The Ashley Madison 'hack'

The organization. The Ashley Madison hack refers to the July 2015 cyber-attack on dating websites that caters to people who are already in relationships. The Canadian parent company at that time *Avid Life Media* had its flagship dating websites *AshleyMadison.com*(AM) connecting dishonest men and/or women seeking affairs with one out of 37 million likeminded people [7]. In 2015, they were advertising their service with a big emphasis on "discreetness" and an attached "trusted security award" medal – especially what their customers are looking for when needing an affair [5].

A continuous stream of money way generated via subscriptions and a deletion service. Customers could create a free profile at AM, but initiating contact with a female was attached with a nifty price tag for men starting at 49$ [6]. According to the leaked data 86% of the users were male [7]. In addition to the subscription, users could pay 19$ to fully delete their data. Nevertheless, leaked data suggests that user data was never deleted however [7].

Timeline. On the 12 July of 2015, employees of Avid Life Media encountered a message that their system have been breached when logging into the computer system. The breaching party *Impact Team* demanded that the site AM and affiliate sites should be shut down and threated with a full data release of customer and employee records. Shortly after, the Impact team started to publish small amounts of data because their demand was not met. At that time, the CEO of Avid Life Media didn't discuss the data's validity but confirmed the data breach. The published data only contained login email addresses which were not verified at registration by AM. The following data dump ,however, contained internal company data like source code and internal emails of the CEO. Again, Impact team's demands were not met prompting to a harsher response: Releasing actual user data. The data contained parts of the credit card information, billing information like address and full name, and dating information [10].

However, it is heavily disputed to what extend user data has been published online. The company itself claims that many copycats aim to get attention by releasing fake data. *KrebsOnSecurity* could verify for a small sample of users that their user information has indeed been exposed [4].To this day, the acteurs stay unrevealed [3]. Although the CEO of the company pointed out an insider attack, no one was arrested for the attack. In General, the company portrayed itself as a victim to a fierce attack on its service. Instead of a responsible notification to their users, the company's statements tried to scrutinize the validity of the breached data [7].

Impact. After understanding how AM generates money, we can identify their biggest business risk: *Trust*. Their target audience were unhappy married woman and men who do seek discretion, because divorces are expensive and publicizing such sensitive data hurts one's reputation among friends and community. Till date, the number of affected divorces is unknown. There also surface suicide reports from former members which could not be undoubtedly linked [7]. This highlights again why companies should take data privacy and security seriously. Even if spouses of active members didn't find out about their loved one's activity, cyber criminals took the chance to pressure them into paying for keeping silence. These type of *blackmailing and sextortion* campaigns even surface long after the initial breach [1, Min. 9]. The company had to pay a fine of $17.5 million to the Federal Trade Commission. However, the fine was reduced to $1.66 million to not put the company out of business [2].

After the data breach, the company also chose to rebrand themselves [8]. This move is done such that customers don't associate the company with the data breach. Furthermore, the renaming can be interpreted as a fool towards investors and employees. Despite all the troubles, the company didn't go bankrupt.

Security Analysis. There is no public technical analysis of how the impact team broke into AM systems. According to the intruders, they were monitoring the system for already long period of time [7]. Furthermore, Avid Life Media aggressively used DMCA-Takedowns to hide the breached data. According to Gabor Szathmari [11], the source code contained *hardcoded credentials* in the form of *database password*, *AWS token*, *private SSL key* and other API tokens. The author did not further explore the storage bucket (S3) associated with the AWS token, but this could likely have been a storage for user chats, images, and videos. Therefore, the developers at AM followed the classic perimeter model. Once the access is gained, the intrudors are free to explore the inner workings.

The incident can be taken as a prime example to show that hard coded credentials don't belong into the source code and not on source controlling systems (e.g. GitHub) where they will remain forever. Furthermore, credentials should be supplied to system dynamically via deployment inside environment variables or key vaults. Each deployment of different system parts should only receive the least necessary credentials [12]. Additionally, the user passwords were saved via a safe bcrypt password hash and an unsafe md5 hash. One group proceeded to crack the md5 hashes to attack the related bcrypt hash to obtain around 11 million passwords. To prevent such issues, developers should stick to strong password hash functions and have independent code reviews in place [9]. Lastly, the developers could have prevented the attack by employing better detection measures. One could monitor database or S3 access to find malicious access.

Summary. In general, the Ashley Madison hack was a major event that had far-reaching consequences for the individuals and companies involved, as well as for the broader issue of online privacy and security. The

website's user data, including email addresses, names, and credit card information, were published to the public, potentially exposing the personal information of millions of users. In particular, the breach specifically highlighted far reaching consequences: It ruined many relations and probably put people into suicide. Furthermore, the response from the breached company left many people in the dark and put the blame on someone else at the end: **the users**.

References

[1] Destination Certification. The hack of ashley madison. https://www.youtube.com/watch?v=2o1OFN1DVU4.

[2] Reuters in Toronto. Ashley madison let off with $1.66m fine over huge hack. https://www.theguardian.com/technology/2016/dec/15/ashley-madison-let-off-with-166m-fine-over-huge-hack.

[3] Brian Krebs. A retrospective on the 2015 ashley madison breach. https://krebsonsecurity.com/2022/07/a-retrospective-on-the-2015-ashley-madison-breach/.

[4] Brian Krebs. Was the ashley madison database leaked? https://krebsonsecurity.com/2015/08/was-the-ashley-madison-database-leaked/.

[5] Ashley Madison. Old website. https://web.archive.org/web/20150623205546/https://www.ashleymadison.com/.

[6] Liam Mannix. Ashley madison made a lot of money, and created very few affairs. https://www.smh.com.au/technology/ashley-madison-made-a-lot-of-money-and-created-very-few-affairs-20150824-gj6er7.html.

[7] Steve Mansfield-Devine. The ashley madison affair. *Network Security*, 2015(9):8–16, 2015. https://doi.org/10.1016/S1353-4858(15)30080-5.

[8] Avid Life Media. We've rebranded! follow us at rubylifeinc. https://twitter.com/avidlifemedia/status/752846714550136832?cxt=HHwWgMCl8aOGO_IUAAAA.

[9] CynoSure Prime. How we cracked millions of ashley madison bcrypt hashes efficiently. https://blog.cynosureprime.com/2015/09/how-we-cracked-millions-of-ashley.html.

[10] Errata Security. Notes on the ashley-madison dump. https://blog.erratasec.com/2015/08/notes-on-ashley-madison-dump.html#.VdeFVyRiM20.

[11] Gabor Szathmari. Credentials in the ashley madison sources. https://blog.gaborszathmari.me/credentials-in-the-ashley-madison-sources/.

[12] CWE Content Team. Cwe-798: Use of hard-coded credentials. https://cwe.mitre.org/data/definitions/798.html.