Quick Guide Issues – Portuguese

**CURRENT WORD COUNT:** 197

**IDEAL WORD COUNT:** 150

As validações não devem ocorrer a menos que sejam exigidas para a operação do aplicativo ou do serviço. Essa é a maneira mais fácil de garantir que todos os nomes de domínio válidos sejam aceitos nos sistemas.

Se a validação for obrigatória, considere o seguinte:

Verifique a porção TLD de um nome de domínio em comparação a uma tabela oficial:

http://www.internic.net/domain/root.zone

http://www.dns.icann.org/services/authoritative-dns/index.html

http://data.iana.org/TLD/tlds-alpha-by-domain.txt

Consulte também o SAC070: https://tinyurl.com/sac070.

Consulte o nome de domínio com relação ao DNS (Domain Name System, Sistema de Nomes de Domínio).

Exija que o endereço de e-mail seja informado mais de uma vez para descartar os erros de digitação.

Valide os caracteres nos rótulos para determinar se o rótulo U não contém pontos de código "NÃO PERMITIDO" ou pontos de código que não estão atribuídos na sua versão em Unicode. Visite: <https://tools.ietf.org/html/rfc5892>.

Limite a validação de rótulos para uma quantidade pequena de regras de rótulos inteiros definidas nas RFCs (Request for Comments, Solicitações de Comentários). Visite: <https://tools.ietf.org/html/rfc5894>.

Se uma cadeia de caracteres semelhante a um nome de domínio contiver o caractere de ponto final ideográfico ‘。’, ele deve ser convertido para um ‘.’ antes de realizar a validação.

-----

**CURRENT WORD COUNT:** 91

**IDEAL WORD COUNT:** 60

O processamento ocorre sempre que um endereço de e-mail ou nome de domínio é usado por um aplicativo ou serviço para executar uma atividade (por exemplo, para pesquisar ou classificar uma lista) ou é transformado em um formato alternativo (por exemplo, para armazenar ACSII como Unicode). É possível que seja realizada validação adicional durante o processamento.

Os nomes de domínio e endereços de e-mail podem ser processados de maneiras ilimitadas\*, o que reforça a necessidade de haver convenções que garantam que os dados estão sendo compreendidos e classificados de modo consistente.

-----

**CURRENT WORD COUNT:** 214

**IDEAL WORD COUNT:** 170

Uma vez que o padrão Unicode é expandido continuamente, os pontos de código não definidos quando o aplicativo ou serviço foi criado devem ser verificados para garantir que eles não “interromperão” a experiência do usuário. Fontes ausentes no sistema operacional subjacente podem fazer com que alguns caracteres não sejam exibidos (muitas vezes o caractere “” é usado para representá-los), mas essa situação não deve resultar em uma falha fatal.

Use APIs compatíveis com Unicode.

Use os documentos mais recentes do protocolo [http://tools.ietf.org/html/rfc5891] e tabelas [http://tools.ietf.org/html/rfc5892] de IDNA (Internationalized Domain Names in Applications, Nomes de Domínio Internacionalizados em Aplicativos) para IDNs (Internationalized Domain Names, Nomes de Domínio Internacionalizados).

Processe no formato UTF-8 sempre que possível.

Certifique-se de que o produto ou recurso lide com os números da maneira esperada. Por exemplo, numerais ASCII e representações de números ideográficas asiáticas devem ser tratados como números. [RFC5892, link acima]

Faça o upgrade de aplicativos e servidores/serviços juntos. Se o servidor for Unicode e o cliente não for Unicode, ou vice-versa, será necessário converter os dados para cada página de código sempre que os dados forem transferidos entre o servidor e o cliente.

Faça análises da codificação para enviar ataques de buffer overflow. Ao fazer a transformação de caracteres, as cadeias de texto podem aumentar ou diminuir significativamente.

-----

**CURRENT WORD COUNT:** 17

**IDEAL WORD COUNT:** 14

\*Exemplos: Identificar pessoas na Nova Zelândia pesquisando no ccTLD .nz; identificar farmacêuticos pesquisando endereços de e-mail usuário@\*.farmacêutico.

-----

**CURRENT WORD COUNT:** 270

**IDEAL WORD COUNT:** 240

Exiba todos os códigos de pontos Unicode compatíveis com o sistema operacional subjacente. Se um aplicativo tiver seus próprios conjuntos de fontes, é necessário oferecer um suporte abrangente a Unicode para a coleta de fontes disponíveis no sistema operacional.

Ao desenvolver um aplicativo ou um serviço ou ao operar um registro, considere os idiomas compatíveis e certifique-se de que o SO e os aplicativos aceitem esses idiomas.

Converta dados diferentes de Unicode para Unicode antes de exibir. Por exemplo, o usuário final deve ver “todos.みんな”, em vez de “todos.xn--q9jyb4c”. (Essa conversão é um exemplo de processamento compatível com a UA).

Exibir Unicode por padrão. Usar texto com Punycode para o usuário apenas quando isso for vantajoso para ele. Aumente a exibição de Unicode com texto em Punycode exibido ao passar o mouse sobre ele como uma forma de mitigação.

Considere que endereços com misturas de escritas serão mais comuns. Alguns caracteres Unicode podem parecer iguais ao olho humano, mas diferente para os computadores. Não presuma que as cadeias de caracteres com mistura de escritas são destinadas a fins maliciosos, como phishing, e se a interface do usuário chamar a atenção do usuário para as cadeias de caracteres, certifique-se de que isso seja feito de maneira que não prejudique os usuários de escritas não latinas. Saiba mais sobre considerações de segurança com Unicode em: <http://unicode.org/reports/tr36/>.

Use o processamento de compatibilidade de IDNA para Unicode a fim atender às expectativas dos usuários. Para saber mais, acesse: <http://unicode.org/reports/tr46/>.

Esteja atento a caracteres não atribuídos ou não permitidos. Saiba mais na RFC 5892: https://tools.ietf.org/rfc/rfc5892.txt.

-----

**CURRENT WORD COUNT:** 328

**IDEAL WORD COUNT:** 280

Revisões de código fonte e teste de unidade

O processo de inspeção do código fonte e verificação de que apenas as técnicas de programação, bibliotecas de software e interfaces (também conhecidas como “APIs”) corretas foram usadas. Depois disso, o administrador poderá verificar se o aplicativo ou serviço funciona testando-o com as capacidades específicas (aceitar, validar etc.) listadas anteriormente. Esse método é geralmente usado somente por desenvolvedores de aplicativos e provedores de serviços on-line.

Como parte do trabalho de conscientização do UASG, o grupo está entrando em contato diretamente com os desenvolvedores de aplicativos e os maiores provedores de serviços on-line para incentivá-los a fazer as revisões de código fonte e testes para aceitação universal e compartilhar uma lista de critérios que podem ser usados para desenvolver os casos de teste.

Testes manuais

Exigem a execução de vários testes em domínios novos e que não utilizam ASCII, como o envio de um endereço de e-mail ao registrar para um serviço on-line e verificação se ele foi aceito. Uma vez que existem vários possíveis serviços on-line que aceitam inscrições, bem como possíveis novas combinações de endereços de e-mail, esse método exige a tentativa de diferentes combinações de aplicativos, serviços, endereços de e-mail e/ou nomes de domínio para fornecer um escopo maior de casos de uso. Esse método pode ser executado por qualquer pessoa, mas é mais trabalhoso.

O UASG também está ajudando a promover esse método desenvolvendo uma lista dos principais sites, aplicativos, endereços de e-mail e nomes de domínio apropriados para esses testes.

Testes automatizados

O uso de scripts automatizados ou diretivas para testar uma variedade de URLs. Esse método exige um trabalho técnico mais direto, mas pode ser dimensionado com mais facilidade para trabalhos maiores de medição e monitoramento. Um exemplo do mundo real é a recente investigação de gTLDs realizada pelo APNIC em nome da ICANN: https://tinyurl.com/new-gtld-ua.

O UASG está examinando alguns métodos de testes automatizados para a aceitação universal e divulgará suas descobertas quando estiverem disponíveis.