Gathering and Analyzing Identity Leaks for a proactive Warning of affected Users

Timo Malderle
University of Bonn
Bonn, Germany
malderle@cs.uni-bonn.de

Matthias Wübbeling
University of Bonn / Fraunhofer FKIE
Bonn, Germany
wueb@cs.uni-bonn.de

Sven Knauer
University of Bonn / Fraunhofer FKIE
Bonn, Germany
knauer@cs.uni-bonn.de

Arnold Sykosch
University of Bonn / Fraunhofer FKIE
Bonn, Germany
sykosch@cs.uni-bonn.de

Michael Meier
University of Bonn / Fraunhofer FKIE
Bonn, Germany
mm@cs.uni-bonn.de

ABSTRACT

Identity theft is a common consequence of successful cyber-attacks. Criminals steal identity data in order to either (mis)use the data themselves or sell entire identity collections of such data to other parties. Warning the victims of identity theft is crucial to avoid or limit the damage caused by identity misuse. However, in order to provide proactive warnings to victims in a timely fashion, the leaked identity data has to be available. Within this paper we present a methodology to gather and analyze leaked identity data to enable proactive warnings of victims.

CCS CONCEPTS

• **Security and privacy → Authentication**; *Privacy protections*;

KEYWORDS

Identity theft, identity leaks, identity leakage, identity breaches, identity fraud, password leak, parsing leaks, cyber ecosystem, cyber crime, authentication

ACM Reference Format:
Timo Malderle, Matthias Wübbeling, Sven Knauer, Arnold Sykosch, and Michael Meier. 2018. Gathering and Analyzing Identity Leaks for a proactive Warning of affected Users. In *CF '18: CF '18: Computing Frontiers Conference, May8–10, 2018, Ischia, Italy*. ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3203217.3203269

1 INTRODUCTION

Digital identity data like email addresses and passwords are of particular value to criminals. It is common, that stolen identity data is sold or published after a data breach. Hence, the number of leaked and available identity data is constantly growing. Aside from being sold or published, digital identity data is linked together by the attacker to an individual profile [7].

CF '18, May 8–10, 2018, Ischia, Italy

ACM ISBN 978-1-4503-5761-6/18/05…$15.00
https://doi.org/10.1145/3203217.3203269

In most cases the affected person notices becoming victim of a criminal act, too late. Therefore, it is desirable to inform affected identity owners about the theft of their digital identity so that preventive or reactive measures may be taken. To ensure the notification of the majority of affected people about an identity leak, it is necessary to develop a service that automates the notification for affected people about the identity leak.

In this paper we introduce a process to gather identity leaks. Subsequently, we analyze the compiled collection and display the amount of publicly available identities. Additionally, we introduce a way that describes how an identity leak collection may be normalized and kept up-to-date. Finally, we outline our future work.

2 RELATED WORK

The research in the field of identity leakage originated a variety of services and projects. Services like *have i been pwned* [8], *vigilante.pw* [16], *hacked emails* [2] or the *HPI-Leak-Checker* [4, 6] are provided to internet users to inform themselves whether their identities have been leaked. Users have to know these services and use them regularly and frequently.

Even if the users do so, they do not know whether they can use the service to find out about the latest identity leak. The underlying methods of these services are not public and the naming of the listed leaks are not consistent, if available at all. Subscription services provide the possibility of notifying users about the occurrence in an indexed leak by email. However, this particular account may be compromised. Hence, consistent and robust proactive warnings can not be provided by these services.

Recent studies explore the possibility to aggregate identity data to profiles. Thomas et al. conduct a study in which they gathered identity data and estimated the risk of the identity owner to lose control of their entire identity [15]. Heen and Neumann documented their approach to deanonymize users by linking their describing attributes across different identity leaks [7]. To verify found login credentials Casal "tested a subset of these passwords" at multiple services [1]. He was able to prove that most of them are correct and can be used to impose another user's identity to the provider of these services. It is our strong believe that ethical and legal concerns are to be taken into closer consideration.

Onaolapo et al. provides a different perspective, recording the events happening on email accounts after the respective credentials were leaked [12]. This honey token study provides evidential

proof in the fact that accounts are used by unauthorized individuals given leaked credentials. DeBlasio et al. developed an identity leak detection service for external web services [3]. This is achieved by registration of honey accounts at the web services in question. These honey accounts share the same login credentials with a prepared mail provider. When the credentials are used at the email provider a breach notification is triggered.

Han et al. and Subrayan et al. found that around 33% of users, who have accounts for two different web services, reuse their passwords [5, 14]. If passwords are not reused, users tend to choose passwords, which share a pre- or suffix [5].

3 GATHERING IDENTITY LEAKS

A warning service should have access to all potential identity leaks. Therefore, a method to identify and collect identity leaks is developed.

We will identify all tasks required by the identity leak gathering process. It may be automated to ensure better operability and scalability. It has to be noted that this work focuses on publicly available leaks.

A service or storage location on the Internet where data leaks are traded, deposited or published are called *data sinks* throughout this paper. A data sink can be public, semi-public or closed for a special group of users. The first category characterizes data sinks which can be fetched automatically. The second category of data sinks describes sources which must be identified and loaded manually. Data sinks which can be found automatically are for example single entries on *paste pages* [11] like *pastebin* [13]. Criminals use these kind of services to distribute their robbed leaks, or advertise them by sharing some samples of an identity leak, to prove their existence or value [11].

Paste page services commonly allow to fetch a number of the last publicly published entries. Fetching may be repeated frequently to receive every new published entry. The software project *dumpmon* follows this approach [17], using filters, looking for defined pattern like email addresses. This software serves as a foundation for this project and is developed further.

Data sinks that must be identified and loaded manually require much more effort. For example a particular forum has to be visited to access this type of data sinks. Social or semantic hurdles have to be overcome to enter such a forum. A social hurdle may be the need for an invitation, a certain reputation in a particular community, or mandatory sharing of leaked identity data to get in. An example for a semantic hurdle may be found in a CAPTCHA which has to be solved to submit the login. Automating these kind of tasks is not feasible. Inhomogeneous storage locations, files, and web formats hinder an automatic approach further.

A URL describing the exact storage location of a data sink can be spread through *Leak-Announcement-Pages* [11]. Different types of these have to be distinguished: forums, leak monitoring pages and social media [11]. The content of *Leak-Announcement-Pages* refers to storage locations of data sinks by a URL. The access to these pages is restricted to authorized users.

An online investigation resulted in a list of 15 *Leak-Announcement-Pages*. To prevent abuse, the list will be provided on justifiable request. No financial resources were used to buy any data. Following services are used as sources for the automated retrievable data sinks: *pastebin.org* [17], *pastie.org* [17], *slexy.org* [17], *micropaste.com*, *siph0n.net*, *pastelink.net*, *stronghold (TOR)*.

4 PARSING LEAK FILES

Identity leaks appear in various formats without a common structure. Identity leak files include a variety of attributes like email addresses, user names and passwords. Attributes are usually split by a separating character. Attribute fields may be encapsuled by a surrounding character. Data records are usually separated by a newline. Describing header fields are most often missing.

An automated parser eases the extraction of the attributes. The most straightforward way of implementing this parser is to use regular expressions. However, this is only possible for attributes with a fixed syntax, e.g. the @ character in an email address or the format of a credit card number. These constructs are called *known pattern*. To enable a proactive warning of the affected persons, more attributes like names or addresses have to be extracted, to allow inference on an appropriate communication channel to the user. Therefore recognition of syntactic separation of attributes and semantic interpretation is required.

First of all, the separator must be identified which isolates all attributes from each other (*Separator Detector*). A character is assumed to be a separator if it appears before and after a *known pattern* (optionally encapsuled by quotation marks). This assumption is verified if this separator appears in every line. It is possible that a leak uses various separators in different areas. These areas are analyzed independently. If this approach shows itself unsuccessful a frequency analysis of characters is performed. The separator is the character which occurs with the highest frequency and the lowest standard deviation by line throughout the entire leak.

After the separator detection, a leak must be examined for different parts, which are called *blocks* (*Block Separator*). Within a *block* only one separator is used and every line contains the same number of attributes. If there is a deviation in the number of attributes then another *block* is found. If not all separators for all blocks are known, *separator detection* and *block separation* are repeated for the blocks without a known *separator*.

After all *blocks* and *separators* are known, the attributes in each block must be categorized (*Column Categorization*). *Known patterns* may be recognized directly. It is also checked if these attributes are at the same position in each line. If not, the block has to be split. Each other kind of attribute has to be identified in a different way. Some of these attributes have the same syntax but a different semantic like user names, forenames and passwords. They may include letters, numbers and special characters and are of undefined length. For the identification of these categories different dictionaries are used that include the most frequently occurring attributes of one category. These dictionaries can be used for a correlation analysis with each column of attributes in one block (*Dictionary Module*). If a column of attributes has a high rate of matches with one dictionary then this column is assumed to be of that type. This approach identifies the following types of categories: user name, forename, surname and passwords. Another approach is to identify the correct category through a frequency analysis of occurring characters (*Frequency Module*).

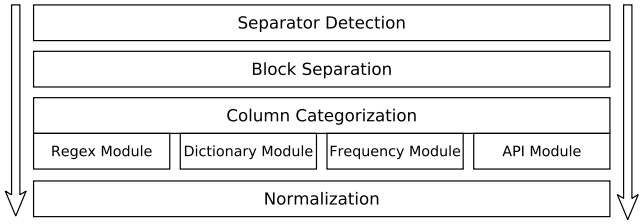

Figure 1: Process of Parsing.

Other attributes like a postal address do not have a fixed syntax. An API of an online map service is used to detect postal addresses (*API Module*). If the service answers with a location for the majority of a category it is confirmed that the attribute is a real address. In addition, this method can be used to homogenize the address format.

Figure 1 shows the described process of parsing. If the results are valid they are passed to the *Normalization*-module to be transformed into the same format.

5 LEAK ANALYSIS

In this section the results of the identity leak analysis will be outlined. 520 identity leaks were collected manually. These files include 3,332,270,763 email addresses. Some of these email addresses occur multiple times in the same or different leaks, resulting in 1,563,002,958 unique email addresses. The services explained in Chapter 2 enlist a comparable number of email addresses. The service *have i been pwned* claims to register 4.86 billion email addresses but includes spam lists [8, 9]. The service *vigilante.pw* claims to register 3.56 billion email addresses [16]. This comparison shows that we gathered identity leaks in the same order of magnitude as other services. The multiple occurrence of some email addresses may be caused by two reasons. Some of the gathered leaks are combined or enriched lists of the same data. Some users use email addresses for multiple services which have been affected by data leaks.

Figure 2 shows the proportions of the number of email addresses included in the identity leaks. The top ten leaks are displayed separately and the rest of the leaks are shown cumulated. These leaks were identified by their reference name. This name is derived from comments about the leak, the specific file name or a heading within the file.

The top ten leaks include 76.66% of all gathered email addresses. The top three leaks contain more than 50% of the found email addresses. While most leaks contain stolen identities, acquired from one service during a security breach, some leaks are composed of multiple leaks. These leaks are known as combo lists [10]. The largest known combo lists are *Exploit.In* [10] and *Anti Public Combo List* [10].

In order to examine their composition, we matched their attributes to known leaks. Our analysis favors email addresses because they are the most frequent attribute found in different leaks. By intersecting the two combo lists with the largest other known leaks it is discovered that approximately 50% of the email addresses can be found in other leaks, while the rest is unique to the combo lists or originates from leaks that are unknown to us at the time of writing. The intersection of both combo lists contains 162,743,673 unique email addresses. That is about 26.39% (Exploit.In) and 38.11% (Anti Public) of all records. It is noteworthy, that email addresses from every single one of the 20 biggest leaks except Adobe can be found in the combo lists although the percentage of overlap varies heavily. A small percentage of overlap is expected because some emails are used for multiple services. In contrast, there are percentages up to nearly 80% in the case of VK in *Exploit.In*. This indicates a reuse of whole parts of the leak in the combo list. Interestingly not one of the biggest 20 leaks could be identified completely in one of the combo lists, closest was the Leak Zoosk with 96.28%. However, in some cases the attributes of a leak are not adopted completely to a combo list, but are edited. In the case of *last.fm*, this leak includes, but is not limited to, email addresses and hashed passwords. These exact email addresses can be found in a combo list, too. They are listed beside a plain text password. Someone must have cracked these passwords.

Frequent seperators are *colon*, *semicolon*, *comma*, *tab*, \r and *space*.

The automatic collection of identity leaks in the period from April 2017 to January 2018 resulted in 18,003 files. This reflects around 69 (arithmetic mean) new files per day. Within these files 9,268,809 email addresses are listed. The majority of the gathered files originates from *pastebin* 69.5%. *Slexy* makes up for 26.0% of the dataset and *siph0n* for 3.9%. The format of these leaks shows no noteworthy differences to the manually identified leaks. However, there is a specialty in the beginning or ending of some files. There is continuous text which gives additional information about the leak. In some cases the text says the presented data is only a part of the whole leak and the whole leak is for sale at a referred URL. Sometimes leaks include more information which is given in a separate section with continuous text or ASCII-Art. In the same time period the manual approach gathered 3.3 billion email addresses and the automatic approach only 9.3 million.

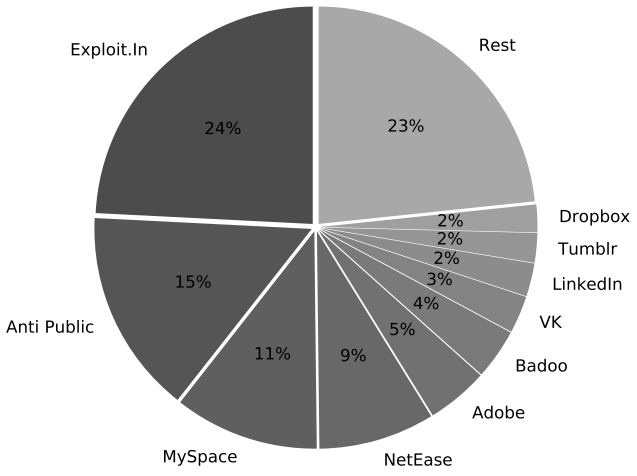

Figure 2: Size distribution of the gathered leaks on the basis of the number of email addresses.

6 CONCEPT FOR FUTURE WORK

The gathered digital identities are the foundation of a service that sends a notification including further instructions on how to react to the victims (*warning service*). The challenge is to identify the correct communication channel and a suitable warning message to ensure that the warning leads to the desired reaction of the notified identity owner. The most trivial solution is to send emails to the identity owner because in most cases an email address is included in identity leaks. However, this kind of email might be mistaken as spam by the receiver. This type of communication channel is not an expedient solution.

Different kinds of communication channels may be distinguished. Communication channels that deliver the warning on a direct way from the warning service to the affected victim is called *Direct communication channels*. Conversely, *indirect communication channels* transport the warning via a third instance. In general, suitable communication channels for warnings are only those that clearly identify a single person with a high probability. Therefore an identity owner identifier must be known for the chosen channel like an email address. That is why a warning with a *direct communication channel* is not feasible in most cases.

The project EIDI, funded by the German Ministry of Education and Research (BMBF), engages this issue by involvement of different organizations that may pose as *indirect communication channels*. These organizations consists of identity owners like social media platforms, banks and telecommunication providers. In a later stage email providers and online shops shall be incorporated. These organizations usually hold more information about identity owners than most identity leaks. The additional information may be used for a warning by aggregating received leaked data with their own identity owner data. A warning can be implemented by a notification during the next login on the organization's platform, a postal letter, or even phone call.

Our Vision:

The overall vision of our research project is to implement a system that automatically notifies affected persons by gathering and processing identity leaks. Applicable privacy laws are taken into account by design.

7 CONCLUSION

User accounts and login details represent service-specific digital identities of Internet service users. In several ways, e.g. by successful attacks on computers of service providers, these identity data get in unauthorized hands and are misused for criminal activities or sold by data dealers. Over the course of these activities extensive collections of identity data become publicly available on the Internet. For example freely accessible Internet-based data storage or exchange services are (mis)used as data sinks for collections of identity data. Typically all this happens unnoticed by the affected identity owners who are exposed to the resulting threats largely unprotected.

Following the vision to resolve this deficiency we systematize the processes and involved services and components around these activities. We propose a process for systematic gathering and analyzing collections of identity data for the purpose of proactive warning of affected identity owners. Several types of data sinks and possible strategies for data discovery and gathering have been investigated and challenges in handling varying formats of identity data collections have been discussed.

The proposed concepts are evaluated by an exemplary analysis of already gathered identity data collections. Difficulties of our highly automatized approach are the data formats of identity data collections which differ heavily. Another problem is that without additional information no assessment of the data quality is possible. Future work will focus on content analysis of collected data leaks and address the problem of identifying affected victims. Possible communication channels towards the victims as well as the design of warnings will be investigated.

The work presented here was funded by the German Federal Ministry of Education and Research under contract-no 16KIS0696K.

REFERENCES

[1] Julio Casal. 2017. 1.4 Billion Clear Text Credentials Discovered in a Single Database. (Dec. 2017). Retrieved January 31, 2018 from https://medium.com/4iqdelvedeep/1-4-billion-clear-text-credentials-discovered-in-a-single-database-3131d0a1ae14

[2] J. M. Chia. 2017. Hacked-Emails. (2017). Retrieved January 31, 2018 from https://hacked-emails.com/

[3] J. DeBlasio, S. Savage, G. M. Voelker, and A. C. Snoeren. 2017. Tripwire: Inferring Internet Site Compromise. In *Proceedings of the 2017 Internet Measurement Conference (IMC '17)*. ACM, New York, NY, USA, 341–354.

[4] H. Graupner, D. Jaeger, F. Cheng, and C. Meinel. 2016. Automated Parsing and Interpretation of Identity Leaks. In *Proceedings of the ACM International Conference on Computing Frontiers (CF '16)*. ACM, New York, NY, USA, 127–134.

[5] W. Han, Z. Li, Minyue Ni, G. Gu, and W. Xu. 2016. Shadow Attacks based on Password Reuses: A Quantitative Empirical View. *IEEE Transactions on Dependable and Secure Computing* X, X (2016), 1–1.

[6] Hasso-Plattner-Institut für Digital Engineering gGmbH. 2017. HPI Leak Checker. (2017). https://sec.hpi.de/leak-checker Sichtung: 23.08.2017.

[7] O. Heen and C. Neumann. 2017. On the Privacy Impacts of Publicly Leaked Password Databases. In *Detection of Intrusions and Malware, and Vulnerability Assessment*, Michalis Polychronakis and Michael Meier (Eds.). Springer International Publishing, Cham, 347–365.

[8] T. Hunt. 2017. have i been pwned? (2017). Retrieved January 31, 2018 from https://haveibeenpwned.com

[9] T. Hunt. 2017. Inside the Massive 711 Million Record Onliner Spambot Dump. (2017). Retrieved January 31, 2018 from https://www.troyhunt.com/inside-the-massive-711-million-record-onliner-spambot-dump/

[10] T. Hunt. 2017. Password reuse, credential stuffing and another billion records in Have I been pwned. (May 2017). Retrieved January 31, 2018 from https://www.troyhunt.com/password-reuse-credential-stuffing-and-another-1-billion-records-in-have-i-been-pwned/

[11] D. Jaeger, H. Graupner, A. Sapegin, F. Cheng, and C. Meinel. 2015. Gathering and Analyzing Identity Leaks for Security Awareness. In *Technology and Practice of Passwords*, Stig F. Mjølsnes (Ed.). Springer International Publishing, Cham, 102–115.

[12] J. Onaolapo, E. Mariconti, and G. Stringhini. 2016. What Happens After You Are Pwnd: Understanding the Use of Leaked Webmail Credentials in the Wild. *IMC '16 Proceedings of the 2016 Internet Measurement Conference* (2016), 65–79.

[13] Pastebin. 2017. Pastebin. (2017). Retrieved January 31, 2018 from https://pastebin.com

[14] S. Subrayan, S. Mugilan, B. Sivanesan, and S. Kalaivani. 2017. Multi-factor Authentication Scheme for Shadow Attacks in Social Network. *2017 International Conference on Technical Advancements in Computers and Communications (ICTACC)* (2017), 36–40.

[15] K. Thomas, F. Li, A. Zand, J. Barrett, J. Ranieri, L. Invernizzi, Y. Markov, O. Comanescu, V. Eranti, A. Moscicki, D. Margolis, V. Paxson, and E. Bursztein. 2017. Data Breaches, Phishing, or Malware?: Understanding the Risks of Stolen Credentials. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security (CCS '17)*. ACM, New York, NY, USA, 1421–1434.

[16] vigilante. 2017. vigilante.pw. (2017). Retrieved January 31, 2018 from https://vigilante.pw/

[17] J. Wright. 2017. Dumpmon. (2017). Retrieved January 31, 2018 from https://github.com/jordan-wright/dumpmon