APPLICATIONS OF HOMOMORPHIC ENCRYPTION

David Archer, Lily Chen, Jung Hee Cheon, Ran Gilad-Bachrach, Roger A. Hallman, Zhicong Huang, Xiaoqian Jiang, Ranjit Kumaresan, Bradley A. Malin, Heidi Sofia, Yongsoo Song, Shuang Wang

This document presents a list of potential applications for homomorphic encryption. The list of application is not comprehensive, instead, it tries to demonstrate the breadth of potential applications in various domains and therefore to demonstrate the importance of this technology.

The list was curated during the Crypto Standardization Workshop that was hosted by Microsoft Research during July 13-14, 2017.

The following table presents some key attributes of the different applications discussed in this document:

Domain	Genomics	Health	National Security	Education
Topic	Match Maker	Billing and Reporting	Smart Grid (Municipal Service)	School Dropouts
Data Owner	Medical Institutions	Small Hospital, Clinic	Nodes and Network	School, Hospital, Welfare
Latency of Service	Hours	Hours	Quasi-Real Time	Week
Data volume (size x no)	DB O(1000X1MB) Query O(1KB)	O(10M) x O(1M)	O(1M) x O(1M)	O(1K) x O(1M)
Data persistency	Add Only	Add Only	Add Only	Add Only
Technical Issues	Comparison, Sorting Auditing Privacy	Tabulation, Linear Algebra	Comparison	Comparison Matrix Analysis
When is possible	1 years	2-3 years	Now	2-3 years
Why HE?	HIPAA	Cyber insurance	Privacy	FERPA
Who pays?	Health Insurance	Hospital	Energy Company	DoE

HE applications in Genomics

Sharing data with privacy has become a limiting step for the field of genomics. DNA and RNA sequences can be generated rapidly and cheaply and as a result large quantities of such sequences are accumulating in different labs and medical institutes. It is estimated that in the next decade or two a significant fraction of the world population will have full genome sequences which are a powerful tool in the study of biology, medicine and human history. Many studies in complex disease or epidemiology require thousands of samples to detect patterns and achieve power on the results. However, there are challenges in broad sharing of these data. Human DNA and RNA sequences are biometric identifiers like a fingerprint. Once these are released they can never be retrieved or pulled back. They can convey medically significant information such as disease risk or socially sensitive markers of identity of family or national origin, such as the presence of an Alzheimer's allele or the discovery of nonpaternity.

Current strategies for protecting genomics data have proven to place a high overhead on researchers. NIH funded projects are required to deposit genomics data for controlled access in the dbGaP database at NCBI under government control, or at a small number of “Trusted Partners” in the community. Going forwards, large-scale genomics data will not be automatically accepted into dbGaP, even though there may not be other places to store them. New efforts to build cloud-based alternatives are proposed but are not currently in place. These developments have put the field in a state of flux and so potentially create strategic opportunities for new, better solutions.

Some driving use cases for genomics data sharing can map to simple operations on the data and may be highly suitable for homomorphic encryption. Two of these involve data sharing to understand clinical significance of genetic variants. These use cases will be discussed further and referred to as ClinShare and Matchmaking. Other examples may involve Beacons or other tools created for the Global Alliance for Genomics and Healthcare (GA4GH). More complex analysis such as GWAS and other statistical analyses of combined genotype and phenotype can be built up with reuse of some simple elements.

Humans are almost identical to each other across 3B base pairs of genome sequence which means that genomic data can be reduced to a simple vector of differences. Changes in the genomic sequence, variants or mutations, can be picked out of the sequence and shared in a simple format called VCF. Phenotype data, which includes relevant clinical results, can be packaged in the emerging exchange format called phenopackets. This combination of genotype and phenotype is useful to share in many settings.

For example, physicians who test the germline BRCA gene in breast cancer patients or their family members often detect novel variants of unknown significance and are not able to advise their patients on recurrence or familial risk. Is the variant that was detected actually pathogenic or is it part of the normal background of variation? If these simple genotype and phenotype data could be collected and analyzed for simple frequency from the many thousands of clinics across the country or the world, then many more of these “variants of unknown significance” or VUS could be understood. The NIH database ClinVar reports the clinical significance of particular variants.

Matchmaking is similar but is based on a slightly different use case. Children with genetic disorders can be tested for the cause of the disorder by looking for de novo variants that are not found in either parent. In this way candidate variants can be picked out but one example is not enough to prove the cause of the disorder. At least one other person with the same disorder who is determined to have the same gene defect is needed to be definitive. Identifying the genetic cause of the disorder can sometimes lead to much improved treatment. A dramatic example of this is the Beery twins who now lead normal lives after many years of severe physical disability.

Both these examples rely on similar fundamental data operations and primitives that can be supported with homomorphic encryptions. The use of homomorphic encryptions can allow different genomic datasets to be uploaded to the cloud and used for providing precision medicine and thus improving the health and wellbeing of patients. These tasks are representatives of many genomic applications that can benefit from homomorphic encryption technologies.

National Security/Critical Infrastructure Use Case

Suppose a network with *n* nodes, for instance a smart grid network, where each node is producing an amount of data. (Each node could represent an individual generator/building or an individual microgrid, etc.) Each node produces data which must be monitored by the larger smart grid/municipality/government. The monitoring body can outsource monitoring and computational work to a public cloud and, using HE, make computations on the state data from every node on the grid. If each node represents an individual microgrid then state measurements could include electrical generation and use, physical equipment temperatures, energy flows, etc. If the nodes represent distinct smart buildings, then state measurements might include current energy use. (Such information could be used to detect anomalies and intrusions, e.g., the presence of malware-infected devices, in the building systems.)

Due to the critical role of this infrastructure it is of great importance to protect the information coming from it and making sure that it cannot be tempered. Note that the analysis performed on the data coming from the grid is used to control the grid and distribution of power. Therefore, by tempering with the data, an adversary might cause a failure in the grid. Therefore, to allow the usage of cloud computing for analyzing the data, it is essential to make sure that the cloud is resilient to potential attacks. HE can deliver such security.

In this vision, measurements from each node in the grid are taken continually and sent, homomorphically encrypted, to the cloud-based platform for computation and analysis.

A node on the network experiences an anomaly such as a spike in energy usage and these metrics are sent to the cloud-based platform where they are computed (details of computation to follow). The encrypted results of the computation are sent to the Decision Center for analysis and may be made available to appropriate queries. (For instance, a representative from another government agency such as ICS-CERT might wish to contact a node administrator to verify the anomaly and track evidence of malware infection.)

Why HE?

There are multiple methods of encrypted computing, but HE is the most appropriate method for this use case. From a simple economic perspective, HE only requires one (not necessarily trusted) server in the cloud, with all end users having a client interface while other encrypted computing methods (e.g., MPC) require a heavier, and more expensive, technology footprint (i.e., more servers). Although it does not seem applicable to this scenario, multiple participants in a MPC system could collude to make inferences about other participants.

Hardware based solutions for protecting data (Intel's SGX for example) have an unclear trust model. Moreover, since the methods used in the implementation of SGX are not public, it is impossible to verify the correctness of the algorithms and protocols, as well as verifying the

implementation. Therefore, for critical applications, such as managing the power grid or water supply of the nation, it may be better to use HE.

Generalization

Smart cities present similar scenarios which are of great importance. An example of such application is route planning for first responders and ambulances. For instance, if there was an automobile accident which required the city's Police, Fire Dept., and multiple ambulances to respond, the city's cloud-based platform could quickly load a server which sends out information requests to specific city departments (e.g., Police, Fire, Ambulance, Transportation, etc…) to assign assets from each department, plan the best routes from the incident site to suitable hospitals.

These applications require different types of computation. Some of them are relatively easy to perform with HE, such as calculating descriptive statistics. However, some aspects of the computation may require additional research such as entity resolution and comparisons

Education applications

Every year, over 1.2 million students drop out of high school in the United States alone.[1] That's a student every 26 seconds – or 7,000 a day. About 25% of high school freshmen fail to graduate from high school on time. To mitigate this problem, we would like to be able to predict at risk students and propose the right intervention for them. However, schools are unlikely to have sufficient information to make such predictions at the highest possible levels. For example, a student might drop because of health conditions in her family or because of their welfare state. Therefore, making accurate predictions require integrating data across different institutions, in this case, schools, hospitals, welfare systems, police departments, and more. These institutes are obliged to preserve the privacy of their data and therefore the integration problem is a severe problem.

There are many risks associated with creating a single repository for this sensitive information. First, it makes it a good target for attacks. Moreover, the school should not have access to welfare information that can be used to discriminate students. However, in the context of predicting attrition, the school should be able to use the information for the specific purpose of preventing dropouts.

We propose the use of Homomorphic Encryption (HE) to resolve this issue. The data can be brought together under the security provided by HE. This makes the data available for computation for approved purposes but without paying the risk of having a single repository and without having to breach the law.

Why HE?

The data used to predict dropout risk is private and sensitive. Therefore, some encryption is needed to make sure data is not leaking. However, encryption at rest and encryption in transit are insufficient since there is a substantial risk of data leaking during processing. Moreover, there is no single entity that is allowed to have access to all the data.

Secure Multi-Party Computation (MPC) may be used to address this challenge. However, many MPC techniques, such as garbled circuits, may present some non-trivial challenges when applied in the real world. For example, all the parties involved must have an online presence during the computation. Moreover, one institute needs to perform heavy computation on behalf of another institute which makes it unclear who would the pay the bill.

Secret sharing presents another possible solution to this task, when combined with MPC for some of the computation. Here, the trust model requires finding at least 2 parties that are

[1] https://www.dosomething.org/us/facts/11-facts-about-high-school-dropout-rates

trusted not to collude. This may be challenging. Moreover, these parties must have high throughput, low latency, communication channel between them which makes this solution expensive and/or slow.

HE presents a feasible solution to this challenge. While HE computation tends to be slower than clear text computation, it can benefit from the ability to batch requests. For example, in this case, the ability to check the dropout risk for all students in an institute simultaneously. Moreover, the task is not very latency sensitive. That is, even if the computation takes days or weeks, it is still valuable.

Description of the computation

The computation has several steps. In the first step, the data about a student should be retrieved from the different repositories. In some application, the request itself may not be sensitive especially if it is made about every student in school. Retrieving data about a student may require some form of entity resolution since the data may be stored under different identifiers in different databases. While the query may, or may not, be encrypted, the retrieved data must be encrypted. Since it is encrypted by different institutes, it does not share the same secret key.

Once the data has been retrieved, it is joined to generate a feature vector. The feature vector is used by a model to make predictions. The model may consist of a linear predictive model, a tree based model, or a neural network.

Challenges

There are several challenges involved in this application. On the technical side, the data comes from multiple entities and therefore is encrypted with different keys. Hence this problem calls for MPC-HE solution. Entity resolution is yet another challenge, however, in some cases this can be avoided. Finally applying the model may be hard depending on the type of model used. For example, deep models (deep nets or deep trees) are harder to implement with HE compared to shallow models.

Another challenge is the incentive models for different parties to participate. While HE presents the lowest barrier for participation but it is still an issue to be considered. Moreover, legal constraints on the different parties should be considered. Finally, some parties might be reluctant to participate fearing that some aggregate information may paint them in not too nice colors.

Generalization

The need to make predictions on private yet distributed datasets is a common theme in many domains. For example, medical information might be distributed between multiple clinics. Border control may wish to collect information from multiple sources to assess the risk of allowing someone to enter the country, and welfare agencies may need to collect information from

different sources to select the best way to support a person in need. Therefore, we see this application as a representative of a large list of important applications.

HealthCare Use Cases for FHE

Health care systems operate in an environment where sensitive information must be protected from disclosure, yet available as input to computations necessary for everyday operations. A testament to the difficulty of managing this balance between risk and utility is that in 2016, there were 13 HIPAA enforcement actions with an average fine of $1.8 million [ref]. Larger breaches, such as the 2015 Anthem Inc. data breach that exposed 80 million records, also attest to the cost when this balance is not maintained correctly: the cost of total damages in that breach is expected to be in excess of $1 billion.

While “cyber-insurance” may provide some protection against such damages, small hospitals and clinics typically find such insurance unavailable. Indeed, minimum policy values are aimed at companies with revenues of at least $2 billion, and premiums are substantial: typically, a rate of $3,500 per $1 million insured. The lack of availability of practical protections results in significant business loss: the National Cyber Security Alliance reports that 60% of small companies subject to major data breaches closed down within 6 months of the breach.

Homomorphic encryption (HE) can help to address the balance of risk and utility in information sharing for some applications in the healthcare industry. Billing and report generation are two such applications. In both cases, analysts need access to individual medical records to compute over some part of their content. By allowing such computation without revealing those records “in the clear”, breaches might be avoided without disrupting such applications that are key to daily operations. In Figure 1, we illustrate how HE enables a “breach-proof” workflow for such applications in a clinic setting. An analyst (at left in the figure) queries current medical records to gather information such as statistics on prescriptions issued or medical encounters provided by the clinic. A potentially untrusted server (in the figure right) holds an encrypted corpus of relevant data, potentially including individual medical records subject to HIPAA protections or other relevant privacy statutes and policies. Homomorphic encryption allows the queries to be computed over that data *while it remains encrypted*, and returns an encrypted answer to the analyst. The analyst then decrypts the answer on a trusted platform and can include query results in relevant reports or invoices. Because the data corpus remains encrypted both while at rest and while used in computation, any adversaries (upper right in the figure) learn nothing about the data or the results of such queries.

The HE approach to protecting sensitive data extends naturally to other domains in healthcare. Below, we describe one such exemplar from the field of precision medicine.

In cancer, tumors are often as distinctive as patients. Tumor heterogeneity makes therapeutic selection challenging. We need to not only stratify treatments based on drug sensitivity, but also avoid over-treatment (for example if a tumor is not responsive to one or more therapies) and predict adverse health events. As a result, matching patients to personalized treatments requires knowledge of the patient’s (that is, the tumor’s) genome, the patient’s medical history and phenotypic characteristics, and the specifics of candidate drugs. Incorporating this

knowledge into the therapy selection process requires intensive computation on data that is highly identifiable. Such analysis is called *pharmacogenomics*. The broader practice of selecting therapies based on such analysis is termed *precision medicine*.

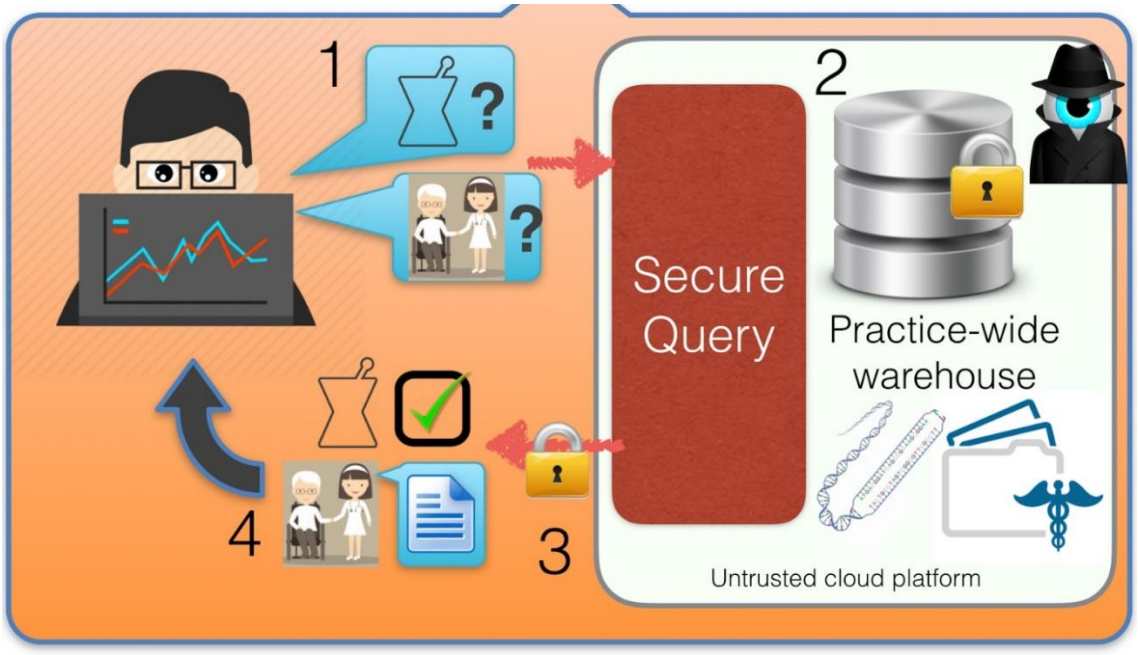


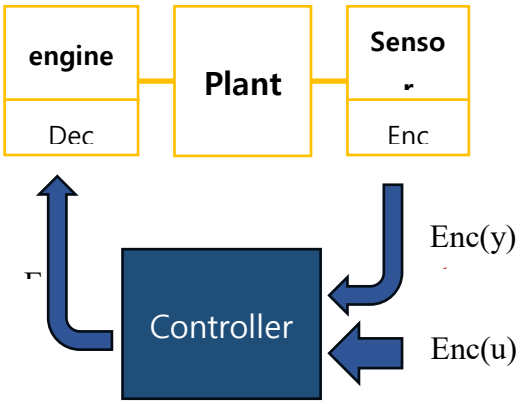
Figure 1: Homomorphic encryption enables clinic analytic workflows over sensitive data

An exemplar precision medicine workflow begins with collection of patient phenotypic characteristics and with sequencing of a patient's genome. Example phenotypic characteristics include the presence of such features as hypertension. Genome sequencing and classification involves identifying a patient's single nucleotide polymorphisms (SNPs) - diversity from the norm in specific areas of a genome. SNPs may for example indicate how well a patient will metabolize certain compounds. Additional molecular diagnostics may be able to further classify a patient's likely response to a drug, or the sub-type of disease that they present. In the next step of our workflow, results of these analyses are put into structured form suitable for computation. In the next step, a drug company prepares information from laboratory studies or prior trials for input to computation, and a computation is run over these two input sets to determine how effective the drug is likely to be for the patient, and what contraindications may be present. Finally, the patient's caregiver (and perhaps the drug company) learn about the suitability of therapy with the drug for the patient.

In such workflows we see a challenge: therapy safety and efficacy must be determined, yet patients are concerned about privacy and agency, hospitals must ensure compliance with relevant laws (such as HIPAA), and pharmaceutical companies are concerned about protecting their intellectual property, especially prior to issuance of relevant patents. Applying homomorphic encryption to the therapy evaluation process allows for determining therapy safety and efficacy while preserving patient and pharma privacy.

In summary, data privacy and utility needs of healthcare organizations, particularly small ones, currently require unappealing trade-offs, sometimes with disastrous outcomes for both organizations and their patients. HE may provide a novel solution to some of these trade-offs, at a cost that is minimal compared to such outcomes.

Protecting Control System with FHE



A control system or a cyber physical system is a computer system that controls signals operating physical system, sometimes networked. It consists of a plant with sensors and actuators and a controller. A controller receives sensing data from the sensors, processes it with user input to compute a command data and sends to actuators that operate a plant following the command. It covers numerous systems including smart cars, drones, and industrial/nuclear plants.

There have been lots of reports about hacking on control systems: In 2015, a hacker showed how to remotely control brakes and accelerators of cars. In 2010, it has been shown that a malicious computer worm entered and computer system in the uranium enrichment facility and varied the rotational speed of the centrifuges to destroy them. Preventing hacking control systems is very important, but regarded as a hard problem. It is recommended that the sensors send sensing data to the controller after encryption and the commander sends control data to the actuator after encryption. It prevents a hijacking of sensing data/control commands by hackers, however cannot prevent a disclosure of the data by malware inside the controller.

Recently, several researchers suggested a use of HE for protecting control systems [1], that is encrypt sensing data with HE. In that case, the controller does not need to decrypt sensing data to process and so can be confidential to the controller itself. Furthermore, any manipulation of the encrypted data by the hackers is probable to be detected by the detection system of the actuator. For more guarantee, it can be considered to a homomophic authenticated encryption (HAE).

Description of the computation

The computation is a matrix multiplication for linear controller. It can be more complicated for smooth fight.

Challenges

It should be run in real time.

Appendix A

Organizers

Jung Hee Cheon	jungheecheon@gmail.com
Ran Gilad-Bachrach	rang@microsoft.com

Contributors

David Archer	dwa@galois.com
Lily Chen	lily.chen@nist.gov
Roger A. Hallman	roger.hallman@navy.mil
Zhicong Huang	zhicong.huang@epfl.ch
Xiaoqian Jiang	xiaoqian.jiang@gmail.com
Ranjit Kumaresan	Ranjit.Kumaresan@microsoft.com
Bradley A. Malin	b.malin@vanderbilt.edu
Heidi Sofia	heidi.sofia@nih.gov
Yongsoo Song	amedonis@gmail.com
Shuang Wang	shw070@ucsd.edu

References

[1] J. Kim, et al. Encrypting Controller using Fully Homomorphic Encryption for Security of Cyber-Physical Systems, IFAC, Vol. 49, Issue. 22, pp.175-180, 2016.