Assessing Real-World Applicability of Redesigned Developer Documentation for Certificate Validation Errors

Martin Ukrop
Masaryk University
Centre for Research on Cryptography and Security
Brno, Czech Republic
mukrop@mail.muni.cz

Michaela Balážová
Masaryk University
Centre for Research on Cryptography and Security
Brno, Czech Republic
433706@mail.muni.cz

Pavol Žáčik
Masaryk University
Centre for Research on Cryptography and Security
Brno, Czech Republic
pzacik@mail.muni.cz

Eric Vincent Valčík
Masaryk University
Centre for Research on Cryptography and Security
Brno, Czech Republic
492944@mail.muni.cz

Vashek Matyas
Masaryk University
Centre for Research on Cryptography and Security
Brno, Czech Republic
matyas@fi.muni.cz

ABSTRACT

We face certificate validation errors commonly, yet the related tools and documentation had been shown to have very poor usability. Previous research suggests that just improving the error messages and corresponding documentation can have significantly positive effects. Our work aims at increasing the usability of certificate validation by 1) redesigning the API error messages and the corresponding documentation, and 2) validating the real-world applicability of the redesign by investigating the opinions of 180 IT professionals. We focus on the perceived obstacles, desired ideal form and overall satisfaction. The redesigned documentation exhibits a reliable significant decrease in perceived incompleteness, with a small amount of perceived bloat and tangle. The redesigned documentation, now published on a dedicated website, is preferred by 89% of our study participants.

CCS CONCEPTS

• **Security and privacy → Usability in security and privacy**; • **Software and its engineering → Documentation**; • **General and reference** → Validation.

KEYWORDS

documentation, TLS certificate, usable security, warning design

ACM Reference Format:
Martin Ukrop, Michaela Balážová, Pavol Žáčik, Eric Vincent Valčík, and Vashek Matyas. 2022. Assessing Real-World Applicability of Redesigned Developer Documentation for Certificate Validation Errors. In *2022 European Symposium on Usable Security (EuroUSEC 2022), September 29–30, 2022, Karlsruhe, Germany.* ACM, New York, NY, USA, 14 pages. https://doi.org/10.1145/3549015.3554296

EuroUSEC 2022, September 29–30, 2022, Karlsruhe, Germany

1 INTRODUCTION

Transport layer security (TLS) is a widely deployed protocol suite: In 2020, over 85% of page loads on the Internet were over TLS [13]. Within the protocol, server authentication is the only protection against the man-in-the-middle attack and other server attacks. Server authentication in TLS depends typically on a single step of validating its X.509 certificate [10].

Nevertheless, failing to validate certificates properly is quite common [4, 5]. Writing TLS-related code is notoriously problematic as lots of things can go wrong [9]. The relevant tools have poor usability [32], often leading to insecure code as demonstrated by Georgiev et al. [12] after inspecting TLS-related code in many non-browser applications. Finding the certificate validation often misconfigured or completely avoided, the authors nicknamed the non-browser code validating TLS certificates as "the most dangerous code in the world." Thus, it is crucial to understand what individual validation errors mean and how severe they are to differentiate harmless errors from malicious attacks.

Most previous work on certificate validation errors focused on perceptions of end users in Internet browsers [6, 11, 28]. However large in number, the mistakes of end users mostly affect just themselves. Our focus goes past the browser GUI to the underlying security libraries and past end users to IT professionals.[1] Consider mobile applications making TLS connections as an example: Testers or DevOps engineers (who come across certificate validation errors for example in the logs) need to understand them to evaluate the errors' severity and propose the necessary fixes. Apart from the error message in the log, they usually consult the official documentation or browse the Internet. On the one hand, as of 2021, the documentation of the certificate validation errors was extremely poor – the median length of an error message in commonly used libraries was just six words, with the official documentation having on average only thirteen words [30]. On the other hand, using informal online resources instead of the documentation was shown to lead to less secure code [2].

[1] This includes people responsible for development, testing, deployment and administration alike since the same errors are encountered by all.

There have already been some academic attempts to solve similar issues. Gorski et al. [15] investigated the potential of API-integrated security advice and Meng et al. [20] looked into adjusting the documentation content and structure along the published usability guidelines. We aim to test the real-world applicability of more usable certificate validation documentation, redesigned following existing guidelines. The closest to our setup is a study by Ukrop et al. [31] demonstrating that just adjusting the existing command line interface (CLI) error messages and the corresponding documentation can significantly positively affect the developer experience.

As our main contribution, we validate the real-world applicability of three redesigned certificate validation messages (and their corresponding documentation). Note that our study does not aim to yield new guidelines or compare the effects of the existing ones, but focuses on real-world applicability of existing academic results. We do this by investigating the opinions of 180 IT professionals seeing the original documentation and the redesign, guided by the following three research questions (RQs):

(1) *Perceived Obstacles.* What are the perceived obstacles of the current documentation? Can we mitigate them? Does the redesigned documentation introduce other obstacles?
(2) *Ideal Form.* What length would the ideal documentation have? How would it be structured?
(3) *Overall Opinions.* Does the new documentation cause a better understanding? Do IT professionals find it more satisfying and helpful? Is it preferred over the original?

Firstly, we discuss the related work on the certificate ecosystem and usable documentation (Section 2). The detailed study design and analysis of the participant sample are described in Section 3 (the full questionnaire attached in Appendices B and A). Comparing the obstacles IT professionals perceive, we show a decrease of incompleteness and a small increase of bloat and tangle for the redesigned documentation and significantly increased self-reported satisfaction. Asking about the ideal form and structure, the redesign also fares well. In general, the redesign seems nicely applicable (detailed results are available in Section 4). The section ends with the description of deployment of the new documentation prototypes, along with artifacts and knowledge acquired in the process.[2] Previous experience, order effects and other aspects of study validity are discussed in Section 5, before concluding the paper.

2 RELATED WORK

Firstly, we review works on the certificate ecosystem, followed by the research attempts to improve it. Lastly, we summarize relevant guidelines for writing better error messages and documentation.

2.1 The World of X.509 Certificates

The ecosystem of X.509 certificates and related protocols [10] is notoriously complicated. Clark et al. [9] give a good overview of things that can go wrong in TLS/HTTPS, from as technical as certificate parsing errors, through subject name manipulation attacks to as institutional as trust anchoring and certificate authority compromise.

[2]See x509errors.org. The anonymized study data are available from crocs.fi.muni.cz/papers/eurousec2022.

Unfortunately, the developer tools in this ecosystem were also found to have poor usability – Ukrop et al. [32] investigated the usability of OpenSSL as the most widely used certificate-manipulating library. Tools and interfaces with bad usability can lead to insecure code: Georgiev et al. [12] inspected TLS code in many non-browser applications, finding the certificate validation often misconfigured or completely avoided.

To combat the poor usability, IT professionals resort to formal resources (documentation) as well as informal ones (forums and tutorials). As researched by Acar et al. [1, 2], turning to online forums and tutorials often helps IT professionals get the code functional but not necessarily secure. Such research hints that accessible official documentation with security information may be of crucial importance.

However, the current documentation of certificate validation errors is too short. The survey of OpenSSL, GnuTLS, Botan, mbedTLS and Microsoft CryptoAPI by Ukrop et al. [30] shows neither of these libraries has a median length of the certificate validation error message over eight words. The corresponding section in the documentation (if it exists) is only a tad longer, with the median of eight (mbedTLS), nine (OpenSSL, GnuTLS) or sixteen (MS CryptoAPI) words.

2.2 Attempting a Change

Adjusting the certificate warnings (content, form, accompanying explanations) has already been researched, but almost exclusively in browsers focusing on the end users [6, 11, 28]. Only little similar research has been done concerning IT professionals, although the importance of clean and consistent error reporting is vital [12, 14].

Some research has been conducted to investigate security warnings related to API usage. In 2018, Gorski et al. [15] studied the effectiveness of API-integrated security advice. They found that such an approach significantly improves code security, with 73% of the participants fixing their insecure code after getting the advice. Nevertheless, the proposed system would not scale easily as it requires custom-tailored patches for each API. A later study by Meng et al. [20] investigated the effects of improved documentation on API usage. In a comparative study, half of the participants interacted with documentation optimized following the design guidelines proposed in the literature, with the other half seeing only the non-optimized version. Results favor the optimized version with fewer errors made and higher speed of planning and executing the tasks.

A similar study on the effects of redesigned documentation was performed in 2019 by Ukrop et al. [31]. 75 attendees of an industrial conference interacted with certificate validation error messages and documentation in either original or redesigned version, focusing on the perceived trust in differently flawed certificates. Results show that even small changes in existing error messages and documentation can positively influence resource use, comprehension and trust assessment. Although being an inspiration to our study, we focus on the perceived documentation usability.

2.3 Usable Documentation

Uddin and Robillard had conducted a survey [29] to investigate common flaws in API documentation in general. In the first phase,

they categorized 79 documentation flaw examples from 69 respondents into ten common obstacles: content-related (incompleteness, ambiguity, unexplained examples, obsoleteness, inconsistency, incorrectness) and presentation-related (bloat, fragmentation, excess structural information, tangled information). The subsequent validation study with 254 participants shows content issues are seen as more important than presentation. The most frequent obstacles were incompleteness and ambiguity, which perfectly aligns with our study results (see Section 4.1).

Focusing more specifically on warning message design, multiple works have attempted to create good-practice guidelines (often specialized to security-related cases). We provide a summary of points relevant to our study below.

- *Decision description.* The warning message should clearly state what the decision to be made is. It should give relevant information (and recommend the safest option). [23]
- *Risk description.* The warning message should clearly explain the risk involved in the decision and the consequences of not complying. [7, 8, 23, 33]
- *Next steps.* The warning messages should be actionable, containing specific steps the user can follow. [7, 8, 23, 33]
- *Brevity.* The warning messages should be kept brief but accurate, as more people read shorter texts. [7, 8, 18]
- *Structure.* All text should be clearly structured, as such warnings are more effective and can maintain attention longer than other formats. [18, 33]
- *Language.* The warning messages should use short sentences and simple grammar to ease the understanding for native as well as non-native speakers. [17]

3 STUDY DESIGN

The study had the form of a questionnaire comparing original and redesigned documentation (study overview in Figure 1). It had a mixed design: 1) There were two versions of the documentation (original and redesigned, in this order) shown to all participants (the within-subjects factor). 2) The task was set up for three different errors for higher validity (each participant was randomly assigned to one error, the between-subjects factor). These were an expired certificate, a certificate with a hostname mismatch and a certificate with an unhandled critical extension.

3.1 Evaluation Questionnaire

The questionnaire had four main parts: 1) Evaluation of the original documentation, 2) evaluation of the redesigned documentation, 3) the length and structure of the ideal documentation with a discussion of the overall preference between these two versions and 4) general questions on previous experience and demographics. The full questionnaire is available in Appendix A.

Both parts for the evaluation of documentation (original and redesigned) started with the text of the documentation itself, followed by questions on perceived understanding (four-point scale from *'Yes'* to *'No'*), satisfaction and helpfulness (five-point scale from *'Extremely'* to *'Not at all'*). After that came a list of possible obstacles asking for their presence (four-point scale from *'Yes'* to *'No'*). For every obstacle, if the participant noted it as present, a free-text question asking for their reasoning appeared. The list of obstacles was based on the survey of content and presentation flaws in API documentation by Uddin and Robillard [29]. As their work was aimed at APIs, not all defined flaws were relevant to us. We kept the following flaws, along with their definitions, taken from the original paper (only the definition of incompleteness was modified to fit our setting better). We omitted unexplained examples, obsoleteness, fragmentation and excess structural information.

- *Incompleteness.* Some information is missing in the documentation. *(our definition)*
- *Ambiguity.* The description was mostly complete but unclear.
- *Inconsistency.* The documentation of elements meant to be combined did not agree.
- *Incorrectness.* Some information was incorrect.
- *Bloat.* The description was verbose or excessively extensive.
- *Tangle.* The description was tangled with information the respondent did not need.

Afterward, the participants were asked about the importance and possible omission of individual parts of the documentation and their overall preference. This was followed by asking for the ideal length (number of lines, reminding the number of lines of our documentation for comparison). Next, there were questions on gender, student status, achieved formal IT-related education, length of their IT-related employment, current position and country of work. The final questions asked for the self-reported knowledge of computer security and X.509 certificates (five-point scale from *'Excellent'* to *'Poor'*) and having used OpenSSL before (four-point scale from *'More than five times'* to *'Never'*).

The collected opinions on documentation quality are only self-reported. This limitation, however, is challenging to overcome. A real programming scenario would lengthen the study considerably and disallow us to have the sample as large and relevant (180 actual IT professionals, not convenience samples such as students). A differently-focused study investigating IT professionals' actions is a logical future work after this study.

3.2 Original and Redesigned Documentation

For the original documentation, we decided for the contemporary version of OpenSSL (January 2020), as it seems to be (by far) the most used cryptographic library [22]. Thus, its improvement would have the biggest impact on real-world security.

To have a more representative sample, we included errors of different types. We chose an expired certificate (a simple and common case, X509_V_Err_Cert_Has_Expired), a certificate with a hostname mismatch (a more complicated but still common case, X509_V_Err_Hostname_Mismatch) and a certificate with an unhandled critical extension (a complicated and uncommon case, X509_V_Err_Unhandled_Critical_Extension).[3] The difference of cases is also supported by previous research [4, 5], showing about 4–7% are caused by expired certificates and approximately 10–18% of the certificate errors happen due to a name mismatch (unhandled critical extension is never mentioned in the statistics of the top-occurring errors).

[3]Results for the individual error cases are always presented in this order. Case names are shortened to *expired, hostname mismatch* and *unhandled extension* for brevity.

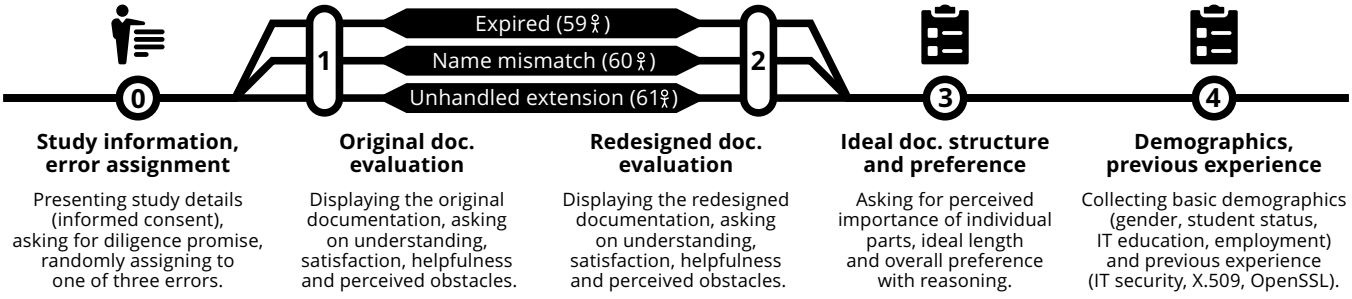

Figure 1: Overview of the experimental procedure. 180 participants evaluated the original and redesigned documentation (within-subjects factor), split into three cases with different certificate errors (between-subjects factor).

The redesigned documentation for the errors was created based on the recommendations for error messages listed in Section 2.3, much inspired by the documentation by Ukrop et al. [31] and consulted between a developer and a security researcher. The final design consists of the following parts (the full documentation available in Appendix B):

- The error code name as a header.
- A single-line description of the problem, also intended to be displayed as the error message directly in the command line when the error occurs.
- Four paragraphs with further details: *explanation*, *security perspective*, *what to do* and *consequences* with each describing the particular aspect of the error.

While the original documentation had only a single line in all three error cases, the redesigned documentation was 27 lines long for the expired certificate error and 23 lines for the other two cases.

In the study, the order of the displayed documentation designs was not counterbalanced: As the redesigned documentation contains more detailed information than the original, seeing the redesign first would significantly disadvantage the original documentation. Nevertheless, to investigate the possible order effects, we conducted a second auxiliary study described in Section 5.1.

3.3 Conducting the Study

The study took place in January 2020 at DevConf.CZ conference.[4] From our booth, we encouraged conference participants to join research on the usability of error documentation. The survey, held in English, was introduced as aiming to "improve documentation" for certificate errors without mentioning OpenSSL or the fact that one design is current and the other is newly created for the study to avoid demand effects. For completion, participants received merchandise (i.e., no monetary compensation).

Multiple privacy-preserving precautions following our institution's ethical research principles were made: the survey was anonymous and hosted at our own instance of the LimeSurvey server, accessible on a trusted university domain. All questions were voluntary. The first page of the survey contained the informed consent stating the organizing institution and explaining the voluntary nature of the questions and intended data use.

To ensure well-phrased questions, three rounds of pilot testing with 28 people were performed. To mitigate bias in the quantitative analyses, the data was cleaned beforehand. From the 220 participants opening the survey, we dropped participants not finishing all screens and having nonsensical answers (huge number of lines for the ideal documentation, specifically 4558 and 1337).

The free-text answers (obstacle and documentation preference reasoning) were analyzed using the qualitative content analysis [25]. Firstly, the principal researcher familiarized with the data and processed it using open coding [24] (looking for reoccurring themes and assigning codes to the individual answers). The analysis was broadly framed by the frequent documentation flaws identified in the previous work [29]. After creating a stable codebook, the second researcher re-coded the data. Thereafter, both researchers resolved all the differences, fully agreeing on the final coding. In line with contemporary human factors research, we therefore omit inter-coder agreement calculations [19].

3.4 Participant Demographics

In total, we had 180 participants who filled in the whole questionnaire. Participants in the survey comprised 86% (154 [5]) of men, 12% (21) of women and 2% (4) of other gender. Only 15% (27) were active students of an IT-related discipline at the time of the study. The largest part of the participants (45%, 80) reached a Master's degree in an IT-related discipline, with 30% (53) having a Bachelor's degree, 2% (3) a postgraduate degree and the rest (24%, 43) not having a formal education in IT.

Almost half of the respondents (49%, 89) worked as developers or software engineers, 8% (14) were employed as testers or quality assurance engineers. The third most numerous category was managerial positions (7%, 13). On average, respondents were employed in an IT-related field for 9.6 ± 6.8 years (median 8) with the minimum being no employment and the maximum as high as 31 years. The study participants were work-based on all continents (excl. Antarctica), with the majority (82%, 147) based in Europe. Most of the participants were from the Czech Republic (32%, 57), followed by Polish participants (16%, 29) and participants from India (11%, 19).

[4] DevConf.CZ is an international community conference for developers, admins, DevOps engineers, testers and others interested in open source technologies. The conference is organized by Red Hat, Inc. and had 1600 attendees in 2020.

[5] From now on, we use the symbol to denote the participants.

The median self-reported knowledge of computer security in general was *'good'*, with the minimum being *'poor'* (3 🧍) and maximum *'excellent'* (10 🧍). The median knowledge of X.509 certificates was lower, only *'fair'* with the same range but shifted towards less experience (49 🧍 *'poor'* and only 3 🧍 *'excellent'*). With respect to previous OpenSSL use, the participant sample was quite experienced: Almost two-thirds (63%, 113 🧍) had used the OpenSSL library more than five times before, almost a quarter (23%, 42 🧍) two to five times, 8% (14 🧍) only once and 6% (11 🧍) had never used OpenSSL before.

To determine if our participant sample does not significantly deviate from the global developer population, we compare its characteristics with the 2020 Stack Overflow Developer Survey [27]. In the survey by Stack Overflow, 8% of the participants were female (12% in our study), with 12% being students (15% in our study). The global developer population seems a bit less formally educated in IT, with the most participants (46%) having a Bachelor's degree (the most frequent in our study was a Master's degree with 45%). Concerning professional experience, the samples are comparable with the median between five and nine years (median 8 in our study). Apart from the major base in Europe, the global population has about 20% people in the USA (only 6% on our study) and 13% in India (11% in our study). In summary, our participant sample seems to be reasonably representative of the global developer population, although maybe a bit more formally educated and a bit more geographically biased towards Europe.

Participants were split almost evenly into three error cases of different types: 59 🧍for the expired case, 60 🧍for the name mismatch case and 61 🧍for the unhandled critical extension case. The appropriate statistical tests show no significant differences among the sub-samples with respect to gender, student status, achieved IT education, years of IT employment and previous OpenSSL usage. Significant differences were present only for self-reported IT security knowledge and X.509 knowledge, with the expired error sub-sample being a bit more knowledgeable (median *'Very good'* computer security knowledge compared to *'Good'* for other errors but the same median of *'Fair'* X.509 certificate knowledge for all errors). In summary, the error sub-samples can be considered mostly comparable.

4 VALIDATION STUDY RESULTS

This section presents study results, organized by the three research questions covering perceived obstacles, ideal form and overall opinions. Note that as all survey questions were optional, the base number of responses may sometimes be slightly lower than 180 🧍. The section ends with the description of results deployment.

4.1 RQ1: Perceived Obstacles

The following paragraphs will discuss results organized by obstacles. Quantification of perceived obstacles is summarized in Figure 2 (answers to questions 5 and 15 of the survey, see Appendix A). Next, Figure 3 summarizes the qualitative content analysis of the obstacles reasoning. It is based on free-text answers coded by two independent researchers and reported separately for original and redesigned documentation. Codes are provided with a simplified definition and an example quotation.

Incompleteness & Ambiguity. These were the obstacles perceived as most severe in the original documentation: 63% of the participants (113 🧍) answered the question if they found the documentation incomplete with *'Yes'* or *'Rather yes'*. For ambiguity, this was 36% (65 🧍). The redesigned documentation shows a statistically significant decrease for both obstacles with 6% (11 🧍) and 5% (9 🧍) (Wilcoxon signed-rank test; for incompleteness $z = -10.13$, $p < 0.001$ with the median changing from *'Rather yes'* to *'No'*; for ambiguity $z = -7.74$, $p < 0.001$ with the median changing from *'Rather no'* to *'No'*). The most prominent deficiency was the lack of accuracy (the code Accuracy, 35 🧍for original documentation / 5 🧍for the redesigned) and desire for more information (LittleInfo, 28/1 🧍). Looking into repeating patterns, we found people notably lacking explanations (Explanation, 22/2 🧍), description of the error cause (ErrorCause, 20/1 🧍) and guidance on the next steps to perform (NextSteps, 19/2 🧍). Less prominent was a call for concrete examples (Examples, 5/0 🧍). All these codes were much less present in the redesigned documentation – supporting the decrease of perceived severity seen in the quantitative evaluation. As illustrated by the following quote, the content analysis also emphasizes that IT professionals want to see the precise details to pinpoint the error quickly (consistent with findings in the related work [31]).

> *"As a developer, I want to know specifically which fields are being compared (CN and SAN) [...]"* [P104, hostname mismatch, redesigned]

Inconsistency & Incorrectness. Neither of these obstacles was much present in either the original or redesigned documentation. Nevertheless, the decrease of perceived inconsistency for the redesigned documentation is significant (Wilcoxon signed-rank, $z = -4.37$, $p < 0.001$ with the median *'No'* in both conditions). There were almost no codes related to these issues apart from the Incorrect, (0/4 🧍) where participants pointed out a minor imprecision in our documentation (saying *servers* instead of *hostnames*).

Bloat. This was the most severely perceived obstacle for the redesigned documentation (20%, 36 🧍answering *'Yes'* or *'Rather yes'*). The increase compared to the original documentation (8%, 14 🧍) is significant (Wilcoxon signed-rank test, $z = 5.19$, $p < 0.001$ with the median changing from *'No'* to *'Rather no'*). Two codes represented this issue: One mentioning repeated information (Repeating, 17/3 🧍) that was mainly present in the original documentation (note that, although short, the original documentation often has the documentation very similar to error code, see Appendix B). The other is the documentation containing unnecessary pieces of information (Unnecessary, 5/35 🧍). A major increase in this code for the redesigned documentation is in line with the perceived increased bloat.

Tangle. Also being quite prominent in the redesigned documentation, 18% of the participants (32 🧍) answered *'Yes'* or *'Rather yes'* when asked if they found it tangled. The increase compared to the original documentation (10%, 18 🧍) was significant (Wilcoxon signed-rank test, $z = 2.23$, $p = 0.02$ with the median *'No'* in both conditions). Deficiencies were seen in structure (Form, 3/7 🧍) and slightly increased after the redesign. This was mainly due to the new structure seeming excessive to some, although opinions were often contradictory (see below). The documentation structure is discussed in more detail later, in Section 4.2. Apart from structure, the

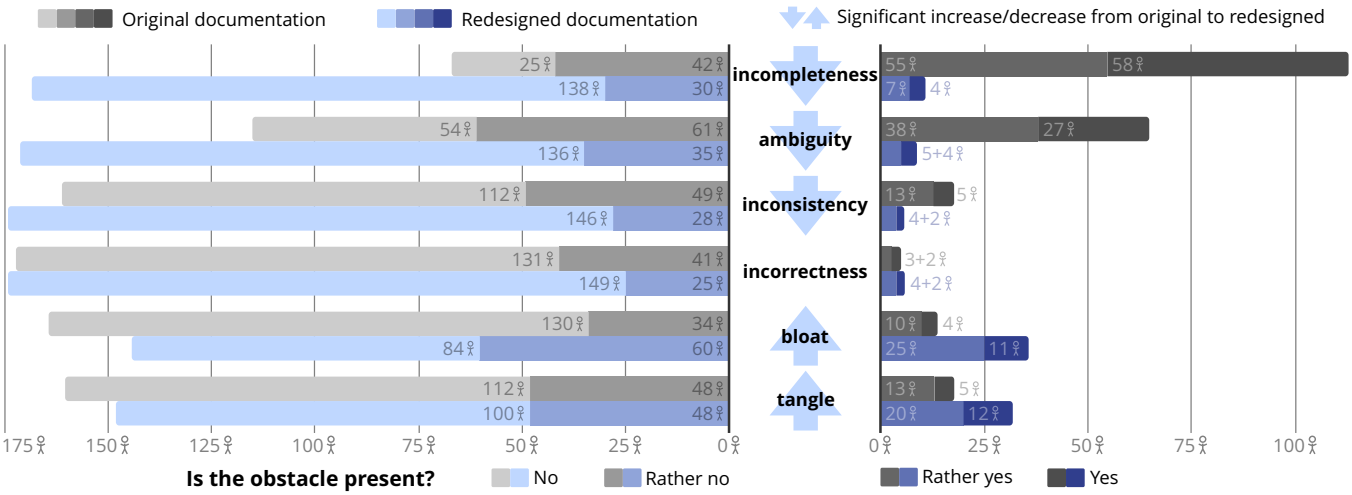

Figure 2: Comparing the perceived severity of the six selected obstacles between the original and redesigned documentation, showing a significant decrease in incompleteness, ambiguity and inconsistency and a significant increase in bloat and tangle.

reasoning analysis shows deficiencies in linguistic clarity (Clarity, 15/8 ♀) in both style and quality.

> *"This [documentation] doesn't sound like it was written by a native English technical writer."* [P104, hostname mismatch, redesigned]

General Observations. The qualitative analysis of obstacle reasoning showed that participant opinions were sometimes inconsistent. The following two examples show contradictory thoughts regarding length and the description of the security perspective.

> *"I think that the more information, the better. Skipping irrelevant parts isn't difficult, and when errors occur, I always want to have as much information as possible. [...]"* [P211, hostname mismatch, redesigned]
> *"The explanation can be more minimalistic."* [P69, unhandled critical extension, redesigned]

> *"[...] If I am an everyday person or junior, the "Security perspective" section is most important and [...]"* [P72, hostname mismatch, redesigned]
> *"I don't really care about the security perspective."* [P209, hostname mismatch, redesigned]

In both conditions, multiple participants had specific suggestions for improvement (Suggestion, 6/11 ♀). These included concrete text reformulations, formatting tips, adding links or other adjustments:

> *"The context of what a SAN and CN are for could be replaced with a link to the x509 specification."* [P104, hostname mismatch, redesigned]
> *"For example, this whole sentence can be removed without losing information: 'It is also this case – the certificate was issued to the subject specified in the certificate.'"* [P223, hostname mismatch, redesigned]

Results Summary for RQ1. Overall, the perceived severity of flaws in the original documentation is in line with the previous work [29] with incompleteness and ambiguity seen as by far the most severe. Both were significantly less present in the redesigned documentation. On the other hand, the redesigned documentation shows a statistically significant increase for bloat and tangle, although much less prominent than the severity of the original documentation issues.

4.2 RQ2: Ideal Form

Multiple participants (28 ♀) skipped the questions on the desired length of the documentation (this is not surprising as imagining/estimating documentation without specific content is quite vague). Nevertheless, we may gain some information from the relative result: 57% of the answering participants (87 ♀) wanted shorter documentation than the displayed redesign, 27% (41 ♀) desired the same length and 16% (24 ♀) wanted even longer. The median desired length was 20 lines – only slightly shorter than our redesigned documentation (23 and 27 lines). Some developers mentioned the length is irrelevant provided the structure is sound.

> *"I don't think the number of lines is important. But it has to be structured well."* [P122, hostname mismatch, redesigned]

Asking about the importance of the specific parts of the redesigned documentation revealed that all used sections were considered important (see Figure 4, medians *'Extremely important'* for the *error code name* and *the short description* and *'Very important'* for the rest). Nevertheless, the importance perception was significantly different among the parts (related-samples Friedman's two-way analysis of variance by ranks, $\chi^2(5) = 82.03$, $p < 0.001$). Post hoc analysis (pairwise comparisons with Bonferroni corrections) shows significant differences for *error code name* and the *short description* to all other parts ($p < 0.010$).

When asked if the participants would omit any part, 67% (116 ♀) would omit none (see Figure 4 on the right side). However, omission want was significantly different for different parts (Cochran's Q, $\chi^2(6) = 340.07$, $p < 0.001$), with pairwise (Bonferroni-corrected) significant differences between wanting to omit no part and wanting to omit any other part (66% vs. less than 18%) and between wanting to omit *consequences* compared to the *error code name* (18% vs. 2%) and the *short description* (18% vs. 2%). The two parts most often mentioned for omission were *consequences* (18%, 31 ♀) and *security perspective* (11%, 19 ♀). These were also mentioned to be related and, therefore, it may be worth combining them.

Code	Short definition	*Example quote*
Accuracy	The documentation should be more accurate.	*"I don't know which hostnames are being compared. Is it the SAN or the CN?"* [P104, hostname mismatch, original]
LittleInfo	Want more details, but not specifying what exactly.	*"More details should be provided"* [P131, expired, original] *"In fact the documentation is not present at all :D"* [P188, name mism., orig.]
Explanation	The doc. lacks definition, explanation or context.	*"There is very little detail about what a critical extension is."* [P47, unhandled extension, original]
ErrorCause	The doc. lacks cause of the error.	*"[The documentation] does not provide an explanation of the root cause, or possible source of the error"* [P63, hostname mismatch, original]
NextSteps	The doc. lacks next steps or problem solution.	*"Because there is not any information, how fix this error"* [P116, unhandled extension, original]
Examples	The documentation lacks examples.	*"It is showing you a simple sentence but could also show some example or definition of what is 'notAfter'."* [P8, expired, original]
Incorrect	Part of the doc. is incorrect.	*"[...] The server hostname can be different than the hostname used to access the server. [...]"* [P212, hostname mismatch, redesigned]
Repeating	Some information is stated multiple times.	*"Doesn't give more information than it is already in the error name"* [P64, unhandled extension, original]
Unnecessary	There is unnecessary information or is too long.	*"[...] I feel that too much information is contained on this documentation for the error."* [P59, unhandled extension, redesigned]
Clarity	Comments on text quality, clarity or style.	*"Wording is just weird..."* [P124, expired, original] *"What to do section is completely tangled. [...]"* [P223, name mism., redesig.]
Form	Comments on formatting and structure.	*"The security could be in a separate link or a drop down. I do not need it the section [now] but it could be available."* [P90, unhandled ext., redesigned]
Suggestion	A specific actionable suggestion.	*"A link to at least the specification of handling critical extensions would be helpful"* [P204, unhandled extension, redesigned]

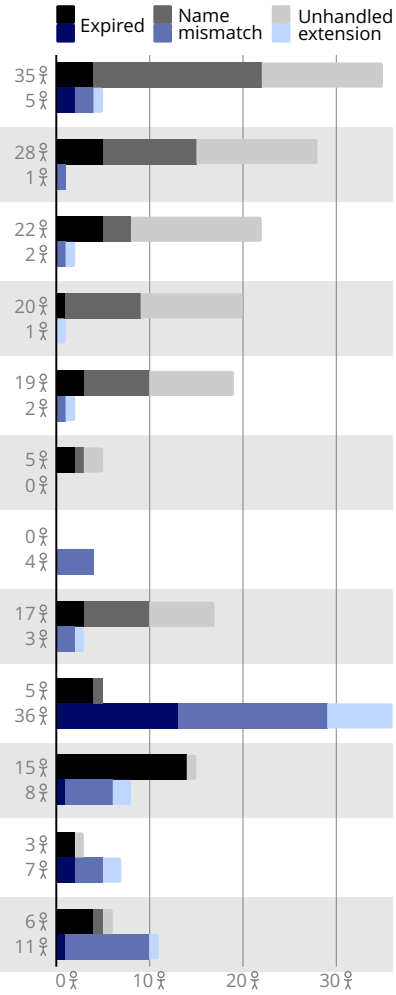

Figure 3: Overview of the reasoning codes for obstacles in the original and redesigned documentation.

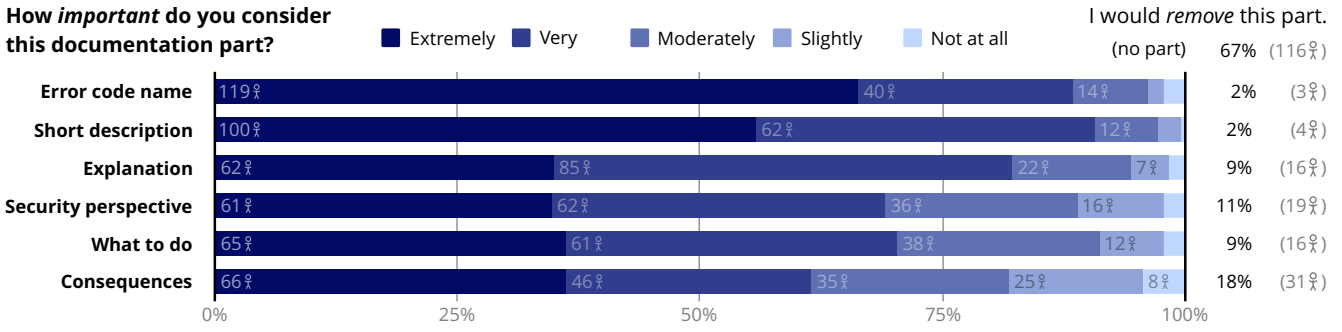

Figure 4: Comparison of the perceived importance among the individual parts of the redesigned documentation, including the ratio of participants who would omit the particular part. Note that about two thirds of the participants would retain all the parts.

> *"I also feel like 'Consequences' is closely tied to 'Security perspective' (i.e., they flow better when read one after the other)."* [P72, hostname mismatch, redesigned]

Additional comments included styling certificate fields as variables to ease text comprehension, pointing to the relevant Request for Comments documents (RFCs), having clickable sections to decrease visual clutter and improving the text style.

> *"I have to read it twice to figure out 'notAfter' is a variable."* [P139, expired, original]

> *"[That is a] very strange sentence after the first look, maybe quotation of "notAfter" would help?"* [P136, expired, original]

> *"To not overwhelm the user by the huge amount of text. it could be hidden and be displayed by clicking the sections [...]"* [P41, hostname mismatch, redesigned]

Results Summary for RQ2. The proposed length and structure seems appropriate; maybe shortening it a bit or combining security perspective and consequences would help. Improvement suggestions include adding links to underlying standards/RFCs and improving the writing style (formatting variables and certificate field names differently, having the style proofread by a technical writer).

4.3 RQ3: Overall Opinions

This section compares self-reported understanding, satisfaction and helpfulness of the IT professionals for both documentation conditions (the overview given in Figure 5). Afterward, it focuses on the overall preference and reasoning thereof (qualitatively coded).

Understanding of the error after seeing the original documentation was already quite high, with 73% of the participants (132 🧍) answering *'Yes'* or *'Rather yes'* when asked if they understood the problem. After seeing the redesigned documentation, the understanding significantly increased to 98% (177 🧍) answering *'Yes'* or *'Rather yes'* (Wilcoxon signed-rank test, $z = 9.03$, $p < 0.001$, with median changing from *'Rather yes'* to *'Yes'*). Note, however, that understanding after seeing the original documentation was significantly different among the error cases (median *'Yes'* for expired, *'Rather yes'* for name mismatch and *'Rather no'* for unhandled extension).

Equivalent questions regarding satisfaction and helpfulness were answered lower for the original documentation: only 23% of the participants (42 🧍) were *'Extremely'* or *'Very'* satisfied and the same number found this documentation *'Extremely'* or *'Very'* helpful. The redesigned documentation shows significant increase in both qualities (Wilcoxon signed-rank test) with 84% (150 🧍) being *'Extremely'* or *'Very'* satisfied ($z = 10.32$, $p < 0.001$) and 90% (163 🧍) finding it *'Extremely'* or *'Very'* helpful ($z = 10.37$, $p < 0.001$) with median changing from *'Moderately'* to *'Very'* in both cases. Note that satisfaction and helpfulness may express the same quality as they show a strong positive association for both original ($\tau_b = 0.78$, $p < 0.001$) and redesigned documentation ($\tau_b = 0.69$, $p < 0.001$).

As for the overall preference of one documentation version over the other, only 11% (19 🧍) preferred the original version with the remaining 89% (156 🧍) preferring the redesigned. The most frequent reasons based on the content analysis of the free-text answers are presented in Figure 6. The original version was mostly praised for its concision (code Concise, 9 🧍for the original documentation / 2 🧍for the redesigned). The redesign was (by far) the most complimented for being more detailed (Details, 2/107 🧍) with many participants requiring no other information (Complete, 2/26 🧍). Investigating the praised details in the redesigned version, we see multiple mentions of the wider context of the error (Context, 0/9 🧍), actionable further steps (NextSteps, 0/37 🧍) and security implications (Security, 0/19 🧍). It is also mentioned as being more clearly written (Clarity, 7/34 🧍), having a better structure (Structure, 0/8 🧍) and thus being more approachable for junior developers (ForBeginners, 3/11 🧍).

Results Summary for RQ3. Overall, IT professionals seem to understand the problem better with the redesigned documentation, are more satisfied with it and find it more helpful compared to the original. The redesigned version is praised mainly thanks to its details, better clarity and structure.

4.4 Results Deployment

We aim to improve the poor situation in error documentation for certificate validation. We aggregate and compare information about certificate validation errors from multiple TLS-enabled libraries and publish them on a public website. Such a collection enables easier testing, library migration and improving (and possibly unifying) the documentation across libraries. Such unifying attempts are rare but feasible – take POSIX [16] or the unified web development documentation [21].

At x509errors.org, we host the current prototype of the website. As of the paper publication, it contains the work-in-progress data of certificate validation errors for four commonly used [22] libraries (OpenSSL, GnuTLS, Botan, mbedTLS and OpenJDK). Each error has the original documentation and many have an example certificate and mapping to errors in other libraries. Selected OpenSSL error entries also feature the redesigned documentation.

Based on the study results discussed in this section, we decided to make these final modifications to the redesigned warning messages and their documentation:

- *External links.* As suggested by multiple participants, mentions of certificate parts and best practices will be supplemented by links leading to relevant RFCs and guidelines, providing a more detailed and authoritative description.
- *Security perspective.* We merge the *security perspective* and *consequences* sections as was suggested by some of the participants as their content is related.
- *Structure.* We keep the proposed length but highlight certificate field names for easier comprehension.

Apart from introducing the first six pieces of our redesigned documentation, it provides 61 example certificates resulting in different certificate errors for easily-reproducible testing, the in-progress mapping of the cross-library relationships on 89 errors across five libraries and three TLS developer guides. Covering the topic at multiple industrial presentations, the website already reached more than 650 unique visitors over the last half a year.

5 VALIDITY OF RESULTS

This section discusses the validity of presented results. An auxiliary study investigating possible order effects is presented first, followed by discussion of previous experience. Lastly, we investigate the stability of redesign effects across the three errors.

5.1 Condition Order Effects

To increase the reliability of the study, we conducted a smaller auxiliary study on 74 🧍at Masaryk University in late 2020 to investigate possible order effects. The study was an exact replication of the main study, with 42% (31 🧍) having the redesigned condition first (there was no indication that this was a reversed order). The participants of the auxiliary study were graduate students (confirmed by the lower

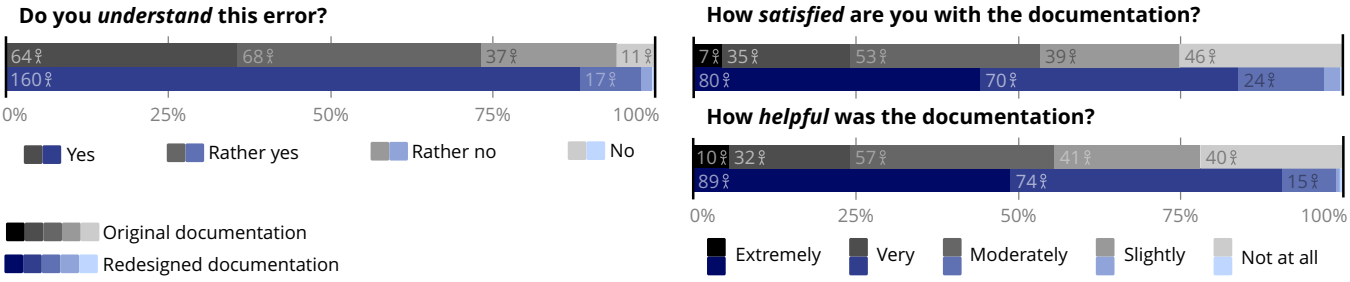

Figure 5: Comparison of the overall error understanding and documentation satisfaction and helpfulness between the original and redesigned condition.

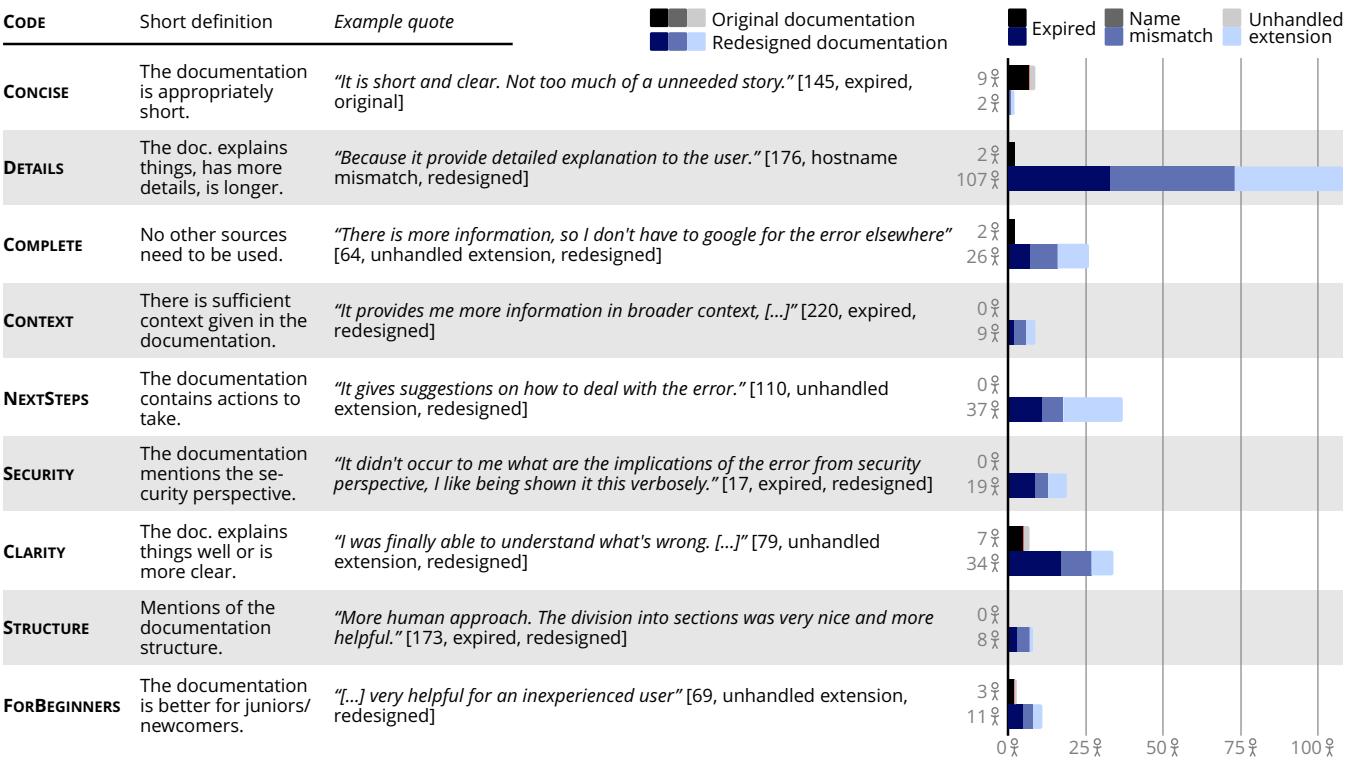

Figure 6: Overview of the reasoning codes for documentation preference in the original and redesigned conditions, showing the (mostly balanced) representation of individual cases (errors).

self-reported previous experience). Still, previous studies [3, 26] suggest that convenience sampling is usually well applicable in usable security research. There were no significant demographic or experience differences between the two order conditions. Let us now investigate the potential order effects on the study research questions.

Comparing the perceived obstacles between the order conditions, the original documentation was seen as more incomplete when displayed second (Mann-Whitney U test, $p = 0.002$, median answer *'Yes'* for both conditions). The redesigned documentation, when displayed second, was viewed as less incomplete ($p < 0.001$, median *'No'* compared to *'Rather no'*), less ambiguous ($p = 0.032$, median *'No'* compared to *'Rather no'*) and less incorrect ($p = 0.006$, median *'No'* for both conditions). Despite these differences, both conditions feature the same relationships between original and the redesign as the main study for incompleteness (decrease, $p < 0.001$), incorrectness (no significant change), bloat (increase, $p < 0.001$) and tangle (increase, $p < 0.013$). Nevertheless, order effects decrease the reliability of the decrease in ambiguity and inconsistency; these were significant only in the condition of original first ($p = 0.001$, $p = 0.039$).

As for the ideal form of the redesigned documentation, there seem to be practically no order effects. Comparing the order conditions, there were no significant differences for the desire of longer or shorter documentation, the same median of 20 lines as the main

study in both conditions, perceived part importance, nor will of omitting certain parts (Mann-Whitney U tests, Fisher's exact tests).

Lastly, some order effects were present for the overall opinions. If the original documentation was displayed first, participants reported significantly higher satisfaction with it (Mann-Whitney U test, $p = 0.001$). However, the median answers were low in both conditions (*'Slightly'*/ *'Not at all'*). When the redesign was displayed first, significantly higher understanding and helpfulness were reported for it ($p = 0.023$, medians *'Yes'*/ *'Rather yes'* and $p = 0.001$, medians *'Very helpful'* in both cases). Despite these small differences, understanding, satisfaction and helpfulness were significantly higher for the redesign compared to the original in both conditions (Wilcoxon signed-rank test, $p < 0.001$). Similarly, the fact that there were no significant differences (Fisher's exact test) for overall version preference, with 100% preferring the redesign when displayed second and 94% when displayed first, strengthens the good perceptions of the redesign.

Summarizing the order effects: 1) The conclusions of the auxiliary study threaten the reliability of decreased ambiguity and inconsistency of the redesigned documentation but strengthen the results for greatly reduced incompleteness and slightly increased bloat and tangle. 2) There seem to be no order effects on the ideal form of the redesign. 3) Despite small order effects on overall opinions, the redesign seems to reliably increase self-reported understanding, satisfaction and perceived usefulness with participants reliably preferring it over the original.

5.2 Previous Experience of Participants

To avoid drawing imprecise conclusions, let us investigate the influence of previous experience of the participants on the main results: the overall opinions (error understanding, documentation satisfaction and helpfulness in both conditions) and obstacles perceived as severe (incompleteness and ambiguity for the original documentation and bloat and tangle in the redesigned documentation). We performed ordinal logistic regression on the stated variables investigating the effects of formal IT education, years working in IT, self-reported computer security and X.509 knowledge, previous OpenSSL usage (as possible covariates) and gender, student status, working in Czechia and seeing the particular error before (as potential cofactors).

For the original documentation, the model predicted understanding, satisfaction and helpfulness significantly better than the intercept-only model ($p < 0.001$) with a systematic positive influence of seeing the error before ($p < 0.006$, odds ratio over 3). For the redesigned documentation, only models for understanding and helpfulness were significant ($p < 0.05$), with notable negative effects of years of IT employment (both models, $p < 0.05$, odds ratio 0.92–0.93) and achieved education level ($p = 0.044$, odds ratio 0.68). As for the documentation obstacles, only the model for incompleteness in the original documentation was significant ($p = 0.040$), with seeing the error before as a predictor ($p = 0.011$, odds ratio 0.38), but the models for ambiguity in the original documentation and bloat, tangle for the redesigned were not significant.

In summary, the performed regression found no significant effects of gender, student status, self-reported IT security and X.509 knowledge or previous OpenSSL usage. The only systematic influence across multiple dependent variables was having seen the presented error before, causing better error understanding, documentation satisfaction and documentation helpfulness, but mainly in the original condition. Although a more in-depth investigation into the effects of previous experience may be necessary, it seems its effects do not pose serious validity risks in this study.

5.3 Documentation Redesign Stability

We examine how stable the effects of the redesign among the three errors are, reiterating the results from Section 4.

RQ1: Perceived Obstacles. Looking at perceived obstacles, we compared the effect of redesigned documentation on the perceived severity of obstacles also error-wise (Wilcoxon signed-rank test). Generally, there was the same behavior as for all participants together (see Figure 2) with only a few exceptions. Incompleteness and ambiguity had a significant decrease for all errors ($p < 0.001$ for all tests). Inconsistency had significant decrease only in the expired and unhandled extension cases ($p = 0.001$ and $p < 0.001$, respectively), while incorrectness only for the expired case ($p = 0.002$). Bloat exhibited significant increase for all errors ($p = 0.004$, $p = 0.001$, $p = 0.004$), while tangle only for the name mismatch case ($p = 0.039$). Looking into the reasoning (see Figure 3), the situation is similar. The expired case had less prevalent incompleteness and ambiguity codes (Accuracy, LittleInfo, Explanation, ErrorCause and NextSteps) but shows significant clarity issues. As mentioned, this was often due to `notAfter` not being formatted/highlighted as a variable. Secondly, the Incorrect code was present only for the name mismatch error in the redesigned condition, where there was, in fact, an imprecision (the documentation said *servers* instead of *hostnames*).

RQ2: Ideal Form. Checking the differences in desired documentation length and structure (see Figure 4), there were no significant differences among the wanted documentation length (one-way ANCOVA with the number of lines of displayed documentation as a possible covariate). As for the documentation part differences among errors, the median importance was significantly different only for the *explanation* part (Kruskal-Wallis H test, $H(2) = 7.23$, $p = 0.027$) and the *what to do* part ($H(2) = 6.30$, $p = 0.043$). Post hoc analysis (Bonferroni-corrected) shows these parts a bit more important for the unhandled extension case than the name mismatch case. However, the median for both cases was *'Very important'*. Comparing the omission desire among errors shows significant differences only for the *what to do* part (Fisher's exact test, $p = 0.017$). Post hoc analysis reveals that significantly fewer participants would omit the *what to do* part for the unhandled extension case compared to the expired case (2% vs. 16%, $p = 0.008$). These results support the initial assumption that the unhandled extension case is a more complicated error than the other two.

RQ3: Overall Opinions. For the error understanding, documentation satisfaction and helpfulness (see Figure 5, all the reported effects of documentation redesign hold also error-wise (Wilcoxon signed-rank test, $p < 0.001$). Neither were significant changes seen for the overall documentation preference (Chi-square test of homogeneity). Content analysis of the preference reasoning reveals only minor

differences (see Figure 6). Participants preferring the original documentation for its concision were mainly from the expired error case (7/0/2 for individual cases). However, this result's validity is limited as the overall number of participants preferring the original documentation is low (19).

Results Summary. All the results reported in Section 4 seem to be relatively consistent throughout all three error cases. The effects of the redesign can, therefore, be considered stable.

6 CONCLUSIONS AND FUTURE WORK

This research aimed to improve the usability of certificate validation by improving the available resources. We performed a validation study on the redesigned documentation for three OpenSSL errors on 180 IT professionals recruited at a developer conference.

The study was guided by three research questions, inspecting obstacles, form and overall opinions of the original and redesigned documentation. The most frequent perceived flaws in the original documentation turned out to be incompleteness and ambiguity. The redesign caused a reliable significant decrease in incompleteness. On the other hand, the redesigned documentation featured a small, though significantly increased, perceived bloat and tangle. The results are backed up by content analysis of free-text reasoning, showing the same trends. The most frequent issue was the lack of accuracy in the original documentation and, conversely, unnecessary content in the redesign. These effects were stable across all three error cases.

Investigating the form shows that desired documentation length is only a few lines shorter than ours. The structuring turned out to be important and appropriate, with two thirds not wanting to omit any part. Several minor suggestions were identified, from proofreading and formatting adjustments to adding links to relevant RFCs. It turns out that multiple reviews by different roles (security specialist, technical writer, quality engineer) may catch more errors and imperfections.

The overall opinions analysis shows a reliable significant increase in the self-reported error understanding and perceived documentation satisfaction and helpfulness. The results are again backed up by content analysis of the free-text reasoning, emphasizing the higher amount of details and better structure. In summary, 89% of the participants preferred the redesigned documentation over the original.

To increase the impact of our work, we published the new designs online at x509errors.org along with 72 automatically generated differently-flawed certificates demonstrating the errors and work-in-progress cross-library error mapping among OpenSSL, GnuTLS, Botan, mbed TLS and OpenJDK. Apart from this, the website also contains the "TLS guides", summarizing the knowledge we obtained while creating the certificate validation clients. The website gradually gains traction with more than 4 000 visits in the last year (average duration on the page being over 1.5 minute).

As a research project of both academic and industrial nature, multiple directions for future development are available:

- *Upstream Documentation.* We plan to redesign the documentation for other errors and file pull requests upstream to the original libraries (primarily OpenSSL).
- *Better Error Reporting.* To fulfill the participants' call for more accurate error messages, they would have to contain case-specific information (e.g., the specific date/time when the particular certificate expired). Although definitely useful, such a change would require a notable refactoring in the library's error reporting system.
- *Investigating Behavior.* The described study would be nicely extended by a second experiment more intensely focused on IT professionals' behavior resulting from seeing the error message or documentation.

In final conclusion, we see the redesigned developer documentation as fit for real-world applications. Starting from our published page, we believe applying existing usable documentation guidelines can help make documentation more usable for IT professionals.

ACKNOWLEDGMENTS

We would like to thank the DevConf organizers for continual support of our developer-related experiments. The research was financially supported by Red Hat Czech and Kiwi.com. We are also grateful to Agáta Kružíková and Lydia Kraus for their help with the experiment and Petr Švenda for helping us write a better paper.

REFERENCES

[1] Yasemin Acar, Michael Backes, Sascha Fahl, Simson Garfinkel, Doowon Kim, Michelle Mazurek, and Christian Stransky. 2017. Comparing the usability of cryptographic APIs. In *Proceedings of the 2017 IEEE Symposium on Security and Privacy (S&P)*. IEEE, Piscataway, NJ, USA, 154–171. https://doi.org/10.1109/sp.2017.52

[2] Yasemin Acar, Michael Backes, Sascha Fahl, Doowon Kim, Michelle Mazurek, and Christian Stransky. 2016. You get where you're looking for: The impact of information sources on code security. In *Proceedings of the 2016 IEEE Symposium on Security and Privacy (S&P)*. IEEE, Piscataway, NJ, USA, 289–305. https://doi.org/10.1109/sp.2016.25

[3] Yasemin Acar, Christian Stransky, Dominik Wermke, Michelle Mazurek, and Sascha Fahl. 2017. Security Developer Studies with GitHub Users: Exploring a Convenience Sample. In *Proceedings of the 13th USENIX Conference on Usable Privacy and Security (SOUPS)*. USENIX Association, Berkeley, CA, USA, 81–95.

[4] Mustafa Acer, Emily Stark, Adrienne Felt, Sascha Fahl, Radhika Bhargava, Bhanu Dev, Matt Braithwaite, Ryan Sleevi, and Parisa Tabriz. 2017. Where the Wild Warnings Are: Root Causes of Chrome HTTPS Certificate Errors. In *Proceedings of the 24th ACM Conference on Computer and Communications Security (CCS)*. ACM Press, New York, NY, USA, 1407–1420. https://doi.org/10.1145/3133956.3134007

[5] Devdatta Akhawe, Bernhard Amann, Matthias Vallentin, and Robin Sommer. 2013. Here's my cert, so trust me, maybe?: Understanding TLS errors on the web. In *Proceedings of the 22nd International Conference on World Wide Web (WWW)*. ACM Press, New York, NY, USA, 59–70. https://doi.org/10.1145/2488388.2488395

[6] Devdatta Akhawe and Adrienne Felt. 2013. Alice in Warningland: A Large-Scale Field Study of Browser Security Warning Effectiveness. In *Proceedings of the 22nd USENIX Security Symposium*. USENIX Association, Berkeley, CA, USA, 257–272.

[7] Lujo Bauer, Cristian Bravo-Lillo, Lorrie Cranor, and Elli Fragkaki. 2013. *Warning design guidelines.* Technical Report CMU-CyLab-13-002. CyLab, Carnegie Mellon University. Retrieved 2021-06-01 from https://www.cylab.cmu.edu/_files/pdfs/tech_reports/CMUCyLab13002.pdf

[8] Cristian Bravo-Lillo, Lorrie Faith Cranor, Julie Downs, Saranga Komanduri, and Manya Sleeper. 2011. Improving computer security dialogs. In *IFIP Conference on Human-Computer Interaction*. Springer International Publishing, New York, NY, USA, 18–35. https://doi.org/10.1007/978-3-642-23768-3_2

[9] Jeremy Clark and Paul van Oorschot. 2013. SoK: SSL and HTTPS: Revisiting Past Challenges and Evaluating Certificate Trust Model Enhancements. In *Proceedings of the 2013 IEEE Symposium on Security and Privacy (S&P)*. IEEE, Piscataway, NJ, USA, 511–525. https://doi.org/10.1109/sp.2013.41

[10] D. Cooper, S. Santesson, S. Farrell, S. Boeyen, R. Housley, and W. Polk. 2008. *Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile.* RFC 5280. Internet Engineering Task Force. https://doi.org/10.17487/RFC5280

[11] Adrienne Felt, Alex Ainslie, Robert Reeder, Sunny Consolvo, Somas Thyagaraja, Alan Bettes, Helen Harris, and Jeff Grimes. 2015. Improving SSL Warnings: Comprehension and Adherence. In *Proceedings of the 33rd Annual ACM Conference*

on Human Factors in Computing Systems (CHI). ACM Press, New York, NY, USA, 2893–2902. https://doi.org/10.1145/2702123.2702442

[12] Martin Georgiev, Subodh Iyengar, Suman Jana, Rishita Anubhai, Dan Boneh, and Vitaly Shmatikov. 2012. The most dangerous code in the world: Validating SSL certificates in non-browser software. In *Proceedings of the 2012 ACM conference on Computer and communications security*. ACM Press, New York, NY, USA, 38–49. https://doi.org/10.1145/2382196.2382204

[13] Google LLC 2019. *Transparency report: HTTPS encryption on the web.* Google LLC. Retrieved 2021-06-01 from https://transparencyreport.google.com/https

[14] Peter Gorski and Luigi Lo Iacono. 2016. Towards the Usability Evaluation of Security APIs. In *Proceedings of the 10th International Symposium on Human Aspects of Information Security and Assurance (HAISA)*. University of Plymouth, Plymouth, UK, 252–265.

[15] Peter Leo Gorski, Luigi Lo Iacono, Dominik Wermke, Christian Stransky, Sebastian Möller, Yasemin Acar, and Sascha Fahl. 2018. Developers Deserve Security Warnings, Too: On the Effect of Integrated Security Advice on Cryptographic API Misuse. In *Proceedings of the 14th USENIX Conference on Usable Privacy and Security (SOUPS)*. USENIX Association, Berkeley, CA, USA, 265–281.

[16] POSIX Austin Joint Working Group. 2018. *Portable Operating System Interface (POSIX(TM)) Base Specifications.* Standard for Information Technology 7. IEEE. 1–3951 pages. https://doi.org/10.1109/IEEESTD.2018.8277153

[17] Marian Harbach, Sascha Fahl, Polina Yakovleva, and Matthew Smith. 2013. Sorry, I don't get it: An analysis of warning message texts. In *International Conference on Financial Cryptography and Data Security*. Springer International Publishing, New York, NY, USA, 94–111. https://doi.org/10.1007/978-3-642-41320-9_7

[18] Kenneth Laughery and Michael Wogalter. 2014. A three-stage model summarizes product warning and environmental sign research. *Safety science* 61 (2014), 3–10. https://doi.org/10.1016/j.ssci.2011.02.012

[19] Nora McDonald, Sarita Schoenebeck, and Andrea Forte. 2019. Reliability and Inter-Rater Reliability in Qualitative Research: Norms and Guidelines for CSCW and HCI Practice. *Proceedings of the ACM on Human-Computer Interaction* 3, CSCW, Article 72 (nov 2019), 23 pages. https://doi.org/10.1145/3359174

[20] Michael Meng, Stephanie Steinhardt, and Andreas Schubert. 2020. Optimizing API Documentation: Some Guidelines and Effects. In *Proceedings of the 38th ACM International Conference on Design of Communication (SIGDOC '20)*. ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3380851.3416759

[21] Mozilla Corporation 2017. *Mozilla brings Microsoft, Google, the W3C, Samsung together to create cross-browser documentation on MDN.* Mozilla Corporation. Retrieved 2021-06-01 from https://blog.mozilla.org/blog/2017/10/18/mozilla-brings-microsoft-google-w3c-samsung-together-create-cross-browser-documentation-mdn/

[22] Matus Nemec, Dusan Klinec, Petr Svenda, Peter Sekan, and Vashek Matyas. 2017. Measuring popularity of cryptographic libraries in Internet-wide scans. In *Proceedings of the 33rd Annual Computer Security Applications Conference (ACSAC)*. ACM Press, New York, NY, USA, 162–175. https://doi.org/10.1145/3134600.3134612

[23] Rob Reeder, Cram Kowalczyk, and Adam Shostack. 2011. Helping engineers design NEAT security warnings. In *Proceedings of the Symposium On Usable Privacy and Security (SOUPS), Pittsburgh, PA*. USENIX Association, Berkeley, CA, USA, 2 pages. Retrieved 2021-06-01 from https://cups.cs.cmu.edu/soups/2011/posters/soups_posters-Reeder.pdf

[24] Johnny Saldaña. 2015. *The coding manual for qualitative researchers* (3rd ed.). SAGE Publishing, Thousand Oaks, CA, USA.

[25] Margarete Sandelowski. 2000. Whatever happened to qualitative description? *Research in nursing & health* 23, 4 (2000), 334–340.

[26] Andreas Sotirakopoulos, Kirstie Hawkey, and Konstantin Beznosov. 2011. On the Challenges in Usable Security Lab Studies: Lessons Learned from Replicating a Study on SSL Warnings. In *Proceedings of the Seventh Symposium on Usable Privacy and Security* (Pittsburgh, Pennsylvania) *(SOUPS '11)*. ACM, New York, NY, USA, Article 3, 18 pages. https://doi.org/10.1145/2078827.2078831

[27] Stack Exchange, Inc. 2020. *Stack Overflow Developer survey.* Stack Exchange, Inc. Retrieved 2021-06-01 from https://insights.stackoverflow.com/survey/2020/

[28] Joshua Sunshine, Serge Egelman, Hazim Almuhimedi, Neha Atri, and Lorrie Cranor. 2009. Crying Wolf: An Empirical Study of SSL Warning Effectiveness. In *Proceedings of the 18th USENIX Security Symposium*. USENIX Association, Berkeley, CA, USA, 399–416.

[29] Gias Uddin and Martin Robillard. 2015. How API documentation fails. *IEEE Software* 32, 4 (2015), 68–75. https://doi.org/10.1109/MS.2014.80

[30] Martin Ukrop, Lydia Kraus, and Vashek Matyas. 2020. Will You Trust This TLS Certificate? Perceptions of People Working in IT (Extended Version). *Digital Threats: Research and Practice* 1, 4 (2020), 30 pages. https://doi.org/10.1145/3419472 Forthcoming, author pre-print available..

[31] Martin Ukrop, Lydia Kraus, Vashek Matyas, and Heider Ahmad Mutleq Wahsheh. 2019. Will You Trust This TLS Certificate? Perceptions of People Working in IT. In *Proceedings of the 35th Annual Computer Security Applications Conference* (San Juan, Puerto Rico) *(ACSAC '19)*. ACM, New York, NY, USA, 718–731. https://doi.org/10.1145/3359789.3359800

[32] Martin Ukrop and Vashek Matyas. 2018. Why Johnny the Developer Can't Work with Public Key Certificates: An Experimental Study of OpenSSL Usability. In *Topics in Cryptology – CT-RSA 2018: The Cryptographers' Track at the RSA Conference 2018*. Springer International Publishing, New York, NY, USA, 45–64. https://doi.org/10.1007/978-3-319-76953-0_3

[33] Michael Wogalter, Vincent Conzola, and Tonya Smith-Jackson. 2002. Research-based guidelines for warning design and evaluation. *Applied ergonomics* 33, 3 (2002), 219–230. https://doi.org/10.1016/S0003-6870(02)00009-1

A SURVEY QUESTIONNAIRE

This appendix contains the full questionnaire, as it appeared in the study apart from the initial screen with study information and diligence promise. The full texts of the documentation present in the questionnaire can be found in Appendix B.

A.1 Informed Consent

This survey is part of a diploma thesis written in cooperation with CRoCS (Centre for Research on Cryptography and Security) Laboratory of Masaryk University and Red Hat Czech. Participating in the survey is entirely voluntary, and you can cancel your participation at any time. All responses are anonymous and will be published as an open-source dataset in the diploma thesis and the follow-up research publications.

{I have read the Informed Consent, and I agree with the participation in the survey under given conditions.}

To have valuable outcomes from the research, we need precise answers reflecting your true opinions.

{I promise I will fill in the questionnaire diligently and according to my opinions (it takes about 15 minutes).}

A.2 Documentation Variant Number One

Instructions

Imagine that you are a developer, and you are working with a protocol that makes use of X.509 certificates. While trying to use the protocol, an X.509 error is displayed, so you open documentation for the error...

Now you will be gradually shown two variants of documentation for an X.509 error. Please, answer the questions concerning each documentation variant.

[Original documentation of the chosen error was shown here (see Appendix B).]

(1) Have you seen this error before? *{Yes; No; I do not remember}*
(2) Do you understand the error? *{Yes; Rather yes; Rather no; No}*
(3) How much are you satisfied with the documentation for the error? *{Extremely satisfied; Very satisfied; Moderately satisfied; Slightly satisfied; Not at all satisfied}*
(4) How much was the documentation helpful? *{Extremely helpful; Very helpful; Moderately helpful; Slightly helpful; Not at all helpful}*
(5) For each of the following possible documentation flaws, decide whether you agree or not. *{Yes; Rather yes; Rather no; No}*
   (a) Do you consider the documentation for the error **incomplete**? *(Incompleteness = Some information is missing in the documentation.)*

(b) Do you consider the documentation for the error **ambiguous**? *(Ambiguity = The description was mostly complete but unclear.)*
(c) Do you consider the documentation for the error **inconsistent**? *(Inconsistency = The documentation of elements meant to be combined didn't agree.)*
(d) Do you consider the documentation for the error **incorrect**? *(Incorrectness = Some information was incorrect.)*
(e) Do you consider the documentation for the error **bloated**? *(Bloated = The description was verbose or excessively extensive.)*
(f) Do you consider the documentation for the error **tangled**? *(Tangled = The description was tangled with information the respondent didn't need.)*

[Questions 6–11 were displayed only for the flaws for which the answer in question 5 was Yes or Rather yes.]

(6) Why do you consider the documentation for the error **incomplete**? *(Incompleteness = Some information is missing in the documentation.) {Free text}*
(7) Why do you consider the documentation for the error **ambiguous**? *(Ambiguity = The description was mostly complete but unclear.) {Free text}*
(8) Why do you consider the documentation for the error **inconsistent**? *(Inconsistency = The documentation of elements meant to be combined didn't agree.) {Free text}*
(9) Why do you consider the documentation for the error **incorrect**? *(Incorrectness = Some information was incorrect.) {Free text}*
(10) Why do you consider the documentation for the error **bloated**? *(Bloated = The description was verbose or excessively extensive.) {Free text}*
(11) Why do you consider the documentation for the error **tangled**? *(Tangled = The description was tangled with information the respondent didn't need.) {Free text}*

A.3 Documentation Variant Number Two

Instructions

Now imagine the same situation: you are a developer, and you are working with a protocol that makes use of X.509 certificates. While trying to use the protocol, an X.509 error is displayed, but now you get the documentation variant shown below.

[Redesigned documentation of the chosen error was shown here (see Appendix B).]

(12) Do you understand the error after reading the documentation for the error? *{Yes; Rather yes; Rather no; No}*

[Questions 13–21 were identical with questions 3–11 from the evaluation of the original documentation.]

(22) How important do you consider these parts of the documentation for this error? *{Extremely important; Very important; Moderately important; Slightly important; Not at all important}*
- Error code name (written in capitals)
- Short description (follows error code name)
- Explanation
- Security perspective
- What to do
- Consequences

(23) Would you shorten the last documentation for the error by removing a part/some parts of it? Please, choose all the appropriate options.
- *Yes, by removing error code name (written in capitals);*
- *Yes, by removing short description (follows error code name);*
- *Yes, by removing Explanation part;*
- *Yes, by removing Security perspective part;*
- *Yes, by removing What to do part;*
- *Yes, by removing Consequences part;*
- *No*

(24) Which documentation of the error do you prefer?
- *The first one (the short one);*
- *The second one (the long one)*

(25) Why do you prefer this documentation for the error? *{Free text}*
(26) How many lines of documentation would you prefer for the error? (The documentation above has *[27 for expired, 23 for name mismatch and unhandled extension]* lines.) *{Numerical answer}*
(27) Any comment regarding understanding or improving documentation for the errors? *{Free text}*

A.4 General Part

Please, answer the last few general questions.

(28) Gender *{Man; Woman; Other}*
(29) Are you currently a student of IT-related discipline? *{Yes; No}*
(30) What is your highest reached degree in IT related discipline? *None; Bachelor degree (e.g., Bc.); Master degree (e.g. Mgr., Ing.); Postgraduate degree (e.g., RNDr., Ph.D.)*
(31) How many years have you been employed in the IT field (including part-time jobs and internships)? *{Numerical answer}*
(32) What is your current IT position? (If you are a student and employed at the same time, refer to your job position.) *{Developer, Software Engineer; Software Architect; Tester, Quality Assurance Engineer; Security Specialist; Network Specialist; Database Specialist; UX Designer; Technical Writer; IT Support, Help Desk Specialist; Product Manager; Manager; Academic Researcher; Student; Other}*
(33) In which country did you spend most of your working life (consider only IT-related work)? (If you are a student, refer to your student life related to IT.) *{Drop-down list with all the countries}*
(34) How do you consider your knowledge of computer security in general? *{Excellent; Very good; Good; Fair; Poor}*
(35) How do you consider your knowledge of X.509 certificates? *{Excellent; Very good; Good; Fair; Poor}*
(36) How many times have you used the OpenSSL library? Consider both CLI (Command-Line Interface) and usage in the source code. *{More than 5 times; 2 - 5 times; Once; Never}*

B DOCUMENTATION

The second appendix contains the full texts of the original and redesigned documentation used in the survey questionnaire.

B.1 Expired: Original documentation

X509_V_Err_Cert_Has_Expired

The certificate has expired: that is the notAfter date is before the current time.

B.2 Expired: Redesigned documentation

X509_Err_Cert_Has_Expired

Validity of the certificate has expired.

Explanation

Every certificate is delivered for a certain time period (determined by notBefore and notAfter fields in the certificate). The time period determines the validity of the certificate. When the time period elapses, the certificate becomes expired.

Security perspective

The certificate is not valid anymore, which means that issuing Certification Authority (CA) does not maintain information about the certificate and does not guarantee the correctness of information provided in the certificate. Moreover, expired certificates are removed from Certificate Revocation Lists (CRLs), which means that a certificate might be revoked in the past (e.g., because of the revealed private key), but we do not get this information about the expired certificate.

What to do

Ensure that date, time and time zone are set correctly on your device. If the time settings are correct and you are responsible for the certificate, you should get a new valid certificate from the CA. In this case, contact either the CA which issued the previous certificate or another CA. If the time settings are correct and you are not responsible for the certificate, contact the responsible person. If it is a web page with an expired certificate, do not provide any personal or secret information to this site.

Consequences

If you are responsible for the certificate and you decide not to renew it, the expired certificate is untrustworthy and your clients do not have to trust you or your business. If you are not responsible for the certificate and you decide to trust it, you may communicate with another person/entity than you think, which may lead to theft of personal information.

B.3 Name mismatch: Original documentation

X509_V_Err_Hostname_Mismatch

Hostname mismatch.

B.4 Name mismatch: Redesigned documentation

X509_Err_Hostname_Mismatch

The requested hostname does not match the subject name in the certificate.

Explanation

The subject field in the certificate carries information about the certificate's holder (an entity that is associated with the certificate's public key). Certificates are issued to subjects specified in the subject field. It is also this case – the certificate was issued to the subject specified in the certificate. However, the problem is that the subject name is different than the server hostname – the server has a certificate that is not associated with the server, the certificate was issued for another server.

Security perspective

The server pretends to be another server. It can be caused by an attacker who may want to steal your information shared with the server (e.g., username and password). Another reason can be a misconfiguration of the server or incomplete information in the certificate.

What to do

If you are responsible for the certificate, check whether all possible hostnames are listed in the certificate, either in the subject name or in the subject alternative name (e.g., 'example.com' and also 'www.example.com'). Another possibility is to redirect all associated traffic to the hostname indicated in the subject name (e.g., redirect 'example.com' to 'www.example.com'). If you are not responsible for the certificate, contact the responsible person. Try to type full site name, including www. If the problem persists, do not provide any personal or secret information to this site.

Consequences

If you access another server than you think, you may receive wrong or malicious content. Moreover, all information provided to this server can be misused.

B.5 Unhandled extension: Original documentation

X509_V_Err_Unhandled_Critical_Extension

Unhandled critical extension.

B.6 Unhandled extension: Redesigned documentation

X509_Err_Unhandled_Critical_Extension

Either critical extension was not recognized, or information in critical extension could not be processed.

Explanation

Certificate extensions can be used for incorporating additional information into a certificate. The extensions can be critical or non-critical. All extensions marked as critical must be processed. If a system, which processes a certificate, cannot recognize a critical extension, it must reject the certificate. It has to reject the certificate also when it recognizes the critical extension, but it cannot process the information contained in the extension.

Security perspective

An extension can carry arbitrary information, and marking it as critical means that it is crucial to process it. If it cannot be processed, there is a security risk that a certificate's key will be used in a manner it must not be, e.g., that a certificate's key will be used for another purpose that it was aimed or that a Certification Authority will issue a certificate for a subject name for which it is not allowed to issue certificates or many other security risks.

What to do

If you are responsible for the certificate, make sure that only necessary extensions are marked as critical and that the values of critical extensions are meaningful. If you are not responsible for the certificate, you can check the critical extensions and the values which contain, but it is not recommended to continue processing the certificate.

Consequences

If you ignore critical extensions that cannot be processed, it may result in unauthorized use of the certificate.