

Avoiding the not secure warning in chrome

Eric Lawrence

 (<https://twitter.com/ericlaw>)  (<https://github.com/ericlaw1979>)  (<https://textslashplain.com/>)

As [announced in September](https://security.googleblog.com/2016/09/moving-towards-more-secure-web.html), Chrome will soon mark non-secure pages containing **password** and **credit card** input fields as **Not Secure** in the URL bar.

This document is intended to aid Web Developers in updating their sites to avoid this warning.

Enable warnings

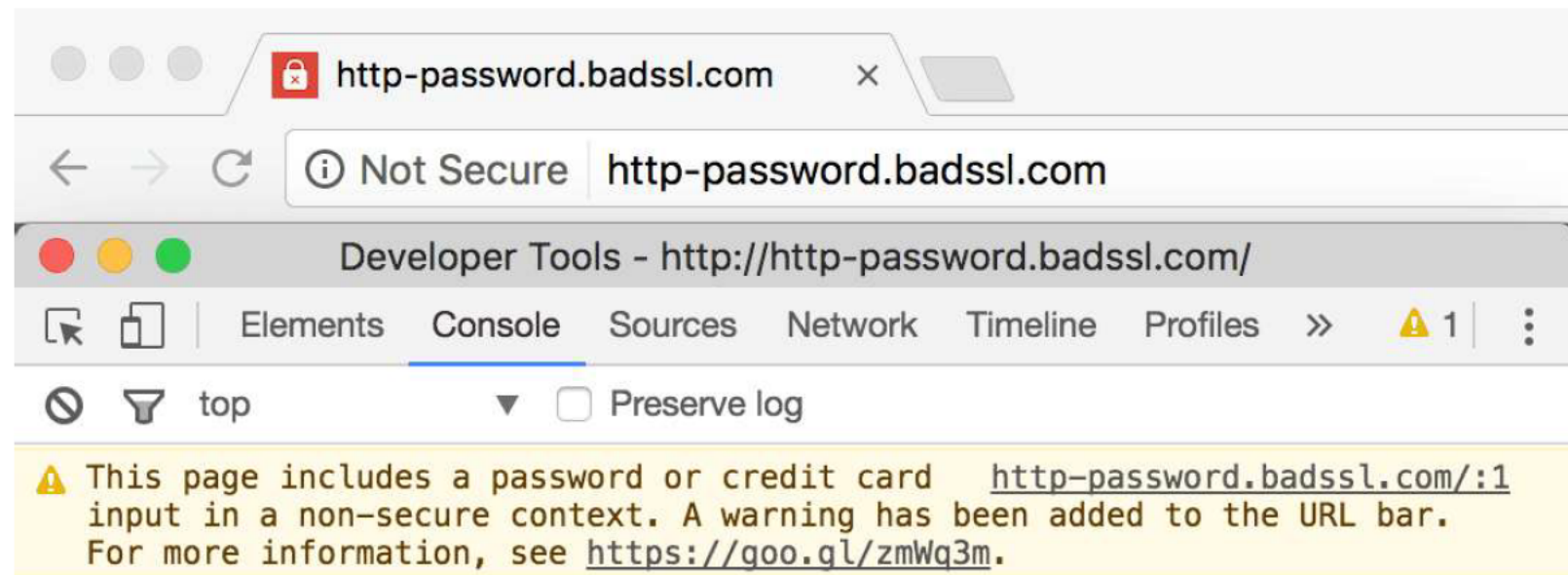
Warnings will be enabled by default for everyone in Chrome 56, slated for release in January 2017.

To test the upcoming user experience before that time, install the latest [Google Chrome Canary](https://www.google.com/chrome/browser/canary.html) build.

To configure Chrome to show the warning as it will appear in January 2017, open `chrome://flags/#mark-non-secure-as` and set the `Mark non-secure origins as non-secure` option to `Display a verbose state when password or credit card fields are detected on an HTTP page`. Then relaunch your browser.

You can see an example of the browser's warning behavior on [this page](http://http-password.badssl.com/).

When the Not Secure state is shown, the DevTools console shows the message `This page includes a password or credit card input in a non-secure context. A warning has been added to the URL bar.`



Resolve warnings

To ensure that the Not Secure warning is not displayed for your pages, you must ensure that all forms containing `<input type=password>` elements and any inputs detected as credit card fields

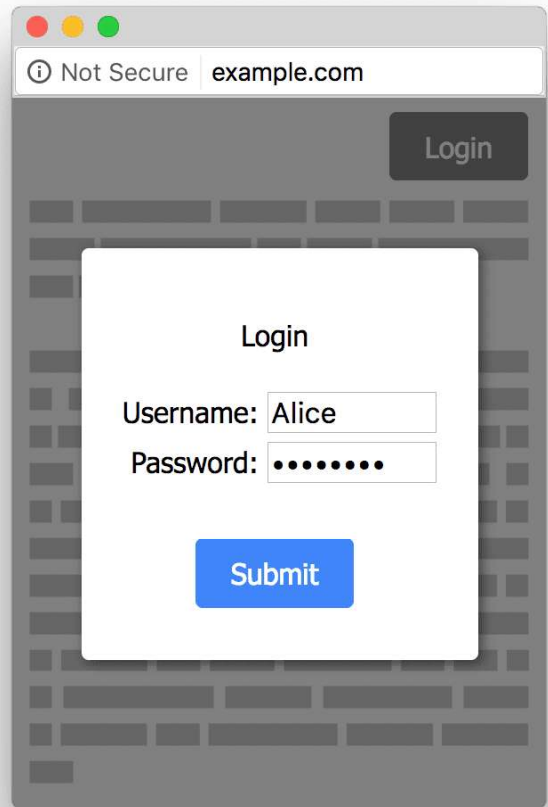
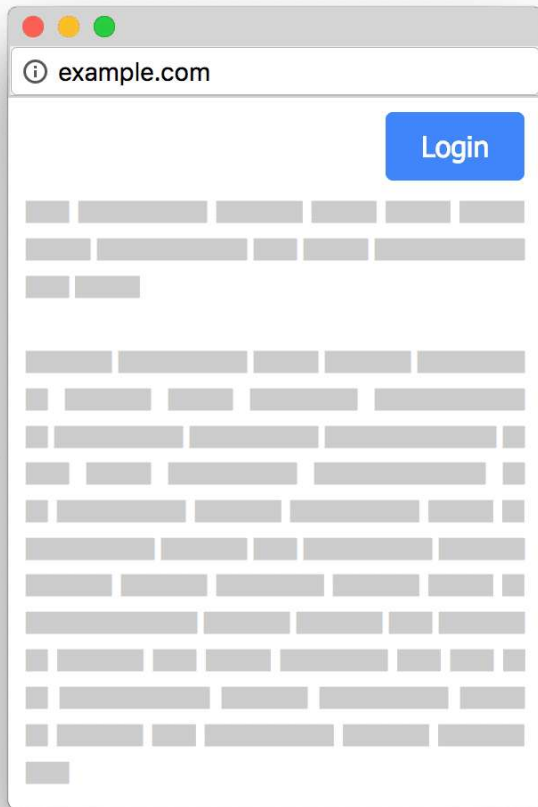
(https://web.dev/learn/forms/autofill#help_users_fill_in_their_credit_card_information) are present *only* on secure origins.

This means that the top-level page must be HTTPS and, if the `input` is in an `iframe`, that `iframe` must also be served over HTTPS.

Warning: It is **NOT** sufficient to place an HTTPS `iframe` inside a HTTP page; the top-level page itself must be HTTPS as well.

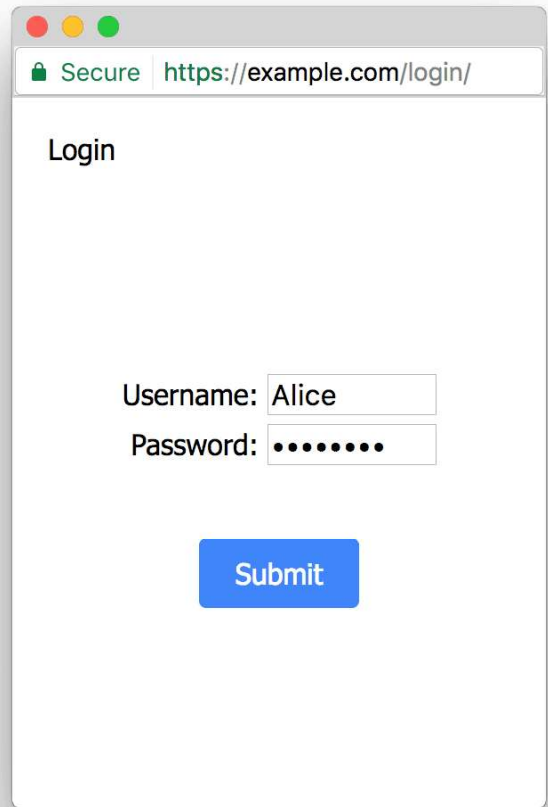
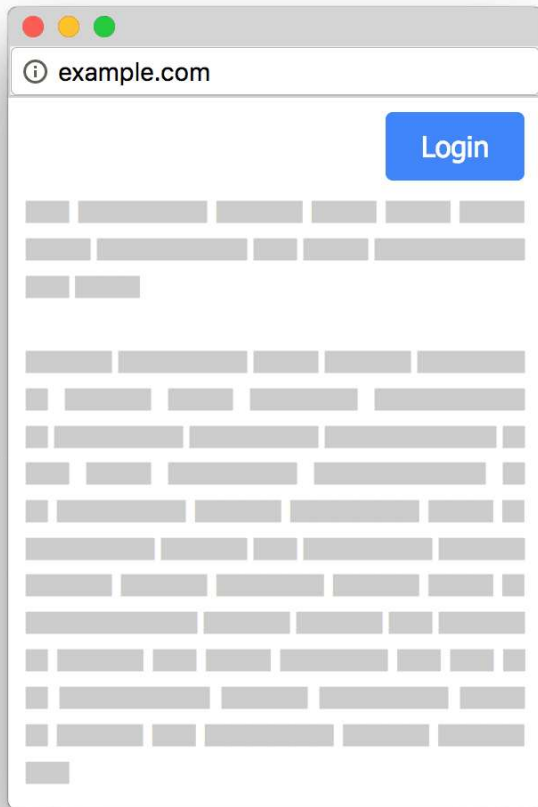
If your site overlays an HTTPS login frame over HTTP pages...

Non-secure login forms trigger the new **Not Secure** UI treatment.



You will need to change the site to either use HTTPS for the entire site (ideal) or redirect the browser window to an HTTPS page containing the login form:

Instead, prefer secure login forms.



Long term - Use HTTPS everywhere

Eventually, Chrome will show a Not Secure warning for *all* pages served over HTTP, regardless of whether or not the page contains sensitive input fields. Even if you adopt one of the more targeted resolutions above, you should plan to migrate your site to use HTTPS for all pages.

Except as otherwise noted, the content of this page is licensed under the [Creative Commons Attribution 4.0 License](https://creativecommons.org/licenses/by/4.0/) (<https://creativecommons.org/licenses/by/4.0/>), and code samples are licensed under the [Apache 2.0 License](https://www.apache.org/licenses/LICENSE-2.0) (<https://www.apache.org/licenses/LICENSE-2.0>). For details, see the [Google Developers Site Policies](https://developers.google.com/site-policies) (<https://developers.google.com/site-policies>). Java is a registered trademark of Oracle and/or its affiliates.

Last updated 2016-10-25 UTC.