Deloitte.

A research paper by
Advanced Research and Solutions Group
Cyber Risk Services

Vetting Leaks

Finding the Truth when the Adversary Lies

CONTRIBUTOR:

Allison Nixon, Threat Researcher, Deloitte & Touche LLP, anixon@deloitte.com

SUMMARY

As the frequency of data breaches continues to climb, it is important for organizations to be aware that some "breaches" are not actually real, and that diligence should be undertaken to determine whether a breach has actually occurred.

Traditionally, "*hacktivists*" have used database dumps as a tool to make a forceful statement on the Internet. Large database dumps can garner significant attention and sometimes change a company's behavior. Stolen databases, for sale in the underground, frequently fetch a high value. Significant releases also give the author status and fame.

Stealing copies of databases is clearly an effective and lucrative operation. Recently, however, there has been a trend in which attackers simply state that they hacked a website, and then present a fake database dump as "proof." Journalists may then hastily report the claim without verification. Even if incident response processes confirm that the leak is fake and the truth is revealed, some amount of damage has likely already been done, and the incorrect reports involving the company can remain on the Internet indefinitely. Motivations for doing this are varied, but most often involve scamming or attempts at gaining notoriety.

It is possible to shorten this painful process to nothing more than a minor inconvenience. With some fast and simple fact-checking techniques, a third-party individual can efficiently assess the probability that a leak is valid, resulting in an efficient and more appropriate response, while reducing unwarranted damage to reputation caused by media frenzy and public concern.

It is important to note that these techniques only demonstrate a leak is fake, not that a compromise has or hasn't occurred. Although attackers can use the techniques contained in this paper to produce higher quality fake leaks (example: fact checking techniques will not help if an attacker uses a "*combolist*"[1] and an account checker to produce a list of valid accounts and then claim they actually hacked the company), awareness provided by this document will provide a greater overall benefit to the public than to the attackers alone.

Additionally, fake leaks can be released after genuine online breaches occur. The following techniques outlined for consideration in this document should be treated as situational investigative tools and should be carefully applied in a broad manner. Only the victim company can provide a full and accurate analysis.

[1] Some malicious actors will collect and aggregate usernames and passwords from sites they hack or database dumps they encounter. The resulting dataset, called a "combolist," is used for password reuse attacks.

Technical Details

There are several techniques one can use to fact-check leaks that do not involve using victims' passwords to log in. These techniques were developed with the intention of causing minimal impact on systems or victims. Fundamentally, these techniques involve establishing a model for what a real dump looks like, and determining if the suspected dump differs from it in a significant way.

Check for recycled leaks

Recycled leaks are the most likely source of a leak and should be checked first. If the contents of a claimed leak are recycled from some time in the past, there is no new data exposed. Seek out unique-looking artifacts such as passwords, different names, text snippets from the rant in the preamble, et cetera, and simply perform a search for them. Since little effort was made in the creation of these types of "leaks", it should require little effort to expose them as fake. Some past dumps are no longer publicly indexed by the search engines, but have been public at some point in the past, and may have been indexed by various data loss tracking projects. In the future, data loss projects could serve as detection for recycled leaks that no longer appear through search engine results.

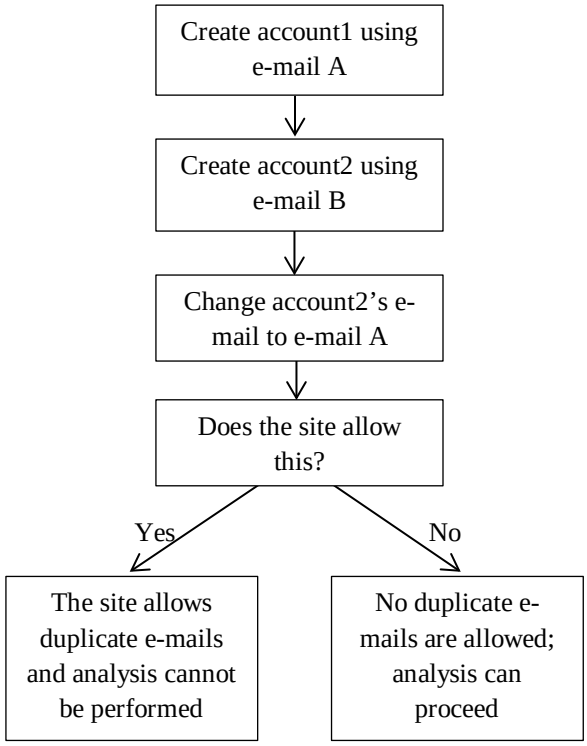
Check for e-mail uniqueness

Many services do not allow two different user accounts to have the same e-mail address. Before using this technique, several things need to be tested to understand the behavior of the victim website, and to avoid causing the victim company to send unsolicited e-mail to victim e-mail owners.

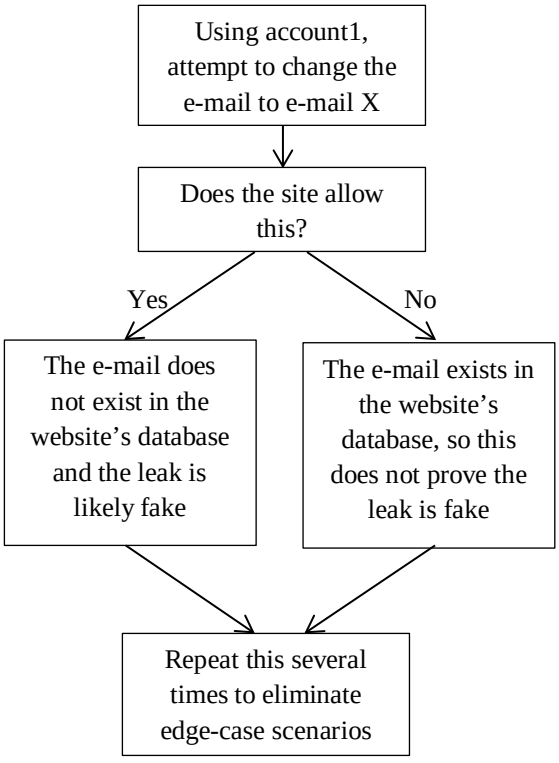
First, the victim company's website rules regarding e-mail uniqueness need to be tested. If the company does enforce e-mail uniqueness, the veracity of the leak can be tested by changing an account's e-mail to randomly selected e-mails in the leak. Almost all e-mails should be traceable to the company's site; untraceable emails indicate that the leak is very likely fake.

The flowchart to the right illustrates a decision process that can be used to determine if an e-mail exists on a website's database. This assumes e-mails are a unique value and the website allows account creation. The materials required are two different e-mail accounts owned by the tester (hereafter referred to as e-mail A and e-mail B), and the sample e-mail accounts from the dump (hereafter referred to as e-mail X).

First step: To test if e-mail checking can occur:



Next Step: Check leaked e-mails against the site:



Check for username existence

If it's possible to view or enumerate the other users on the website, it is worth checking to see if the users in a leak actually exist. For example, if a website gives different responses for [existing username : incorrect password] and [nonexistent username : incorrect password], then it has an information leakage issue that divulges the existence of an account username. If account creation is allowed on the site, attempting to create accounts with usernames from the dump should result in error messages if duplicates are not allowed. This technique works using the same fundamental concept as the "e-mail uniqueness" section above: If unique identifiers are shown in the leaked dump, attempt to duplicate them on the live site.

Additionally, some websites deliberately allow users to view others' profiles. If this is the case, fact-checking the contents of the leak is easier.

Check for password policy adherence

Many sites impose a password policy on users. A tricky issue here is that the password policy may not be enforced consistently for all users. It would be suspicious if the policy is generally enforced, but a large number of leaked credentials are not in adherence to the site's password policy

Conversely, if no password policy exists, and no users have absurdly simple passwords like "123456", the leak should be treated with suspicion.

If a site performs password resets or assigns user passwords, a leak may contain randomly generated passwords that all conform to the pattern imposed by the password reset mechanism. If these types of passwords do not exist, it may be suspicious.

Passwords outside the realm of possibility

If the dataset originated from hashed passwords, certain assumptions can be made about the resulting list of cracked passwords. It is difficult to crack an MD5 hash longer than 13 characters without advanced wordlists and dictionary word combinations. Therefore, highly complex passwords coming from a supposedly cracked hashlist are suspect. Plaintext passwords like "el1cXNBynBqzA7lFveQc" are typically generated from password management programs and a percentage of users will have a password like this. However, it is suspicious if a hacking group claims to have cracked a password with that level of complexity from a hash.

Conversely, it is suspect if the list came from cleartext passwords, and plaintext passwords like those generated from password management programs are never observed.

Username and Password Style

Past real leaks show that people often create passwords in a predictable manner, with a majority of those passwords containing dictionary words, and sometimes specific predictable words. In many cases, people use the name of the service they are using in the actual account credentials. For example, breaches have been identified because lists of usernames and passwords contained multiple references to a breached company's name. For example, a breach of "example.com" may include passwords like "examplepassword" or "monkeyexample". Users may also include the date in a password, which can be useful in dating a leak.

Adherence to known "most common passwords" lists is also worth checking. The following site contains some statistics on the frequency of commonly used passwords, and can be compared to a suspect leak. It is important to take into account the password policy of the victim website while performing this analysis.

https://xato.net/passwords/more-top-worst-passwords

Passwords created by humans have a distinct style. This is an area worthy of further study as there is not currently a way to programmatically differentiate between passwords created by humans or computers.

"Leaked" credit cards analysis

Credit cards must conform to a number of formatting rules. Each of these rules can be applied to a credit card leak to determine if the leak is valid. The most accurate analysis may come from someone who works at the affected credit card company, but third parties can perform several independent checks.

Several different types of credit card formats

A number of available public guides describe card data formats. While some real leaks can have junk data, if a leak generally does not conform to a valid data standard, it can be discounted as fake.

Formatting information for magnetic stripe data (useful when validating track1 track2 dumps):

http://www.gae.ucm.es/~padilla/extrawork/tracks.html

Formatting information for credit card numbers (as seen on the front of the card):

http://www.computersolving.com/computer-tips-tricks/what-your-credit-card-numbers-mean/

Importantly, the check digit on the credit card must conform to the Luhn algorithm. The first 6 digits of the card number reveal information about the card issuer, even down to the bank where the card was issued. This information is known as the bank identification number (BIN).This information can be matched against a "BIN List" to find the issuing bank. Analysis of issuing banks may yield additional clues about the validity of a dump. For example, a breach in America yielding credit cards mostly issued by banks in the Netherlands would raise some questions.

Frequency Analysis of Data

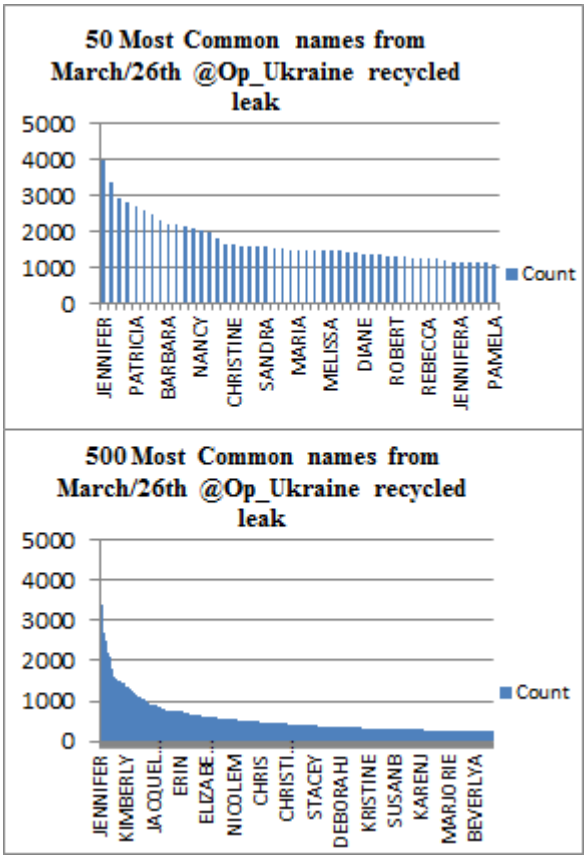
Certain statistical facts about the population will be true when viewing a dump containing real names, given a large enough sample. For example, the distribution of names in a breach should produce a graph with a few highly popular names, and a very large number of increasingly rare names. This "long tail" characteristic of name distribution is going to be present in real breaches, as well as recycled breaches, and random samples of the population. But if the data were randomly generated and care is not taken to fake the frequency distribution, the generated data will differ from observed frequency phenomenon.

Benford's Law is one such phenomenon that has been used to detect faked datasets, such as detecting fraud[2]. Another observed phenomenon is the fact that the commonness of people's names differs based on the year they were born, and this information is public[3].

Additional research should be done to determine how leaked data conforms to frequency phenomenon like Benford's law.

[2] https://iaonline.theiia.org/putting-benfords-law-to-work
[3] http://www.ssa.gov/OACT/babynames/decades

One example here is from a dump produced by the Anonymous threat actor group "OpUkraine" in 2014. Since this was recycled data from a previous breach, the data shows the same characteristics as a natural population. The below frequency distribution table was taken from the Op_Ukraine dump[4], and while not every natural population will have an identical graph, every randomly selected population of humans should exhibit a "long tail" in any graph of the popularity of their first names:



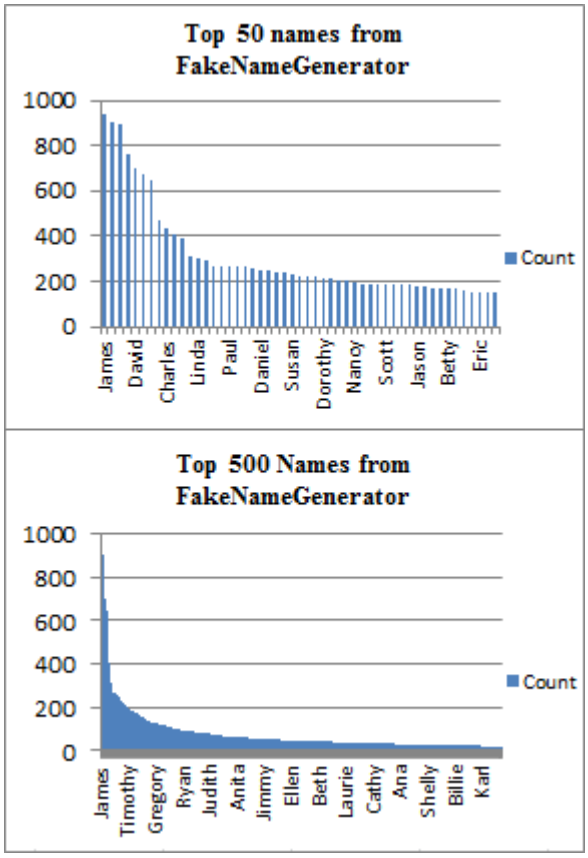
Another method to check dumps that contain real names is to track down the people the dump supposedly represents. If a dump contains a phone number, e-mail, address, or other information related to the person, that same profile should be held by a real person, somewhere.

Randomly Generated Data

The website "Fake Name Generator"[5] generates lists of identities that could easily look like real database dumps, as it conforms to a lot of observed frequency related

[4] Data originally posted on pastebin.com from the March 26 2013 Anonymous @Op_Ukraine Data Breach; information has since been removed from the original publication site.
[5] http://fakenamegenerator.com

phenomenon that affects names, addresses, etc. However, it is easy to detect as fake because none of the people actually exist.



Smell Test

The final measure is the smell test. This is hard to quantify and should be applied on a case by case basis. It involves determining whether or not the action taken is rational and typical of how the action may likely be carried out. The question of what is "typical" is leading answered by someone who is familiar with the underground and has seen similar actions take place.

For example, a claim by a hacking group that it plans to release a vast credit card dump should be met with suspicion. Credit card dumps retain high dollar values on fraud markets and releasing a list for free would not be rational.

Atypical actions also raise suspicion. In one incident, a Pastebin post was made offering eBay credentials up for sale. This is not how stolen databases are normally sold. Stolen databases are usually sold with some assurances that the data will be delivered. Usually, this is done on a forum with trusted administrators that attempt to verify the data, or the seller has a reputation for delivering these products. In this instance, the post was made with no such assurances or reputation, and further inspection confirmed the suspicion that the database was unrelated to the real breach that took place. [6]

The past history of a group is also a good indicator of the validity of leaks they release. It is worth checking the past releases to assess the overall skill level of the group. A group who has been releasing real data is likely to release real data the next time.

Another example of a leak that fails the smell test is shown here, in one of the many "FBI hacked" fake leaks circulating around on Pastebin:

http://pastebin.com/DwDJ0WW8

Not only are the passwords implausible (FBI internal networks are unlikely to allow anyone to use "passwords123"), but the leak contains a lot of extraneous public information such as the results of DNS queries. "Leaking" publicly available data does little to make a statement, and including unnecessary public data is not typical of real leaks. Indeed, including complicated-looking public data may indicate the leaker is an individual without the required skills to pull off the hack they claim.

Conclusion

The techniques described in this paper should provide awareness and insight into attempts to use fake database dumps to scam or generate fear, uncertainty and doubt. This research is designed to assist in expediting the validation process for intelligence analysis and make it easier for analysts to identify forged leaks in the future. It is our hope that the general public views database dumps with a more skeptical eye, and will not suffer from undue panic in the event of a false breach announcement.

[6] source: http://krebsonsecurity.com/2014/05/expert-fake-ebay-customer-list-is-bitcoin-bait/

Contact us

Vikram Bhat, Principal, Cyber Risk Services, Deloitte & Touche LLP, +1 973 602 4270, vbhat@deloitte.com
Christopher Stevenson, Head of Research and Development, Cyber Risk Services, Deloitte & Touche LLP, +1 201 499-0584, chstevenson@deloitte.com
Lance James, Head of Cyber Intelligence, Cyber Risk Services, Deloitte & Touche LLP, +1 760 262 4141, ljames@deloitte.com