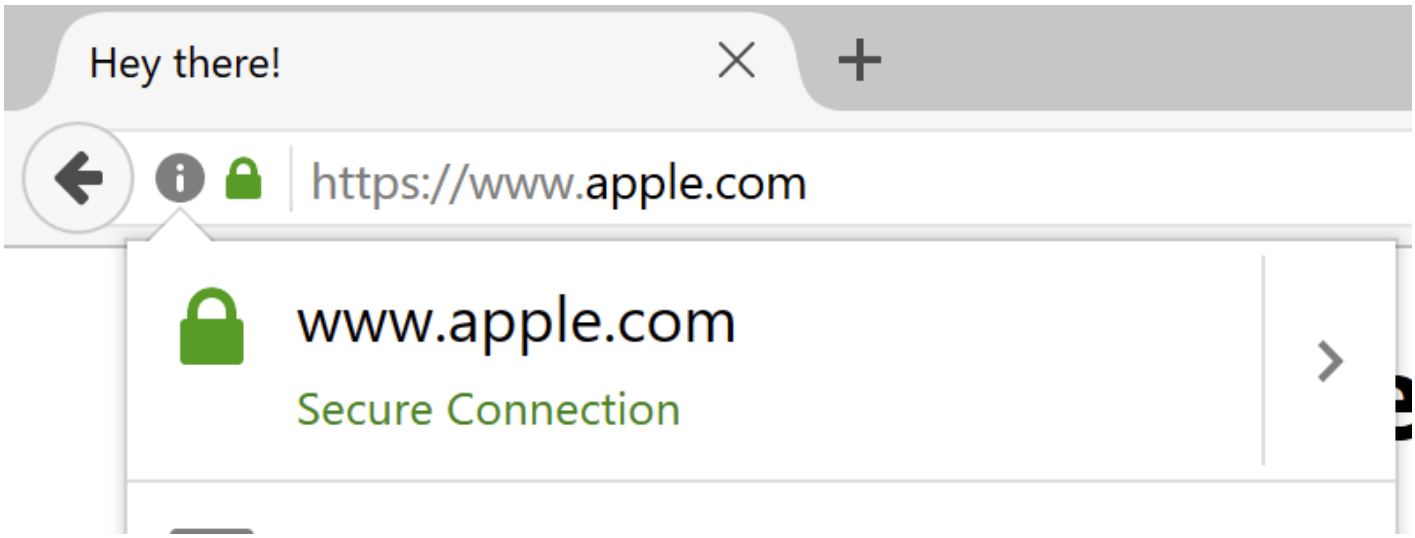
Phishing with Unicode Domains

Posted by Xudong Zheng on April 14, 2017



Before I explain the details of the vulnerability, you should take a look at the proof-of-concept.

Punycode makes it possible to register domains with foreign characters. It works by converting individual domain label to an alternative format using only ASCII characters. For example, the domain "xn--s7y.co" is equivalent to "短.co".

From a security perspective, Unicode domains can be problematic because many Unicode characters are difficult to distinguish from common ASCII characters. It is possible to register domains such as "xn--pple-43d.com", which is equivalent to "apple.com". It may not be obvious at first glance, but "apple.com" uses the Cyrillic "a" (U+0430) rather than the ASCII "a" (U+0041). This is known as a homograph attack.

Fortunately modern browsers have mechanisms in place to limit IDN homograph attacks. The page IDN in Google Chrome highlights the conditions under which an IDN is displayed in its native Unicode form. Generally speaking, the Unicode form will be hidden if a domain label contains characters from multiple different languages. The "apple.com" domain as described above will appear in its Punycode form as "xn--pple-43d.com" to limit confusion with the real "apple.com".

The homograph protection mechanism in Chrome, Firefox, and Opera unfortunately fails if every characters is replaced with a similar character from a single foreign language. The domain "apple.com", registered as "xn--80ak6aa92e.com", bypasses the filter by only using Cyrillic characters. You can check this out yourself in the proof-of-concept using Chrome, Firefox, or Opera.

Visually, the two domains are indistinguishable due to the font used by Chrome and Firefox. As a result, it becomes impossible to identify the site as fraudulent without carefully inspecting the site's URL or SSL certificate. This Go program nicely demonstrates the difference between the two sets of characters. Internet Explorer, Safari, along with several less mainstream browsers are fortunately not vulnerable.

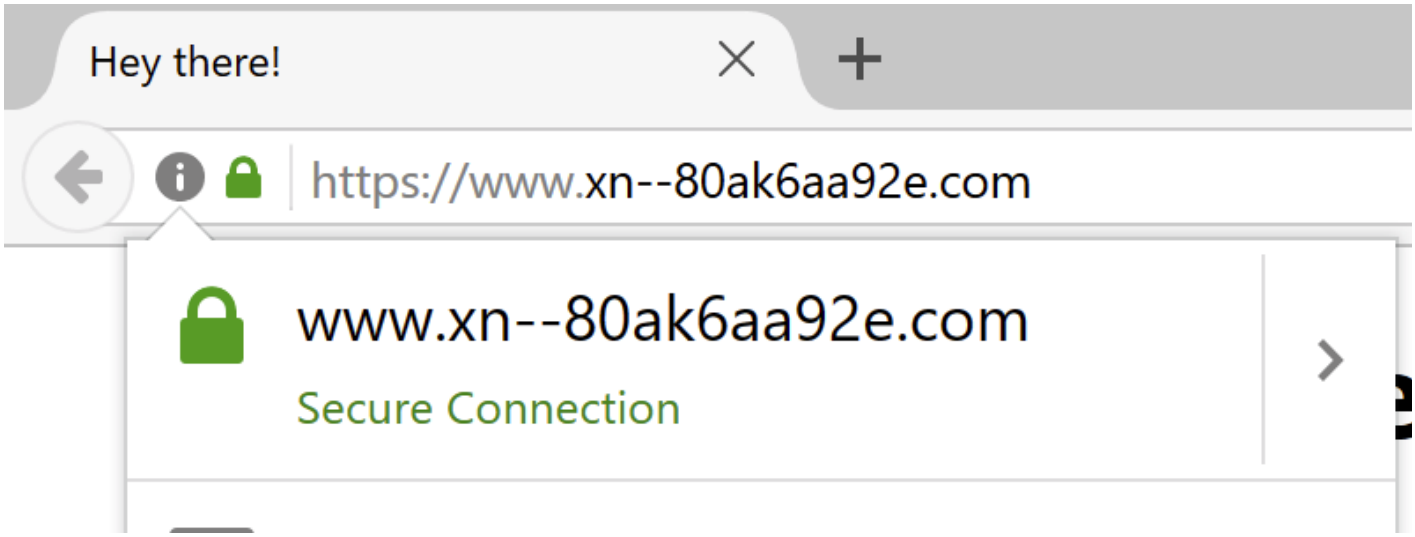
Screenshots: Chrome, Firefox, Firefox SSL

This bug was reported to Chrome and Firefox on January 20, 2017 and was fixed in the trunk of Chrome 59 (currently in Canary) on March 24. The Chrome team has since decided to include the fix in Chrome 58, which should be available around April 25. The existence of the bug in Opera was brought to my attention only after the initial publication of this post. The problem remains unaddressed in Firefox as they remain undecided whether it is within their scope. The Bugzilla issue was initially marked "RESOLVED" and

"WONTFIX", though it has since been reopened, made public, and given the "sec-low" keyword.

> Our IDN threat model specifically excludes whole-script homographs, because they can't be detected programmatically and our "TLD whitelist" approach didn't scale in the face of a large number of new TLDs. If you are buying a domain in a registry which does not have proper anti-spoofing protections (like .com), it is sadly the responsibility of domain owners to check for whole-script homographs and register them.

Firefox users can limit their exposure to this bug by going to `about:config` and setting `network.IDN_show_punycode` to `true`. This will force Firefox to always display IDN domains in its Punycode form, making it possible to identify malicious domains. Thanks to user MARKZILLA from reddit for this temporary solution.



A simple way to limit the damage from bugs such as this is to always use a password manager. In general, users must be very careful and pay attention to the URL when entering personal information. Until this is fixed, users should manually type the URL or navigate to the site via a search engine when in doubt. I hope Firefox will consider implementing a fix to this problem since this can cause serious confusion even for those who are extremely mindful of phishing.

Enjoyed this? Follow me on Twitter @Xudong_Zheng

Share on Facebook