PassSec+ 2.0 – An add-on that protects your passwords, payment data and privacy

Maxime Veit
Karlsruhe Institute of Technology

Melanie Volkamer
Karlsruhe Institute of Technology

Abstract

Entering sensitive data on websites can pose serious security and privacy risks. To mitigate these risks, nowadays, users need to carefully check the information provided in the address bar before entering sensitive data. However, many users are not aware of these checks. Even if they are aware they may judge untrustworthy websites as trustworthy ones, especially in the case of phishing websites that copy the content of the website and choose a URL that looks similar or the same as the authentic one. To support users in detecting untrustworthy websites more efficient and more effective we developed the PassSec+ concept and a corresponding browser add-on in 2015. We recently revisited this approach and noticed some shortcomings. To address these, we adopted the logic as well as the user interfaces.

1 Introduction

There are many security and privacy risks that can occur when entering sensitive data (e.g. passwords or bank details) on a untrustworthy website. Therefore, it is essential to check the information in the address bar of the web browser, i.e. to check the URL as well as whether HTTPS is in place and that there is no issue with the certificate. However, entering sensitive data on websites is part of our everyday life. Thus, if we want to mitigate corresponding risks, we spend a lot of time just checking information in address bars.

In reality, many people are not aware that they should check the address bar or do not know how to check it properly. Even those who are aware and have the knowledge are often too much in a hurry that they do not properly check or simply forget because the relevant information is displayed in the address bar of the web browser and not where there focus is. Furthermore, small changes to the URL are difficult to detect without carefully checking it (e.g. *gooogle.com/login* instead of *google.com/login* can easily be overlooked).

To support users in reducing the risk, we proposed in 2015 the PassSec+ concept [2] and an implementation as browser addon for Firefox and Google Chrome. PassSec+ not only secures the password, as the name suggests, but also credit-card information and personal information. The approach is pretty much inline with the findings in [1], i.e. security interventions are most effective if they appear just-in-time and -place.

Therefore, we re-visited the concept and noticed a number of shortcomings which we address with the new version, presented in this abstract:

- The concept considered only websites as low risk if they have an extended validation certificate. However, extended validation certificates have lost their attraction as web browsers do not highlight the address bar in green anymore. Correspondingly, it is less widely used.
- It can be potentially risky to enter sensitive information on a website although the information in the address bar looks o.k.:
  - The destination to which the data is sent and how it is transmitted can be an issue, too, i.e. data is send unencrypted and/or to a third party website, which may pose user data on serious risks. In such a situation, the previous PassSec+ would have even indicated that the risk is low (because of the information displayed in the address bar), while it actually is not.
  - In some browsers, internationalized domain names can also be used as domains to trick the user by displaying a domain name that looks the same as a

USENIX Symposium on Usable Privacy and Security (SOUPS) 2022.
August 7–9, 2022, Boston, MA, United States.

known domain, but just uses similar-looking characters of a different language (e.g. yahoo.com instead of yahoo.com). This was not considered but is an issue at least with Firefox.

- Personalized icons were used for the low risk cases as well as for the unknown risk case, in order to make it more difficult for attackers. However, it was perceived confusing that the icons were not in the same color as the border of the input fields.
- There is the risk of users accidentally accepting some unknown or high risk cases, as no delay was implemented to make it more likely that users carefully check the situation before deciding.

To address all these issues, we adopted the logic and extended the checks applied before deciding about the risk level communicated to the user. To take a systematic approach on the possible risks in this context, we considered the different cases from the BadSSL website[1]. We also improved the user interfaces to be more consistent and avoid habituation effects. Interfaces for additional cases were developed as well.

2 Overview of PassSec+

In this section, we provide an overview of the general idea of PassSec+ including the proposed modifications and extensions.

Figure 1: Input field indicates high risk

PassSec+ supports users in deciding if entering data on a website is with low or high risk, or with unknown risk (at least unknown to PassSec+, thus users need to reveal the risk level on their own). The information about the risk level are indicated to users in the corresponding input field by a corresponding color: For low risk cases green and blue is used (see Figure 2 for an example), for unknown risk cases grey is used and for high risk cases red is used. For the unknown risk and low risk, a security icon in the corresponding color is shown inside the input field. The security icon is only known by the user and PassSec+ and is set when PassSec+ is installed. This way a malicious website cannot easily spoof the security classification of PassSec+ as the attacker does not know which security icon is used. For the high risk case a red exclamation mark is shown (see Figure 1).

Figure 2: Input field if website is on allowlist by user.

In case of a unknown or high risk an active intervention (see Figure 3 for an example) is displayed below the input field as soon as the focus is put on this field, i.e. the user could potentially input sensitive data. The input is disabled for three seconds, which can be changed by the user in the settings.

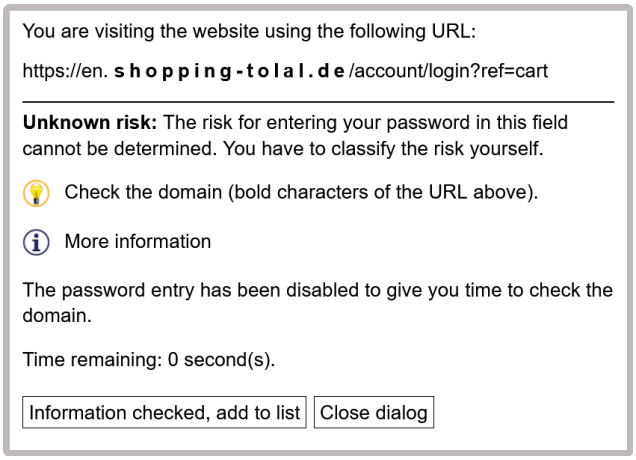

Figure 3: Example of the grey security intervention which asks to check the domain if it was not checked before already.

PassSec+ uses different types of information to judge the risk for the input fields on a website: e.g. the actual (top-level) domain of the URL and whether non-ASCII characters are used there; the default-allowlist[2]; users history, i.e. webpages users consider as low risk.

Furthermore, PassSec+ supports users by highlighting the domain part of the URL in the active security intervention. This enables users to focus on the important part of the URL when checking it. Also, the domain is displayed with extra character spacing to allow revealing typos (e.g. *arnazon.com* instead of *amazon.com*) as it is done in [3]. If there are non-ASCII characters in the domain, then the domain is displayed in Punycode[3] to prevent IDN homographic attacks.

[1] https://badssl.com/ It lists all possible cases with respect to HTTPS/HTTP (e.g. mixed contend and expired certificates) one could observe when visiting websites on the Internet. When clicking on their examples one sees in which cases, the webbrowser would use a passive indicator (e.g. lock icon) and in which an active intervention is used. Our focus is on those cases in which only a passive indicator is used (be it a positive one or a negative one indicating potential risks).

[2] PassSec+ is published under GNU General Public License v3.0; the initial list can be customized by developers. In the version available in the Firefox and Google Chrome store it contains the websites from the Alexa Top 100 as well as German banking websites for historical reasons. See https://github.com/SecUSo/PassSec-plus/blob/master/js/default-preferences.js

[3] Punycode is a representation of internationalized domain names in

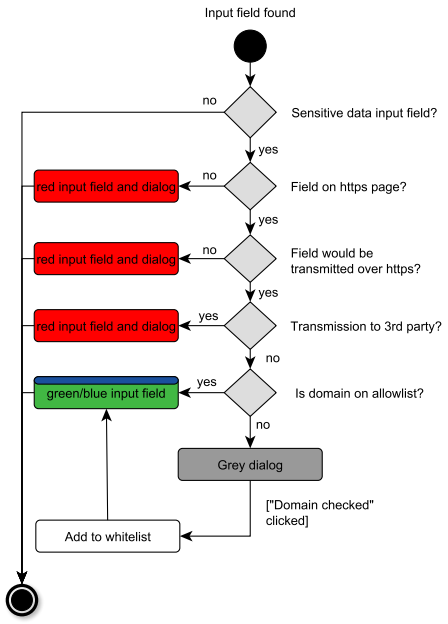

Figure 4: Simplified visualization of how PassSec+ proceeds for each input field on the accessed website.

3 Algorithm Details

In this section, we explain how the algorithm of PassSec+ works and which attacks are prevented that way. Figure 4 depicts a simplified description of how PassSec+ checks each input field when a website is accessed[4]. In the following paragraphs the various steps are explained:

First, PassSec+ checks once during page loading whether there are any input fields on the web page at all ("Sensitive data input field?"). To do so, it checks for password entry fields as well as for terms in the name attribute of the input fields (e.g. bank details, credit card number, address, birthday).

Then PassSec+ is checked if there is a secure connection to the Website ('Field on https page?'). Thereby, it checks whether the data of the website, which also covers the input fields, is actually from the server behind the URL and is not manipulated by an attacker (i.e. no active sniffing is in place). Note, in some cases (e.g. if a website is visited the first time) the attacker can actively downgrade the website to HTTP (SSL stripping). PassSec+ would notice this. In case the connection is not secure, PassSec+ classifies all input fields as high risk. However, PassSec+ checks whether the URL is available using HTTPS. If this is the case, in the active security intervention (which will only be displayed if the user

Figure 5: Dialog for high risk cases allows to switch to https if availible

clicks into one of the fields) PassSec+ will recommend the user to switch first to HTTPS (see Figure 5). Any other option on how to continue is disabled for three seconds. If the user follows the recommendation of switching to HTTPS once, this will be done automatically the next time for this website.[5]

In case, the connection to the webpage is secure, for each relevant input field, PassSec+ checks if the data would be transmitted securely, i.e. using HTTPS ('Field would be transmitted over https?') as well as to the same domain as of the URL the actual webpage is received from ('Transmission to 3rd party'). This is necessary to ensure, that (1) an attacker cannot read along the sensitive data transmitted (i.e. no passive sniffing is possible) and (2) to prevent Man-in-the-Middle attacks which are especially relevant when connected to public hotspots. If any of the two checks fails then the affected input field is highlighted in red (as PassSec+ classifies it as high risk). In both of these cases, an active security intervention appears when users click on the input field. The intervention explains what the problem is and recommends how to proceed. Entering data is disabled for a specific amount of time to give the user time to check the information given accurately.

If none of the above checks failed for any of the input fields, PassSec+ checks the trustworthiness of the URL, i.e. the trustworthiness of the domain. PassSec+ first checks whether the domain is on an allowlist ('Is domain on allowlist?'). The allowlist contains by default the Alexa Top 100 websites and the German banking websites for historical reasons (i.e. called default-allowlist). The allowlist can be extended by the user (this part is called user-allowlist). The default-allowlist can be customized by the developers.[2] If the website is on the allowlist the input field is shown in green or blue depending on which of the two sublist the domain is in. PassSec+ displays the two subcases in different colors to be more transparent to the user why the situation is considered as low risk.

ASCII characters (e.g. *xn–80a2aar51d.com* instead of yahoo.com). See https://tools.ietf.org/html/rfc3492

[4]For a more detailed flow chart we refer our readers to https://bwsyncandshare.kit.edu/s/XwJ3b7istX8gZja.

[5]This is similar to what the HTTPS Everywhere Addon has done in the past. See https://www.eff.org/https-everywhere

In case PassSec+ does not find the domain in the allowlist, it classifies it as unknown risk. As such the grey color is used for the input field. If the user clicks into such a field, the security intervention from Figure 3 is shown. A click on "Information checked, add to list" adds the domain to the user-allowlist.

4 Future Work

The focus of the future work is to improve PassSec+ against active attacks, to evaluate the usability, especially with respect to its effectiveness, and to extend its functionality with respect to webpages using JavaScript extensively.

It is challenging to show the security intervention in place so where the input field is and at the same time stand out from the website content to not get spoofed. Active attacks would try to actively change the website design to prevent PassSec+ somehow from working, i.e. while PassSec+ supports users in detecting Phishing pages as the input fields are likely to be grey and not green or blue, the Phisher would try to design the webpage in a way that the input fields would be green or blue as expected. This can be done e.g. by spoofing the dialog of PassSec+ (including the personal icon) and showing different information. This is possible, as the security interventions are integrated by PassSec+ in the actual website. It can also mean that the website tries to add itself to the allowlist by simulating a click on the button. As future work, we plan to change the implementation of the addon in a way that it displays the security intervention in an extra popup window or HTML inline frame, i.e. detached from the web page, to address the shortcomings just described.

For the evaluation of PassSec+, we plan for at least two user studies: (1) A field study to evaluate the general usability and user acceptance plus adoption. We plan to use a study protocol similar to the one in [2]. (2) The effectiveness is evaluated in a lab study. Here participants are asked to judge for each webpage how risky it is to enter sensetive information on the corresponding wegpage. In order to get a better understanding on how supportive PassSec+ and the fact that information is displayed just-in-time and place is, we aim for using eye-tracking in this study.

References

[1] Justin Petelka, Yixin Zou, and Florian Schaub. Put your warning where your link is: Improving and evaluating email phishing warnings. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, CHI '19, page 1–15, New York, NY, USA, 2019. Association for Computing Machinery.

[2] Melanie Volkamer, Karen Renaud, Gamze Canova, Benjamin Reinheimer, and Kristoffer Braun. Design and field evaluation of passsec: raising and sustaining web surfer risk awareness. In *International Conference on Trust and Trustworthy Computing*, pages 104–122. Springer, 2015.

[3] Melanie Volkamer, Karen Renaud, and Benjamin Reinheimer. Torpedo: tooltip-powered phishing email detection. In *IFIP International Conference on ICT Systems Security and Privacy Protection*, pages 161–175. Springer, 2016.