INSTITUT Internet-Technologien &
-Anwendungen

Georg Knabl

FH | JOANNEUM
University of Applied Sciences

Securing Systems against Machine Learning-driven, Password Guessing Attacks

Introduction

When passwords are attacked by password cracking software like John the Ripper or hashcat, the efficiency of this process is significantly affected by the quality of the password lists that are used. Traditionally, tools like these use rule sets or masks along with dictionaries that include leaked passwords gained by previous successful attacks. However, these pre-identified password creation schemes are chosen and converted to attack patterns either by humans or by static automation algorithms which might miss actual human password patterns. Additionally, these tools have limited capabilities in generating password lists of individuals.

In recent years, machine learning algorithms have evolved that are capable of learning password creation schemes of humans by analyzing password leaks (Melicher et. al., 2016). Additionally, these algorithms can be fed with personal information of an individual to generate tailored password lists (Knabl, 2018). Hence, this technology poses a new threat to password security and needs to be considered when securing systems.

This research points out approaches to harden systems that rely on password security and build a protection layer against machine learning-based attacks.

General Mitigation Strategies

A general way to secure passwords is to add another factor of authentification (e.g. 2FA). Additionally, there are at least three main strategies to secure passwords itself against such machine learning-based attacks.

The first mitigation strategy is to try to **recreate the password list creation process** of attackers and warn users if their password is on such a list. This can be achieved by using similar deep neural network models and datasets. Unfortunately, it is practically impossible to generate the same password list of an attacker as he or she will probably use different training datasets with another set of personal details and different hyperparameters.

The second strategy is to apply common approaches to **increase entropy** such as using long or complex passwords. However, as recent research has shown (Knabl, 2018), recurrent neural networks are capable to learn the use of personal information and obfuscations, such as leetspeak. If complexity is gained by adding long personal details or using simple encoding schemes, these approaches will not contribute to the security when attacked by machine learning algorithms (ibid.).

The third and most important mitigation strategy is to **treat human passwords as unsafe** and establish password policies that warn users or even prevent the use of such passwords. This strategy, however, requires an algorithm to reliably identify human passwords. Additionally, it forces users to abide their common creation schemes and switch to true random password list generators, such as those integrated in password managers.

Identifying Human Passwords

The classification of the human-likeness of passwords can be achieved using machine learning classifiers along with suitable vectorizers. Before the training process, this supervised learning approach requires labeled datasets of passwords. This includes a mixture of machine-generated passwords created by a password generator and passwords created by humans. The latter were collected using password lists of the **10,000 most commonly used passwords** (Miessler and Haddix, 2012).

A more suitable approach would be the use of common leaked password lists, such as the "rockyou"-dataset (Miessler and Haddix, 2012) as those do not limit passwords to the popular ones and include individual passwords. Unfortunately, the use of such lists violate data protection regulations, especially the EU's General Data Protection Regulation (GDPR) and hence were not considered in the evaluation.

Having collected a labeled dataset, the passwords first were translated into vector-space using either a **count-** or a **TF/IDF-vectorizer**. Next, different machine learning classifiers were tested, including **Logistic Regression**, **Multinomial Naïve Bayes**, **Linear Support Vector Machine** and **Random Forest**. These classifier- and vectorizer-combinations were then tested against each other to find the best suitable model for the dataset. The results of the training showed that **all combinations** were able to achieve a **precision, recall and f1-score of 99%** and thus identify human-created passwords highly accurately.

These tests, however, did not include **human-random passwords** that were created by humans pressing "random" buttons on the keyboard. As such a dataset could not be found, a survey was sent out to students, which, altogether collected 2,345 human-random passwords from 469 people. The already-trained models were able to correctly classify 83% of those human-random passwords. To further improve the training, a new dataset was created containing human-random versus truly-random passwords and new models were trained on that labeled list. These models were able to now classify **94%** of the passwords correctly, despite the fact that most of them looked truly-random to a human eye.

```
14061966          0.9961306540
y-JQ6{v;_yb|q     0.0000000000
ZBT4n#z-x         0.0000121259
longball          0.9920406811
vikings           0.9723564484
gunit             0.9683620674
.XP?]b3\6nP]l|    0.0000000000
8J9{Bd^           0.0000107884
123india          0.9986476258
*[qg;t            0.0000058089
```

Excerpt of human-likeness evaluation applied to a password list containing random and human passwords.

```
,asgl213
HGHfwjiofjiw!?
FEA452
dciuowed7983zy_
jksdgf644kjbndf
Xkkeelt7tad5z
sabjas012
123jfmvfkfn49fvk.
```

Excerpt of randomly-typed human passwords.

Conclusions

It was shown that common password hardening strategies do not provide sufficient protection to machine learning-based attacks. Besides adding another factor of identification, human passwords in general should therefore be treated as potentially unsafe. Machine learning algorithms can support the process of classifying human-likeliness of passwords with a high confidence. This even applies to "randomly"-typed passwords by humans. Using such classifiers allows system administrators and software engineers to add password policies that withstand machine learning-based attacks.

References

- Knabl, Georg (2018). "Machine Learning-driven Password Lists". Master's thesis. FH JOANNEUM, Kapfenberg, Austria.
- Melicher, William, Blase Ur, Sean M Segreti, Saranga Komanduri, Lujo Bauer, Nicolas Christin, and Lorrie Faith Cranor (2016). "Fast, Lean, and Accurate: Modeling Password Guessability Using Neural Networks". In: 25th {USENIX} Security Symposium ({USENIX} Security 16). Vancouver: {USENIX} Association, pp. 175–191.
- Miessler, Daniel and Jason Haddix (2012). SecLists. Available from: https://github.com/danielmiessler/SecLists [Mar. 17, 2018]

Contact

DI (FH) Georg Knabl, MSc
georg.knabl@pageonstage.at