Enabling the Efficient, Dependable Cloud-based Storage of Human Genomes

Vinicius Vielmo Cogo, Alysson Bessani
LASIGE, Faculdade de Ciências, Universidade de Lisboa
Lisboa, Portugal
{vvcogo, anbessani}@ciencias.ulisboa.pt

***Abstract*—Efficiently storing large data sets of human genomes is a long-term ambition from both the research and clinical life sciences communities. For instance, biobanks stock thousands to millions of biological physical samples and have been under pressure to store also their resulting digitized genomes. However, these and other life sciences institutions lack the infrastructure and expertise to efficiently store this data. Cloud computing is a natural economic alternative to private infrastructures, but it is not as good an alternative in terms of security and privacy. In this work, we present an end-to-end composite pipeline intended to enable the efficient, dependable cloud-based storage of human genomes by integrating three mechanisms we have recently proposed. These mechanisms encompass (1) a privacy-sensitivity detector for human genomes, (2) a similarity-based deduplication and delta-encoding algorithm for sequencing data, and (3) an auditability scheme to verify who has effectively read data in storage systems that use secure information dispersal. By integrating them with appropriate storage configurations, one can obtain reasonable privacy protection, security, and dependability guarantees at modest costs (e.g., less than $1/Genome/Year). Our preliminary analysis indicates that this pipeline costs only 3% more than non-replicated systems, 48% less than fully-replicating all data, and 31% less than secure information dispersal schemes.**

***Index Terms*—Data Storage, Dependability, Cloud, Genomes**

I. Introduction

Whole genome sequencing (WGS) is the process of digitizing the complete sequence of nucleotide pairs that compose the DNA stored in a cell of an organism at a specific time [1]. The DNA sequence of every human being has more than 3.2 billion nucleobases (e.g., `A` for adenine, `C` for cytosine, `G` for guanine, and `T` for thymine), which results in more than 300GB of data in the FASTQ format (i.e., the standard raw data format from WGS [2]). Research and clinical life sciences communities will directly benefit from solutions that enable the efficient storage of large data sets of human genomes. For instance, there is a pressure on biobanks to store the digitized genomes from the thousands to millions of biological physical samples they already stock in their infrastructures [3].

Sequencing a human genome currently costs around $1000 and this price is expected to drop even further in the near future [4, 5]. This decreasing cost of DNA sequencing motivates the adoption of sequenced data in routine medical procedures. For instance, personalized medicine brings medical decisions to the individual level propelling the use of specific procedures and treatments for each patient [6]. It may benefit from the expansion of genome sequencing, and individuals may have their cells sequenced multiple times during their lives.

The number of to-be-stored genomes is increasing exponentially [7]. Storing genomes efficiently may accelerate medical breakthroughs since researchers would like to analyse thousands of samples at time. However, this sharing augments the risks for donors' privacy (e.g., DNA may be used to obtain identity- and health-related information [8]). The million-scale size and criticality of sets of genomes require systematic solutions to store and share this data in efficient, scalable, and secure ways [8, 9].

In general, life sciences institutions do not have the necessary expertise on data storage nor sufficiently large infrastructures to efficiently store this data [10]. Cloud computing is a natural economic alternative to private infrastructures since it requires low initial capital and allows a scalable growth in an pay-as-you-go manner. However, cloud computing is not as good an alternative in terms of security and privacy concerns [11]. Additionally, the increasing severity of data breaches (e.g., [12]) and the tightening of privacy-related regulations (e.g., GDPR [13]) have been driving the demand for increased security also on cloud-based storage. Secure storage solutions based on multiple clouds (i.e., a cloud-of-clouds) have been proposed in the last decade (e.g., [14, 15]). However, they usually incur in high storage overheads (e.g., 50% [15]), which prevents a bigger adoption in practice.

In this work, we present an end-to-end composite pipeline intended to enable the secure, dependable cloud-based storage of human genomes by integrating three mechanisms we have recently proposed. These mechanisms encompass (1) a privacy-sensitivity detector for human genomes [16], (2) a similarity-based deduplication and delta-encoding algorithm for sequencing data [17], and (3) an auditability scheme to verify who has effectively read data in storage systems that use secure information dispersal [18]. The first mechanism identifies the privacy sensitive portions of human genomes and allows the portions associated with different privacy risk levels to follow different privacy-related paths in the pipeline. The second mechanism focuses on balancing reduction ratio and read performance better than existent genome compression algorithms. Finally, the third mechanism identifies and enforces the additional requirements for auditing who has effectively read data from a modern secure dispersed storage.

We advocate that one can obtain reasonable privacy protection, security, and dependability guarantees at modest costs

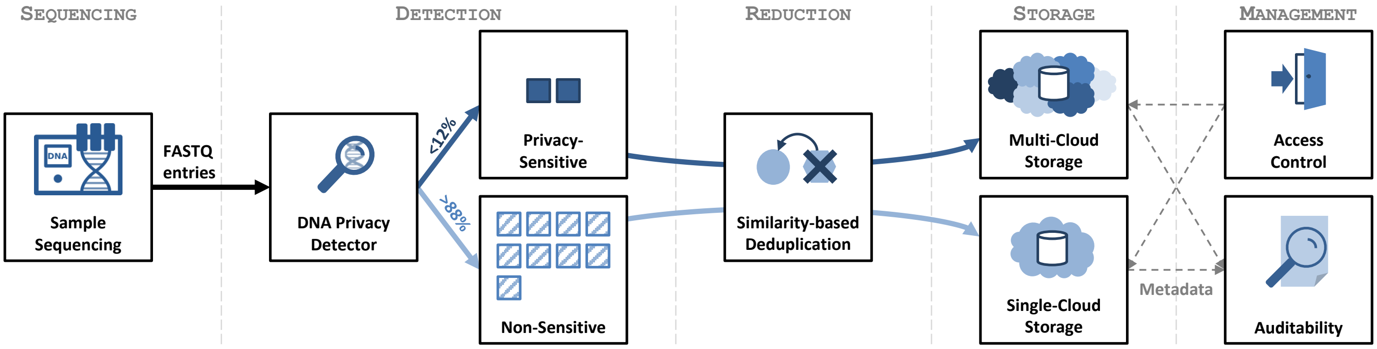

Fig. 1. Overview of our pipeline intended to enable the efficient and dependable storage human genomes in public clouds.

(e.g., less than $1/Genome/Year) by integrating the mentioned mechanisms with appropriate storage configurations. Our pipeline presents a small storage overhead of 3% compared to non-replicated systems, but it costs 48% less than fully-replicating all data and 31% less than secure information dispersal schemes.

II. THE PIPELINE

In this section, we present an end-to-end composite pipeline intended to enable the efficient, dependable cloud-based storage of human genomes. This pipeline inserts privacy-awareness, cost-efficiency, and auditability into the storage ecosystem focused on human genomes. It is composed of five phases, as presented in Figure 1: SEQUENCING, DETECTION, REDUCTION, STORAGE, and MANAGEMENT.

The first phase (SEQUENCING) obtains the digitized genome from biological samples. The second (DETECTION) separates genomes' portions according to their privacy sensitivity. The third (REDUCTION) applies data reduction techniques to improve data density and reduce storage costs. The fourth (STORAGE) retains the genome's portions in appropriate repositories and provides data to clients. Finally, the fifth (MANAGEMENT) provides tools for controlling and monitoring the system. These phases are described in the remaining of this section.

A. SEQUENCING

Sequencing machines digitize genomes by translating the chemical compounds from biological samples to digital information. Next Generation Sequencing (NGS) [19] is the name given to the machines that sequence genomes at a high-throughput [20]. However, NGS machines do not provide the whole human genome in a single contiguous DNA sequence. They generate millions of small *DNA reads*, which are small pieces of DNA containing sequences with hundreds to thousands of nucleotides each. Additionally, every nucleobase from a human genome is sequenced many times and appear in several complementary reads (e.g., 30–45×) to improve the sequencing accuracy.

Data obtained from this process is usually stored in the FASTQ text format [2], in which every *entry* contains four lines. The first line of every FASTQ entry is a comment about the read sequence and starts with a "@" character. The second line contains the DNA sequence read by the machine. The third line starts with a "+" character to determine the end of the nucleotide sequence and can optionally be followed by the same content of the first line. The fourth line contains quality scores, which measure the confidence of the machine on each read nucleotide. Each sequenced FASTQ entry (i.e., 4 lines) is sent separately to the next step in our storage pipeline.

B. DETECTION

Previous works on privacy-preserving genome processing advocated the partitioning of genomic data [21, 22], but assumed it would be done manually [23] or by a tool out of their scope [24]. We closed this gap by proposing a *DNA Privacy Detector* [16], which was the first comprehensive privacy-aware detection method that enabled users to implement such partitioning automatically.

Given a DNA segment of a predefined size, our method detects whether this segment may contain a known privacy-sensitive information or not. It does so based on a knowledge database of published signatures or patterns of privacy-sensitive nucleic and amino acid sequences. The detector decides based on the information present in the knowledge database, and forwards each received FASTQ entry alternatively to a privacy-sensitive output or to a non-sensitive one. Recent works have upgraded this detection method to: evolve the knowledge database to detect previously unknown privacy-sensitive sequences [25]; support FASTQ entries with larger DNA sequences [26]; and support additional privacy-sensitivity levels according to different risk classifications [27].

In this work, data from the DETECTION phase results in two subsets: a small privacy-sensitive portion (i.e., 12% of the FASTQ entries from each human genome) and a large non-sensitive one (i.e., 88%). This 12/88 ratio between these two portions comes from the employed knowledge database, which contains the currently known privacy-sensitive sequences [16]. Reducing the data that requires stronger security and dependability premises (to less than 12%) naturally contributes to the cost-efficiency of any storage solution.

By identifying the privacy-sensitive sequences using our solution and protecting them, one neutralizes the existent threats of re-identifying individuals [28] and of inferring private information about them [8]. Finally, FASTQ entries from both portions (i.e., the privacy-sensitive and the non-sensitive) are sent to the REDUCTION phase, which deduplicates this data to make it even more cost-efficient.

C. REDUCTION

Reducing the size of data from genomes is imperative to enable the efficient storage of large data sets of human genomes. Without a considerable data reduction, most hospitals and biobanks cannot store this data, which may delay advances in medical research and diagnosis [29].

We evaluate and compare several generic (e.g., GZIP [30]) and specialized (e.g., LFQC [31]) compression tools [32] to better depict the state of the art in the reduction of human genomes. We have selected portions of five representative human genomes (SRR400039, SRR618664, SRR618666, SRR618669, SRR622458) from the 1000 Genomes project [33]. They sum up approximately 265GB of data in FASTQ files. GZIP is the fastest generic compressor in our experiments and compresses the selected genomes, on average, $3.21\times$, which results in a reduction ratio $r = 0.3115$. LFQC is the specialized tool with the best reduction ratio and compresses the mentioned genomes, on average, $8.20\times$, which results in a reduction ratio $r = 0.122$. Other evaluated tools provide intermediate results between these two solutions both in terms of throughput and reduction ratio.

Storage of sequencing data is an important, challenging domain mostly unexplored in the systems community [29]. Deduplication is a technique that reduces the storage requirements by eliminating unrelated redundant data [34]. Additionally, deduplication has two advantages when compared to compression algorithms: it may leverage the inter-file similarities, while most compression algorithms consider only intra-file data or use a single generic contiguous reference; and it usually achieves a better read performance than compression. However, due to the fact that FASTQ entries contain unique identifiers, traditional identity-based deduplication (e.g., chunk-based [35]) fails to provide a satisfactory reduction in the storage of genomes.

Similarity-based deduplication is an interesting alternative since there are several entries with very similar structure or content. Solutions for similarity-based deduplication commonly cluster similar entries into buckets and use identity-based deduplication within them [35], or they focus mostly on the delta-encoding problem [36] and employ inefficient global indexes [37]. We have been working on a solution that balances space savings and read performance by integrating efficient similarity-based deduplication based on Locality-Sensitive Hashing (LSH) [38] and specialized delta-encoding based on the Hamming distance for genome sequencing data [17]. This solution finds, separately for the DNA and QS lines of each FASTQ entry, the most similar base chunk in a deduplication index and replaces the original lines by a pointer to the best candidate and the transformations to recover the original sequence from it.

Preliminary analysis of our ongoing work indicates it achieves 60% of the reduction ratio of the best specialized tool (i.e., LFQC) and compresses 50% more than the fastest generic competitor (i.e., GZIP) using a human reference genome as the deduplication index for the DNA lines and 2^{20} synthetic candidates for the QS lines. Additionally, it restores data $83.3\times$ faster than LFQC and $4.4\times$ faster than GZIP. In summary, our solution is currently able to compress the selected genomes, on average, $4.92\times$ (i.e., $r = 0.2032$).

D. STORAGE

Data from the DETECTION phase is divided into two subsets: a privacy-sensitive portion of human genomes and a non-sensitive one. These two portions are deduplicated and delta-encoded in the REDUCTION phase and are handled differently in the present phase. The privacy-sensitive portion requires stronger security premises, while the other portion can use affordable security techniques. From the moment they are delivered by the previous phases, the STORAGE phase applies commonly used dependability and security techniques to store data properly in public clouds.

Cloud computing is an economic alternative to expensive private infrastructures. We consider a system architecture composed of a single public cloud to store the non-sensitive portion of human genomes and a cloud-of-clouds to store the privacy-sensitive portion.

1) Single-Cloud Storage: This is the simplest scenario, where we apply *standard encryption* on data from the REDUCTION phase and store it in a *single public cloud*. This encryption guarantees that only authorized users have access to data, and these users need to know the decryption key. The rationale for this decision is the fact that this data is the less (or non-) sensitive portion of human genomes, and thus the security and dependability provided by a single public cloud is acceptable.

There is an inherent execution cost in this scenario. The cost for reading data will be equals to the cost of transferring the compressed encrypted file from the cloud, decrypting it, decompressing it, requesting the original sequences and quality scores from the deduplication index, and applying the delta-encoded transformations to recover the original data.

2) Multi-Cloud Storage: In this storage configuration, we also initiate by applying *standard encryption* on data. Then, data is *split in blocks* that will be sent to different clouds later [15]. It guarantees that no cloud has the whole genome in its infrastructure, which increases the privacy-protection in case the data stored in a subset of clouds is compromised. We opt to apply *secret sharing* [39] on the encryption key to distribute it together with the data blocks, which makes the system independent of key managers. Storage optimal *erasure codes* [40] are also employed to allow recovering the data in case of failures without the need of replicating all data blocks, which reduces the storage cost compared to full replication. Finally, data is sent to a quorum of clouds from the *cloud-of-clouds*, where each cloud stores different data blocks in a secure setting and provides increased availability.

There is an inherent execution cost in applying all these techniques over data. The cost for reading data will be equals to the cost of transferring the data from a subset of clouds plus: recovering the original blocks from the erasure codes and secret sharing methods, decrypting the data, and decoding the original data based on the entries used in the deduplication system and the delta-encoded transformations. The needed subset of clouds must result in clients receiving at least the minimum number τ of correct blocks to decode a data item.

We employ CHARON [41] as our backend since it is a complete storage solution that provides the two mentioned configurations (single public clouds and a cloud-of-clouds) and allows also the storage of data in private repositories. The original cloud-of-clouds configuration of CHARON assumes $n \geq 3f + 1$, with $\tau \geq f + 1$, which means that this storage configuration incurs in an optimal storage overhead of 50% ($f = 1$). As it will be explained in the next section, to support auditability, we consider a cloud-of-clouds configuration in our pipeline where $n \geq 5f + 1$ and $\tau \geq 3f + 1$ (i.e., resulting in a storage overhead of only 25% with $f = 1$).

E. Management

Several components (e.g., key distribution, performance monitoring, and billing) may fit in this generic phase. However, we are interested only in the access control and auditability ones because these are the main blocks responsible for guaranteeing that only authorized users can and have effectively accessed the data.

1) Access Control: Access control permits certain users to obtain and modify specific data items according to their roles. This mechanism may also have distinct access rules for the different portions of human genomes (i.e., the privacy-sensitive and the non-sensitive portions). Additionally, cryptographic solutions from the Storage phase complement access control mechanisms since an attacker that circumvents the access control does not obtain the data in clear. Finally, the cloud-of-clouds in Charon provides a joint access control combining multiple cloud providers, where its access control is satisfied even if up to f providers have been compromised [41].

2) Auditability: Auditability is the systematic ability of verifying some property in an environment and is a deterrent measure that complement preventive ones, e.g., security, dependability, and privacy-protection. In this work, we are interested in auditing exactly which users have effectively read each human genome stored in the system, which already separates them from the whole group of users that are authorized to read it but have never done so. Auditors need to access only metadata, such as filenames, access logs, and access control rules (i.e., they do not need access the whole data sets).

Usually, auditability systems must keep an indelible tamper-proof track of data accesses to detect, analyse, and sanction misuses. However, the guarantees from this registry directly depend on the configuration of the system. For instance, non-replicated storage systems must trust in the single cloud provider they use to register and provide evidences for every action users perform in the system.

However, storage systems that employ multiple cloud providers have the opportunity to avoid this trust requirement by using the logs from a subset of providers to create a Byzantine fault tolerant track of records. We have recently identified the formal requirements of such auditability for systems based on secure information dispersal schemes [18]. Basically, auditability requires $n \geq 5f+1$ (i.e., $\tau \geq 3f+1$) to provide a weak form of auditability in systems supporting fast reads [42] or a strong form of auditability in systems with slow reads (i.e., reads with more than one communication round).

Privacy-awareness (from §II-B) allows us to provide adequate auditability guarantees for the different portions of human genomes. The fact that only the non-sensitive portions of human genomes are stored on single-cloud storage reduces the impact of loosing auditability information on these configurations. The storage of the privacy-sensitive portion of human genomes guarantees that every effective read is reported by the audit process [18].

III. Feasibility Discussion

Storage costs directly impact the feasibility of collecting large sets of whole human genomes. Furthermore, storage solutions must benchmark their cost-efficiency to not burden institutions and to make dependability affordable [43].

Haussler *et. al* [44] estimated the costs of creating a data warehouse to store (and process) one million human genomes (compressed to 180GB each). Their calculated capital expenditure (CapEx) was $65M for the first year and $35M per subsequent year to maintain and update the infrastructure.

One million human genomes is an interesting example of the scale biobanks will face since they already manage similar numbers of physical samples [7]. Assuming that each human genome sizes s = 300GB (i.e., 30–45× of coverage), one million genomes result in 300PB of data. Storing all this data is expensive, where even the cost of using only cold storage from a single cheap cloud provider (e.g., Microsoft Azure—see Table I) is $3.6M per year. Investing in more dependable solutions (e.g., secure information dispersal using multiple clouds) increases this annual cost to approximately $13M.

In this section, we evaluate the feasibility of the presented composite pipeline. We use the estimated annual cost (in $) to store a single human genome as the metric of interest since it can easily be adapted to deployments of any size. We start by delineating three basic configurations typically used in cloud-based storage and present their pros and cons.

TABLE I
Cloud storage pricing (in $/GB/Month) in June, 2019.

Cloud Storage	Standard	Infrequent	Cold
Microsoft Azure[1]	0.0184	0.01	0.001
Alibaba Cloud[2]	0.0185	0.01	0.0036
Google Cloud[3]	0.02	0.01	0.004
Amazon AWS[4]	0.023	0.01	0.004
IBM Cloud[5]	0.022	0.012	0.006
Oracle Cloud[6]	0.0425	0.0255	0.0026

[1] https://azure.microsoft.com/pricing/details/storage/blobs/

[2] https://www.alibabacloud.com/product/oss/pricing

[3] https://cloud.google.com/storage/pricing

[4] https://aws.amazon.com/s3/pricing/

[5] https://www.ibm.com/cloud-computing/bluemix/pricing-object-storage

[6] https://cloud.oracle.com/storage/pricing

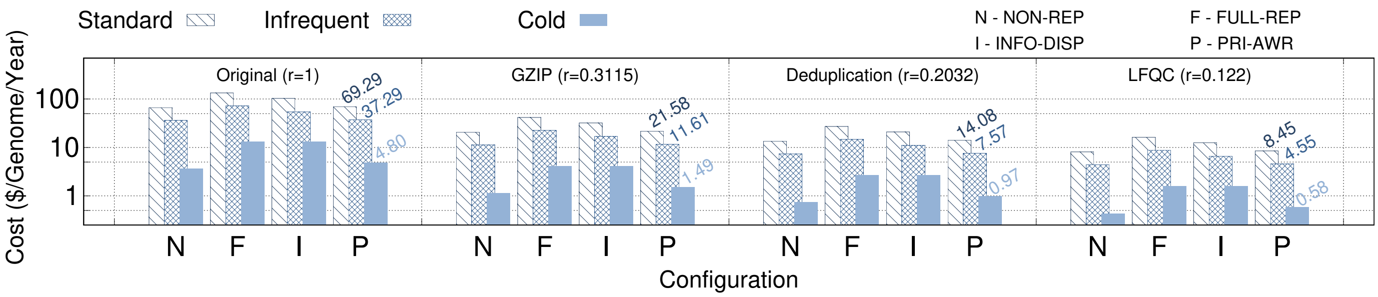

Fig. 2. Estimated annual cost (in $) to store a human genome ($s = 300$GB) considering different configurations and reduction ratios (r).

The first configuration (NON-REP) stores all data only in the cheapest single cloud provider from Table I (i.e., Microsoft Azure). It is the baseline of this evaluation and represents a non-replicated scenario, where the cloud provider has to be trusted and is a single point of failure at the administrative domain level. The second configuration (FULL-REP) replicates all data into trusted cloud providers. It tolerates cloud outages (i.e., crash faults only) of a subset of providers (i.e., $n \geq f + 1$). We consider tolerating one fault (i.e., $f = 1$) in this evaluation, which means that this configuration results in a storage overhead of 100% since $n = 2$. The third configuration (INFO-DISP) distributes data into multiple untrusted providers using secure information dispersal schemes [14, 15], which guarantee that no single cloud stores or has access to the entire data set. It tolerates a subset of malicious clouds (i.e., $n \geq 3f + 1$) and results in a storage overhead of 50% (for $f = 1$) [15] compared to our baseline (i.e., NON-REP).

Our pipeline employs the steps described in §II and stores each genome portion in an appropriate configuration. It results in a fourth configuration (PRI-AWR) that conservatively stores approximately 12% of each human genome (i.e., the privacy-sensitive portion) using a special case of secure information dispersal and the remaining 88% (i.e., the non-sensitive portion) in a non-replicated configuration. Our secure information dispersal uses more clouds than the configuration in INFO-DISP to support the auditability of who has effectively read data, as described in §II-E. This increase in the number of replicas results in a storage overhead of 25% (for $f = 1$) instead of the 50% from INFO-DISP. In the end, our pipeline has a storage overhead of only 3% since $0.12 \times 1.25 \times r \times s + 0.88 \times r \times s = 1.03 \times r \times s$.

Despite the configuration of choice, we assume all solutions encrypt data to protect confidentiality and use the n cheapest clouds on each different configuration. Additionally, this comparison considers that a human genome originally sizes $s = 300$GB (i.e., $r = 1$) and can be reduced by different algorithms with the following compression ratio r: GZIP reduces a genome to 93.45GB ($r = 0.3115$), our similarity-based deduplication reduces it to 60.96GB ($r = 0.2032$), and LFQC reduces it to 36.66GB ($r = 0.1222$).

Figure 2 presents the estimated annual cost (in $) to store a human genome in every configuration described in this section either uncompressed or compressed by one of the three mentioned reduction algorithms. Additionally, this figure considers scenarios using three storage service levels that are available in all evaluated cloud providers and differ in the expected frequency of data accesses: standard, infrequent, and cold storage (the less frequent, the cheaper—see Table I).

While storing an uncompressed human genome can cost $\$66.24$ per year in NON-REP using the cheapest standard cloud storage, it can drop to $\$0.43$ storing this genome compressed by the LFQC in the cheapest cold storage provider. Considering dependable alternatives, fully replicating data always costs more than using secure information dispersal, which by its turn always costs more than using our privacy-aware pipeline. Storing an uncompressed genome using only our privacy-awareness (§II-B) and auditability phases (§II-E) can cost $\$69.29$ per year in NON-REP standard storage, while $\$0.58$ are enough to store it compressed with LFQC using cold storage. However, LFQC has a small restore throughput compared to our deduplication and the other competitors (see §II-C). Using all phases from our pipeline (i.e., deduplicating instead of using LFQC) results in a storage cost of $\$14.08$ using standard storage services and $\$0.97$ using cold storage. It means that storing one million human genomes with our pipeline costs less than $1M per year. These results vouch for the utility of the mechanisms integrated in our composite pipeline to enable the efficient, dependable storage of human genomes in public cloud infrastructures.

IV. Final Remarks

We described an end-to-end composite pipeline that introduces privacy-awareness, cost-efficiency, and auditability into the data storage ecosystem focused on human genomes. By integrating the presented mechanisms with appropriate storage configurations, we showed that it is possible to obtain reasonable privacy protection, security, and dependability guarantees at modest costs (e.g., less than $1/Genome/Year). This pipeline is intended to enable the efficient, dependable cloud-based storage of human genomes since it provides enhanced dependability guarantees with adequate storage overhead (e.g., 3% compared to non-replicated systems). Moreover, the efficiency of the proposed pipeline is also attested by the fact it costs 48% less than fully-replicating data and 31% less than using secure information dispersal schemes exclusively.

Acknowledgements

Authors would like to thank co-authors Francisco Couto, Paulo Veríssimo, Ricardo Mendes, Tiago Oliveira, Nuno

Neves, and João Paulo, as well as many LASIGE colleagues for their inestimable contributions to the mechanisms that compose the presented pipeline. This work was supported by the European Commission through projects BiobankCloud (FP7-ICT-317871), SUPERCLOUD (H2020-ICT-643964), and DiSIEM (H2020-DS-700692) and by Fundação para a Ciência e para a Tecnologia (FCT) through the projects IRCoC (PTDC/ EEISCR/6970/2014) and the LASIGE Research Unit (UID/CEC/00408/2019).

References

[1] E. R. Mardis, "Next-generation DNA sequencing methods," *Annu. Rev. Genomics Hum. Genet.*, vol. 9, pp. 387–402, 2008.

[2] P. Cock *et al.*, "The Sanger FASTQ file format for sequences with quality scores, and the Solexa/Illumina FASTQ variants," *Nucleic Acids Res.*, vol. 38, no. 6, pp. 1767–1771, 2010.

[3] V. V. Cogo and A. Bessani, "From data islands to sharing data in the cloud: the evolution of data integration in biological data repositories," *Communications and Innovations Gazette (ComInG)*, vol. 1, no. 1, pp. 1–11, 2016.

[4] K. A. Wetterstrand, "DNA sequencing costs," available at http://www.genome.gov/sequencingcostsdata. Retrieved on July 3, 2019.

[5] V. Marx, "Biology: The big challenges of big data," *Nature*, vol. 498, no. 7453, pp. 255–260, 2013.

[6] M. A. Hamburg and F. S. Collins, "The path to personalized medicine," *New England Journal of Medicine*, vol. 2010, no. 363, pp. 301–304, 2010.

[7] R. W. G. Watson, E. W. Kay, and D. Smith, "Integrating biobanks: addressing the practical and ethical issues to deliver a valuable tool for cancer research," *Nature Reviews Cancer*, vol. 10, no. 9, p. 646, 2010.

[8] M. Naveed, E. Ayday, E. W. Clayton, and et al., "Privacy in the genomic era," *ACM Computing Surveys (CSUR)*, vol. 48, no. 1, p. 6, 2015.

[9] E. Ayday, E. De Cristofaro, J.-P. Hubaux, and G. Tsudik, "Whole genome sequencing: Revolutionary medicine or privacy nightmare?" *IEEE Computer*, vol. 48, no. 2, pp. 58–66, 2015.

[10] A. Bessani and et al., "BiobankCloud: a platform for the secure storage, sharing, and processing of large biomedical data sets," in *Proc. of the DMAH 2015*, 2015.

[11] Z. Xiao and Y. Xiao, "Security and privacy in cloud computing," *IEEE communications surveys & tutorials*, vol. 15, no. 2, pp. 843–859, 2012.

[12] O. Security, "The 21 biggest data breaches of the 21st century," Available at https://optimumsecurity.ca/21-biggest-data-breach-of-21-century. Retrieved on July 3, 2019, 2019.

[13] "Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation)," *Official Journal of the European Union*, vol. L119, pp. 1–88, 2016.

[14] H. Abu-Libdeh, L. Princehouse, and H. Weatherspoon, "Racs: A case for cloud storage diversity," in *Proc. of the 1st ACM Symposium on Cloud Computing (SoCC)*, 2010, pp. 229–240.

[15] A. Bessani, M. Correia, B. Quaresma, F. Andre, and P. Sousa, "DepSky: Dependable and secure storage in cloud-of-clouds," *ACM Transactions on Storage (TOS)*, vol. 9, no. 4, 2013.

[16] V. V. Cogo, A. Bessani, F. M. Couto, and P. Verissimo, "A high-throughput method to detect privacy-sensitive human genomic data," in *Proc. of the 14th ACM Workshop on Privacy in the Electronic Society (WPES)*, 2015, pp. 101–110.

[17] V. V. Cogo and A. Bessani, "Efficient storage of whole human genomes," Poster in the *11th European Conference on Computer Systems (EuroSys)*, 2016.

[18] ——, "Auditable register emulations," *arXiv:1905.08637*, pp. 1–12, 2019.

[19] J. Shendure and H. Ji, "Next-generation DNA sequencing," *Nature biotechnology*, vol. 26, no. 10, pp. 1135–1145, 2008.

[20] L. Liu, Y. Li, S. Li, N. Hu, Y. He, R. Pong, D. Lin, L. Lu, and M. Law, "Comparison of next-generation sequencing systems," *Journal of biomedicine & biotechnology*, vol. 2012, p. 251364, 2012.

[21] Y. Erlich and A. Narayanan, "Routes for breaching and protecting genetic privacy," *Nature Reviews Genetics*, vol. 15, no. 6, pp. 409–421, 2014.

[22] D. Greenbaum, A. Sboner, X. J. Mu, and M. Gerstein, "Genomics and privacy: Implications of the new reality of closed data for the field," *PLoS Computational Biology*, vol. 7, no. 12, p. e1002278, 2011.

[23] E. Ayday, J. L. Raisaro, U. Hengartner, A. Molyneaux, and J.-P. Hubaux, "Privacy-preserving processing of raw genomic data," in *Proc. of the DPM 2014*, 2014, pp. 133–147.

[24] K. Zhang *et al.*, "Sedic: privacy-aware data intensive computing on hybrid clouds," in *Proc. of the 18th ACM Conference on Computer and Communications Security (CCS)*, 2011, pp. 515–526.

[25] V. V. Cogo, A. Bessani, F. M. Couto, M. Gama-Carvalho, M. Fernandes, and P. Esteves-Verissimo, "How can photo sharing inspire sharing genomes?" in *Proc. of the 11th International Conference on Practical Applications of Computational Biology and Bioinformatics (PACBB)*, 2017.

[26] J. Decouchant, M. Fernandes, M. Voelp, F. M. Couto, and P. Esteves-Verissimo, "Accurate filtering of privacy-sensitive information in raw genomic data," *Journal of biomedical informatics*, vol. 82, pp. 1–12, 2018.

[27] M. Fernandes, J. Decouchant, M. Volp, F. M. Couto, and P. Verissimo, "DNA-SeAl: Sensitivity Levels to Optimize the Performance of Privacy-Preserving DNA Alignment," *IEEE Journal of Biomedical and Health Informatics*, vol. Early Access, pp. 1–8, 2019.

[28] M. Gymrek, A. L. McGuire, D. Golan, E. Halperin, and Y. Erlich, "Identifying personal genomes by surname inference," *Science*, vol. 339, no. 6117, pp. 321–324, 2013.

[29] D. Pavlichin, T. Weissman, and G. Mably, "The quest to save genomics: Unless researchers solve the looming data compression problem, biomedical science could stagnate," *IEEE Spectrum*, vol. 55, no. 9, pp. 27–31, 2018.

[30] P. Deutsch, "GZIP file format specification version 4.3," Internet Requests for Comments, RFC 1952, 1996.

[31] M. Nicolae, S. Pathak, and S. Rajasekaran, "LFQC: a lossless compression algorithm for FASTQ files," *Bioinformatics*, vol. 31, no. 20, pp. 3276–3281, 2015.

[32] I. Numanagić, J. K. Bonfield, F. Hach, J. Voges, J. Ostermann, C. Alberti, M. Mattavelli, and S. C. Sahinalp, "Comparison of high-throughput sequencing data compression tools," *Nature Methods*, vol. 13, no. 12, p. 1005, 2016.

[33] L. Clarke *et al.*, "The 1000 Genomes Project: data management and community access," *Nature methods*, vol. 9, no. 5, pp. 459–462, 2012.

[34] L. Freeman, R. Bolt, and T. Sas, "Evaluation criteria for data de-dupe," 2007, iNFOSTOR.

[35] J. Paulo and J. Pereira, "A survey and classification of storage deduplication systems," *ACM Computing Surveys (CSUR)*, vol. 47, no. 1, p. 11, 2014.

[36] F. Douglis and A. Iyengar, "Application-specific delta-encoding via resemblance detection," in *Proc. of the USENIX Annual Technical Conference (ATC)*, 2003, pp. 113–126.

[37] L. Xu, A. Pavlo, S. Sengupta, and G. R. Ganger, "Online deduplication for databases," in *Proc. of the ACM International Conference on Management of Data (SIGMOD)*, 2017, pp. 1355–1368.

[38] P. Indyk and R. Motwani, "Approximate nearest neighbors: towards removing the curse of dimensionality," in *Proc. of the ACM Symposium on Theory of Computing (STOC)*, 1998, pp. 604–613.

[39] H. Krawczyk, "Secret sharing made short," in *Advances in Cryptology (CRYPTO)*, 1993, pp. 136–146.

[40] J. S. Plank, "Erasure codes for storage systems: A brief primer," *Login: The USENIX Magzine*, vol. 38, no. 6, pp. 44–50, 2013.

[41] R. Mendes, T. Oliveira, V. V. Cogo, N. Neves, and A. Bessani, "CHARON: A secure cloud-of-clouds system for storing and sharing big data," *IEEE Transactions on Cloud Computing (TCC)*, vol. Early Access, pp. 1–12, 2019.

[42] R. Guerraoui and M. Vukolić, "How fast can a very robust read be?" in *Proc. of the 25th Annual ACM Symposium on Principles of Distributed Computing (PODC)*, 2006, pp. 248–257.

[43] R. N. Charette, "Why software fails [software failure]," *IEEE Spectrum*, vol. 42, no. 9, pp. 42–49, 2005.

[44] D. Haussler *et al.*, "A million cancer genome warehouse," University of Berkley, Dept. of Electrical Engineering and Computer Science, Tech. Rep., 2012.