sciendo

Proceedings on Privacy Enhancing Technologies ; 2021 (3):413–432

Peter Ney*, Lee Organick, Jeff Nivala, Luis Ceze, and Tadayoshi Kohno

DNA Sequencing Flow Cells and the Security of the Molecular-Digital Interface

Abstract: DNA sequencing is the molecular-to-digital conversion of DNA molecules, which are made up of a linear sequence of bases (A,C,G,T), into digital information. Central to this conversion are specialized fluidic devices, called sequencing flow cells, that distribute DNA onto a surface where the molecules can be read. As more computing becomes integrated with physical systems, we set out to explore how sequencing flow cell architecture can affect the security and privacy of the sequencing process and downstream data analysis. In the course of our investigation, we found that the unusual nature of molecular processing and flow cell design contributes to two security and privacy issues. First, DNA molecules are 'sticky' and stable for long periods of time. In a manner analogous to data recovery from discarded hard drives, we hypothesized that residual DNA attached to used flow cells could be collected and resequenced to recover a significant portion of the previously sequenced data. In experiments we were able to recover over 23.4% of a previously sequenced genome sample and perfectly decode image files encoded in DNA, suggesting that flow cells may be at risk of data recovery attacks. Second, we hypothesized that methods used to simultaneously sequence separate DNA samples together to increase sequencing throughput (multiplex sequencing), which incidentally leaks small amounts of data between samples, could cause data corruption and allow samples to adversarially manipulate sequencing data. We find that a maliciously crafted synthetic DNA sample can be used to alter targeted genetic variants in other samples using this vulnerability. Such a sample could be used to corrupt sequencing data or even be spiked into tissue samples, whenever untrusted samples are sequenced together. Taken together, these results suggest that, like many computing boundaries, the molecular-to-digital interface raises potential issues that should be considered in future sequencing and molecular sensing systems, especially as they become more ubiquitous.

DOI 10.2478/popets-2021-0054
Received 2020-11-30; revised 2021-03-15; accepted 2021-03-16.

***Corresponding Author: Peter Ney:** University of Washington, E-mail: neyp@cs.washington.edu

1 Introduction

Given the centrality of DNA to life, the ability to process and analyze DNA samples has become instrumental in a number of fields including medicine, genetics, and bioengineering. Reading DNA, using a process known as *DNA sequencing*, is done using complex hybrid computer-sensor instruments, called DNA sequencers, that take DNA molecules as input and return digital files containing the linear sequences of DNA bases (i.e., A, C, G, and T) in the DNA sample. High demand for sequencing has led to the development of high-throughput sequencers capable of handling billions of DNA molecules at a time [34].

At their core, DNA sequencers are a type of molecular-to-digital interface that translates information stored in DNA molecules into digital files. This conversion between information forms — namely, DNA to digital data — takes place in specially designed hardware, called *flow cells*. Flow cells are fluid moving cartridges designed to distribute DNA across a sticky surface so that the DNA molecules can be read (e.g., fluorescently imaged).

As modern computing becomes more coupled to the physical world with processes like molecular sensing, it is important that we explore how computer security issues might manifest in these systems, especially at the boundary between the physical and digital. Previous work has considered how molecules, like DNA, can eventually trigger problems in downstream data processing [33]. However, to our knowledge, no work has substantially explored security issues at the molecular-to-digital interface. As the central hub of this conversion, we seek to understand the security implications of

Lee Organick: University of Washington, E-mail: leeorg@cs.washington.edu
Jeff Nivala: University of Washington, E-mail: jmdn@cs.washington.edu
Luis Ceze: University of Washington, E-mail: luisceze@cs.washington.edu
Tadayoshi Kohno: University of Washington, E-mail: yoshi@cs.washington.edu

sequencing flow cell architecture and how it interfaces with the broader sequencing and analysis workflow. In particular, we focus on the sequencing flow cells from Illumina's high-throughput sequencing instruments, the most popular class of DNA sequencers [45].

The DNA sequencing process is a long pipeline from sample collection, through sequencing on a flow cell, and eventual data processing (see Figure 1; described in detail later). Many of these phases provide different avenues for adversarial manipulation. For example, DNA samples are sent into the sequencing pipeline that could be manipulated by an adversary prior to sequencing, and the sequencer itself produces significant output, as both digital data and physical hardware (e.g., discarded single-use flow cells). We study how these entry and exit points can affect the security of the molecular-to-digital interface. In our investigation, we find that the sequencing process has two classes of vulnerabilities that are familiar in many computing domains: data remanence and integrity issues. In this work, we explore these two vulnerabilities and show how they can result in security and privacy issues.

Problem 1: DNA is 'sticky' and causes data remanence on flow cells: We hypothesized that DNA's high stability would leave enough of a trace to be recovered from a used flow cell after sequencing. As there is currently no guidance from major sequencing companies on how to properly dispose of flow cells — and some companies even suggest returning them to the manufacturer for recycling — any residual information that is recoverable from a discarded flow cell could have privacy implications in areas like medicine or research e.g., [31, 37]. We explored this possibility by developing a simple recovery procedure that works on two Illumina flow cell models. We then quantified how much data could be recovered from two different DNA sequencing applications, including genomics and DNA data storage. Our results show that residual DNA does indeed remain on a flow cell after sequencing, and after these molecules are recovered, they are sufficient to reconstruct a substantial portion of the information in the original sample. We then discuss possibilities to mitigate data remanence issues on sequencing flow cells.

Problem 2: DNA data can 'leak' from one sample into another and corrupt data: It is common to merge DNA from multiple samples and sequence them together to increase throughput. The data from the combined samples is later demultiplexed into the originating samples using special DNA subsequences that are added to each sample prior to sequencing and later read on the flow cell [28, 29]. While this process is mostly accurate, it does occasionally bin DNA data into the incorrect sample (approximately 0.01%-1% rate of misbinning) [25, 27, 30, 43, 46]. This process has been discussed by the sequencing community as a source of random noise and sequencing error. However, we hypothesized that this could affect data integrity by allowing targeted manipulation between samples. To study this, we ran a number of experiments to evaluate whether a maliciously crafted DNA sample could leverage this vulnerability to affect specific genetic changes in other samples that were co-sequenced. We further evaluate whether this attack could be done with raw DNA or manipulated tissue samples (e.g., saliva). We discuss methods like quality filtering, reducing misbinning, and anomaly detection to mitigate this problem.

Stepping back: Issues like the flow cell sticky and leaky vulnerabilities highlight how new systems in biotechnology and molecular sensing can have problems typically associated with traditional computer systems (e.g., data remanence and corruption attacks) at the computing boundary. As sequencers and other wet lab equipment are effectively specialized computers, we believe that problems like these are important for the security and privacy community to consider.

We now begin in Section 2 with an overview of the necessary background in biology and DNA sequencing. In Section 3 we provide an assessment of the broad security and privacy challenges for flow cells and, from that, derive our two key areas for technical investigation. In Section 4 we explore Problem 1, described above, and in Section 5 we explore Problem 2. We discuss additional related works in Section 6, and reflect on our results in Section 7.

2 Background

Here, we cover the background on DNA sequencing and applications necessary to discuss the following security investigations described in Section 4 and Section 5.

2.1 Sequencing-By-Synthesis

In this work we focus on the most popular class of high-throughput sequencing instruments, pioneered by Illumina (formerly Solexa), that rely on a method called

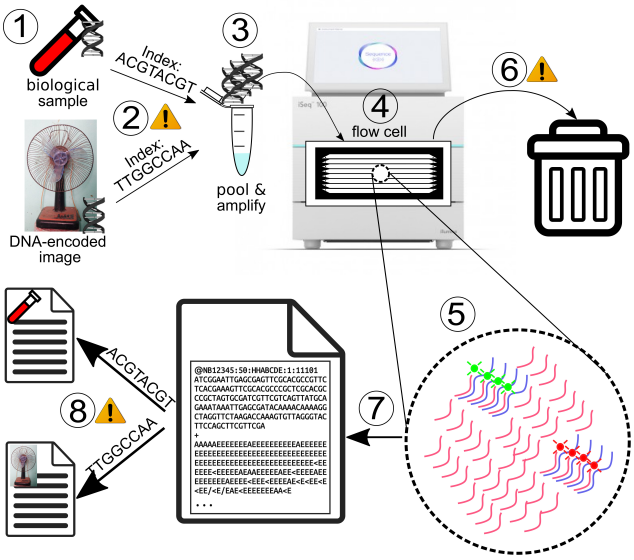

Fig. 1. *DNA sequencing stages* — "Caution symbols" highlight phases of the sequencing process with security implications we study. (1) Purified DNA is obtained from one or more samples. (2) Each sample is prepared with a unique index separately and pooled into a single solution. (3) The pooled sample is optionally amplified. (4) The fully prepared sample is loaded on the flow cell where the DNA is sequenced. (5) Cameras measure the fluorescence on the flow cell to 'read' each DNA strand one base at a time. (6) The flow cell is discarded after sequencing. (7) The fluorescent images are converted into raw digital sequencing data files. (8) The DNA reads are separated into different files corresponding to the originating sample using the index sequence.

sequencing-by-synthesis (SBS). We do not evaluate alternative sequencing technologies, like long-read sequencers because they are not as widely deployed in genomics applications. See Figure 1 for an overview of the SBS process.

Sequencing begins with sample preparation: DNA molecules from a sample (e.g., blood) are isolated and broken into short pieces (fragments) around 150-500 base pairs (bp) in length; the fragments are optionally amplified (copied) to increase the DNA yield; and finally, sequencing adapters and index barcodes are added and multiple samples are pooled together (discussed in Section 2.2). The DNA solution is then injected into a flow cell where the fragments bind to shorter, complementary DNA sequences already attached to the glass surface, called adapter primers. Once attached to the surface, the fragments are duplicated into clusters containing hundreds of clones — strands identical to the original fragment — that are large enough to be read together by high-resolution cameras. The clusters are then sequenced one base at a time using special fluorescent bases that release a different RF-signal for each type of base (e.g., red for A, green for C) that is captured by a camera. Finally, the images are processed to return the sequence of bases in each fragment (known as a *read*); sequencing files can contain hundreds of millions to billions of reads.

Depending on the sequencing technology, flow cells can be either one-time use cartridges which are discarded after sequencing or are reusable for several sequencing runs. We hypothesized that DNA molecules would remain on a flow cell after sequencing, and given that DNA is very stable — even at room temperature — it is very likely that a significant fraction of the DNA, if leftover, could persist intact for years [4]. It is this residual DNA that remains in the flow cell after disposal that we consider in Section 4.

2.2 Multiplex Sequencing

Multiplex sequencing is a method to increase per-base sequencing throughput by mixing multiple DNA samples together and sequencing them in parallel; Illumina's NovaSeq is capable of sequencing 96 samples per lane and 384 per flow cell [34]. Just as processors have used parallelism to deal with the slowing of Moore's law, sequencing-by-synthesis technology has leveraged parallelism to decrease cost and is critical to achieving ubiquitous genomic sequencing with the $100 genome [41]. Multiplex sequencing is achieved by appending short 6-10 bp sequences, called *indexes* or *barcodes*, to the DNA fragments in each sample. Every DNA fragment refers to its originating sample because all fragments in a sample are assigned the same unique index. Then the samples are pooled together and sequenced as one mixed sample. Afterwards, the pooled DNA reads that are stored in sequencing data files can be demultiplexed into the appropriate sample using the index that was sequenced. See Stage-2 and Stage-8 of Figure 1.

Multiplex sequencing does add some noise because a small number of reads (0.01-1.0%) are incorrectly demultiplexed into the wrong sample; essentially, this causes data from different samples to slightly leak into each other [25, 27, 30, 43, 46]. This effect, known as *index cross-talk*, is caused by excessive reactivity of reagents and DNA during sample preparation or due to DNA layout on a flow cell, for example when clusters from two fragments overlap [7, 25, 30, 43]. In many applications, like germline variant calling or genotyping the analysis pipelines are designed to be robust to sequencing error [3], and so a moderate amount of index cross-talk is tolerable; however, cross-talk has caused

problems when it happens at high levels (in some cases reported to be 3% or higher) or in error-sensitive sequencing applications like rare variant calling or ancient DNA analysis [7, 27, 43]. While the index cross-talk phenomenon is known, its implications have not been significantly studied from a security perspective. In Section 5 we explore how the information leakage caused by index cross-talk can be used adversarially.

2.3 Sequencing Applications

DNA sequencing is used in a wide variety of applications from genomics, medicine, forensics, and nascent technologies like DNA data storage systems. Here, we give brief background on the applications that will be relevant in subsequent sections.

2.3.1 Genomics

One of the most popular uses of sequencing is to analyze the human genome for research or medical purposes. Depending on the desired analysis, it is common to either sequence all of the DNA in the genome, called whole genome sequencing (WGS), or to just sequence the exome — the part of the genome that encodes for proteins — called whole exome sequencing (WES). As described previously, the DNA in a genomic sample is randomly fragmented into short DNA strands before sequencing; this results in the DNA being sequenced in random order. Therefore, the first step after sequencing is to reorder the short DNA reads to reconstruct the larger input sequences. When a reference genome is known (e.g., for humans), this can be done using sequencing alignment, which maps reads to the corresponding part of the genome[1].

Genomic DNA samples often have redundancy because samples originate from collections of cells and the DNA is often amplified. Therefore, it is common to have multiple reads (potentially 100s) that map to the same genomic location; the average number of reads per base in the genome is known as the *read coverage* and the number of reads at a specific location as the *read depth*. It is recommended to have 30-50X coverage with WGS and 100X coverage with WES [18].

Most parts of the genome are the same between individuals. Therefore, an important question in both research and medicine is to find which parts of the genome vary[1]. *Variant calling* is a technique that uses aligned reads to identify where a sequenced sample differs from a reference sequence: this can include single base changes (single nucleotide polymorphisms or SNPs) or larger insertions or deletions (indels). The sequence of the reads that are aligned to the region of interest are statistically analyzed to determine the sequence (or call) of the variant in the sample; the more reads that align to a location, the higher the confidence in the call.

2.3.2 DNA Data Storage

One of the more exciting new uses of DNA is its use as a medium to store digital data (e.g., images, videos, books) [6, 14]. DNA is used because it has certain advantages over traditional storage media including: long half-life (thousands of years), high theoretical data density (exabytes per cubic cm), and inherent parallelism of molecules [4]. So called DNA data storage systems write data into DNA-encoded files using synthetic constructed DNA molecules; DNA files are read later via sequencing. State-of-the-art DNA storage systems encode data using error correction schemes so that files can be reliably read — even in the presence of molecular decay and sequencing errors — and some systems support indexing schemes so that files can be randomly accessed in large DNA databases [35].

3 Molecular Processing and Security

Computer hardware and peripheral attachments have long been used to convert between different types of information — for example, hard disk drives are electromechanical devices that convert information encoded magnetically on a surface into electrical signals. What makes flow cells different from typical hardware is that they operate on molecules, which bring unusual properties to the computing domain. We highlight some molecular properties of DNA below that are relevant to the security of flow cells and the sequencing process:

1. *Amplification*: As the information carrier for life, cellular processes have evolved to make high fidelity copies of DNA. This process has been harnessed so DNA can be exponentially copied in solution (via Polymerase Chain Reaction, or PCR). The ability to highly amplify molecules means that even trace amounts of DNA, on the order of 10s of strands, can be amplified and read [36].

2. *Stability*: As previously discussed, DNA is a highly stable molecule. High quality strands can be maintained for years at room temperature and centuries to millennia under optimal conditions [5]. Stability has made DNA attractive in new applications like DNA data storage.
3. *Mixing*: Molecules will mix and move randomly in solution via Brownian motion. Mixing benefits molecular systems because it enables parallel processing of pooled solutions.
4. *Bonding and reactivity*: DNA in its single-stranded form is designed to form weak chemical bonds with complementary strands. In the presence of certain enzymes DNA can even react with other molecules. However, as reactions are a stochastic process governed by thermodynamics, this can lead to unintended side reactions.

These molecular properties of DNA raise security issues when viewed in the context of the sequencing process described in Figure 1.

Consider the molecular properties of amplification, stability, and bonding. These all raise questions about the security of Stage-6 (flow cell disposal). DNA strands from a sequenced sample may be bonded to synthetic adapter sequences or non-specifically stuck to the flow cell surface and remain stable there for years. And because small amounts of DNA can be amplified, even trace quantities of DNA may be recoverable from flow cells and be read to reveal sensitive information. We consider the possibility of flow cell data recovery in Section 4.

The mixing and reactivity properties also raises security questions in Stage-2 (pooling samples) and Stage-8 (demultiplexing data). As previously described, the ability to mix DNA together is beneficial because it can be processed in parallel to increase throughput. However, this provides a vector for an adversary to influence other samples because the adversary knows that their sample will be mixed together with others in the same solution. Further, the fact that DNA can react in solution in the presence of enzymes — both in the combined sample mixture before sequencing and on the flow cell itself — may give the adversary an avenue to influence results in other samples. We study the ability of one sample to influence another when the samples are combined together in solution in Section 5.

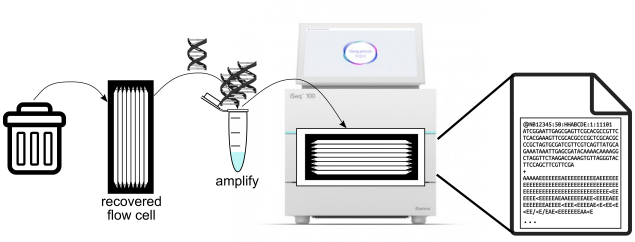

Fig. 2. *Residual DNA recovery on a used flow cell* — A disposed flow cell is recovered, residual DNA collected, and the DNA resequenced. Water is flushed through the input ports on a flow cell to collect the residual DNA. The resulting DNA is amplified and resequenced on a new flow cell, which results in a partial recovery of the original sequencing data.

4 Sticky Bits: DNA Data Recovery

In this section we explore the 'stickiness' property of DNA to understand whether a DNA sample can be sequenced a second time, from a used flow cell. We hypothesized that a significant quantity of residual DNA will remain on a flow cell after sequencing, which may be recovered and resequenced. To better understand this phenomena, we developed a proposed method to recover residual DNA on a used flow cell and then quantified how much data from the original sequencing run could be recovered. The proposed method is easy to perform, and we find that it is possible to recover significant residual information. (See Figure 2 for an illustration.) We conclude with a discussion of the security implications of resequencing and cover potential mitigations.

4.1 Stickiness Creates Residual DNA

We hypothesized that residual DNA might exist and be collected from a single-use flow cell for several reasons. First, DNA's high stability, even in dried conditions, might leave DNA on the flow cell interior surfaces for weeks or longer; moreover, DNA may remain attached because the flow cell surface is designed to stick to DNA. The other hint is that reusable flow cells used in alternative long-read sequencing technologies, like the Oxford Nanopore MinION, have reported potential issues of cross contamination between sequencing runs on the same flow cell, which suggests that there is enough residual material left to impact sequencing [32]. However, it was not clear how much of this residual DNA would remain on the single-use

Fig. 3. Left: iSeq 100 flow cell. **Right**: NextSeq 500 Mid Flow Cell.

flow cells used by sequencing-by-synthesis sequencers and whether residual DNA could be easily recovered. Further, the sequencing-by-synthesis process is partially destructive because it incorporates non-standard DNA bases into strands and standard wash procedures in the sequencer may flush away most of the residual DNA after sequencing. For these reasons, an Illumina technical representative told us recovery was unlikely to work. To our knowledge, there have been no considerations of this phenomenon from a security perspective.

In this work, we studied one-time use flow cells designed for sequencing-by-synthesis from Illumina (see Figure 3 for a picture of Illumina flow cells we studied).

4.1.1 Residual DNA Collection

A DNA sample is loaded through an input port on the flow cell and drained through a separate drainage port during sequencing. We suspected that water could be manually injected through this channel after sequencing to collect residual DNA.

To quantify how much DNA could be recovered via this flushing method, we studied two classes of Illumina flow cells, patterned and non-patterned, designed for the NextSeq 500 and iSeq 100, respectively. We first took a used iSeq flow cell and flushed it successively with 20 μL of water, collecting the runoff liquid each time. Since both ends of the DNA fragments are appended with publicly known flow cell adapter sequences, we can use quantitative PCR (qPCR) to roughly estimate how much DNA was recovered in each flush by comparison to a standard control. The first three flushes contained a significant quantity of residual DNA. By the sixth flush we were unable to detect any more DNA (Figure 4-Left). The second flush contained more DNA than the first one, likely because the first flush included excess fluid left from the prior sequencing run. Even though we could detect no DNA after 6 flushes, it is possible trace amounts of DNA remain on the flow cell surface, but it may need to be collected using different methods.

When we aggregated the flushes for a single flow cell we were able to recover around 400 pg of DNA per flow cell. We next tested if the same DNA extraction approach worked on the non-patterned flow cell type. To do this, we similarly flushed one lane of a used NextSeq 500 flow cell and used the flushed water as input to a standard PCR reaction. After PCR, we performed gel electrophoresis on the sample to visualize if we were able to amplify any of the residual DNA. The gel results confirmed that we were able to amplify DNA of the expected length from the flushed water, again using the Illumina adapter sequences as primers (Figure 4-Right). Since the NextSeq 550 uses the same post-wash step as the clinically approved NextSeq 500DX system, these results suggest that residual DNA recovery could be an issue in medical applications. See Appendix A for experimental methods used in this section.

4.1.2 Information in Residual DNA

The previous findings show that residual DNA containing the adapter sequences could be recovered from different models of used flow cells, but those results are not sufficient to determine whether the residual DNA contains useful information from a previous sequencing run. For example, the residual DNA could have low diversity because it originated from only a few starting strands that were highly amplified on the flow cell during sequencing; the DNA strands could have a high mutation rate, thereby losing most of the original information; or the residual DNA could have come from sources other than the original sample, e.g., PhiX viral DNA that is added to sequencing runs as a quality control. To quantify how much information could be obtained from a used flow cell, we experimented with different levels of physical sample redundancy: one was highly redundant (a DNA data storage file) and the other had low redundancy (a human genome sequenced at low coverage).

High Redundancy: DNA data storage systems encode digital files in mixtures (pools) of synthetic DNA molecules. For some digital storage schemes, the sequences which encode that file may be known in ad-

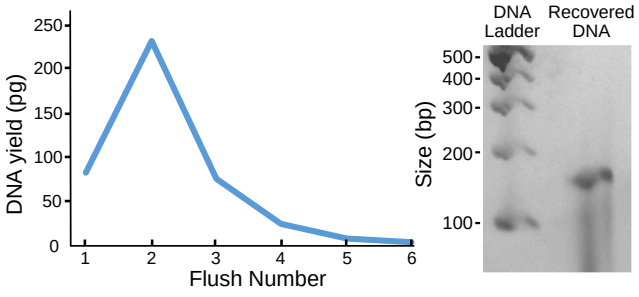

Fig. 4. *Quantifying residual DNA recovery from used flow cells* — **Left**: Quantity of DNA (in pg) recovered from a used iSeq flow cell in each successive wash. **Right**: Agarose gel electrophoresis of total residual DNA collected from a NextSeq 500 mid flow cell (expected product size is 175 bp).

Fig. 5. Images encoded in DNA data storage pool.

vance. This makes it possible to precisely quantify how much of a sample can be recovered, since the expected sequences are known. DNA-encoded files also include extra redundancy so that files can be reliably recovered, even in the presence of errors or missing strands. This makes DNA encoded 'files' a good candidate to study residual recovery under high redundancy conditions.

We began with a DNA pool of images (i.e., collection of images stored in DNA), obtained from the authors of an existing DNA data storage system (Figure 5) [35]. We sequenced the image pool with an iSeq using standard sequencing protocols, and as anticipated, the resulting sequencing data decoded into the correct images. The next day, we gathered residual DNA from the used iSeq flow cell according to the previously described collection procedure, amplified it using PCR to increase its yield, and prepped the DNA for sequencing as before. Finally, the amplified residual DNA was sequenced on a new iSeq flow cell so we could quantify how much of the original file was recoverable.

As discussed, DNA storage encoding schemes are designed with redundancy and error correction, therefore, it is possible to completely recover a file even when strands are missing or have errors. In this experiment, we found that the residual DNA data was sufficient to fully recover all the images with perfect fidelity. Both the original and residual sequencing run had similar levels of strand recovery: 96.6% and 96.5% of the expected strands were found, respectively. This suggests that in highly redundant samples, like data pools, at least one copy of each expected strand can be recovered. However, the DNA that was recovered from the residual sample had a significantly higher error rate (see Supplement C for additional error details). In particular, the substitution-based error rate was significantly higher — approximately 0.1% vs 0.7% substitution errors per base — but the insertion and deletion error rates were only slightly more. We hypothesize that the higher error rates could be the result of chemical degradation of the DNA bases (such as depurination[1]), DNA base modifications resulting from the sequencing chemistry, and additional rounds of amplification during sample preparation and sequencing.

In the case of the image pool, the higher error rate in the residual DNA sequences was not sufficient to prevent proper decoding of the file. High sequencing redundancy is common with many sequencing applications to increase accuracy. For example, with whole genome sequencing, it is recommended to have 30X-50X average coverage of every base in the genome. Therefore, as we saw with the image pool, redundancy may make efficient recovery possible in other applications (e.g., whole genome sequencing), even in the presence of higher errors. We study this next.

Low Redundancy: Here, we explore residual DNA recovery in a more traditional sequencing application: whole genome sequencing. To keep the conditions consistent, we studied recovery using the same type of iSeq patterned flow cell. The iSeq is a low throughput sequencing instrument that produces significantly less data than required for 30-50X coverage typically used with whole genome sequencing. As a consequence, there was much less redundancy in the sample than normal, which let us explore the limits of residual recovery in low redundancy samples.

We ordered a human genome sample from the Coriell Institute (GM12878) to evaluate whole genome

1 Loss of an A or G from the DNA backbone caused by hydrolysis.

sequencing recovery [13]. Communications with our IRB determined that this did not require human subjects review because the sample and data are publicly available but the sample is not identified. The whole genome sample was prepared for sequencing using standard protocols as before, and sequenced with an iSeq. After alignment to the reference human genome (hg38), the average genome-wide coverage (average number of reads overlapping each base in the genome) was low (0.26X). Using the same collection procedure as before, we extracted residual DNA from the used flow cell the next day and resequenced it. The resequenced sample had a much higher proportion of duplicate reads than the original sample: 9.7% were duplicates in the original vs 58.7% in the resequenced (Figure 6-Left). Note that in genomic analysis, two reads are considered PCR duplicates if the beginning of the reads align to the same base of the reference genome. This is because such duplicates are likely to have originated from the same source strand; however, two duplicate reads can differ in their exact base calls (e.g., due to sequencing error).

The higher rate of duplication is somewhat expected because the DNA has been amplified multiple times: first when the original sample is being sequenced on the initial flow cell, using the sequencing-by-synthesis process (which includes a phase to amplify DNA into clusters), and when the residual DNA from the used flow cell is amplified a second time before resequencing. Repeated amplification helps to increase the residual DNA yield but is likely to reduce the underlying sequence diversity and create duplication.

After removing duplicate reads, the original sample had 4.4 million unique reads and the resequenced one had 1.8 million. Of the 4.4 million unique reads originally sequenced, we recovered approximately 1.0 million of them (23.4%) from the used flow cell. Interestingly, there were a significant number of reads in the resequenced residual sample (~780,000) that never appeared in the original one (Figure 6-Right). This could be caused by residual DNA on the flow cell that was not sequenced or was low quality and filtered out by the sequencing software. The coverage of the resequenced sample was also 3.25 times lower than the original one (see Figure 7 for coverage distribution of both samples). If we extrapolate this to normal coverage whole genomes (30X-50X) we could expect that resequencing residual DNA could achieve 10X-15X coverage.

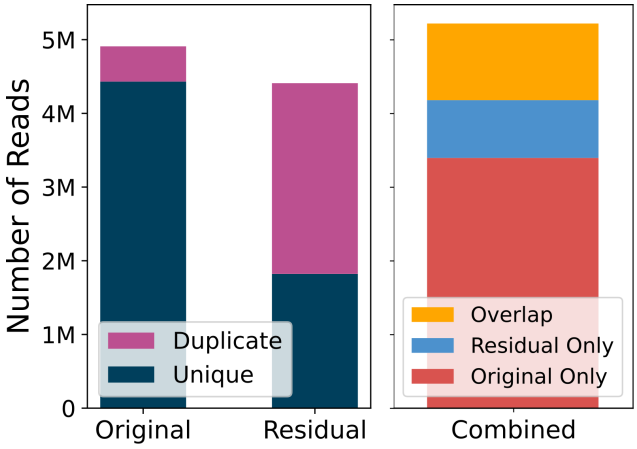

Fig. 6. *Sequencing reads in the original and residual sequencing run* — **Left**: Number of unique and duplicate reads in the original and residual sequencing run. **Right**: Number of unique reads differing and in common between the two sequencing runs.

4.2 DNA Data Recovery and Privacy Leaks

Given that a significant portion of sequenced DNA information can be recovered from used flow cells via residual DNA recovery, we next explore potential privacy risks arising from this phenomena and possible mitigations to these issues.

4.2.1 Security Risks

Presently, there are no general recommendations for used flow cell disposal. Illumina does not suggest a particular method to safely dispose of the flow cell cartridge and instead recommends that users work with their local Environmental Health and Safety officials to develop proper disposal protocols due to local waste disposal regulations (confirmed via correspondence with Illumina). As a consequence, used flow cells could be found anywhere from biohazard waste, glass or sharps disposal, or regular non-hazardous garbage. PacBio, another sequencing machine manufacturer, provides documentation stating that their flow cells are not expected to release any hazardous substances, even upon disposal, and so disposal is unregulated [37]. Some companies even suggest returning flow cells for recycling [31].

Residual recovery gives an adversary a relatively simple method to recover potentially sensitive data from a used flow cell. For example, sequencing is routinely done for medical or research purposes, which can contain sensitive health data or intellectual property [44].

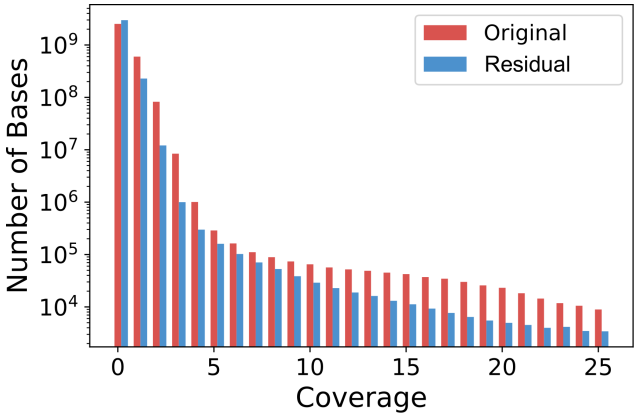

Fig. 7. Number of bases in the human genome with a given coverage in the original and residual sequencing run.

An obvious risk is that medical or patient genotypes could be recovered from a discarded medical flow cell. Such a flow cell could contain recoverable private genetic information or be used to identify a patient (e.g., using [8]), which could link an individual to a study or sensitive medical group. These problems may raise regulatory issues if medical information is not properly disposed. Similarly, DNA containing proprietary sequences, like a new Genetically Modified Organism (GMO), can also be sequenced for research purposes.

4.2.2 Mitigations

Defense against data remanence in other domains has relied on a number of methods including physically destroying the storage medium, using techniques to securely erase or overwrite data (e.g., Gutmann method), or encryption of the data before storage [15]. Our findings suggest that now is the time to develop best practices for addressing the risks with residual DNA on flow cells. We discuss analogous possibilities in the DNA flow cell context below:

- **Destruction:** The safest way to destroy remaining DNA information is to physically destroy the residual DNA. Mechanical flow cell destruction is not an adequate solution, as DNA could still be recovered from fractured flow cell surfaces. Thermal treatment, for example via baking, will be a much more effective way to destroy the underlying DNA molecules as complete degradation of DNA can be achieved in as little as five minutes at 190 degrees C [23]. Chemical treatment, like flushing the flow cell with bleach, may also be effective [39]. Even thorough flushing of the flow cell with water after sequencing may be sufficient to remove most residual DNA. In our experiments, six flushes was sufficient to remove detectable quantities of DNA.

- **Erasure:** A separate approach that could be done in conjunction with destruction is to wash the flow cell with random DNA after sequencing. In effect, random DNA from a similar distribution as the sequenced sample could act like 'cover traffic' — e.g., an 'erasure' mixture of human genomes prepared in the same manner as the original sample. This could add additional levels of obfuscation if a small quantity of DNA remained after destruction. We emphasize that this approach would need to solve a number of challenges to be effective. For example, if the sequences in the erasure mixture are known to an adversary, then it may be possible to filter out the known sequences, leaving just the original sample. Further, even if the erasure sample was random, it would need to come from the same distribution as the samples being sequenced (e.g., appear like a random human genome) otherwise "random" looking reads could be trivially filtered out.

- **Encryption:** Domains like DNA data storage are a natural use case for encryption, since the data that is read (i.e., DNA) can be arbitrarily encoded. Encryption also provides other benefits, like randomization, which makes DNA sequences easier to synthesize. However, we note that encryption does not solve privacy issues when sequencing DNA originating from biological samples, which is naturally fixed, and thus, cannot be encrypted like synthetic DNA encoding a file.

4.3 Summary

In this section we experimented with flow cells to understand whether residual DNA molecules and resulting data could be recovered from used flow cells. We found that significant quantities of residual DNA could be collected from two classes of flow cells (designed for the Illumina NextSeq and iSeq) and quantified how much of the information could be recovered. We find that the residual DNA recovery from used flow cells is sufficient to cause significant privacy risks that could affect important sequencing applications like medicine or genomics. These results show how sequencing flow cell design and the sticky nature of DNA molecules can lead to familiar

data recovery and secure deletion issues that have been seen in other computing domains.

5 Leaky bits: Molecular Data Corruption

Here we study the 'leakiness' of DNA data caused by index cross-talk that happens when multiple samples are sequenced together (recall Section 2 and Stage-2 and Stage-8 of Figure 1). While index cross-talk is known to lower data quality, we suspected that it could be used for targeted data corruption. Prior work has observed cross-talk between samples and suggested that it could create privacy issues, but to our knowledge, the security implications of index cross-talk have not been investigated [33]. In this section, we consider additional risks from index cross-talk, namely, that a DNA sample may be able to leverage cross-talk to influence other co-sequenced samples in a directed and controlled manner. We experiment with an artificially designed DNA sample and evaluate how it can be used to alter the genetic interpretation of other samples that are sequenced together, which has implications for third-party DNA sequencing. Specifically, in our experiments, we study the adversarial possibility of changing someone's sickle cell condition from "normal" to being a sickle cell carrier.

5.1 Altering Samples with Index Cross-Talk

As previously described, index cross-talk happens when a small fraction (0.01%-1.0%) of DNA reads are assigned to the wrong sample when multiple samples are sequenced together on the same flow cell. While not normally an issue, this effect has caused problems in sequencing applications where the samples have low coverage and are very sensitive to noise (e.g., single-cell sequencing) [25, 43]. For example, in these high-sensitivity applications, a small amount of data contamination can create false correlations between samples in downstream bioinformatics analysis. In more typical genomic processing, like variant calling, cross-talk is not an issue because the read coverage is high and the tools are designed to be robust to random sequencing noise. However, this reasoning assumes that the reads leaking between samples are randomly distributed from diverse samples, and as a consequence, no location in the genome is significantly impacted. For example, if two human genomes are sequenced together, cross-talk between them will come from DNA reads randomly sampled from across the genome, which will result in at most a few reads covering any given portion of the genome — an amount unlikely to affect genomic analysis with moderate coverage.

We hypothesized that an atypical, adversarially-created sample with low diversity could affect the genetic interpretation of other samples — even in applications, like variant calling, that are robust to noise — because the reads leaked between samples are not randomly distributed. Further, we suspected that any desired variant could be altered by leveraging cross-talk in this manner.

To test this possibility, we set out to design and create a DNA sample to look like a well known human genetic variant, in this case, the most common mutation responsible for sickle cell disease. This mutation is a type of single base substitution called a single nucleotide polymorphism (SNP). In humans the sickle cell SNP is in the *β-globin* gene and has two alleles (or variants): wild-type (i.e., normal) A and sickle cell T. Every person has two copies of each gene, so an individual can be AA (homozygous dominant: normal), AT (heterozygous: sickle cell carrier), and TT (homozygous recessive: sickle cell disease). Our goal was to study, experimentally, whether index cross-talk from the designed sickle cell sample could be sufficient to alter the sickle cell variant called in a wild-type human genome sample that was sequenced concurrently. Doing so would change the diagnosis of an individual from healthy (AA) to either a sickle cell carrier (AT) or sickle cell disease (TT).

5.1.1 Sample Design

Variants, like the sickle cell SNP, are identified in a human sample by aligning all reads to the reference human genome and identifying those that overlap with the variant of interest. The sequence of the aligned reads are compared against the reference sequence to find any differences; the more reads that contain that change, the more likely the variant is real and not the result of noise. In practice, variant calling algorithms also make adjustments to correct for sequencing errors and known statistical priors. To give some intuition, if 60 reads overlap a SNP in the genome and 29 contain a C at that position and the other 31 contain a G, then the SNP is likely heterozygous CG in that individual.

If we construct synthetic DNA molecules containing the sequence of the variant and surrounding bases from

the reference genome, and then sequence this synthetic strand, the resulting reads will align to the variant's location in the genome and would contribute to a variant call. In the case of the sickle cell variant, this would be a strand containing the sickle cell mutation (T) in the middle surrounded by flanking sequences from the reference. If a sample contains many copies of this synthetic fragment, and the sample is co-sequenced alongside other samples, then even a small quantity of cross-talk may be sufficient to modify the variant which is called in the other samples (e.g., sickle cell trait). See Figure 8 for an illustration.

In practice, this strategy will not work because downstream programs are designed to identify and remove duplicate reads before variant calling to reduce noise. (As discussed in Section 4, this is because duplicate reads are likely artifacts from amplification and not reflective of true variation in the sample.) If just one sickle cell DNA strand was designed, all reads would align to the same position in the genome and all but one would be filtered out as a duplicate. Therefore, for this attack to work many distinct strands must be used, each which aligns to a different location.

We can easily solve this by designing a longer fragment containing the 200 bp on either side of the sickle cell SNP locus. A synthetic DNA mixture of this sickle cell fragment can then be randomly sheared (physically broken) so there will be many unique fragments surrounding this locus, which will each be treated as unique reads and not be filtered out as duplicates. We choose 200bp for the fragment length because fragmentation produces DNA strands around 150 bases, and so most random fragments will contain the variant position.

We ordered the 400bp sickle-cell synthetic fragment from a third-party synthetic gene service and prepared it as discussed above. See Appendix B for the experimental methods used in this section.

5.1.2 Multiplex Sequencing and Downstream Analysis

Next, we prepared a two sample sequencing experiment to test whether the previously designed synthetic sickle cell sample would, in fact, alter the sickle cell SNP called in another independent sample. We ordered a human whole exome sample from a person that was sickle cell wild-type (AA). This is the same anonymous individual (female) from Utah used in Section 4 whose genome has been used extensively in sequencing studies [13]. The whole exome sample was prepared for sequencing under one sequencing index and the sickle cell fragment

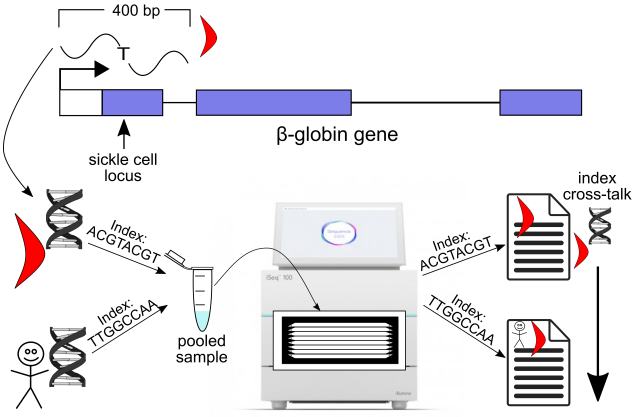

Fig. 8. *Altering genetic variants using the index cross-talk* — A synthetic sample is designed to look like a variant of interest (e.g., sickle cell) and surrounding region in the genome. The designed sample is pooled and sequenced with a target sample. Reads leak from the synthetic sample into the target one due to index cross-talk and alter the variant that is called (e.g., homozygous wild-type (AA) to sickle cell trait (AT)). See also Figure 1.

was prepared under another index, as discussed in Section 2. The two samples were then pooled and multiplex sequenced on an Illumina NextSeq 500, which uses a non-patterned flow cell known to have low to moderate amounts of index cross-talk [25, 46].

The resulting sequencing data was demultiplexed into two sample files using the corresponding indexes for the two samples (see Appendix C for sequencing statistics). If the variant calling resulted in the Utah individual having a sickle cell variant call of AT (heterozygous) or TT (homozygous recessive) then cross-talk between the samples was sufficient to alter the variant.

We ran the demultiplexed Utah exome data through the GATK variant calling analysis pipeline according to their recommendations; this pipeline is designed to identify probable variants across the genome. The sickle cell SNP was altered to a heterozygous call — AT (i.e., a recessive carrier of sickle cell disease), but nearby variants did not change. The manipulated heterozygous variant passed all quality filters and had a high quality score (2281 phred).

To summarize, our findings suggest that an adversary can use cross-talk to adversarially influence the output of DNA sequencing processes, in our case making it seem as though a victim has a false genetic variant. Having found this to be possible, we now explore this attack vector in more depth.

Sample Info	
Sickle Cell Locus	**chr11:5227002 (hg38)**
dbSNP ID	**rs334**
Utah Sample Genotype	**(A/A)**
Sickle Cell Sample Genotype	**(T/T)**
Variant Calling Results	
Called Variant	**(A/T)**
Variant Quality (phred)	**2281**
Total Read Depth	**820**
Depth with Base A	**242**
Depth with Base T	**559**

Table 1. Cross-Talk Experimental Summary.

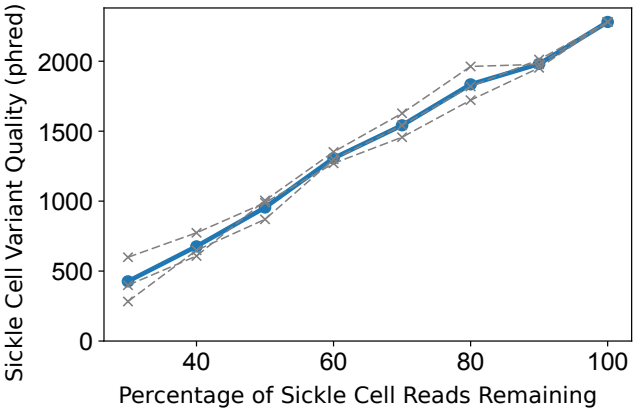

Fig. 9. *Amount of cross-talk necessary to modify a variant* — Reads that aligned to the sickle cell locus with the sickle-cell mutation were removed randomly in varying proportions to simulate lower levels of cross-talk. Simulations were run three times every at 10% intervals, each simulation is shown by the grey dotted lines; average by the blue line.

5.1.3 Cross-Talk at the Sickle Cell Locus

The average coverage of the Utah sample was higher than normal for exome sequencing (321X average), since we only sequenced two samples on a high throughput sequencing instrument. We suspect that the high coverage in our experiments made altering variants even more difficult because there are more legitimate reads covering each genetic loci, and therefore, more reads would need to leak from other samples to alter variants. When looking at the sickle cell position, the coverage was especially high, likely due to a mixture of legitimate and leaky reads from the sickle cell sample. There were 820 aligned reads, 559 (68%) encoding the sickle cell variant (T), which is quite low in comparison to the 97.2 million total reads in the sickle cell sample and 98.4 million reads in the exome sample (see Table 1 for a summary of results).

Similar to what has been reported in other cross-talk studies [46], the quality of the sickle cell reads and index bases were lower than the normal ones: median read quality was 33 vs 25 (phred) and median index quality was 33 vs 20 (phred) for the wild-type and sickle cell reads, respectively. This suggests a possible method to identify or remove malicious attack reads, which we discuss in Section 5.3.

One consequence of these results is that it may be difficult to alter another sample to be the opposite homozygous variant call (e.g., change from AA to TT) because a sufficient number of normal reads (i.e., non-leaky ones) remain from the original sample to influence the variant call. Therefore, this vulnerability may be more significant for dominant mutations, such as the tumor suppressor gene *BRCA1*, or for decisions where carrier status is important, like parenting decisions.

5.1.4 Cross-Talk Simulations

Through the above study, we found that a very low quantity of leaked sickle cell reads was enough to alter a variant (just 559 out of 98.4 million in the exome sample). Next we wanted to study the limits of cross-talk by simulating even lower levels of leakage.

We can simulate different levels of cross-talk by randomly removing sickle cell reads that appear in the exome DNA data file and calling variants like before. We found that the quality of the sickle cell heterozygous variant that was called was roughly proportional to the number of sickle cell reads that appeared in the data file (Figure 9). A heterozygous sickle-cell variant passed standard quality filters was still called as long as at least 40% of the reads remain, suggesting that cross-talk could be further reduced and still allow variants to be altered.

5.1.5 Extensions to Other Variant Types

Our experimental design only considered the modification of a single SNP. Yet, genomic applications often evaluate more complex variants, like insertions or deletions (indels), or assay multiple variants at once [19]. We believe that our approach will apply generally to these other cases because synthetic DNA strands can be constructed to appear like any variant, and variant calling algorithms rely on similar principles across different variant types. Moreover, more than one variant can be targeted at once by combining multiple attack

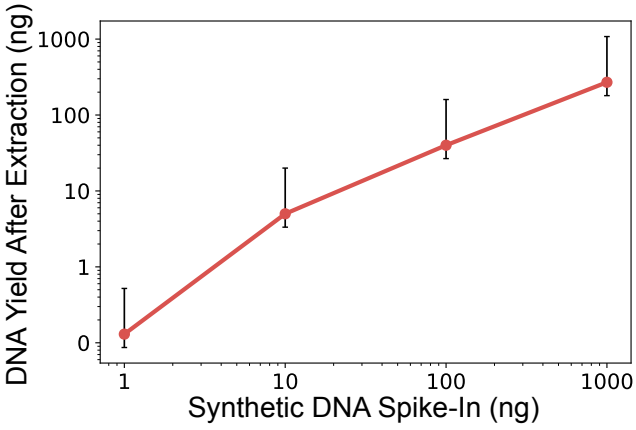

Fig. 10. Yield of spiked-in sickle cell strands recovered from saliva samples after genomic DNA extraction (quantified with qPCR). Error bars signify max uncertainty observed with standard curve control.

strands in the same sample. However, we leave an experimental evaluation of this possibility to future work.

5.2 Security of Third-Party Sequencing

The results of the previous section show how low diversity samples can be used to manipulate the variants called in other co-sequenced samples, where "variants" are biologically-relevant properties (like a genetic disease). We now discuss what security and privacy implications this could have on sequencing.

Ney et al. discussed using this vulnerability to 'read' data that leaked over from other samples [33]. Our results extend on this idea to show that an adversary also has the capability to 'write' specific information into other samples. These capabilities are only relevant in situations where the adversary does not control the sequencer; otherwise, the adversary could more directly attack the sequencing process. Therefore, the index cross-talk vulnerability is most relevant when sequencing is done by third-parties that receive samples from different sources, somewhat analogous to multi-tenancy with untrusted parties in cloud computing. Third-party sequencing is routinely done for research, in places like core facilities or outsourced lab providers, and in sequencing applications like medical testing. Multiplex sequencing is necessary for high sequencing throughput and cost effective sequencing, therefore, we expect it to remain a staple in industry and consumer facing sequencing applications.

The biggest challenge when attacking other samples is that the adversary may not know what other samples are being sequenced, which limits the adversary's ability to target specific samples. In cases like this, the attack would be more akin to a denial-of-service attack where the goal is to corrupt the results of any other samples that are concurrently sequencing, which might harm competitors or patients doing medical testing.

5.2.1 Tissue Sample Spike-In

In some applications, like medical testing, the adversary may be restricted to submitting tissue samples (e.g., blood or saliva), not pure DNA samples like before. One possibility for the adversary is to spike in the malicious DNA into a tissue sample directly (e.g., into saliva or blood). However, it is not clear whether this will work because none of the strands may remain after genomic DNA is extracted from the tissue sample, which is designed to isolate DNA from cells, not DNA in solution. To test this possibility we studied whether the sickle cell DNA strands could be spiked into a saliva sample and survive a genomic DNA purification procedure.

We took DNA from the sickle cell sample that was designed in Section 5.1.1 and spiked it into a saliva sample at different concentrations. DNA from each saliva sample was purified using a stock Qiagen genomic DNA extraction kit. Following purification, we measured the amount of sickle cell DNA remaining in the purified saliva sample using qPCR. The sickle cell strand could be detected in the processed saliva sample at high levels similar to the quantity originally spiked-in (Figure 10). Most importantly, the amount detected was proportional to the amount spiked-in. This suggests that an adversary can tune precisely how much of the synthetic strand to appear in the final purified DNA solution by spiking in DNA at a corresponding concentration.

When spike-in is combined with the ability to corrupt data with index cross-talk, an adversary can attack a wider range of services than before because standard tissue samples can be submitted for sequencing. This is especially relevant now that widely accessible direct-to-consumer tests, provided by companies like Ancestry, are beginning to incorporate next-generation sequencing into their medical products [17].

5.3 Reducing Data Corruption

Here, we consider three defensive strategies to prevent this data corruption vulnerability: minimizing cross-talk, quality filtering, and anomaly detection.

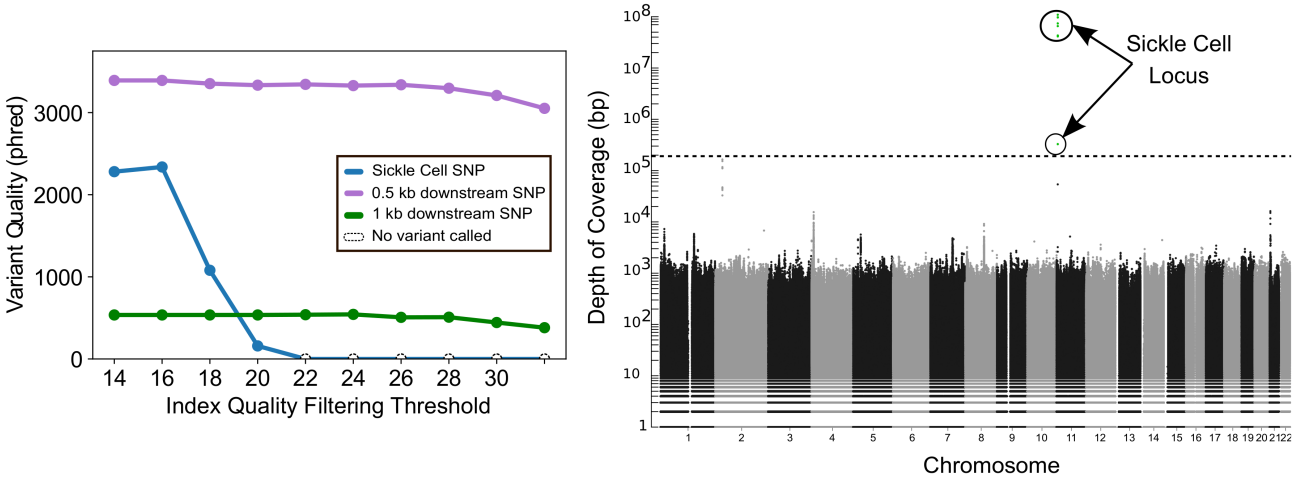

Fig. 11. *Cross-talk mitigations* — **Left**: Variant quality score of the sickle cell and nearby variants when filtering out reads with low index quality scores. No sickle cell variant is called when filtering out reads with an index quality < 22. **Right**: Read depth along the non-sex chromosomes using all reads (no demultiplexing). Each point represents the depth at a single position (points with zero-coverage are not displayed). Dotted horizontal line shows the highest coverage not located at the sickle-cell locus. Circled green points are in the 400 bp region used to design the sickle-cell fragment.

5.3.1 Minimize Index Cross-Talk

The research community has experimented with a number of methods to minimize index cross-talk because of its negative effect on sequencing quality, particularly in error-sensitive applications. These approaches have included: adjusting the indexing scheme to use two unique indexes (unique dual indexing), the use of newer indexes that contain unique molecular identifiers (UMIs), or improving or modifying the sequencing process [7, 22, 25–27, 30]. Unique dual indexes are now optionally supported by Illumina and have been shown reduce the observed cross-talk rate to around 0.001%, which is in line with our observed rate of leaky sickle cell reads on the NextSeq (0.0006%) [7, 20, 21]. As our results demonstrate, low cross-talk at these levels is still sufficient to alter variants, and thus, is not a sufficient defense by itself. However, reducing cross-talk still reduces the overall amount of data that can be corrupted. Thus, while not a complete solution, we recommend that unique dual indexes be used if adversarial manipulation or data theft is a concern.

5.3.2 Quality Filtering

Similar to what has been reported by other groups, both the sequencing bases and bases in the index have a lower quality score than other reads [25, 46]. We wanted to test removing reads with a low index base quality to see if that would disproportionately remove leaky reads while not affecting legitimate variant calls. Using data from the Utah exome sample that was sequenced previously with the sickle cell sample (Section 5.1), we filtered out sequencing reads with low index quality scores at various thresholds and called variants as before. We wanted to see if this would remove the false heterozygous sickle cell variant but leave other variants unaffected. When comparing the variant call at different thresholds, we found that a moderate degree of filtering — removing reads with an average index base quality score less than 22 — was sufficient to remove the false sickle cell variant but left nearby legitimate variants in the genome unaffected (Figure 11-Left). This suggests that some basic filtering by read quality will remove false variants from leaky samples but leave legitimate ones unchanged. However, more work is needed to understand whether this approach would scale to genome-wide variant calling without excessive false positives.

5.3.3 Anomaly Detection

The sequencer can also detect anomalies before returning demultiplexed data. Manipulating variants using cross-talk requires that the malicious sample have especially high coverage at the targeted variants. If the data from the malicious sample is merged with all others, then the combined reads will have abnormally high coverage at the target locus when aligned to a reference.

Reads from abnormally high coverage positions could be removed or flagged as anomalous. In the experimental sequencing run, we found that a number of locations have high coverage (>1000) due to natural variation or biases in amplification or sample preparation. However, the sickle cell locus had a depth of nearly 100 million base pairs, 3 orders of magnitude higher than any other position (Figure 11-Right), making it highly anomalous compared to any other loci. Aligning all samples may be too computationally expensive to use this regularly, but it does give an effective means to identify regions likely to be influenced by leaky reads, regardless of whether it happens accidentally or maliciously.

5.4 Summary

We experimented with multiplex sequencing to show how DNA leakage can cause separate DNA samples to affect each other. We proved that a malicious sample can be designed and sequenced with other samples to create targeted genetic changes in another genome sample. This risk increases because malicious DNA can be spiked directly into tissue samples and survive purification, implying that manipulation may be possible through tissue samples directly. As sequencing becomes more ubiquitous, and eventually a commodity product, we expect that multiplex sequencing will be necessary to achieve cost effective sequencing. Therefore, we believe that vulnerabilities like index cross-talk mediated data corruption are important to consider, especially as sequencing is used by a wider audience in end-user applications, like personalized medicine or consumer testing.

6 Related Work

In recent years DNA-based biotechnology and molecular systems more generally have begun to get more attention from the computer security community. DNA sequencers in particular have gotten significant scrutiny because they play a crucial role in genomics and have complex hardware and software threat surfaces [10, 42]. DNA itself has even been shown to be a vector for possible computer attacks [33]. Similarly, physical biotechnology hardware can also have issues with privacy leaks via information leakage and side-channels. For example, Faezi et al. used audio captured from a DNA synthesis machine to recover the strands that were created with 88% accuracy [9].

Most analogous to the residual DNA data attack on flow cells (Section 4) is secure data deletion on traditional storage devices. Studies have shown that easily obtained used storage hardware, like hard disks, have repeatedly been found to contain sensitive information that was not properly deleted [11, 12]. To deal with these issues a number of methods have been developed to securely delete data from physical media [40], the most famous being the Gutmann method for secure hard drive deletion [15]. An example of data persistence, under specific conditions, is the cold-boot attack [16].

Information leakage and data corruption attacks have a long history of being used to manipulate computers. Most relevant to this work are attacks that affect multi-user systems (e.g., shared cloud environments). In a canonical example, Zhang et al. found that side-channel attacks could be used to extract cryptographic keys from other users in a multi-tenant virtual machine [48]. Data manipulation attacks have also been shown capable of altering a machine's state. For example, row hammer attacks can be used to flip adjacent bits in DRAM, which can be used to elevate privilege or remove memory protections [24, 47]. Although not information leakage between peer-type systems, another related concept is fault-injection attacks, such as those on cryptographic devices [2].

7 Discussion and Conclusions

In this work we have experimented with sequencing flow cells, the center of the molecular-to-digital conversion, and find that the sequencing process can be vulnerable to data remanence and data corruption issues. In particular, it is the unusual properties of molecular processing, namely amplification, stability, mixing, and bonding/reactivity, that make these attacks possible. To our knowledge, these are the first examples to show how the physical properties of molecules can contribute to computer security issues. While we believe it is important to consider emerging molecular security issues as technologies like DNA sequencers become more ubiquitous, we do not believe that these findings require immediate action from Illumina; we have, nevertheless, disclosed our results to Illumina prior to publication.

As technologies like DNA sequencers and other molecular instruments continue to be improved and applied in more computational settings we suspect that more examples like the issues highlighted in this study will arise. In particular, we believe that a computer se-

curity mindset is especially important in new fields like molecular informatics — the use of molecules for storage and computation — that continue to blur the molecular-computational boundary.

Acknowledgements

We thank Sandy Kaplan, the Molecular Information Systems Lab, and the Security and Privacy Research Lab for helpful discussions and comments on this paper. We thank our shepherd, Erman Ayday, and the anonymous reviewers. This research was supported in part by a grant from the DARPA Molecular Informatics Program, NSF Grant CNS-1565252, the University of Washington Tech Policy Lab (which receives support from the William and Flora Hewlett Foundation, the John D. and Catherine T. MacArthur Foundation, Microsoft, the Pierre and Pamela Omidyar Fund at the Silicon Valley Community Foundation), the Short-Dooley Professorship, and the Torode Family Professorship.

References

[1] Joel Armstrong, Ian T. Fiddes, Mark Diekhans, and Benedict Paten. Whole-Genome Alignment and Comparative Annotation. *Annual Review of Animal Biosciences*, 2019.

[2] Alessandro Barenghi, Luca Breveglieri, Israel Koren, and David Naccache. Fault injection attacks on cryptographic devices: Theory, practice, and countermeasures. *Proceedings of the IEEE*, 100(11):3056–3076, 2012.

[3] Zachary S Bohannan and Antonina Mitrofanova. Calling variants in the clinic: Informed variant calling decisions based on biological, clinical, and laboratory variables. *Computational and structural biotechnology journal*, 2019.

[4] Luis Ceze, Jeff Nivala, and Karin Strauss. Molecular digital data storage using DNA. *Nature Reviews Genetics*, 2019.

[5] Weida D. Chen, A. Xavier Kohll, Bichlien H. Nguyen, Julian Koch, Reinhard Heckel, et al. Combining Data Longevity with High Storage Capacity—Layer-by-Layer DNA Encapsulated in Magnetic Nanoparticles. *Advanced Functional Materials*, 2019.

[6] George M Church, Yuan Gao, and Sriram Kosuri. Next-generation digital information storage in dna. *Science*, 337(6102):1628–1628, 2012.

[7] Maura Costello, Mark Fleharty, Justin Abreu, Yossi Farjoun, Steven Ferriera, et al. Characterization and remediation of sample index swaps by non-redundant dual indexing on massively parallel sequencing platforms. *BMC genomics*, 19(1):332, 2018.

[8] Yaniv Erlich, Tal Shor, Itsik Pe'er, and Shai Carmi. Identity inference of genomic data using long-range familial searches. *Science*, 362(6415):690–694, 2018.

[9] Sina Faezi, Sujit Rokka Chhetri, Arnav Vaibhav Malawade, John Charles Chaput, William H Grover, Philip Brisk, and Mohammad Abdullah Al Faruque. Oligo-snoop: A non-invasive side channel attack against dna synthesis machines. In *NDSS*, 2019.

[10] Iliya Fayans, Yair Motro, Lior Rokach, Yossi Oren, and Jacob Moran-Gilad. Cyber security threats in the microbial genomics era: implications for public health. *Eurosurveillance*, 25(6):1900574, 2020.

[11] Simson L Garfinkel. Forensic feature extraction and cross-drive analysis. *digital investigation*, 3:71–81, 2006.

[12] Simson L Garfinkel and Abhi Shelat. Remembrance of data passed: A study of disk sanitization practices. *IEEE Security & Privacy*, 1(1):17–27, 2003.

[13] GM12878. Coriell Institute. https://www.coriell.org/0/Sections/Search/Sample_Detail.aspx?Ref=GM12878.

[14] Nick Goldman, Paul Bertone, Siyuan Chen, Christophe Dessimoz, Emily M. Leproust, Botond Sipos, and Ewan Birney. Towards practical, high-capacity, low-maintenance information storage in synthesized DNA. *Nature*, 2013.

[15] Peter Gutmann. Secure deletion of data from magnetic and solid-state memory. In *Proceedings of the Sixth USENIX Security Symposium, San Jose, CA*, volume 14, pages 77–89, 1996.

[16] J Alex Halderman, Seth D Schoen, Nadia Heninger, William Clarkson, William Paul, et al. Lest we remember: cold-boot attacks on encryption keys. *Communications of the ACM*, 52(5):91–98, 2009.

[17] Matthew Herper. Ancestry launches consumer genetics tests for health, intensifying rivalry with 23andme. Stat, October 16, 2019.

[18] Sequencing coverage for NGS experiments. Illumina. https://www.illumina.com/science/technology/next-generation-sequencing/plan-experiments/coverage.html.

[19] Truesight cystic fibrosis data sheet. Illumina.

[20] Ampliseq for illumina BRCA panel reference guide. Illumina, 2019.

[21] Ampliseq for illumina exome panel reference guide. Illumina, 2019.

[22] Effects of index misassignment on multiplexing and downstream analysis, 2020. https://www.illumina.com/content/dam/illumina-marketing/documents/products/whitepapers/index-hopping-white-paper-770-2017-004.pdf. Accessed: 2020-06-12.

[23] Moshe Karni, Dolev Zidon, Pazit Polak, Zeev Zalevsky, and Orit Shefi. Thermal degradation of DNA. *DNA and Cell Biology*, 2013.

[24] Yoongu Kim, Ross Daly, Jeremie Kim, Chris Fallin, Ji Hye Lee, Donghyuk Lee, Chris Wilkerson, Konrad Lai, and Onur Mutlu. Flipping bits in memory without accessing them: An experimental study of dram disturbance errors. *ACM SIGARCH Computer Architecture News*, 42(3):361–372, 2014.

[25] Martin Kircher, Susanna Sawyer, and Matthias Meyer. Double indexing overcomes inaccuracies in multiplex sequencing on the Illumina platform. *Nucleic acids research*, 40(1), 2012.

[26] Qiaoling Li, Xia Zhao, Wenwei Zhang, Lin Wang, Jingjing Wang, et al. Reliable multiplex sequencing with rare index mis-assignment on DNB-based NGS platform. *BMC*

genomics, 20(1):215, 2019.

[27] Laura E MacConaill, Robert T Burns, Anwesha Nag, Haley A Coleman, Michael K Slevin, et al. Unique, dual-indexed sequencing adapters with umis effectively eliminate index cross-talk and significantly improve sensitivity of massively parallel sequencing. *BMC genomics*, 19(1):30, 2018.

[28] Matthias Meyer and Martin Kircher. Illumina sequencing library preparation for highly multiplexed target capture and sequencing. *Cold Spring Harbor Protocols*, 2010(6), 2010.

[29] Matthias Meyer, Udo Stenzel, Sean Myles, Kay Prüfer, and Michael Hofreiter. Targeted high-throughput sequencing of tagged nucleic acid samples. *Nucleic Acids Research*, 35(15), 2007.

[30] Abhishek Mitra, Magdalena Skrzypczak, Krzysztof Ginalski, and Maga Rowicka. Strategies for achieving high sequencing accuracy for low diversity samples and avoiding sample bleeding using Illumina platform. *PloS one*, 10(4), 2015.

[31] Why do i need to return my flow cells? Nanopore. https://store.nanoporetech.com/us/nanohelp/faq/why-do-i-need-to-return-my-flow-cells.

[32] New kit extends yields of flow cells. https://nanoporetech.com/about-us/news/new-kit-extends-yields-flow-cells. Accessed: 2020-06-12.

[33] Peter Ney, Karl Koscher, Lee Organick, Luis Ceze, and Tadayoshi Kohno. Computer security, privacy, and DNA sequencing: Compromising computers with synthesized DNA, privacy leaks, and more. In *26th USENIX Security Symposium (USENIX Security 17)*, pages 765–779, Vancouver, BC, 2017. USENIX Association.

[34] Novaseq system specifications. https://www.illumina.com/systems/sequencing-platforms/novaseq/specifications.html. Accessed: 2020-06-11.

[35] Lee Organick, Siena Dumas Ang, Yuan Jyue Chen, Randolph Lopez, Sergey Yekhanin, et al. Random access in large-scale DNA data storage. *Nature Biotechnology*, 2018.

[36] Lee Organick, Yuan Jyue Chen, Siena Dumas Ang, Randolph Lopez, Xiaomeng Liu, et al. Probing the physical limits of reliable DNA data retrieval. *Nature Communications*, 2020.

[37] Smrt cell 8m tray safety data sheet. PacBio, 2019. https://www.pacb.com/wp-content/uploads/SDS-SMRT-Cell-8M-Tray.pdf.

[38] Brent S Pedersen and Aaron R Quinlan. Mosdepth: quick coverage calculation for genomes and exomes. *Bioinformatics*, 34(5):867–868, 2018.

[39] A. M. Prince and L. Andrus. PCR: How to kill unwanted DNA. *BioTechniques*, 1992.

[40] Joel Reardon, David Basin, and Srdjan Capkun. Sok: Secure data deletion. In *2013 IEEE symposium on security and privacy*, pages 301–315. IEEE, 2013.

[41] Antonio Regalado. China's bgi says it can sequence a genome for just $100. MIT Technology Review, February 26, 2020. https://www.technologyreview.com/2020/02/26/905658/china-bgi-100-dollar-genome/. Accessed: 2020-06-12.

[42] Garrett J Schumacher, Sterling Sawaya, Demetrius Nelson, and Aaron J Hansen. Genetic information insecurity as state of the art. *bioRxiv*, 2020.

[43] Rahul Sinha, Geoff Stanley, Gunsagar Singh Gulati, Camille Ezran, Kyle Joseph Travaglini, et al. Index switching causes "spreading-of-signal" among multiplexed samples in Illumina HiSeq 4000 DNA sequencing. *BioRxiv*, 2017. https://doi.org/10.1101/125724.

[44] Wendy Weijia Soon, Manoj Hariharan, and Michael P Snyder. High-throughput sequencing for biology and medicine. *Molecular systems biology*, 9(1), 2013.

[45] Julie Utterback. Illumina remains the clear leader of the genomic sequencing market. Morningstar, April 30, 2020.

[46] Erik Scott Wright and Kalin Horen Vetsigian. Quality filtering of Illumina index reads mitigates sample cross-talk. *BMC genomics*, 17(1):876, 2016.

[47] Yuan Xiao, Xiaokuan Zhang, Yinqian Zhang, and Radu Teodorescu. One bit flips, one cloud flops: Cross-vm row hammer attacks and privilege escalation. In *25th USENIX Security Symposium (USENIX Security 16)*, pages 19–35, 2016.

[48] Yinqian Zhang, Ari Juels, Michael K Reiter, and Thomas Ristenpart. Cross-vm side channels and their use to extract private keys. In *Proceedings of the 2012 ACM conference on Computer and communications security*, pages 305–316, 2012.

A DNA Data Recovery Experimental Methods

Below are the wet lab and analysis protocols used for experiments in Section 4.

A.1 DNA Extraction from Used Flow Cells

Used flow cells were removed from the sequencing machine after their initial sequencing run and kept at room temperature for 1 to 7 days before extraction of the residual DNA was performed. Residual DNA recovery was tested with a used iSeq 100 (i1 kit) and NextSeq 500 v2.5 (mid kit) flow cells. Residual DNA extraction was performed by using a pipette with a P20 or P200 tip to flush molecular biology grade water into one of the two flow cell ports. This was typically done in increments of 20 microliters, that is, 20 microliters of water was injected into one of the ports and the liquid flow through coming ejected out of the other port was collected in a clean 1.5 mL Eppendorf tube. Five to six 20 uL flushes were typically performed for each flow cell. After collection of all the flushes, the DNA in the samples was either quantified using qPCR (with forward P5 primer 5'AATGATACGGCGACCACCGA, and reverse P7 primer 5'CAAGCAGAAGACGGCATACGAGAT), analyzed by agarose gel electrophoresis, and/or pooled into a single sample to be resequenced.

A.2 Original Sequencing Runs (Image Pool and Whole Genome Sample)

A single file encoding 10 images was selected out of a larger image pool by PCR. Purified genomic DNA for the whole exome sample was obtained from Coriell Institute (sample NA12878). A sample of this genomic DNA was then sheared for sequencing using NEBNext dsDNA Fragmentase (New England Biolabs). Both the DNA data sample and fragmented genomic samples were prepped for sequencing with Illumina adapters using NEBNext Ultra II DNA Library Prep Kit (NEB) according to the manufacturer's protocol. The prepped libraries were then sequenced using an Illumina iSeq 100 i1 Reagents kit according to the manufacturer's protocol. The image pool was sequenced with dual 8bp indexes and single-ended 177 cycling protocol (the length of the synthetic fragments in the pool is 150bp) and the whole genome was sequenced using dual 8bp indexes and a single-ended 300 cycling protocol.

A.3 Resequencing Residual DNA

Following collection of the residual DNA from a used flow cell, 5 ul was used as input to a 50 uL total volume PCR reaction (Kappa Systems) and amplified for 25 cycles with an annealing temperature of 58 degrees Celsius. After PCR, the reaction was column purified with QIAquick PCR purification kit (Qiagen). The purified PCR product was then quantified (Nanodrop) and sequenced using an Illumina iSeq 100 i1 according to the manufacturer's protocol using the same dual 8bp indexes and 300 cycle single direction protocol as before.

A.4 Analysis

Reads were aligned to the human genome (GRChg38) using bwa-mem (v0.7.15). PCR duplicates were marked using the Picard MarkDuplicates utility (v2.9.0). Coverage analysis was done using mosdepth (v0.2.9) [38].

B Molecular Cross-Talk Experimental Methods

Below are the wet lab and analysis protocols used for experiments in Section 5.

B.1 Library Preparation

The sickle-cell ultramer and primers for amplification were ordered from IDT (see Supplementary Table 2 for sequence and primers). It amplified with primers, 100 μL of 2x Kapa HiFi enzyme mix, 80 μL of molecular grade water, 5 μL of each primer at 10μM diluted in 1x TE buffer, and 10 μL of the synthesized ultramer at 1 ng/μL diluted with 1x TE buffer, for a mixture totalling 200 μL. The mixture was vortexed on a benchtop vortexer for 10 seconds, then split into two 0.2 mL PCR tubes and placed in the thermocycler with the following protocol: (1) 95 °C for 3 min, (2) 98 °C for 20 s, (3) 60 °C for 20 s, (4) 72 °C for 30 s, (5) go to step (2) 11 additional times for a total of 12 cycles, and (6) 72 °C for 30 s. The resulting product had no side products when examined with a QIAGEN QIAxcel fragment analyzer, and it was approximately 165 ng/μL.

The human genome NA12878 was ordered through Coriell Institute and was not modified prior to shipping to Genewiz for library preparation.

Both the whole genome and the sickle-cell amplicon were sent to Genewiz for further preparation. The whole genome was prepared with the Agilent SureSelect Exome library preparation kit (v6) to prepare only the exome for sequencing, using index A11 with sequence *CCAGTTCA*. (A single index was used because at the time commercial exome library preparation kits supported only single indexes.) Fragment sizes ranged from 290 bp to 784 bp as measured by the Quiagen Fragment Analyzer. The amplicon was prepared with fragmentation using the NexteraXT kit, using index N703 with sequence *AGGCAGAA* and index S516 with sequence *ACTCTAGG*. Fragment sizes ranged from 168 bp to 608 bp (using the Quiagen Fragment Analyzer).

B.2 Sequencing

The prepared exome and sickle-cell samples were found to be 6.2 ng/μL (23 nM) and 2.2 ng/μL (11nM), respectively, with the Qubit 3.0 fluorometer. The run was 48 percent exome sample (0.9μL) and 48 percent amplicon sample (2μL), with a 4 percent PhiX spike-in as a sequencing control. Samples were diluted and denatured prior to sequencing using the NextSeq System Denature and Dilute Libraries Guide. Sequencing was done on the NextSeq 500 and used a 300 cycle Mid kit (flow cell v2), with 150 cycles in each read and two 8 bp index reads.

B.3 Downstream Processing

All reads were demultiplexed with the Illumina bcl2fastq conversion software (v2.20.0) using the default configuration (one base pair mismatches was allowed). The create-fastq-for-index-reads flag was used to retrieve index quality scores. The exome sample was demuxed with (i7:CCAGTTCA) and the sickle-cell sample with (i7:AGGCAGAA; i5:ACTCTAGG).

To call all variants, reads were aligned to the human genome (GRChg38) using bwa-mem (v0.7.15). PCR and optical duplicates were removed with the Picard MarkDuplicates utility (v2.9.0). Base scores were recalibrated with GATK (v3.7) BaseRecalibrator with the following vcf files from the GATK resource bundle: dbSNP v138, OMNI 2.5, HapMap 3.3, and Mills and 1000G Gold Standard Indels. Variants were called with GATK HaplotypeCaller in discovery mode and SNPs were hard filtered according to GATK's generic filtering recommendations (`QD < 2.0`, `MQ < 40.0`, `FS > 60.0`, `SOR > 3.0`, `MQRankSum < -12.5`, and `ReadPosRankSum < -8.0`). Exome coverage was computed using the bedtools coverage utility (v2.25.0) with the Agilent SureSelect Exome v6 bed files.

B.4 Variant Quality Simulation

Reads containing the sickle-cell SNP were any that covered the rs334 position (chr11:5227002) after alignment and had the sickle-cell base (T) at that position. All such reads were identified and removed in varying proportions from the demuxed FASTQ file to simulate lower levels of index misassignment. For example, to simulate 90% levels of misassignment, 10% of the sickle-cell reads (rounded up) were removed, at random, from the FASTQ file. Then the reads were aligned and variants called on the FASTQ file as usual. Simulation was run every 10% from 0-100% three times with a different random seed each time.

B.5 Defenses

To filter out reads based on index quality, the average i7 base phred quality score was computed for each read pair. Any reads which were less than the given quality threshold were removed from the FASTQ file. The remaining reads, which passed the i7 quality threshold, were aligned and had variants called. Variants were called using even quality filter thresholds from 14-32.

B.6 Saliva Spike-In

Four 1 mL samples of saliva were collected in 1.5 mL tubes and varying amounts of the sickle cell-encoding synthetic DNA fragment was added to each sample (1ug, 100 ng, 10 ng, and 1 ng). The DNA from each sample was then extracted using a QIAamp DNA Blood Minikit using the manufacturer's recommended protocol for saliva. The concentration of the sickle cell fragment in the final elution step was then quantified by qPCR using primers specific for the fragment (Table 2).

Sickle-Cell DNA Fragment and Primer Sequences	
Sickle-Cell Oligo	5′ AAGGGTGGGAAAATAGACCAATAGGCAG AGAGAGTCAGTGCCTATCAGAAACCCAAGA GTCTTCTCTGTCTCCACATGCCCAGTTTCT ATTGGTCTCCTTAAACCTGTCTTGTAACCT TGATACCAACCTGCCCAGGGCCTCACCACC AACTTCATCCACGTTCACCTTGCCCCACAG GGCAGTAACGGCAGACTTCTCCACAGGAGT CAGATGCACCATGGTGTCTGTTTGAGGTTG CTAGTGAACACAGTTGTGTCAGAAGCAAAT GTAAGCAATAGATGGCTCTGCCCTGACTTT TATGCCCAGCCCTGGCTCCTGCCCTCCCTG CTCCTGGGAGTAGATTGGCCAACCCTAGGG TGTGGCTCCACAGGGTGAGGTCTAAGTGAT GACAGCCGTACC 3′
Forward Primer	5′ AAGGGTGGGAAAATAGACCA 3′
Reverse Primer	5′ GGTACGGCTGTCATCACTTA 3′

Table 2. Sickle-Cell DNA Fragment and Primer Sequences.

C Error and Sequencing Statistics

This supplement contains DNA sequencing statistics (Table 3) and sequencing error rates (Table 4; Figure 12) for the DNA image pool run and corresponding residual DNA resequence from Section 4.1 and the index cross-talk run from Section 5.1, respectively.

D Cross-Talk Quality Scores

Figure 13 is a box plot comparing the quality scores of the sickle cell vs wild type read from the index cross-talk experiments done in Section 5.1.

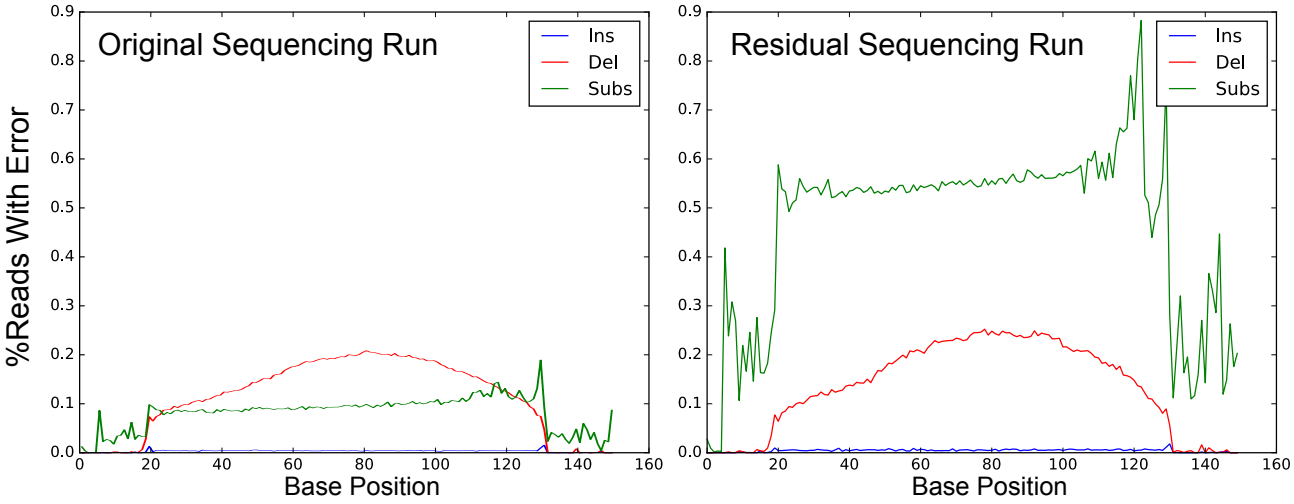

Fig. 12. Insertion (blue), deletion (red), and substitution (green) error rates in the original (left) and residual DNA (right) runs.

Cross-Talk Sequencing Run Metrics	
Total PE Reads (PF)	**209,948,900**
Total Indexed Reads (PF)	**195,632,495 (93.18%)**
%≥Q30	**82.79%**
Exome Sample	
Number PE Reads	**98,448,354 (46.9%)**
Percent Aligned	**99.68%**
Average Insert Size	**147.55**
Average Coverage	**321.42X**
Sickle-Cell Sample	
Number PE Reads	**97,184,141 (46.3%)**
Percent Aligned	**99.38%**
Average Insert Size	**135.68**

Table 3. Sequencing statistics for the index cross-talk sequencing run with a multiplexed sickle cell and Utah exome sample.

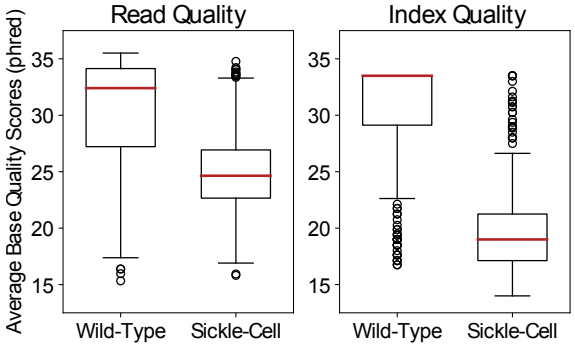

Fig. 13. Box plot of read and index quality scores. Center red line is the median, box limits are the upper and lower quartile, whiskers are 1.5x the interquartile range, and points are outliers.

DNA Data Storage Sequencing Error Rate		
Sequencing Error Statistic	**Original (%)**	**Residual (%)**
Insertion Average	**0.0058**	**0.0068**
Fraction Ins-A	**20.6108**	**25.4876**
Fraction Ins-C	**12.2551**	**12.4957**
Fraction Ins-T	**8.9324**	**11.1448**
Fraction Ins-G	**58.2017**	**50.8719**
Deletion Average	**0.1733**	**0.2031**
Fraction Del-A	**22.9114**	**22.8680**
Fraction Del-C	**28.4231**	**28.6607**
Fraction Del-T	**27.4847**	**27.1486**
Fraction Del-G	**22.1766**	**22.3033**
Substitution Average	**0.1232**	**0.6978**
Fraction Sub A-to-C	**0.7494**	**0.6877**
Fraction Sub A-to-G	**11.1042**	**11.8425**
Fraction Sub A-to-T	**6.7783**	**5.1560**
Fraction Sub C-to-A	**1.8594**	**4.8470**
Fraction Sub C-to-G	**4.2276**	**9.3614**
Fraction Sub C-to-T	**15.2903**	**13.7343**
Fraction Sub G-to-A	**26.6947**	**16.8878**
Fraction Sub G-to-C	**4.1482**	**4.3416**
Fraction Sub G-to-T	**6.4364**	**2.8684**
Fraction Sub T-to-A	**8.4081**	**16.1015**
Fraction Sub T-to-C	**11.3836**	**14.4384**
Fraction Sub T-to-G	**0.3440**	**0.7131**

Table 4. Base error rates for the original and resequenced residual DNA sequencing run. Note the substantially higher substitution error rate ($>$ 5.5X) with the residual sequencing run.