August 1, 2026

To whom it may concern:

We are writing to disclose a vulnerability to tampering inherent to files produced during the course of forensic DNA analysis. This is a severe vulnerability compromising the integrity of forensic DNA testing conducted in criminal investigations/prosecutions and in national defense. We are notifying you as an interested party so that you can take appropriate action.

In brief, we have established a proof-of-concept that DNA profiling data contained in .fsa/.hid files can be edited or entirely fabricated.

We describe stakeholder concerns below. It is the vendor's responsibility to address these concerns.

We do not believe that this vulnerability was widely known in the industry. The quality of .hid or .fsa files are only as good as the quality of the lab that produces them. These files should never be trusted in isolation. We will describe the technical background of the .fsa/.hid file formats, the nature of the vulnerability, steps required to tamper with the data, a non-exhaustive list of attack scenarios, and recommendations to consider to mitigate this risk. We do not believe that .fsa and .hid files need to be inherently untrustworthy. The issues we've uncovered would be mitigated by implementing two key strategies: (1) leveraging hardware support to generate files that are cryptographically verifiable, tamper-evident, and unforgeable, and (2) establishing robust information management protocols for the management of files.

Our research was motivated by several developments: increasingly frequent attempts to interpret marginal DNA profiling data, recent publications describing algorithmic methods to simulate complex DNA profiling data, and the accessibility of generative AI. While none of these developments affects the existence of this vulnerability, each development increases the likelihood that a motivated party could exploit this vulnerability or reduces the likelihood that tampered data could be identified by expert review.

Background

The .fsa and .hid file formats extend the ABIF file schema.[1] They are produced by Data Collection Software running on a PC connected to a genetic analyzer. Genetic analyzers most commonly used in forensic DNA testing detect emissions from DNA fragments, labeled with fluorescent tags when produced using commercially available PCR amplification kits, such as AmpFlSTR®Identifiler® Plus (Part Number 4427368) and GlobalFiler™ (Part Number 4476135). ABI PRISM 3100 Genetic Analyzer series (and their predecessor from the 1990s) produced .fsa files and were widely used in forensic DNA laboratories until the mid-2010s. The

[1] https://archive.gfjc.fiu.edu/workshops/resources/literature/Genetic_Analysis_and_Interpretation/06_ABIF_File_Format.pdf

Applied Biosystems 3500 Genetic Analyzer series followed, producing .fsa files or the newer .hid format, more commonly used at this time. To our knowledge, .fsa/.hid files were produced in the *vast* majority of forensic DNA testing conducted around the world for the past thirty years.

For each sample "injected" (processed) by the analyzer, an .fsa and/or .hid file is produced. Both .fsa and .hid formats contain a multitude of metadata as well as fluorescence signal data. The metadata records information such as date/time of injection, sample name, analysis parameters, etc. Fluorescence data are recorded at 4-6 wavelengths several times per second, resulting in 4-6 series of ~10,000 signal data points. Typical .fsa files amount to ~100kB (kilobytes). .hid files contain more data than .fsa files, including results of interim analysis and pre-processed signal data, around ~1MB (megabyte).

Following data collection, .fsa/.hid files might be saved onto a networked file server, a USB flash drive for physical transfer, or loaded into a Laboratory Information Management System (LIMS) database. In some instances, they will also be provided to third parties outside of the laboratory for research, additional analysis, or review.

.fsa and .hid files contain "tags" that demarcate sections of data, like chapters in a book that contains no index. Multiple analysis software programs are available, such as commercially available Applied Biosystems GeneMapper ID-X or SoftGenetics GeneMarker HID, as well as the open-source OSIRIS.[2]

After transfer from the data collection PC, .fsa/.hid files will be loaded into one of these "secondary analysis software applications" so that the signal data can be processed. The genetic variants targeted by this type of testing vary in size (fragment length), correlated to the migration time during the injection at which their signals are observed. The intensity of the signal corresponds to the quality and quantity of DNA recovered from a sample and used for testing - higher quality and quantity of DNA means more signal.

Nature of the vulnerability

.fsa files are trivially vulnerable to tampering. The fluorescence signal data are stored unencrypted in a series of ~10,000 data points following a "DATA" tag. Signal data at any point(s) in these series can be directly modified using a hex editor.[3] It is possible to modify the signal data without altering any metadata.

.hid files are somewhat less trivially vulnerable, though they are not secure from tampering. Like .fsa files, they contain unencrypted signal data separated by "DATA" tags. These data can be modified directly using a hex editor in the same manner as .fsa file signal data. .hid files

[2] https://www.ncbi.nlm.nih.gov/osiris/
[3] A hex editor allows direct editing of binary code. Numerous freeware hex editors are available for public download.

include additional unencrypted analysis data. Since these additional data are unencrypted, they can be similarly modified.

A major difference between .fsa and .hid files is the existence of a second, encrypted set of signal data embedded within the file. However, the encryption used is DES, a 50-year-old cipher, which has not been considered secure since the 1990s.[4] DESencryption keys can be easily broken by brute force. Brute forcing is unnecessary for .hid files because the encryption key used by .hid files produced by AB 3500-series analyzers has been publicly posted on a Github source code repository since at least 2013 - several years before most US forensic DNA laboratories switched to the .hid file format.[5] Since the DES cipher uses the same key to encrypt as it does to decrypt data, anyone with a .hid DES key can decrypt the signal data, modify the now-unencrypted signal data, then reencrypt and rewrite the modified data over the original encrypted data.

Steps taken to demonstrate an attack

Due to the increasing availability and popularity of generative AI systems that can perform complex analysis and code generation directed by conversation, we wondered whether vulnerabilities in the .fsa and/or .hid file formats could be effectively exploited by an experienced forensic DNA analyst without any need to read or write source code themselves. We found, in short: yes.

Using the PROVEDit public data set of .fsa and .hid files and a $20/month subscription to Anthropic's Claude chatbot, we were able to modify fluorescence signal data within .fsa files.[7] By editing only signal data, we were able to produce files with modified signal data that retained the metadata of a real .fsa file. We swapped signal data between files, simulated a mixture by summing signal data from two single-source DNA samples, and added a third individual's DNA profile to a preexisting two-person mixture. The .fsa files produced can be analyzed in GeneMapper ID-X v1.4, GeneMarker, and OSIRIS v2.16. These tampered files appear to pass internal quality control (QC) checks. These attacks required domain-relevant knowledge of forensic DNA genetic analysis but did not require any knowledge of programming.

Since .hid files contain duplicated data in both unencrypted and encrypted forms, the same attack is accordingly more complicated. However, since OSIRIS and GeneMarker can read the unencrypted signal data, an attack on OSIRIS- or GeneMarker-analyzed .hid files can involve modification of only the unencrypted signal data. GeneMapper ignores the unencrypted data in favor of the encrypted data. Attacks involving GeneMapper-analyzed data therefore require decryption->modification->reencryption steps. In our limited trials, Claude required coaxing to import the publicly available DES key in order to complete an attack on the GeneMapper-analysis process. We acknowledge that the ability to identify the publicly available source code

[4] https://w2.eff.org/Privacy/Crypto/Crypto_misc/DESCracker/
[5] https://github.com/545ch4/ABIF/blob/master/lib/ABIF/file.rb
[7] https://lftdi.camden.rutgers.edu/provedit/files/

as a promising avenue for developing this attack could pose some impediment to a non-programmer.
We instructed Claude to produce Python scripts that would allow repeatable attacks without internet access. All of our demonstration attacks were *completely* coded by Claude.[8] To be clear, we have demonstrated that a human knowledgeable about forensic DNA and with genAI bot access can exploit this vulnerability and even generate programs that allow repeated off-line attacks. Attacks include simulating a mixture by summing signal data from two single-source profiles and adding a third contributor's DNA profile to a preexisting two-person mixture.

It should be noted that both .fsa files and .hid files include data fields containing internal checksums, intended to promote data integrity.[9] Despite modifying signal data in a number of .fsa and .hid files, we did not observe GeneMapper, GeneMarker, or OSIRIS refusing to analyze tampered files or even to highlight that checksum verification was performed, never mind that it should indicate compromised data integrity. Checksum values are only as useful as they are trustworthy. If the source of the checksum value is a tampered file, the checksum value can be modified to incorrectly validate fraudulent signal data.

Non-exhaustive attack scenarios

Attacks appear to be possibly performed by anyone who has physical and/or network access to these files. Detection of the attack could be demonstrable only when an oversight/accountability system has logged a .fsa/.hid file characteristic that could identify bit-level modifications prior to the attack taking place AND when that system is referenced post-attack for data integrity checks.

No singular system for laboratory data management exists within the US. It is quite possible that some laboratories are robustly protected against the attacks we describe. Others are likely vulnerable to physical and/or networked attacks, including foreign attacks on internet-connected systems.

Increasing use of probabilistic genotyping software such as STRmix™ and TrueAllele(R) has encouraged laboratories to attempt interpretation of increasingly marginal DNA profiling data. DNA profiles that might already appear to be of marginal utility could conceivably be slightly modified to the detriment/benefit of an individual without any red flags raised by the software or an analyst's experience.

Recent research has demonstrated an ability to generate entire forensic DNA profiles, including complex mixed profiles.[10] While these research projects have not attempted to insert simulated DNA profile data into .fsa/.hid files, we consider this a trivial extension of our proof-of-concept. If

[8] https://github.com/ForensicBioinformatics/ABIFtampering
[9] A checksum is a fixed-size (e.g., 128-bit) output of a standardized mathematical operation on a file or data set. Modifying a single bit of the input file/data affects the checksum value produced. Checksums are used to guard against data corruption by providing a reference checksum against which a file/data can be checked.
[10] https://github.com/NetherlandsForensicInstitute/DNANet/blob/main/synthetic_profiles/README.md

these technologies were coupled, **theoretically any profile can be fabricated and inserted into a .fsa/.hid file**.

Vulnerable organizations

Approximately 220 forensic DNA labs presently operate in the United States. In addition to criminal casework, many of these laboratories also engage in paternity testing or testing unidentified remains. Military laboratories are involved in gathering foreign and military intelligence on adversaries overseas. Each of these laboratories should be considered vulnerable to these attacks until demonstrated to be otherwise.

Recommendations moving forward

We recommend asymmetric cryptographic signing. This approach involves using a private key, preferably from hardware, to "sign" the data. Integrity can then be confirmed using a public key, ensuring the files remain identical to their original state.[11] For optimal security, this signing process should be performed by the source hardware, the producing lab or, preferably, both entities.

Additionally, improving security on the “pipeline” of data that passes from analyzer to collection PC can facilitate uploading of the data files to a LIMS database that can establish a snapshot of data file contents. Checks against this database could be used for internal laboratory checks for data tampering.

While we understand that the actual content of these data files should be protected due to their sensitive nature, we encourage stakeholders to consider lessons learned from other disciplines such as electronic voting that have established protocols for verifying results without exposing private data.

The vendor has issued a security bulletin addressing this vulnerability.[12] Stakeholders should consider this bulletin when developing their approach to this issue.

Conclusion

This critical and longstanding vulnerability now requires no specialized programming knowledge to attempt and can be difficult to detect. Although downstream controls can help mitigate the risk, only the vendor can completely eliminate the vulnerability.

[11] Public key cryptography is the technical basis of electronic signatures. Asymmetric key cryptography predates the field of forensic DNA by a decade and was popularized in the 1990s for use in the World Wide Web.

[12] https://documents.thermofisher.com/TFS-Assets/CORP/Product-Guides/fsa_hid_bulletin.pdf

We remain open to further discussions concerning stakeholder priorities, community values, and mitigation strategies.

Sincerely,
Nathaniel Adams, Systems Engineer, Forensic Bioinformatic Services, Inc.
Laura Gaydosh Combs, Ph.D., Associate Professor of Forensic Science, Henry C. Lee College of Criminal Justice and Forensic Sciences, University of New Haven
Kevin P. Dyer, Ph.D., Independent Researcher, Security Engineer
Dan E. Krane, Ph.D., Professor of Biological Sciences, Wright State University