Cybersecurity: Protecting Genomic Data by Improving the Security Hygiene of DNA Processing Programs

A Technical Report submitted to the Department of Computer Science

Presented to the Faculty of the School of Engineering and Applied Science

University of Virginia • Charlottesville, Virginia

In Partial Fulfillment of the Requirements for the Degree

Bachelor of Science, School of Engineering

Stephanie Skahen

Spring, 2023

On my honor as a University Student, I have neither given nor received unauthorized aid on this assignment as defined by the Honor Guidelines for Thesis-Related Assignments

Advisor

Briana Morrison, Department of Computer Science

Cybersecurity: Protecting Genomic Data by Improving the Security Hygiene of DNA Processing Programs

CS4991 Capstone Report, 2023

Stephanie Skahen
Computer Science
The University of Virginia
School of Engineering and Applied Science
Charlottesville, Virginia USA
sms4yv@virginia.edu

ABSTRACT

There is a high volume of genetic information from Direct to Consumer (DTC) DNA testing sites such as 23andMe and Ancestry.com that is processed, stored, accessed, and shared on the internet. Without proper protections and security practices, the risk that genetic information can be stolen and used with ill intent increases. Security can be improved by creating policies and mandates in line with security hygiene practices taught in CS 3710 and other cyber security courses at UVA. For the initial hygiene status of DTC synthesis and storage companies, I conducted a literature review and analysis of policies and research into their vulnerabilities. Based on preliminary research, it appears that until recently, there were no proper policies in place to prevent companies from selling or sharing the anonymized data they collect. The databases containing this information contain "messy" code and opening doors for possible breaches. Based on my analysis I propose that these companies adopt new cybersecurity practices including removing buffer overflow opportunities and creating tighter identify access protocols. A follow-on analysis should be done on the effectiveness of these practices on large scale databases.

1. INTRODUCTION

Currently, the consumer genetic testing market, such as 23AndMe, is expected to grow by 12.5% every year according to a Market Research Report from Souring Intelligence, LLP (2021). Information stored in these databases have been used by law enforcement in cases such as the Golden State Killer (St. John, 2020). In this example, law enforcement submitted DNA of the unknown killer and was able to determine that it was Joseph DeAngelo through the DNA matches with relative. In 2020, the online DNA database and matching service GEDmatch was hacked altering the privacy settings of users (Mullin, 2020).

The genetic data held within the databases used by gene tracing companies can be used to identify individuals, their family members, and contains medical information that could be used by foul players. Keeping private information private becomes increasingly difficult with the digitization of data and the storing of data in the cloud. Without proper cybersecurity and database security practices, this private information can be leaked by and to anyone.

2. RELATED WORKS

Arshad, et al. (2021) conducted an analysis of the cybersecurity of open-source bioinformatics software and databases. I utilized their findings to discover and analyze vulnerabilities present in direct-to-consumer DNA testing sites.

Schumacher et al, (2020) examined the process of genetic testing and storage. They outline potential friction points and vulnerabilities in the process such as the use of third-party software and equipment.

Initiatives and standards proposed by NIST, CISA, IEEE, and the FCC are used to inform the solution presented. These entities all have proposed standards that can be combined and adapted to fit the case of the consumer genetic testing market.

3. PROPOSED DESIGN

The proposed solution consists of two parts: 1) mitigating buffer overflows during data computation and storage; and 2) improving access control measures. A buffer overflow often occurs when there is malformed input data into a set memory block size, which overwrites adjacent code or memory. Access control includes which users to specific information, who is allowed to be considered a user, and how involved the identification process is for each user.

3.1 Review of System Architecture

Many existing open source DNA processing algorithms are written in C/C++ and contain buffer overflow vulnerabilities through the use of known vulnerable functions and a lack of data validation (Peter Ney et al., 2017). Without having access to the code that popular DTC companies use, the open-source code analysis is being used to identify vulnerabilities. DTC companies and storage sites allow access to data once a DNA sample has been submitted and processed. 23andMe validates to ensure that the DNA tested is human and does not allow data access with invalid samples.

3.2 Requirements

The databases containing genetic information and processing software should protect against the use of buffer overflow attacks that may cause data leaks, inject malicious code, or disrupt the databases' functioning. It should also protect against unauthorized access by outside individuals and users.

3.2.1 System Limitations

The system is limited by the fact that users are connected to other users with similar genetic profiles indicating family therefore similar data profiles will have access to a limited amount of information about other users. The system also must process the DNA, store the DNA, and analyze the DNA opening many opportunities for points of failure or leaks.

3.3 Key Components

The proposed system has a series of cybersecurity specifications, challenges associated with these specifications, and proposed solutions to these problems. I outline these components below.

3.3.1 Specifications

Processing Programs written in C/C++ should not include the known vulnerable functions: *vsprintf(), scanf(), strcpy(), printf(), sprintf(), strcat(),* and *gets()*. These functions are known vulnerabilities.

Buffers declared to contain data should not be static preventing memory overwrite opportunities.

Data should be validated to ensure that it is formatted properly. This check should occur both on DNA processing to ensure that the specimen is valid, and also before analysis or storage occurs to ensure that the data type, length, and makeup is what is expected.

The system should disallow access to any data for users that submit invalid samples. The system should also require two-factor authentication for database maintenance personnel and require the answering of security questions to change privacy settings.

3.3.2 Challenges

This solution requires the rewriting of many common open source DNA processing projects which would create labor costs both for the software patch and the extensive testing done to ensure the patch is secure and compatible with the previous version.

Two-factor authentication would require an additional service to authenticate through that the companies would need to develop or purchase a license to utilize.

3.3.3 Solutions

Most of the vulnerable functions such as strcpy() and printf() have secure alternatives native to C/C++ libraries. These alternatives are strlcpy() and snsprintf() which require the size of the buffer to be input into the function as well. These function alternatives should be used when possible with additional data validation statements to ensure the proper size is used. The very similar alternative functions will mitigate the amount of large changes needed to protect against buffer overflows as well as limit the amount of debugging necessary during the testing phase.

Popular two-factor authentication providers include Microsoft, Google, and Duo Security. These options provide apps and webpages through which to confirm authentication and take care of the security between their own system and the users. Cheaper, but less secure options include using E-mail or SMS to send users codes for each login attempt.

4. ANTICIPATED OUTCOMES

Following this model for security of databases and user data during the processing state should fully prevent data leaks caused by buffer overflows.

The use of two-factor authentication should prevent automated bot attacks, 96% of bulk phishing attacks, and 90% of targeted attacks with non-SMS based two-factor authentication or 76% for SMS based 2FA targeted attacks on genetic database companies and their users according to a Google study on Google's Authenticator (Kurt Thomas & Angelika Moscicki, n.d.).

Attacks utilizing buffer overflow capabilities or accessing user accounts through bots, phishing, or targeted attacks should also be mitigated.

5. CONCLUSION

Online computing and data storage is continuing to grow, and with that growth, many industries dealing in user data are growing their online footprint as well. DTC genetic testing, storage, and analysis companies can be the avenue for large amounts of personal, potentially harmful data leaks unless proper precautions are taken. Following database security best practices and ensuring that buffer overflow attacks are mitigated, and proper access control measures and validation are incorporated, is essential to mitigating the effects of data leaks in the event of an attack.

6. FUTURE WORK

This body of work is a proposal for a project and with expected outcomes, however, actual testing of the effectiveness of this model is undone. A model of this system needs to be created and tested against an array of attacks including buffer overflow, sql injection, and phishing. This work focuses on two aspects of database security: buffer overflows and access control. Future work should consider additional focuses such as password strength requirements and Denial of Service attack preventions or vulnerability.

REFERENCES

Arshad, S., Arshad, J., Khan, M. M., & Parkinson, S. (2021). Analysis of security and privacy challenges for DNA-genomics applications and databases. *Journal of Biomedical Informatics*, *119*, 103815. https://doi.org/10.1016/j.jbi.2021.103815

Thomas, K. & Moscicki, A. (n.d.). New research: How effective is basic account hygiene at preventing hijacking. *Google Security Blog*. https://security.googleblog.com/2019/05/new-research-how-effective-is-basic.html

Mullin, E. (2020, July 30). *The Era of DNA Database Hacks Is Here*. OneZero. https://onezero.medium.com/the-era-of-dna-database-hacks-is-here-85a860190622

Ney, P., Koscher, K., Organick, L., Ceze, L. & Kohno, T. (2017). *Computer Security, Privacy, and DNA Sequencing: Compromising Computers with Synthesized DNA, Privacy Leaks, and More* (USENIX Association, Ed.; pp. 765–778).

Schumacher, G. J., Sawaya, S., Nelson, D., & Hansen, A. J. (2020). Genetic Information Insecurity as State of the Art. *Frontiers in Bioengineering and Biotechnology*, *8*, 591980. https://doi.org/10.3389/fbioe.2020.591980

St. John, P. (2020, December 8). *The untold story of how the Golden State Killer was found: A covert operation and private DNA*. Los Angeles Times. https://www.latimes.com/california/story/2020-12-08/man-in-the-window