Review of group 1 report on Online Password Store

Applied Cryptography

November 26, 2015

Troels Thomsen 152165
Rasmus Haarslev 152175
Vilius Maskovas 107154
Justas Mikalajunas 107083

1 Formal mistakes

This analysis is based solely on the report.

Language errors
The report contains several misspellings of words such as “refereed/referees” (on page 16) and misuse of words such as “notations” (when meaning notes on page 15), “logging” (when meaning log-in on page 17). The report contains several usages of superlatives such as “absurd”, “very”, “extremely” - which who do not add any value to the report and should be excluded from a formal report.

Audience
The audience of the report is a student group who is taking advanced security course, however the report is written for audience which is not familiar nor with security nor IT.

Usage of definitions
On page 3 an imaginary company is introduced called “Stensikker A/S”, but this company is not defined until page 11. This serves as a source of confusion to the reader as the reader would expect to be introduced to any unknown terminology.
On page 3 the abbreviation “RNG” is used. Being students of the same course we know that this is an abbreviation of “Random Number Generator”, but this term is also used in non abbreviated form making its meaning confusion to the reader. The reader expects consistency - either choose the abbreviation or the whole term.
On page 11, in table 3 the abbreviation “API” is defined as the term “Web Application Interface”. Since API is a common computer science abbreviation defined as “Application Programming Interface” it should not be redefined. The terms “Web API”, “SOAP API”, “RESTfull API”, etc. already exist to cover the use case you are looking for.

Contradictions
Sections 2.1, 2.3 and 2.4 or in general section 2 contain contradictions for the consequences of passwords being compromised. The report tries to argue that passwords for high-ranking government officials are important to the security of the country, or at least more important than passwords of regular citizens. Afterwards the report argues that from the perspective of the imaginary company it does not matter if a single user has their password stolen, but what if that user is a high-ranking official? It further argues that if a user has their browser compromised it is also a “negligible” risk to the imaginary company. But if a user has the plugin for their browser compromised, would it not render all other users of that plugin at risk, and would that not render the whole company at risk. This seems like a contradiction.
Further it seems pointless to argue security or negligibility from the perspective of the company. If the company does not have its individual customers security as a high priority there is a conflict of interest in the system.

Language errors
In report it is said that the application should be easy to use, but usability scope is not specified. Usability could be broken down into memorability, learnability, efficiency rather than accepting general scope. The protocol is suppose to be used for a browser plugin, mobile application and a website. However, the report concentrates only on browser plugin leaving out mobile application and the website.

2 Attacker model

2.1 Unrealistic / too narrow attacker model

In the description of the attacker, it is stated, that the worst possible attacker (when government agencies are out of scope) is an insider at Stensikker A/S, that tries to access the client data illegitimately. This is not entirely true. The report does not take into account the Dolev-Yao attacker model, which is the attacker, that can overhear, intercept, and synthesize messages. When the browser extension starts a session with the server, the user has to input his master password. The master password will at some point be stored in memory in clear text, which a compromised browser would be able to read. It is mentioned that the system want to prevent this by flushing memory, but it does not mention anything about how they are going to do this. However, even if the password is flushed, there will be a period of time, where it is stored in clear text.

2.2 Passwords policies and generation

Password generation – secure RNG. Secure is not defined, therefore we can assume that dice roll is used as RNG The password generation exploits "secure RNG" according the report. Secure RNG does not define how the RNG works allowing to assume that the RNG uses a coin flip mechanism for one time giving 2 different passwords. What is really safe password? 8 char is not. Need to check how long it takes to break it. The explanation for why 8 character password was chosen as a minimal master password requirement is missing. Math behind the choice would be the best explanation why this choice was made.

2.3 Protocol

The report does not give clear understanding how two-factor authentication works. Also it is written that they use two methods for two-factor authentication, sms and email, and only at the end of the report it is mentioned that they actually use only sms.
The protocol requires devices to be authenticated, but it does not explain how it is done - how do you recognize that a request is coming from a trusted device without being vulnerable to replay attacks?
Figure 6 is unclear and confusing. It is missing an explanation of session authentication with the server. It is unclear how the token is used.
Login() message from plugin to AuthServer is missing something or is vulnerable to replay attacks.
requestEncPW() does not include user identification, therefore how does server keep track which user sent the request.
Authentication cookie in Figure 3 is not explained or mentioned anywhere in the report.
It is unclear how uniqueness of the salt guaranteed.
It is unclear how salt appears in user's initial knowledge.
No arguments to have globally unique salt for every user.
Why to use symmetric encryption. Why not use a cryptographically secure hash?
On page 17, it is mentioned that passwords on the server is stored in encrypted form, using symmetric encryption with a key generated by a hash of the password and a salt. If a password is encrypted with itself, the server will not be able to decrypt the password unless it stores the key, which is contradictory to their protocol.

It is stated that the system will use bcrypt to hash passwords, but the implementation / usage details are missing such as amount of rounds.
The report defines that the HTTPS protocol will be used to secure communication, however the SSL / TLS version is not specified therefore it can be assumed that SSL 2.0 is used which was acknowledged as not secure a decade ago.
Usage of Kerberos / authentication server.
Kerberos is mentioned in the report, however the implementation and usage of the kerberos is unclear.

3 Summary

The report contains several contradicting and confusing sections making the reader unsure how the system works. Most of the choices in the system are lacking justifications, therefore creating an image of uncertainty. It was great idea to try to implement Kerberos into the system, however the description was unclear.