqueta inicial o a ".org" como un TLD.

La vinculación es la acción mediante la cual una aplicación acepta una cadena de caracteres y determina de forma dinámica si debe crear un hipervínculo a una ubicación de Internet (http:// o https://) o una dirección de correo electrónico (mailto:). La vinculación, si se produce, debería funcionar de forma coherente para todas las direcciones web, nombres de correo electrónico y rutas de red que se formen correctamente.

La vinculación utiliza algoritmos y reglas creadas por los desarrolladores de software para determinar si una cadena se debe interpretar como un enlace, o no. En relación a esto es la forma en que la gente puede identificar una cadena de caracteres como un nombre de dominio. Si bien los navegadores, clientes de correo electrónico y procesadores de texto son lugares obvios, existen muchas más aplicaciones que toman estas decisiones.

Recomendaciones de buenas prácticas



1. Intente vincular en base a prefijos de protocolos explícitos (*por ejemplo*, “https://”, ftp://”, “mailto:”) pero complete la acción únicamente si el resto de la cadena de caracteres está formada correctamente.

Cadena de caracteres de ejemplo	Comportamiento esperado / Resultado
ejemplo.com	Sin hipervínculo porque el protocolo está ausente y no se infiere.
http://ejemplo.com	Crea hipervínculo porque el protocolo es explícito.
http:ejemplo.com	Sin hipervínculo debido a sintaxis incorrecta (falta //).
http://ejemplo.a	Ninguna vinculación porque “a” no es un TLD.
http://ejemplo.ab	Sin hipervínculo debido a sintaxis incorrecta (puntos consecutivos).
http://普遍接受-测试.世界	Crea hipervínculo porque el protocolo es explícito.

2. Intente generar hipervínculos en base a prefijos de protocolos implícitos (*por ejemplo* “www” infiere “http://www”).

Cadena de caracteres de ejemplo	Comportamiento esperado / Resultado
www.ejemplo.com	Crea hipervínculo porque el protocolo está implícito ⁸ .
label@example.com	Crea mailto: label@example.com porque el protocolo está implícito.

3. El HTML que rodea los URL que contienen texto bidireccional puede incluir códigos que afectan la dirección en la que se muestra el texto. La versión vinculada debe mantener la misma dirección de visualización.
4. Si los TLD se utilizan como un 'identificador especial' para determinar la capacidad de generar el vínculo, entonces deben incluirse todos los TLD. Una lista de TLD debería actualizarse con frecuencia.

Unicode: antecedentes y atributos de puntos de código

⁸ Nota: el sitio web real podría ser solo de https y requerir https:// en lugar de http://. Si este es el caso, entonces el hipervínculo puede no resolverse.



El estándar Unicode ha ido evolucionando desde que se publicó por primera vez como Unicode 1.0 en 1991. Cada versión desde entonces ha agregado más caracteres y puntos de código para manejar más idiomas y códigos de escritura. La versión actual es la 12.1.

En Unicode, cada punto de código tiene un conjunto de propiedades, como `Uppercase_Letter`, `Decimal_Number` o `Nonspacing_Mark`. Muchos caracteres tienen una propiedad de código de escritura como el latín, el han (chino) o el árabe, mientras que otros, como la puntuación, no la tienen.

Como se describe a continuación, IDNA utiliza atributos de punto de código para determinar qué caracteres están permitidos en los IDN. [UAX#44](#), Base de datos de caracteres de Unicode, describe la base de datos de los atributos de puntos de código.

UTF8, UTF16 y otros métodos de codificación

Un punto de código de Unicode puede tener un valor numérico que va de cero a 0x10FFFF. Dado que un único byte solo puede contener valores de 0 a 0xFF, se necesita algún tipo de codificación multibyte para almacenar puntos de código de Unicode.

La versión original de Unicode tenía puntos de código de menos de 64K (0xFFFF), por lo que cada punto de código podía caber en un entero de 16 bits. Esto dio lugar a una codificación de dos bytes conocida como UCS o UCS-2. Cuando Unicode se expandió más allá de los puntos de código de 64K, UCS se extendió a UTF-16⁹, que utiliza pares de puntos de código de 16 bits que de otro modo no serían válidos, conocidos como *sustitutos*, para representar valores superiores a 64K. Si bien esto funciona, ha generado problemas de depuración dado que los sustitutos agregan complejidad a cualquier código que cuente el número de puntos de código en una cadena de caracteres, o que clasifique las cadenas de caracteres en orden de puntos de código. Un problema adicional es que algunos ordenadores como los fabricados por IBM almacenan primero el byte alto de un valor de 16 bits ("big-endian"), y otros como los fabricados por Intel almacenan primero el byte bajo ("little-endian"). Como resultado, UTF-16 tiene dos variantes de almacenamiento: UTF-16BE y UTF-16LE. Existen técnicas para detectar y corregir problemas de endian, pero pueden dar lugar a errores. En este punto, UTF-16 se utiliza principalmente en aplicaciones existentes con API de Microsoft Windows y en los lenguajes Java y Javascript.

Una codificación alternativa es UTF-8, que codifica cada punto de código como una cadena de longitud variable de uno a cuatro bytes. UTF-8 tiene varias ventajas sobre UTF-16, entre ellas, que el subconjunto ASCII de Unicode está codificado como un solo byte, por lo que cualquier cadena ASCII es automáticamente una cadena UTF-8. UTF-8 suele ser más compacto que su equivalente en UTF-16, y es más fácil de clasificar porque las cadenas UTF-8 clasificadas en orden de bytes están automáticamente en orden de puntos de código. IDNA y EAI requieren codificación UTF-8.

UTF-32 es un formato simple que almacena cada punto de código en un entero de 32 bits. Es conveniente para el procesamiento interno en programas dado que los puntos de código en una matriz de UTF-32 se pueden indexar directamente, pero rara vez se utiliza en el almacenamiento debido a su volumen.

⁹ Consulte la sección 3.10 del estándar Unicode para obtener los detalles técnicos de UTF-8, UTF-16 y UTF-32, en <https://www.unicode.org/versions/Unicode12.0.0/ch03.pdf>.



IDNA – Breve reseña histórica y su estado actual

El IETF definió por primera vez los Nombres de Dominio Internacionalizados en Aplicaciones (IDNA) en 2003 como lo que ahora se conoce como IDNA2003¹⁰. Incluía un algoritmo para asignar los puntos de código Unicode en una forma estándar en las etiquetas de nombre de dominio conocido como Nameprep, y un algoritmo para codificar las etiquetas de puntos de código de Unicode en ASCII conocido como Punycode. Nameprep incluye transformaciones como la asignación de mayúsculas a minúsculas.

Después de adquirir cierta experiencia con IDNA, el IETF desarrolló y publicó una especificación revisada conocida como IDNA2008 en 2010¹¹. IDNA2008 creó los términos etiqueta U y etiqueta A y eliminó el paso Nameprep, y aconsejaba que las aplicaciones deberían hacer una asignación apropiada para el entorno de aplicación y configuración regional. IDNA2008 se actualizó para Unicode 6.0 mediante el documento RFC 6452 en 2011 y sigue bajo revisión por parte del IETF.

En la práctica, demasiadas implementaciones siguen utilizando IDNA2003. Algunas bibliotecas utilizan tablas (como las que se incluyen en IDNA2003) creadas para IDNA2008. No existen asignaciones de configuración regional para IDNA2008 excepto las reglas estándar de normalización y conversión de mayúsculas y minúsculas incluidas en el Estándar Unicode.

Una excepción es que hay algunas asignaciones de UTS#46, [Procesamiento de Compatibilidad de IDNA de Unicode](#). Esto especifica si deben aceptarse o asignarse algunos caracteres comunes que se asignan en IDNA2003 pero que se permiten como caracteres en IDNA2008. Es importante que las aplicaciones traten estos caracteres de acuerdo con IDNA2008 y no con IDNA2003, y que si se utiliza UTS#46, se utilice de forma compatible con IDNA2008.

Casos de uso para pruebas

El software que está destinado a manejar IDN y direcciones de correo de EAI debería ser puesto a prueba con una amplia gama de nombres de dominio y direcciones. Consulte [UASG 004](#), *Casos de uso para la Evaluación de la preparación para la UA*, para obtener un conjunto de casos de prueba.

Actualización de software para EAI

El cumplimiento con EAI requiere actualizaciones de servidores de correo, software de envío y entrega, agentes de usuarios de correo y correo web, y cualquier aplicación que maneje direcciones de correo electrónico y envíe correo.

¹⁰ La definición está en los documentos RFC [3490](#), [3491](#) y [3492](#).

¹¹ La definición está en los documentos RFC [5890](#), [5891](#), [5892](#), [5893](#), [5894](#) y [5895](#).



Para obtener una descripción general de la EAI, sus cuestiones y cómo implementarla, consulte [UASG 012](#), *Internacionalización de Direcciones de Correo Electrónico (EAI): Una descripción técnica*.

Temas avanzados

Códigos de escritura complejos

Los detalles de los códigos de escritura complejos pueden ser de interés limitado para aquellos que no sean desarrolladores y que creen sus propias bibliotecas de visualización o análisis de cadenas de caracteres. Sin embargo, se incluye un resumen aquí para garantizar que todos los lectores tengan suficiente conocimiento para reconocer errores de código relacionados con estos códigos de escritura cuando se encuentren en las experiencias del usuario.

Para el texto HTML con formato en páginas web y correo electrónico, los estándares HTML tienen características elaboradas para manejar y mostrar texto complejo y bidireccional, que los desarrolladores deberían entender y utilizar para representar el texto. Consulte la sección del estándar de HTML de WHATWG sobre representación¹², y la sección correspondiente del estándar de HTML de W3C¹³.

Idiomas de derecha a izquierda y conformidad con Unicode

Algunos códigos de escritura, como el latín y el devanagari, muestran caracteres de izquierda a derecha cuando el texto se presenta en líneas horizontales. Otros códigos de escritura, como el árabe o el hebreo, muestran caracteres de derecha a izquierda. El texto también puede ser bidireccional cuando un código de escritura de derecha a izquierda utiliza dígitos que se escriben de izquierda a derecha o cuando utiliza palabras incrustadas del inglés u otros idiomas que se escriben utilizando códigos de escritura de izquierda a derecha.

Se pueden generar desafíos y ambigüedades cuando la dirección horizontal del texto no es uniforme. Para resolver este problema, existe un algoritmo para determinar la direccionalidad del texto Unicode bidireccional.

Existe un conjunto de reglas que la aplicación debe implementar para producir el orden correcto en el momento de la visualización, que se describen mediante el Algoritmo Bidireccional de Unicode. Generalmente nos referimos a esto como el "algoritmo Bidi".

El algoritmo Bidi

El algoritmo Bidi describe cómo el software debe procesar el texto que contiene secuencias de caracteres de izquierda a derecha (LTR) y de derecha a izquierda (RTL). La dirección

¹² Disponible en <https://html.spec.whatwg.org/multipage/rendering.html>

¹³ Disponible en <https://www.w3.org/TR/2018/WD-html53-20181018/rendering.html>



base¹⁴ asignada a la frase determinará el orden en que se visualizará el texto. Puede ser de izquierda a derecha o de derecha a izquierda y define el orden en el que se muestran las secuencias de caracteres. En este documento, la dirección base es de izquierda a derecha, de modo que todas las secuencias de caracteres se muestran con la primera secuencia a la izquierda de la segunda secuencia.

Para saber si una secuencia es de izquierda a derecha o de derecha a izquierda, cada carácter en Unicode tiene una propiedad direccional asociada. La mayoría de las letras son fuertemente tipadas (caracteres fuertes) como LTR (de izquierda a derecha) o RTL (de derecha a izquierda) en función del código de escritura del que forman parte. Una secuencia de caracteres RTL fuertemente tipados se visualizará de derecha a izquierda. Esto es independiente de la dirección base circundante. Por ejemplo:

(LTR) ejemplo - مثال (RTL).

El texto con diferente direccionalidad se puede combinar en línea. En dichos casos, el algoritmo Bidi produce un sentido direccional separado de cada secuencia de caracteres contiguos con la misma direccionalidad.

Los espacios y la mayoría de la puntuación no son fuertemente tipados como LTR o RTL en Unicode porque se pueden usar en cualquier tipo de código de escritura. Por lo tanto, se clasifican como caracteres neutrales o débiles. Los caracteres débiles son los que se utilizan generalmente en una dirección, pero en algunos contextos pueden utilizarse en la otra. Los ejemplos de este tipo de caracteres incluyen:

- Dígitos europeos.
- Dígitos arábigos-índicos.
- Símbolos aritméticos y símbolos de divisas.
- Símbolos de puntuación que son comunes a muchos códigos de escritura, como los dos puntos, la coma, el punto y el espacio de no separación.

La direccionalidad de los caracteres neutrales es indeterminada sin contexto. Algunos ejemplos incluyen:

- Pestañas.
- Separadores de párrafos.
- La mayoría de los otros caracteres de espacios en blanco.

Cuando un carácter neutral se encuentra entre dos caracteres fuertemente tipados que tienen el mismo tipo direccional, también adoptará esa direccionalidad. Por ejemplo, un carácter neutral entre dos caracteres RTL se tratará como un carácter RTL en sí mismo, y tendrá el efecto de extender el sentido direccional:

- مثال.نطاق

Incluso si hay varios caracteres neutrales entre los dos caracteres fuertemente tipados, todos serán tratados de la misma manera.

¹⁴ En HTML, la dirección base se hereda de la dirección predeterminada del documento, que es de izquierda a derecha, o se establece explícitamente con el elemento primario más cercano que utiliza el atributo de dirección "dir".



Cuando un espacio o puntuación se encuentra entre dos caracteres fuertemente tipados que tienen diferente direccionalidad, el carácter (o caracteres) neutro se tratará como si tuviera la misma direccionalidad que la dirección base predominante. Por ejemplo:

- ejemplo. مثال

Recuerde que este documento tiene su base como izquierda a derecha y, por lo tanto, *ejemplo* es el dominio de segundo nivel y *مثال* es el TLD.

A menos que exista un reemplazo direccional, los números siempre se codifican y se introducen primero los dígitos de orden superior, y los numerales se representan de izquierda a derecha. La direccionalidad débil solo se aplica a la ubicación del número en su totalidad.

Los detalles completos del algoritmo Bidi se describen en el [Informe técnico N.º 9 de Unicode](#).

La regla Bidi para nombres de dominio

Un nombre de dominio Bidi es un nombre de dominio que contiene al menos una etiqueta RTL. La regla Bidi para nombres de dominio, especificada en el documento RFC 5893¹⁵, limita los puntos de código en los nombres para que no haya dos nombres que sean secuencias diferentes de puntos de código pero que muestren lo mismo debido a las reglas de visualización bidireccional.

Uniones

Algunos lenguajes utilizan códigos de escritura alfabéticos en los que los fonemas individuales se escriben utilizando dos caracteres denominados dígrafo. En otras palabras, un dígrafo es un grupo de dos letras sucesivas que representan un solo sonido (o fonema).

Ejemplos de dígrafos en inglés

ch (como en church)	th (then)	sh (shoe)
ph (como en phony)	th (think)	gh (rough)

Algunos dígrafos están completamente unidos como ligaduras. En escritura y tipografía, una ligadura se produce donde dos o más grafemas o letras se unen como un único glifo. Un ejemplo es el carácter et (&), que evolucionó a partir de las letras latinas adjuntas *e* y *t* ("et" que significa "y"). En la tipografía inglesa, *fi* y *ffi* se muestran a menudo como ligaduras.

Si las ligaduras y los dígrafos tienen la misma interpretación en todos los idiomas que utilizan un código de escritura determinado, la normalización Unicode generalmente resuelve las diferencias y las hace coincidir. Cuando tienen diferentes interpretaciones, para lograr la coincidencia se deben utilizar métodos alternativos (probablemente elegidos a nivel

¹⁵ Códigos de escritura de derecha a izquierda para Nombres de Dominio Internacionalizados en Aplicaciones (IDNA), RFC 5893, <https://www.rfc-editor.org/info/rfc5893>



de registro) o los usuarios deben ser instruidos para comprender que la coincidencia no tendrá lugar. Un ejemplo de interpretación diferente se puede encontrar en la Sección 4.3 de RFC 5894¹⁶. El Consorcio Unicode enumera dos estrategias principales para determinar el comportamiento de unión de un carácter en particular después de aplicar el algoritmo Bidi para manejar caracteres de unión y separador de longitud nula conocidos como ZWJ y ZWNJ. (Para obtener más información sobre estos caracteres, consulte <http://www.unicode.org/L2/L2005/05307-zwj-zwnj.pdf>).

- Cuando se configura, una implementación puede hacer referencia al almacenamiento de respaldo original para ver si había caracteres ZWNJ o ZWJ adyacentes.
- De forma alternativa, la implementación puede reemplazar ZWJ y ZWNJ por una propiedad de carácter fuera de banda asociada con esos caracteres adyacentes, de modo que la información no interfiera con el algoritmo Bidi y la información se conserve a través de la reorganización de esos caracteres. Una vez que se ha aplicado el algoritmo Bidi, esa información fuera de banda se puede utilizar para dar la forma adecuada.

Los registros de nombres de dominio y cualquier otra entidad que permita la creación de nombres de dominio (por ejemplo, las aplicaciones que crean etiquetas de tercer nivel y niveles inferiores) deben seguir la Regla Bidi para Nombres de Dominio para garantizar que los nombres se muestren de forma coherente y para evitar nombres confusos que puedan utilizarse para ataques homógrafos.

Para obtener más información sobre las uniones, consulte la Sección 4.3 de [RFC 5894](#).

Homóglifos y caracteres similares

Los homóglifos son caracteres que, debido a las similitudes en tamaño y forma, pueden parecer idénticos o confusamente similares. Frecuentemente ocurren cuando se mezclan las letras latinas, cirílicas y griegas. Por ejemplo, “o” del alfabeto latino (código U+006f), “o” minúscula del cirílico (código U+043e) y “o” ómicron minúscula del griego (código U+03bf). En algunos casos, hay homóglifos en una sola escritura, como la letra minúscula croata “lj” (código U+01c9) y las dos letras “lj” (código U+006c U+006a). Consulte la tabla en <http://homoglyphs.net/> para obtener más ejemplos.

Para evitar nombres de dominio con homóglifos, los registros deberían utilizar Reglas para la Generación de Etiquetas (LGR) que limiten los puntos de código en una etiqueta a un conjunto de un único código de escritura o códigos de escritura compatibles. Cada registro debería tener LGR para cada código de escritura en el que acepte registros¹⁷.

Para obtener más información sobre los mecanismos de seguridad de Unicode para la detección de confusiones, consulte:

- http://www.unicode.org/reports/tr39/#Confusable_Detection

¹⁶ *Nombres de Dominio Internacionalizados en Aplicaciones (IDNA): antecedentes, explicación y fundamento*, RFC 5894, <https://www.rfc-editor.org/rfc/rfc5894.html#section-4.2>

¹⁷ La IANA tiene una colección de LGR de registro en su Repositorio de Prácticas de IDN en <https://www.iana.org/domains/idn-tables>.



Para obtener más información sobre los caracteres confusamente similares y las prácticas recomendadas, consulte:

- Tutorial y descripción general del uso indebido de Unicode del M3AAWG <https://www.m3aawg.org/sites/default/files/m3aawg-unicode-tutorial-2016-02.pdf>
- Prácticas recomendadas del M3AAWG para la prevención del uso indebido de Unicode <https://www.m3aawg.org/sites/default/files/m3aawg-unicode-best-practices-2016-02.pdf>

Normalización, conversión de mayúsculas y minúsculas y preparación de cadenas de caracteres

La normalización de Unicode ayuda a determinar si dos cadenas de caracteres Unicode son equivalentes entre sí y proporciona formas estándar para procesar y almacenar cadenas de caracteres. Algunos caracteres pueden ser representados en Unicode por varias secuencias de códigos. Esto se denomina equivalencia Unicode. Unicode proporciona dos tipos de equivalencias:

- Canónica
- Compatibilidad

Las secuencias que representan el mismo carácter visual se denominan canónicamente equivalentes. Estas secuencias tienen el mismo aspecto y significado cuando se imprimen o visualizan. Por ejemplo:

U+006E (“n” latina minúscula) seguido de U+0303 (la combinación de la tilde “̃”)	= ñ
U+00F1 (letra “ñ” minúscula del alfabeto español)	= ñ

Unicode define NFC (Forma de Normalización C) como Descomposición Canónica, seguida de la Composición Canónica. Esto reduce el texto a un número mínimo de puntos de código sin cambiar su apariencia. Debe tenerse en cuenta que en este ejemplo, los tres caracteres anteriores son válidos para ser utilizados de acuerdo con IDNA2008.

Los equivalentes de compatibilidad son secuencias que pueden tener diferentes apariencias, pero en algunos contextos tienen el mismo significado. Es un tipo de equivalencia más débil entre caracteres o secuencias de caracteres. Por ejemplo:

U+FB00 (la ligadura tipográfica “ff”)	= ff
U+0066 U+0066 (dos letras “f” latinas)	= ff

En el ejemplo anterior, el punto de código U+FB00 se define como compatible, pero no canónicamente equivalente a la secuencia U+0066 U+0066. Las secuencias que son canónicamente equivalentes también son compatibles, pero lo contrario no siempre es cierto.



Debe tenerse en cuenta que el punto de código U+FB00 no es válido de acuerdo con IDNA2008.

Unicode define NFKC (Forma de Normalización KC) como Descomposición de Compatibilidad, seguida de la Composición Canónica. Esto reduce el texto a un conjunto estándar de puntos de código y puede cambiar su apariencia. Por ejemplo, NKFC convierte la ligadura “ff” en dos letras “f f” y al símbolo de antemeridiano a.m. (U+33C2) en cuatro caracteres “a.m.” (U+0061 U+002E U+006D U+002E.)

Para evitar problemas de interoperabilidad derivados del uso de secuencias de caracteres canónicamente equivalentes, pero diferentes, el W3C recomienda utilizar el NFC para todo el texto.

Para ver una lista de todos los caracteres que pueden cambiar en cualquiera de los Formularios de Normalización, consulte: <http://www.unicode.org/charts/normalization>.

Algunos otros puntos a tener en cuenta:

- Los caracteres de las etiquetas de IDN deben estar en formato NFC.
- Cuando dos aplicaciones comparten datos Unicode, pero los normalizan de forma diferente, pueden producirse errores y pérdida de datos.
- El consorcio Unicode afirma que las Formas de Normalización deben permanecer estables en el tiempo. En otras palabras, una cadena de caracteres debe permanecer normalizada en todas las versiones futuras de Unicode para proporcionar compatibilidad con versiones anteriores.
- Como se ha indicado anteriormente, sea conservador al mirar qué puntos de código permite en un nombre de dominio.

Consejos prácticos para desarrolladores de software

✘	No intente realizar la normalización convirtiendo a mayúsculas o ignorando los caracteres que no sean espacios, ya que esto dificulta la clasificación, la copia de datos, la importación y exportación de datos, la recuperación de datos por parte de las aplicaciones cliente y puede generar la pérdida o corrupción de datos.
✘	Nunca permita puntos de código en nombres de dominio que no estén permitidos de acuerdo con IDNA2008.

Para obtener más información sobre la normalización de Unicode, consulte:

- <http://www.w3.org/TR/charmod-norm>
- <http://unicode.org/reports/tr15>

Asignación y conversión de mayúsculas y minúsculas

La asignación y conversión de mayúsculas y minúsculas es el proceso de convertir todos los caracteres de una cadena de caracteres en el mismo tipo de letra, generalmente minúscula. La asignación de mayúsculas [A-Z] a minúsculas [a-z] funciona para documentos de texto solo ASCII, pero es mucho más compleja en idiomas que utilizan caracteres adicionales. La asignación de mayúsculas y minúsculas puede depender del contexto, y el carácter asignado puede depender del contexto en el que se encuentre, por



ejemplo, varias formas de la sigma del griego. También puede depender de la configuración regional, y el carácter asignado puede depender de la configuración regional en la que se interprete el texto, por ejemplo, la i mayúscula y minúscula turca con punto y sin punto. La conversión de mayúsculas y minúsculas es independiente de la configuración regional, para las cadenas que serán interpretadas por el software, mientras que la asignación de mayúsculas y minúsculas depende de la configuración regional y tiene por objeto generar texto que la gente pueda leer. Por último, la asignación a mayúsculas y la asignación a minúsculas **no** son funciones inversas.

Para los IDN, IDNA2008 permite a las aplicaciones utilizar cualquier asignación de casos apropiado porque la asignación se realiza antes de la validación de los puntos de código. En la práctica, las asignaciones de identificadores específicos según la configuración regional no existen y todos utilizan las asignaciones de UTS#46 de Unicode¹⁸.

Consejos prácticos para desarrolladores de software

✓	Considere el objetivo deseado antes de intentar la asignación de mayúsculas y minúsculas: ¿es una asignación genérica para etiquetas, una cadena de caracteres en un idioma conocido o algo más?
✓	Realice la Normalización de Unicode antes de la conversión de mayúsculas y minúsculas.

Glosario y otros recursos

Glosario

Etiqueta A	Representación con codificación compatible con ASCII (ACE) de una etiqueta en un Nombre de Dominio Internacionalizado, utilizado internamente dentro del protocolo del DNS. Las etiquetas A siempre comienzan con el prefijo “xn--”. Una etiqueta A puede convertirse en una etiqueta U y viceversa sin pérdida de información.
Prefijo ACE	Prefijo “xn--” de codificación compatible con ASCII.

¹⁸ UTS#46, *Procesamiento de compatibilidad de IDNA de Unicode*, <https://www.unicode.org/reports/tr46/#Mapping>



ASCII	Código Estadounidense Estándar para el Intercambio de Información. ASCII incluye caracteres del alfabeto latino sin acento y los dígitos europeo-arábigos. ASCII es un subconjunto de Unicode: cada carácter ASCII es también un carácter de Unicode.
API	Una Interfaz de Programación de Aplicaciones (API) es un conjunto de rutinas, protocolos y herramientas para crear software y aplicaciones. Una API puede ser para un sistema basado en web, sistema operativo o sistema de base de datos, y proporciona instalaciones para desarrollar aplicaciones para ese sistema con un lenguaje de programación determinado.
Espacio de código	Rango que define los límites inferior y superior para una codificación.
Punto de código	Un punto de código es un valor numérico en un espacio de código. Los puntos de código se utilizan para distinguir el valor numérico de su codificación como una secuencia de bits, y para distinguir el carácter abstracto de una representación gráfica particular del mismo (glifo).
Zona Raíz del DNS	La zona raíz es el directorio central para el DNS, que es un componente clave para buscar cosas en el DNS; por ejemplo, traducir los nombres de host a direcciones IP.
EAI	La Internacionalización de Direcciones de Correo Electrónico permite caracteres UTF-8 en una dirección de correo electrónico: el nombre de dominio, la parte local o ambos.
IANA	Autoridad de Números Asignados en Internet. Sus funciones incluyen: <ul style="list-style-type: none">▪ Administración de la raíz del DNS, los dominios .int y .arpa y un recurso de prácticas de IDN.▪ Coordinación de la reserva global de direcciones IP y números AS, suministrándolos principalmente a los Registros Regionales de Internet (RIR).▪ Los sistemas de numeración de los protocolos de Internet son administrados conjuntamente con los organismos de normalización.
ICANN	La misión de la ICANN es ayudar a garantizar una Internet global, unificada, estable y segura. Para contactar a otra persona en Internet debemos ingresar una dirección –un nombre o un número– en nuestra computadora u otro dispositivo. Esa dirección debe ser única para que las computadoras puedan localizarse unas a otras. La ICANN ayuda a coordinar y brindar soporte a estos identificadores únicos en todo el mundo. La ICANN fue creada en 1998 como una corporación de bien público y sin fines de lucro, con una comunidad integrada por participantes de todo el mundo.



IDN	Nombre de Dominio Internacionalizado. Los IDN son nombres de dominio que incluyen caracteres UTF-8 más allá de las veintiséis letras básicas del alfabeto latino "a-z", los números 0-9 y el guión "-"
IDNA	Nombres de Dominio Internacionalizados en Aplicaciones.
IDN ccTLD	Dominio de alto nivel con código de país que incluye caracteres más allá de las veintiséis letras del alfabeto latino básico "a-z". Ejemplos: <ul style="list-style-type: none">▪ .рф (Rusia)▪ .مصر (Egipto)▪ .السعودية (Arabia Saudita)
IETF	El Grupo de Trabajo en Ingeniería de Internet (IETF) es una gran comunidad internacional abierta de diseñadores de red, operadores, proveedores e investigadores involucrados en la evolución de la arquitectura y el funcionamiento continuo de Internet. Está abierto a la todas las personas interesadas. El IETF desarrolla estándares de Internet, en particular, los estándares relacionados con el conjunto de protocolos de Internet (TCP/IP) y los protocolos utilizados para la web como HTTP y TLS.
Idioma	El método de comunicación humana, ya sea hablado o escrito, que consiste en el uso de palabras de una manera estructurada y convencional.
Punycode	Un algoritmo que representa UTF-8 en el subconjunto de caracteres limitados de ASCII admitido por el Sistema de Nombres de Dominio (DNS). Punycode se utiliza en etiquetas A en el marco de los Nombres de Dominio Internacionalizados en Aplicaciones (IDNA).
Registrador	Una organización donde los usuarios registran nombres de dominio. El registrador guarda registros de la información de contacto y presenta la información técnica a un directorio central conocido como "registro".
Registro	La base de datos principal y autoritativa de todos los nombres de dominio registrados en cada dominio de alto nivel (TLD).
RFC	Una Solicitud de Comentarios (RFC) es un documento formal del Grupo de Trabajo en Ingeniería de Internet (IETF) que es el resultado de la redacción del comité y la posterior revisión de las partes interesadas. Algunas (aunque no todas) RFC documentan las normas de Internet aprobadas.
Texto	El conjunto de letras o caracteres utilizados en la escritura, que representan los sonidos de un idioma.



Nombre de dominio de segundo nivel	En la jerarquía del Sistema de Nombres de Dominio (DNS), un dominio de segundo nivel (SLD o 2LD) es un dominio que está directamente debajo de un dominio de nivel superior (TLD). Por ejemplo, en ejemplo.com, ejemplo es el dominio de segundo nivel del TLD .com.
Etiqueta U	Una etiqueta U es una cadena de caracteres Unicode válida según IDNA que incluye al menos un carácter no ASCII. Se puede convertir en una etiqueta A y viceversa sin pérdida de información.
Software listo para la Aceptación Universal	Software que tiene la capacidad de aceptar, almacenar, procesar, validar y visualizar todos los dominios de alto nivel, IDN y direcciones de correo electrónico por igual.
Unicode	Un estándar universal de codificación de caracteres. Define la forma en que se representan los caracteres individuales en archivos de texto, páginas web y otros tipos de documentos. Unicode fue diseñado para admitir caracteres de todos los idiomas del mundo. Puede admitir aproximadamente 1 000 000 caracteres. Véase: http://unicode.org .
UTF	Formato de Transformación Unicode. Es una forma de representar puntos de código Unicode como una secuencia de bytes. UTF-8 es el UTF preferido para manejar IDN y EAI. UTF-8 convierte Unicode a bytes de 8 bits.
M3AAWG	El Grupo de Trabajo Anti-Abuso de Mensajes, Malware y Móvil (M ³ AAWG) es donde la industria se reúne para trabajar contra: botnets, malware, spam, virus, ataques de denegación de servicio (DoS) y otras acciones abusivas en línea. Véase: https://www.m3aawg.org/ .
W3C	El Consorcio Mundial de Internet (W3C) es una comunidad internacional en la cual las organizaciones miembro , el personal de tiempo completo y el público trabajan en conjunto para desarrollar estándares para la web como HTML. Véase: https://www.w3.org/ .
WHATWG	El Grupo de Trabajo de Tecnología de Aplicaciones de Hipertexto Web (WHATWG) es una comunidad de personas interesadas en desarrollar la web a través de estándares y pruebas. El WHATWG fue fundado por integrantes de Apple, la Fundación Mozilla y Opera Software en 2004, después de un taller del W3C. Véase https://whatwg.org/ .



ZWJ	Zero-Width Joiner o Unión de longitud nula es un carácter no imprimible que se utiliza en la composición tipográfica computarizada de algunos códigos de escritura, como el código de escritura arábigo y todos los códigos de escritura índicos. Cuando se coloca entre dos caracteres que de otro modo no estarían conectados, un ZWJ hace que se impriman en su forma conectada.
ZWNJ	Zero-Width Non-Joiner o Separador de longitud nula es un carácter no imprimible utilizado en la informatización de sistemas de escritura que utilizan ligaduras. Para algunos idiomas y códigos de escritura, muchas de las letras del alfabeto se conectan naturalmente con la siguiente letra cuando se escribe en una palabra, formando una ligadura. Sin embargo, para mostrar correctamente ciertos prefijos, sufijos y palabras compuestas, el ZWNJ se utiliza para anular este comportamiento natural de unir letras y evitar que se unan a la siguiente letra (pero sin añadir un espacio entre las dos).

Para obtener un glosario completo de la ICANN, consulte: <https://www.icann.org/icann-acronyms-and-terms/>.

RFC y normas clave

RFC de IDN	
RFC 3492	<p>Punycode: Una codificación Bootstring de Unicode para Nombres de Dominio Internacionalizados en Aplicaciones (IDNA)</p> <p>La RFC 3492 describe a Punycode como:</p> <p><i>"una sintaxis simple y eficiente de codificación de transferencia diseñada para su uso con Nombres de Dominio Internacionalizados en Aplicaciones (IDNA)"</i></p> <p>Punycode transforma de forma única y reversible una cadena de caracteres Unicode en una cadena de caracteres ASCII. Esta RFC define un algoritmo general denominado Bootstring. Este algoritmo permite que una cadena de puntos de código básicos represente de manera única cualquier cadena de puntos de código extraídos de un conjunto más grande.</p> <p>https://tools.ietf.org/html/rfc3492</p>



RFC 5890	Nombres de Dominio Internacionalizados en Aplicaciones (IDNA): Definiciones y marco del documento Esta RFC describe el contexto de uso y el protocolo para una revisión de Nombres de Dominio Internacionalizados en Aplicaciones (IDNA). https://tools.ietf.org/html/rfc5890
RFC 5891	Protocolo de Nombres de Dominio Internacionalizados en Aplicaciones (IDNA) Esta RFC especifica el mecanismo de protocolo, denominado Nombres de Dominio Internacionalizados en Aplicaciones (IDNA), para registrar y buscar los IDN de una manera que no requiera cambios en el propio DNS. https://tools.ietf.org/html/rfc5891
RFC 5892	Los puntos de código Unicode y los Nombres de Dominio Internacionalizados para Aplicaciones (IDNA) La RFC 5892 especifica reglas para decidir si un punto de código, considerado de forma aislada o en contexto, es un candidato para su inclusión en un Nombre de Dominio Internacionalizado (IDN). https://tools.ietf.org/html/rfc5892
RFC 5893	Códigos de escritura de derecha a izquierda para Nombres de Dominio Internacionalizados en Aplicaciones (IDNA) Esta RFC proporciona una nueva regla Bidi para etiquetas de Nombres de Dominio Internacionalizados en Aplicaciones (IDNA), para el uso de códigos de escritura de derecha a izquierda en Nombres de Dominio Internacionalizados. https://tools.ietf.org/html/rfc5893
RFC 5894	Nombres de Dominio Internacionalizados en Aplicaciones (IDNA): antecedentes, explicación y fundamento Este documento informativo proporciona una descripción general de un sistema revisado para abordar las versiones más recientes de Unicode y proporciona material explicativo para sus componentes. https://tools.ietf.org/html/rfc5894



RFC 5895	Asignación de caracteres para Nombres de Dominio Internacionalizados en Aplicaciones (IDNA) 2008 Esta RFC describe las acciones que una implementación puede realizar entre la recepción de la entrada del usuario y la aprobación de puntos de código permitidos para el nuevo protocolo de IDNA (2008). Describe una operación que se aplicará a la entrada del usuario para preparar esa entrada de usuario para su uso en un protocolo "en la red". También incluye un procedimiento de implementación general para la asignación. https://tools.ietf.org/html/rfc5895
RFC de EAI	
RFC 6530	Descripción general y marco para el correo electrónico internacionalizado Este estándar presenta una serie de especificaciones que definen los mecanismos y las extensiones de protocolos necesarios para proporcionar compatibilidad completa a las direcciones de correo electrónico internacionalizadas. Este documento describe cómo encajan entre sí los diversos elementos de la internacionalización de las direcciones de correo electrónico y las relaciones entre las especificaciones principales asociadas con el transporte de mensajes, formatos de encabezado y manejo. https://tools.ietf.org/html/rfc6530
RFC 6531	Extensión SMTP para correo electrónico internacionalizado El documento define una extensión de Protocolo de transferencia de correo simple para que los servidores puedan anunciar la capacidad de aceptar y procesar direcciones de correo electrónico internacionalizadas y encabezados de correo electrónico internacionalizados. https://tools.ietf.org/html/rfc6531
RFC 6532	Encabezados de correo electrónico internacionalizado Este documento especifica una mejora del formato de mensaje de Internet y MIME que permite el uso de Unicode en las direcciones de correo y la mayoría del contenido del campo del encabezado. Este documento especifica una mejora del formato de mensaje de Internet (RFC 5322) y MIME que permite el uso directo de UTF-8, en lugar de ASCII, en valores de campo de encabezado, incluidas las direcciones de correo. Se define un nuevo tipo de medio, mensaje/global, para los mensajes que utilizan este formato extendido. Esta especificación también levanta la restricción MIME al tener codificaciones de transferencia de contenido sin identidad en cualquier subtipo del tipo de mensaje de nivel superior, de modo que las partes de



	<p>mensaje/global se puedan transmitir de manera segura a través de la infraestructura de correo existente.</p> <p>https://tools.ietf.org/html/rfc6532</p>
RFC 6533	<p>Notificaciones internacionalizadas de estado de entrega y eliminación</p> <p>Esta especificación agrega un nuevo tipo de dirección para las direcciones de correo electrónico internacionales para que una dirección de destinatario original con caracteres que no sean ASCII pueda conservarse de forma correcta incluso después de la degradación. Esto también proporciona tipos de medios de retorno de contenido actualizado para notificaciones de estado de entrega y notificaciones de eliminación de mensajes para admitir el uso del nuevo tipo de dirección.</p> <p>https://tools.ietf.org/html/rfc6533</p>
RFC 8398	<p>Direcciones de correo electrónico internacionalizadas en certificados X.509</p> <p>Este documento define una nueva forma de nombre para su inclusión en el campo otherName de una extensión Subject Alternative Name y Issuer Alternative Name de X.509 que permite asociar un sujeto de certificado a una dirección de correo electrónico internacionalizada.</p> <p>https://tools.ietf.org/html/rfc8398.</p>
RFC 8399	<p>Actualizaciones de internacionalización a la RFC 5290</p> <p>Las actualizaciones de RFC 5280 descritas en este documento proporcionan una alineación con la especificación de 2008 para Nombres de Dominio Internacionalizados (IDN) y agregan compatibilidad para direcciones de correo electrónico internacionalizadas en certificados X.509.</p> <p>https://tools.ietf.org/html/rfc8399</p>

Normas clave



ISO 10646 (Unicode)	<p>Para proporcionar una base técnica común para el procesamiento de información electrónica en varios idiomas, la Organización Internacional de Normalización (ISO) ha desarrollado una norma de codificación internacional denominada ISO 10646. La norma ISO 10646 proporciona un estándar unificado para la codificación de caracteres en todos los principales idiomas del mundo, incluidos los caracteres del chino tradicional y simplificado. Este gran conjunto de caracteres se denomina Conjunto Universal de Caracteres (UCS). El mismo conjunto de caracteres está definido por el estándar Unicode, que además define las propiedades de caracteres adicionales y otros detalles de la aplicación de gran interés para los implementadores.</p> <p>Unicode es un sistema de codificación de caracteres diseñado por el Consorcio Unicode para apoyar el intercambio, procesamiento y visualización de los textos escritos de los principales idiomas del mundo. ISO 10646 y Unicode definen varias formas de codificación de su repertorio común: UTF-8, UCS-2, UTF-16, UCS-4 y UTF-32.</p> <p>http://www.iso.org/iso/home/store/catalogue_ics/catalogue_detail_ics.htm?csnumber=63182</p>
GB18030 (China)	<p>GB 18030-2000 es un estándar del gobierno chino que especifica una página de códigos extendida para su uso en el mercado chino, además de UTF-8. El código de procesamiento interno para el repertorio de caracteres puede y debe ser Unicode; sin embargo, la norma estipula que los proveedores de software deben garantizar un ida y vuelta exitoso entre GB18030 y el código de procesamiento interno. Todos los productos que actualmente se venden o que se venderán en China deben planificar la migración de la página de códigos para admitir GB18030 sin excepción. GB18030 es un "estándar obligatorio" y el gobierno chino regula el proceso de certificación para reforzar la implementación de GB18030.</p> <p>http://icu-project.org/docs/papers/unicode-gb18030-faq.html</p>

Recursos en línea



API	<p>Interfaz de Programación de Aplicaciones (API) de Windows https://www.msdn.microsoft.com/enus/library/windows/desktop/ff818516%28v=vs.85%29.aspx</p> <p>API de SharePoint https://msdn.microsoft.com/en-us/library/office/jj860569.aspx</p> <p>Lista de sufijos públicos https://publicsuffix.org/list/public_suffix_list.dat</p> <p>Lista de TLD autoritativa de la ICANN http://data.iana.org/TLD/tlds-alpha-by-domain.txt</p> <p>API de Android http://developer.android.com/guide/index.html</p> <p>API de MAC IOS https://developer.apple.com/library/mac/navigation</p> <p>Marco de .Net https://msdn.microsoft.com/en-us/library/system.text.encoding(v=vs.110).aspx</p>
Seguridad de Unicode	<p>Consideraciones de seguridad relativas a Unicode http://www.unicode.org/reports/tr36</p> <p>Mecanismos de seguridad de Unicode http://www.unicode.org/reports/tr39</p>
Agrupaciones de caracteres de Unicode	<p>Planos de código Unicode https://www.unicode.org/versions/Unicode12.0.0/ch02.pdf; págs. 44-54</p> <p>Descripción general de GB18030 http://icu-project.org/docs/papers/gb18030.html</p> <p>Tabla autoritativa de correspondencia entre GB18030-2000 y Unicode http://source.icu-project.org/repos/icu/data/trunk/charset/data/xml/gb-18030-2000.xml</p> <p>Normalización de Unicode https://unicode.org/reports/tr15/</p>
Ataques de Unicode	<p>Sección 3.1, "Ataques de UTF-8" en el Informe técnico N.º 36 de Unicode http://unicode.org/reports/tr36/#UTF-8_Exploit</p> <p>Prácticas recomendadas del M3AAGW para la prevención del uso indebido de Unicode https://www.m3aawg.org/sites/default/files/m3aawg-unicode-best-practices-2016-02.pdf</p> <p>Tutorial y descripción general del uso indebido de Unicode del M3AAGW https://www.m3aawg.org/sites/default/files/m3aawg-unicode-tutorial-2016-02.pdf</p>



	<p>Véase también: http://www.unicode.org</p>
Misceláneos	<p>URI http://tools.ietf.org/html/rfc3986</p> <p>El Sistema de Nombres de Dominio: Una explicación no técnica: por qué la resolución universal es importante http://www.internic.net/faqs/authoritative-dns.html</p> <p>Glosario de la ICANN https://www.icann.org/icann-acronyms-and-terms/</p>

¿Necesita más información?

El Grupo Directivo sobre Aceptación Universal (UASG) y la comunidad están disponibles para asesorar a los desarrolladores e implementadores de software.

Contáctenos para compartir sus ideas y sugerencias sobre el tema en info@uasg.tech.
Únase a la lista de debate sobre Aceptación Universal en <http://tinyurl.com/ua-discuss>.
Para obtener más información sobre la iniciativa, visite <http://www.icann.org/universalacceptance>.