How to Design a Blacklist for a Password Meter

Ludovic Jacquin, Amrit Kumar, and Cédric Lauradoux
INRIA
firstname.lastname@inria.fr

1 Extended abstract

The evolution of password cracking tools commands to administrators and operating system designers to help users to choose strong passwords. Password meters and filters are nowadays widely deployed. They are used online by the prominent Internet companies, included in the main operating systems (passfilt.dll for Microsoft Windows and apg for Linux) and passwords managers. In addition, there is also several free libraries to evaluate passwords: vtpassword, zxcvbn or Navajo. Xavier De Carné de Carnavalet has given an extensive analysis of all those passwords meters in his master thesis [1] which was published partially at NDSS 2014 [2]. Based on the observations of De Carné de Carnavalet [1], most password meters works as follow:

1. the presence or not of characters from the four ASCII charsets (lower, upper, digit, symbol or better known as LUDS) is tested.
2. Some metrics are computed such as the Shannon entropy.
3. The password is checked against a blacklist.

In addition, some transformations can be applied to the passwords like mangling. After these steps, the meter declares if the password security is weak, medium or strong (or any other metrics). The confidence and trust in a password meter is something very relative. The situation is very similar to the NIST test suite for random number generator. If a password fails the test, then the user must change it. Otherwise, the user cannot be sure that his password is safe from cracking tools. Our work focuses on the design of blacklist mechanism.

Including a blacklist in a password meter is not an easy task. The designer faces a dilemma between building an exhaustive list and efficiency. An exhaustive list ensures the best security level for the user but increases the cost of managing the list (access time and memory). We review how to create lists and how to store them. We benchmark the various techniques proposed in the literature including Bloom filters, Count-min sketch and approximate pattern matching.

References