A Trusted Healthcare Data Analytics Cloud Platform

Arun Iyengar, Ashish Kundu, Upendra Sharma and Ping Zhang
IBM T. J. Watson Research Center
1101 Kitchawan Rd
Yorktown Heights, NY 10598
United States
Email: {aruni, akundu, upendra.sharma, pzhang}@us.ibm.com

***Abstract*—This paper presents a cloud-based system for health care applications. Our system has advanced features for preserving privacy which are essential for health care applications that deal with confidential data. We describe some of the bioinformatics applications which our system is designed for. Performance is significantly enhanced by caching, and enhanced clients for performing part of the computations are a key component of our system.**

Cloud, due to its pay-as-you-go pricing and API based deployment model, has become widely used for delivering and maintaining infrastructure technology for businesses. However, there are significant challenges with using the cloud for applications with strict privacy and compliance requirements; health care applications fall in this domain. This paper describes an architecture and solutions for handling these types of applications.

I. Introduction

A wide variety of services are now offered via the cloud for performing data analytics. These services offer storage, data analysis, and artificial intelligence capabilities such as language, speech, and visual recognition. Such services are increasingly being used for health care applications. There is a large amount of biological data which is available, and people are analyzing such data for scientific research as well as for medical purposes. There are millions of scientific articles available in PubMed, and natural language processing techniques which can automatically extract important information from these papers are being used.

This paper presents key issues in cloud-based systems for health care analytics. We describe a number of health related applications which are of significant importance including drug repositioning, drug safety, collecting and monitoring patient information, and general analysis of biomedical data. We present an overall system for handling these applications.

Our system can be used for storing data with differing privacy requirements. Some of the data are highly confidential. For example, there is confidential patient data which must be protected according to HIPAA requirements. Other data do not have such strong data confidentiality requirements. A key feature that our platform provides is analysis of scientific data which may be contained in publications and publicly available databases. This data can be stored less securely than the highly private data.

Our platform provides a wide variety of analytic capabilities on top of the data. The capabilities include analyzing biological data for important characteristics. An example would be predicting diseases cased by genes. While experimental data exists on some genes which cause diseases, our system can use techniques such as matrix factorization to compute additional associations between genes and diseases.

Our system can collect data from many different sources. Information can be provided from mobile devices such as cell phones. Mobile devices can provide personal health care data from users. A key feature we provide is the ability to perform processing at client devices. The clients can be mobile devices or more powerful computers. Allowing processing to take place at the clients conceptually moves computing to the edges of networks. It offloads computing from servers.

There are other advantages to providing enhanced client functionality. There can be confidential data that clients may not be willing to share with servers. Highly confidential data can be analyzed and encrypted or anonymized at clients before being sent to servers. Clients can also perform processing and analysis while disconnected from servers.

Our clients also perform caching to reduce latency for accessing data from servers. The cost for accessing data from remote cloud servers can be orders of magnitude higher than the cost for accessing data locally [1], [2], [3]. Caching can thus dramatically improve performance. Our system employs caching at multiple levels and not just at the client level.

Our system has several distinguishing features from previous ones.

- The proposed system ”weaves” security, privacy and compliance in the lifecycle of the crown-jewels that need protection: data, systems, users and devices. As another key contribution, we have also defined how blockchain is used to implement secure HCLS (health care and life sciences) data provenance as well as a number of key security, privacy and compliance components of our system.
- Our system provides computational capabilities both in the cloud servers themselves as well as at clients themselves. Conceptually, this moves computation to the edge of the network. This can be important for enhancing privacy as clients can perform data encryption

and anonymization before sending information to servers. It can also improve performance by allowing certain computations to take place at the client without the need to incur latency for communication with a remote cloud server.

- Our system makes use of external data sources and knowledge bases. It also can use external Web services, particularly in the artificial intelligence domain. The use of external data sources and services adds to the functionality offered by our system.

The remainder of the paper is structured as follows. Section II describes the infrastructure of our health cloud platform. Section III presents a system and user-level view of our architecture. Section IV presents security, privacy, and compliance issues. Section V describes representative bioinformatics and health care applications which our system is designed for. Section VI presents related work. Finally, Section VII concludes the paper.

II. Health Cloud Platform

Figure 1 depicts a conceptual architecture of our overall system. Figures 2 and 3 describes the key components (functional and non-functional respectively) of the system in a logical setting. Functional components define the system capabilities for healthcare data analytics and management. Non-functional components define the security, privacy and compliance as well as scalability and performance capabilities of the system in order to support the functional capabilities. Our system provides a high level of security and privacy to store protected health information (PHI) as well as advanced analytics capabilities, namely complete model lifecycle management as well as remote execution of authorized models on enhanced clients.

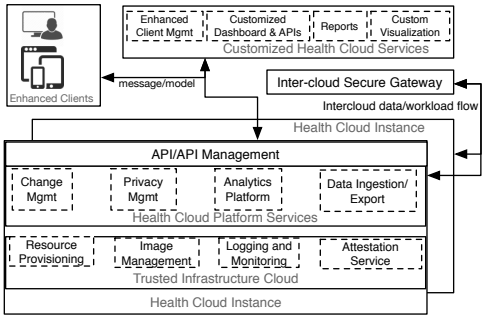

Fig. 1. A conceptual architecture of our system.

A. Infrastructure Cloud

Our system is comprised of native cloud applications hosted on an Infrastructure as a Service (*IaaS*) cloud platform [4], [5] that provides the necessary compute and storage resources with high scalability and availability at low cost. The infrastructure cloud is compliant (HIPAA/GxP/GDPR), which means that all the components of the cloud stack are compliant to the security and compliance policy.

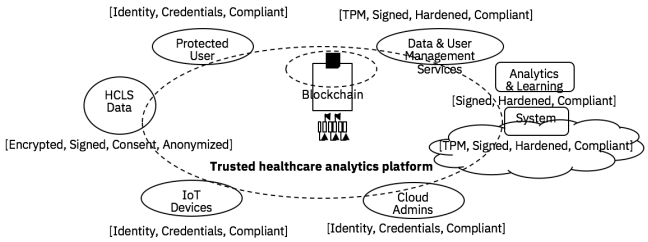

Fig. 2. Key Functional Elements for Healthcare Cloud.

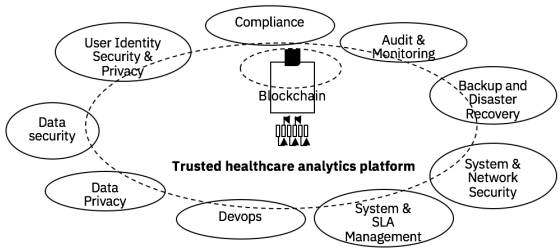

Fig. 3. Key Non-functional Elements for Healthcare Cloud.

The infrastructure cloud is created leveraging virtualization of the physical resources, for instance compute, storage and network. At a high level, the IaaS cloud's stack includes i) bare-metal hardware, ii) host operating system/hypervisor iii) Image and hypervisor management and monitoring services. A compliant cloud is built using verified and attested hardware, hypervisors and all the management services whose integrity is ensured by using a Trusted Platform Module[1] (TPM) [6] installed on the hardware resources and the *Attestation Service*. The basic idea is similar to proposals [7], [8], i.e. create a root of trust at the hardware level (using TPMs and Attestation Service) for each server and then extend it, via a transitive trust model, to the hypervisor. Our system leverages the vTPM [9] to transitively extend the root of trust to the guest OS and the software stack therein.

The *Resource Provisioning* service, *Assertion Service,* along with TPMs/vTPMs, help in creating trusted secure health cloud instances. The *Image Management Service* accepts only those VM images that are signed by an approved list of keys managed by an *attestation service*. The *Logging and Monitoring* service provides secure log and monitoring data for both infrastructure services as well as for platform services.

B. Health Cloud Platform Services

Platform services provide secure generic services, namely a DevOps Service, high availability and disaster recovery service, federated identity management service, Analytics platform and some health care specific services, namely consent management, data ingestion and export service.

Platform DevOps and non-functional services: HIPAA/GxP compliance expects not only the final deployed system to be

[1]A dedicated secure processor that secures hardware through integrated cryptographic keys.

compliant but also the development as well the automated operations to be compliant; this means that not only are the hosts, VMs and the deployed software stack verified and attested but also the development and deployment process of all the components. *Change Management service* is one of the very important services that (under the guidance of a compliant policy) controls changes to any deployed component, infrastructure and software alike. All authorized changes are first described, evaluated and finally approved in the *change management system*; thereafter the CM service accordingly updates the *Attestation Service* regarding the approved changes and their new signatures.

The platform supports a federated identity management system, which means that the platform user's identity could be managed and authenticated by an external (approved) system. Once users are authenticated, their roles and access privileges are managed by the platform's RBAC system.

Privacy Management: Access privileges are controlled by the role based access control (RBAC) system of the platform[2]. The platform supports *Tenant*, *Organizations*, *Groups*, *Environments*, *Users*, *Roles*, and *Permissions*. *Tenant* is a namespace under which all the other entities of RBAC are grouped, for instance tenant could be an enterprise. *Organization* represents departments, particularly from the point of view of resources. Resources that are to be shared, like services, environments etc, are added to organizations. *Groups* represent healthcare studies/programs to which PHI data is consented for. *Environments* are the various development and deployment environments to which users are have access to. *Users* are individuals persons registered under a tenant. Users can have different roles in different environments within an organization which would govern their access privileges. *Permissions* are read and write access control to various resources in the platform under a tenant, organization, or group.

Since the platform supports uploading *protected health information* (PHI) via the *Data Ingestion* service, it is important to secure the consent of the patient/user for the uploaded data via a *consent management* service.

This architecture is particularly advantageous in situations where an analytics compute workload needs to be shipped over to another cluster (possible another cloud) without compromising the trust.

Registration Service: The platform supports an idea of tenant, which is equivalent to an account at an enterprise level for metering and billing of various services. A default *organization* for each tenant is created; under that, a default environment for development and deployment of custom services for instance development and deployment of customized models is created.

Data Ingestion and Export: Data ingestion in essence means the following three steps: i) upload of verified data, ii) validation/curation of the data, and finally ii) storing the data. Health care data has no single data schema/format, and there are a variety of standards for the data format, so the first step is to adopt an electronic healthcare information exchange format, for instance, FHIR, HL7, etc [11]. Our system adopts FHIR as the data ingestion format; this is not a limitation of the system as the system can be easily extended to support any other format by writing adapters that transform data from one exchange format to another, e.g. from HL7 to FHIR and back. The data can be uploaded by authenticated users, either from a device or other system leveraging APIs. The uploaded data is verified, curated and stored in scalable and trusted back-end storage systems.

Data ingestion is a slow process and is thus designed as an asynchronous communication process. Data flows either from a client device or from a source repository to the storage system of the platform. Encrypted data, using a client's public certificate issued by the platform, is *Uploaded* to a secure temporary storage area, and a message is left in the platform's internal messaging system for the background ingestion process to ingest the data. The platform returns a *status URL* to the uploading client, which can be used to know the status of the data ingestion process as it goes through its ingestion flow sequence. The background data-ingestion process picks the encrypted data from the staging area and performs the following three steps under *Ingestion*: i) Decrypts data using the client's private key (generated by the platform at the time of registration and stored in a *key management system*). ii) Validates the uploaded bundle for errors. iii) After successful validation, the data is de-identified and stored in the backend storage system (*Data Lake*) with a reference-id, and the reference-id to identity the mapping is stored in the *metadata*.

The platform also exposes an *Export* service which performs two types of exports, namely i) *Anonymized* export, that anonymizes the data to protect privacy, and ii) *Full export* where the re-identified consented data is provided to the client. This is typically needed by Clinical Research Organizations (CRO) to conduct various types of studies.

API and API management: The platform exposes secure APIs for all its capabilities. The API management system first authenticates the user requesting the APIs, and once successfully authenticated, it consults the *Privacy Management* system and allows API access accordingly.

C. Customized client services

Our trusted health cloud platform is a health care specific cloud that offers compliant services and advanced analytics capabilities to support various health care uses cases. Customized client services can be developed on top of our platform which could be specific to a tenant/client. Clients could develop customized dashboards and use custom report generation tools either by using the analytics cloud provided by the platform or by exporting anonymized data to their own environment and using their own specialized tools.

Customized client services could also take approved and compliant models and push them to enhanced clients for better interaction with the patient.

Intercloud secure gateway

[2]Our RBAC model is motivated from that of Cloudfoundry's [10]

Data in a health cloud is special both from compliance and security; thus it is often the case that data gets collected in one cloud instance while analytics and other services are collected in another cloud platform. Many times the cloud designed to scale for data collection and authoring is not well equipped with other services which would be needed for application and/or model development and deployment. Our design of extending the root of trust to the level of containers allows transfer of trusted analytic workloads (packaged in containers) across different cloud instances (provided each one of them is trusted). This allows the computation to be transferred to data instead of otherwise, thereby making it very efficient and secured. This approach also does not depend on external untrusted libraries as the container would be authored in a trusted environment with trusted libraries. The intercloud secure gateway facilitates transfer of these trusted analytics containers between cloud platforms and also offers a service of *Remote Attestation* for the platform to attest when the analytics workload is started.

III. System and User-Level View of Architecture

Figure 4 depicts our overall system from the perspective of users. The system can scale to a large number of servers. Even though only two are depicted, the actual number can be considerably higher.

Our system performs computation on data with different privacy requirements. The top server is for computations that do not require high degrees of data privacy. The bottom server in the figure is for data with high levels of confidentiality. Although the figure shows just two servers, the system can be scaled to a high number of servers.

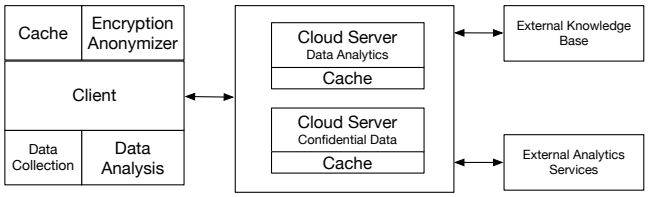

Fig. 4. This figure depicts an architecture of our system from the perspective of users.

Caching is a critically important feature for improving performance. Note that it takes place at multiple parts of the architecture, both at the clients and servers. Caching works best for data which do not change frequently. If the data are changing frequently, cache consistency algorithms need to be applied to keep multiple versions of the data consistent. It may not be feasible to cache rapidly changing data for which it is very important to have updated copies.

Our system has significant analytics capabilities. However, it should be noted that there are many external Web services which can be used to provide additional analytics such as those from IBM, Microsoft, Amaazon, Google, and others. Our system has the ability to use these external Web services and provide the results to users. Many of these services are in areas such as natural language understanding, visual recognition, and speech recognition. The AI services from different providers offer similar functionality but are not identical. We provide users with a choice of services for similar functionality. In addition, we maintain information on the different services to allow users to pick the best ones. This information includes response times and availability of the services.

For some of the services (e.g. text extraction), we have standard tests which we run to test the accuracy of the services. This information is available to users. Users can also provide feedback on services. While we provide user feedback on services to users, we note that such information should be used with caution as it may not be accurate.

Another key aspect of our system is that we make use of data from external databases and knowledge bases. This data can be used in analytics calculations as well as provided directly to users. These external data sources include general knowledge bases such as DBpedia [12], Wikidata [13], and Yago [14]. We also make use of scientific databases such as the DisGeNet database of genes and variants involved in human diseases [15], the PubChem database of chemical structures [16], the DrugBank database on drug and drug targets [17] and the SIDER database on drug side effects [18].

We provide access to papers in PubMed and PubMed Central. We perform text analysis on these papers to extract important scientific facts. We also provide access to knowledge bases related to language such as WordNet [19].

We cache data from these knowledge bases locally. That way, data can be accessed and analyzed more quickly than if it needs to be fetched remotely. For the most up-to-date data, the remote knowledge bases can be directly queried. However, it often is not necessary for an application to have the most up-to-date version of these knowledge bases.

A. Developing Data Analytics Applications

We provide HTTPS (REST) interfaces to our system. Users access our system as Web services. In order to make it easier for users to develop applications using our system, we provide enhanced clients which offer additional functionality for client machines (Figure 4). These enhanced clients provide features such as caching, data analytics, and encryption (Figure 4). We provide software development kits (SDK) which run on client machines to make it easy for clients to access our Web services [20]. These SDKs implement functionality such as client-side encryption, caching, and data analytics and are available for commonly used programming languages, including Java, Python, and JavaScript. That way, an application running on a client machine can easily make method calls in one of these languages to access our Web services or use enhanced client features such as caching, encryption, or data analysis.

Users can also write applications which run on our servers. Such applications may run more quickly, as calling our own services as well as accessing data stored within our system will be faster from within our system than from a client computer.

Our system has a secure, compliant *Analytics Platform* that helps approved users perform local model generation

and testing. The Analytics platform supports various lifecycle stages of analytics models, namely i) data cleaning, ii) initial model generation iii) model testing iv) model deployment and v) model update. The analytics platform offers tools for performing different operations, including authoring tools like Jupyter [21] and version control tools such as git [22].

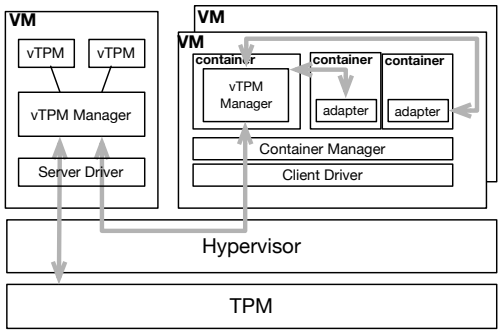

Fig. 5. A conceptual architecture of a secure container cloud over virtual machines for *Analytics Platform*.

For applications requiring a high degree of security, the *analytics platform* is designed to provide a secure environment by extending the root of trust to containers as shown in Figure 5. The approach is a hybrid of approaches advocated in [23] and [9]. The main idea is to have a software implementation of trusted platform modules (TPM) (*vTPM*), execute it in a dedicated VM and take measurements that will be used by an external Attestation Service (shown in Figure 1) to determine the system's trustworthiness. The way the process works is that in each VM, the Core Root of Trust Measurement (CRTM) code runs in the VM's BIOS (BIOS instrumented with TCG extensions [24]). Furthermore, the trusted kernel extends the root of trust transitively to libraries and drivers [25]. Each VM has a client driver that accesses the vTPM instance via the server side driver in the special VM hosting the vTPM. In each VM there is a special container that runs a vTPM Manager (a user space process) that provides the vTPM interface to other containers either through a Unix socket or via IPC (in which case the client container would need an adapter that exposes an IPC interface as a standard character device).

IV. Security, Privacy and Compliance

A. Threat Model

The proposed cloud-based system handles private and sensitive HCLS (Healthcare and Life Sciences) data, which is why it is expected not only to be reliable but also trusted. By trust, we intend here that a trusted system is compliant with respect to the regulatory requirements as mandated, the security of data, systems (including networks) and users as well as the privacy requirements of the patients. To that end, it is essential to implement compliance-specific requirements for HIPAA, GDPR, GxP, and so on.

The goals of the attackers may be to undermine a business competitor or to expose healthcare data, identity and credential information stealing, or to cause socio-political-economical harm to the users, healthcare providers or the geopolitical area involved. Moreover, the goal of attackers may be to compromise the health or treatment of one or more individuals.

In this section, we describe two standard adversary models: honest-but-curious adversaries and malicious adversaries. Formal definitions of these models can be found in [26].

- **Honest-But-Curious Adversaries**: In this model, all players are obliged to follow the protocol and act according to their prescribed action in the protocol. If the protocol is secure, no player gains information about other players' private input sets, other than what can be deduced from the result of the protocol.
- **Malicious Adversaries**: In this model, an adversary may behave arbitrarily. In particular, we cannot hope to prevent malicious players from refusing to participate in the protocol, choose arbitrary values for private input, or abort the protocol prematurely.

These adversaries may be external to the proposed cloud system and IoT devices, and also may be internal to the system such as system admins, devops staff as well as staff who have access to data, system, or policy decisions in any way. Such attacks may cause the system to become "non-compliant" with respect to regulatory requirements. That leads to financial harm as well as business reputation dilution of a given entity involved in offering the services.

The goals of attacks that can be carried out could be:

- Confidentiality, privacy of data and logs: HCLS data may be exposed, exfiltrated, tampered with, or may be modified in a way to cause disruption of the system reliability. Logs are supposed to be non-sensitive; however, they may be analyzed to carry out inference attacks. Some types of logs are to be protected.
- Privacy of users: Users may be patients, relatives of patients, genetically related individuals, doctors and healthcare staff, administrative individuals and so on. Most such user information is sensitive in nature.
- Security of the cloud, IoT, blockchain systems: Intrusion, unauthorized modification, man-in-the middle attacks on the IoT devices as well as on the cloud (compute, memory and storage, network, services) may be carried out. Such systems may be accessed in an unauthorized manner and their functionalities compromised by code-injection and malware attacks not only by external adversaries but also by insider threats.
- Availability and reliability: the system may be attacked in order to make it unusable at a certain time or at an event.

In this paper, we take an end-to-end holistic view of security and design security requirements. Past systems have addressed security, privacy and compliance as discrete requirements. In contrast, the proposed system in this paper "weaves" security, privacy and compliance in the lifecycle of the crown-jewels that need protection: data, systems, users and devices. As another key contribution, we have also defined how blockchain is used to implement secure HCLS data provenance as well as

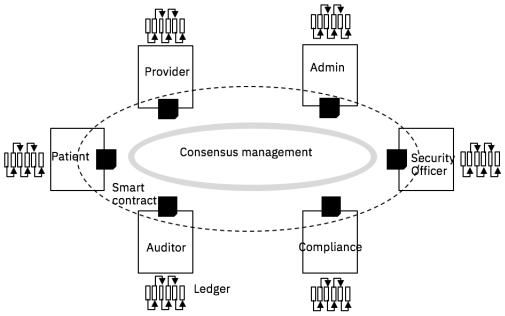

Fig. 6. HCLS Blockchain network.

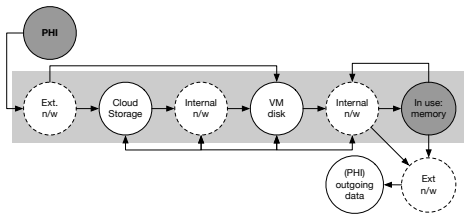

Fig. 7. Cloud Security and Compliance Components.

a number of key security, privacy and compliance components of our system.

In this regard, we have proposed blockchain-based identity management, management of malware and vulnerability, data lifecycle and privacy as well as compliance. Past systems make use of centralized databases without any transparency into how such data is managed and how multiple parties that do not fully trust each other engage on security and privacy of the healthcare system and associated PHI data and patients.

Blockchain enables data provenance and ensures data access and consent provenance as required by GDPR and HIPAA. Moreover blockchain supports audit capabilities for the data management process, which is required as part of regulatory requirements. The blockchain network we are talking of is a permissioned blockchain system such as Hyperledger.

Security Vs Compliance: Security is a bottom-up requirement, while compliance is a top-down requirement. Compliance requirements are already defined by regulatory policies, and they need to be implemented by implementing security and privacy policies and mechanisms. The mechanisms are part of the bottom-up implementation of the security enforcement in the systems.

B. Security

Security of data, system and users are implemented as part of the system design.

1) Secure Data Management: Data flows from external components to a data ingestion system, which processes the data for validation and verification. At that level, data is encrypted at multiple levels – first it is encrypted with a well-established shared key (public key encryption is too expensive to maintain the scalability of the system), and then it is transmitted over a secure channel such as over TLS. An integrity verification mechanism may also have been in place - we recommend using HMACs instead of digital signatures unless the digital signatures are part of the encryption process such as signcryption techniques, or AES CBC mode (encryption and integrity).

The ingestion service decrypts the data using the shared key or established key between the sender and the cloud platform. The ingestion service verifies the integrity/authenticity of the data and stores the signatures and meta-data on a database to maintain receiving meta-data.

The ingestion service carries out the following verifications and validations on the data before forwarding it to the data management and analytics systems.

- integrity and authenticity verification
- scanning of data for malware
- verification of level of privacy/anonymization supported
- verification of consent of the patient

Leakage-free authenticity and integrity verification of HCLS data: Often HCLS data is shared in parts and not as a whole given the compliance requirements and privacy issues involved. Existing systems make use of Merkle hash techniques or traditional hashing of the data and digital signatures to prove authenticity of data. However, they leak information, and leakage-free redactable and sanitizable signatures [27], [28], [29] should be used for such data sharing and analytics purposes. Graph-based HCLS data can also be verified using HMACs [30].

Blockchain-based HCLS data protection management: In another approach as discussed earlier, a data ingestion service can also store the meta-data and events related to an HCLS record on a de-centralized blockchain ledger. In the blockchain network, the parties are: sender (sending client), receiver (receiving service/ingestion service), healthcare provider (peer on behalf of the healthcare provider), data protection service, audit service as well as other services. The different parties using the consensus protocol agree on the data to send and receive, which then leads to commitment of the ledger record to the global ledger. A blockchain ledger is used to maintain a link to the data and meta-data around it. The data is stored and encrypted on a separate server in order to ensure separation-of-duties and implement appropriate access control. Not all parties need to know the PHI (Protected Health Information) always, and the access per HIPAA and security requirements are need-to-know based; thus it is essential not to store the PHI data on the full replicated de-centralized ledger, but to instead store the data in a centralized service.

Upon each event or transaction such as data receipt, data retrieval, data anonymization and such other events, the blockchain ledger is updated with a ”handle/reference” to the encrypted data record, hash of the data, information about the event/transaction, and meta-data. The data record is stored in a database. A hash of the data stored on the ledger is computed using a perfectly secure hash function for stronger privacy and security.

The ingestion service may also validate the data for its format as well as schema used for the data. Per security, the

ingestion service employs a data filtration system to determine if the data contains any malware. If so, the filtration services filter out the record and update the blockchain with the information that the corresponding record identified by an identifier such as a random UUID or a pseudo-random number contains malware. The malware-management blockchain network is a different network that takes the record and carries out policy-driven actions on top of it - such as cleaning, sanitizing it and/or dropping the record and informing the sender and other stakeholders of this information. It can also employ analytics in order to determine risky senders or risky records.

Once the data has been ingested and filtered, the ingestion service may use another service, "anonymization verification service", in order to verify how good the anonymization on the incoming record is. If the anonymization verification service determines that a claimed anonymized record is not properly anonymized, then such a record is dropped, and a response is sent back to the sender. Such information is also recorded in a "privacy blockchain network". Such a blockchain records the privacy levels of each record received. In a different approach, information about a given record on malware, privacy and integrity can be added to a single blockchain network. It is a design decision. Smart contracts can carry out analytics on top of such information and use such information for dynamic ledger management.

After the data is ingested, it is encrypted using a different key or set of keys based on the defined encryption process. Such data can also be re-anonymized independently or together with other data objects. Both the original and anonymized versions of data objects are encrypted and stored. The reason to encrypt anonymized versions of data objects is to ensure that in case there is a breach, the databases cannot be used to retrieve highly valuable data (anonymized data have utility for the purpose of analytics and machine learning and for secondary usage). Attackers need to gain access to the keys to gain access to such data.

Data flows into other components and services in the system such as machine learning model training, analytics, reporting, insurance management, patience management, pharmacy processes and so on. Such services need to gain access to the plain text; thus the key management service in the system ensures that authorized components, services and identities have access to the appropriate set of keys that are generated dynamically and/or statically.

Key Management System: A key management system is a single-tenant isolated system that is dedicated only to a single customer or single instance of the regulated system for HCLS services. It should not be multi-tenant primarily because on a virtual host, the isolation guarantees are not as strong as the air-gapped systems and bare-metal servers. However, the decision to use virtual key management services co-located with other services on the same host is based on business use cases if the the risk of such a deployment meets the compliance and security criteria. However, we envision that such a key management service shall be hardware based (e.g., hardware security modules).

When data needs to flow within the system, it is always transmitted over encrypted channels. If a given system hosts plain text data in memory for processing, access to such a system is monitored and if possible made limited for the period of time when the data is there.

Secure deletion of data: HCLS and sensitive data are deleted from memory, storage and cache as soon as their "need-to-use" time period is over permanently or for the near future; such data is not needed. The deletion process has to follow secure deletion practices. In order to support GDPR and right-to-forget, our system supports encryption-based record deletion and deletion of data relevant to a given patient from all parts of the system.

Identity management of healthcare providers, system administrators and patients are managed with blockchain using self-sovereign identity and privacy-preserving identity-mixer technology.

2) Secure System Management: Each system component is developed using a compliance-assured devops environment and development team. The system components are engineered in a secure manner.

Each system component is signed using a digital signature. A given container or VM image is signed as it is. Another approach is to aggregate the signatures of each package installed on the container/VM and generate an aggregate signature. Such a signature is derived using the private key(s) stored on the TPMs where the images are created, or can be derived from the private keys stored in another key management service. Such signatures are managed by the integrity management architecture or by the remote attestation service.

Malware analysis of the systems is carried out by the malware analysis service and/or the malware blockchain network. Several peers are on the network: cloud vendor, system admin, each software vendor, optionally national vulnerability or other such organizations in relevant geolocations, and compliance officers as well as parties managing the reporting of and fixing of such vulnerabilities.

C. Privacy

The enhanced client can anonymize the data it is sending to the system. Our anonymization verification service verifies the degree of anonymization of the receiving data and data generated by the system. The degree of anonymization is determined by analyzing the data, its semantics, and its attributes. The degree of anonymization/privacy has two parts – one independent of other data objects and another that is determined holistically with respect to other data objects.

A privacy management network using blockchain described earlier can keep track of privacy degrees of the data records.

D. Regulatory Compliance

THe Health Insurance Portability and Accountability Act of 1996 (HIPAA) specifies data privacy and security requirements for health care data in the United States. The HIPAA controls 8 are categorized into four pillars: administrative, physical, technical and policies and documentation.

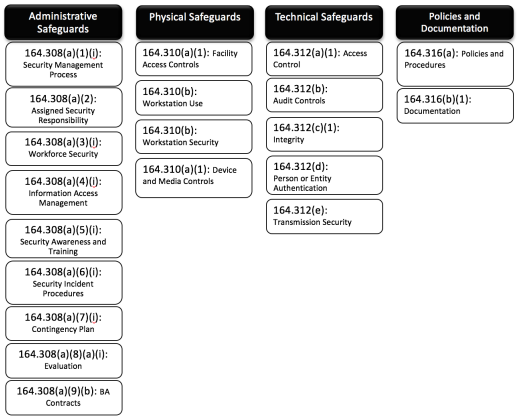

Fig. 8. Key HIPAA Controls.

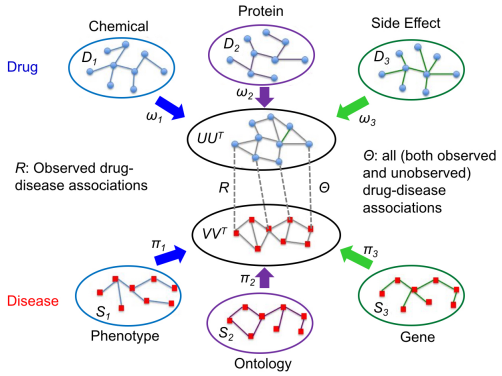

Fig. 9. A graphical illustration of the main idea of JMF. Reproduced from the open access source [38].

GDPR compliance requirements from Europe are specifically for health care systems and data in the EU region. It is more stringent in privacy requirements than HIPAA.

E. Auditability

Regulatory requirements and security forensic analysis requires auditability as a service in our system. External and internal teams may be able to audit the data usage and processing as well as security, privacy and compliance enforcements. Moreover, users need to be audited. Security controls need to be audited for how they are configured and managed. Logs are collected from each of these processes; change logs are managed, and such logged events cannot contain sensitive data.

Log analytics systems are used for audit and forensic purposes. Use of blockchain networks as described earlier helps in audit management. Hyperledger has an auditor view that allows an auditor to get access to the ledgers and search for use and processing of data, system integrity and user provenance.

V. Applications

We now describe bioinformatics applications which we have developed and are well-suited to run on our system. These applications use machine learning techniques to analyze large amounts of biological data. Scalability enabled by our cloud platform is critically important for scaling the applications to handle large data sets. In addition, some of our applications have strict privacy requirements necessitating the need for the features described earlier.

A. Drug Repositioning

Inefficiency of pharmaceutical drug development with high expenditure but low productivity has been widely discussed [31], [32]. Drug repositioning, finding additional indications (i.e., diseases) for existing drugs, presents a promising avenue for identifying better and safer treatments without the full cost or time required for *de novo* drug development. There have been several successful examples (for example, thalidomide to treat leprosy or finasteride for the prevention of baldness); however they have primarily been the result of serendipitous events based on *ad hoc* clinical observation, unfocused screening, and happy accidents. Big data analytics for both drugs and diseases provide an unprecedented opportunity to uncover novel statistical associations between drugs and diseases in a scalable manner.

1) Bioinformatics Data and Analytics: Bioinformatics methods were developed for inferring novel associations between drugs and diseases by the Guilt by Association (GBA) approach [33], matching drug indications by their disease-specific response profiles based on the Connectivity Map (CMap) data [34], utilizing structural features of compounds/proteins (e.g., molecular docking) to predict new drug indications [35], and constructing drug networks and using network neighbors to infer novel drug uses based on phenotypic profiles, such as side effects [36], and gene expression [37]. All of these methods only focus on different aspects of drug/disease activities and therefore result in biases in their predictions. Also, these methods suffer from the noise in the given information source.

We proposed a bioinformatics solution, Joint Matrix Factorization (JMF) [38], for drug repositioning hypothesis generation, by integrating multiple drug information sources and multiple disease information sources to facilitate drug repositioning tasks. Figure 9 depicts a high level idea of our overall algorithm. JMF utilizes drug similarity network, disease similarity network, and known drug-disease associations to explore the potential associations among other unlinked drugs and diseases. Then JMF is formulated and solved as a constrained non-convex optimization problem. As an example, we investigate three types of drug information (i.e., chemical structure, target protein, and side effect) and three types of disease information (i.e., phenotype, ontology, and disease gene). The proposed framework is also extensible, and thus JMF can incorporate additional types of drug/disease information sources.

Compared to prior art, it is worthwhile to highlight the following novel aspects that JMF can achieve simultaneously: (1) JMF can predict additional drug-disease associations by considering both drug information and disease information. (2) JMF can determine interpretable importance of different infor-

mation sources during the prediction. (3) As by-products, JMF can also discover the drug and disease groups, such that the drugs or diseases within the same group are highly correlated with each other, thus providing additional insights for targeted downstream investigations including clinical trials. We applied JMF to predict additional treatments for *Alzheimer's Disease* and *Systemic Lupus Erythematosus*, and some of the new drug-disease associations we predicted have been verified in clinical trials [38].

The techniques that we use for calculations like drug repositioning include determining quantitative similarities of entities such as drugs and diseases. Drug similarities can be calculated by multiple methods such as similarity in chemical structure, drug targets, and side effects. We have used the PubChem database [16] to determine similarities in chemical structures of drugs. We have used the DrugBank database [17] to determine similarity in drug targets. To determine similarity in side effects, we use the SIDER database [18].

Existing data on similarities between different drugs and diseases is incomplete. Therefore, computational methods are needed to infer additional disease and drug similarities from existing data. We have used collaborative filtering techniques such as matrix factorization [39] for inferring drug and disease similarities. Neural networks can be used as well.

Similarity-based techniques have also been used to predict drug-drug interactions. Tiresias is a knowledge-based prediction system that takes in various sources of drug-related data and knowledge as input and provides drug-drug interaction predictions as output [40]. Entities of interest for drug-drug interaction prediction are pairs of drugs instead of single drugs. Tiresias computes similarities on pairs of drugs by combining similarity metrics on individual drugs.

B. Drug Effect Signal Detection from Real World Evidence (RWE) Data

With the advent of access to digitized RWE, catalyzed by wide-spread adoption of electronic medical records (EMRs) as well as the confluence of big data and supporting analytical approaches, a systematic approach to clinically relevant drug-repositioning approaches is also enabled recently.

RWE, often defined as non-interventional data on individual's activities and health, are characterized by large, complex, intricately structured datasets often containing several years of data on millions of patients. Data sources for RWE can stem either from observational, simple trials (i.e. pragmatic trials), as well as from registries, administrative data, health surveys, EMRs, medical chart reviews, or adverse-event reporting and even social media. This type of data can address a wide range of challenges across drug development, and has been mainly used to support health economics research. However, RWE constitutes a fertile, and largely untapped, ground for generating and validating drug repositioning candidates, with the ability to systematically leverage such data being vastly dependent on the advent of sophisticated analytical methods such as artificial intelligence and deep learning, and their application to healthcare.

1) RWE data resources: Analysis can leverage both claims and medical records from the following databases:

The **Explorys SuperMart database** [41] includes medical data of over 50 million patients (approximately 15% of the US population), pooled from multiple different healthcare systems. Data consists of a combination of individual-level, de-identified clinical EMRs, healthcare system outgoing bills, and adjudicated payer claims and is standardized and normalized using common ontologies. The EMR data includes patient demographics, diagnoses, procedures and admissions, prescribed drugs, vitals and laboratory values.

The **Truven Health MarketScan Research Databases** [42] contain individual-level, de-identified, healthcare claims information from employers, health plans, hospitals, Medicare, and Medicaid programs, for the period of January 1st, 2011 to December 31st, 2015. Specifically:

- Truven Health MarketScan Commercial Database contains health insurance claims across the continuum of care (e.g. inpatient, outpatient, and outpatient pharmacy) as well as enrollment data from large employers and health plans across the United States which provide private healthcare coverage for more than 100 million employees, their spouses, and dependents.
- Truven Health MarketScan Medicare Supplemental Database is created for Medicare-eligible retirees with employer-sponsored Medicare Supplemental plans.
- Truven Health MarketScan Multi-State Medicaid Database contains the pooled healthcare experience of approximately seven million Medicaid enrollees from multiple States. It includes inpatient services and prescription drug claims, as well as information on enrollment, long-term care, and other medical care.

Data related to individual patients is integrated from all providers of care, maintaining all healthcare utilization and cost record connections at the patient level.

Much of this data is confidential, and maintaining data privacy while the data are being analyzed is critically important. The privacy-preserving features of our system are critically important for these types of applications.

2) RWE data analytics: Previous studies mainly leverage survival analysis to validate non-chemotherapy drugs associated with improved cancer survival [43] and/or decreased cancer risk [44] of patients from EMRs.

We are interested in mining EMRs in order to identify a potential indication from *multiple* existing drugs simultaneously. As an initial attempt, we extended the Self-Controlled Case Series (SCCS) [45] model to build a predictive model, called Drug Effects on Laboratory Test (DELT) algorithm, which uses the drug prescription history of patients to predict their continuous numeric values of Glycated hemoglobin (HbA1c) level [46]. We examined the drugs (predictors) that have significant blood sugar lowering effects. If some of them are not known to lower blood sugar already, we can consider those drugs as potential candidates for repositioning to control blood sugar, with further inspection.

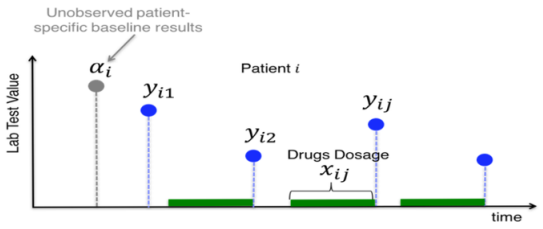

Fig. 10. Laboratory test measurements for patient i over time. y_{ij} is the laboratory test result for the j^{th} measurement for patient i. x_{ij} is a list of drugs that were taken by patient i prior to measurement j. α_i is the patient-specific baseline value if the patient is exposed to no drug. Reproduced from the open access source [46].

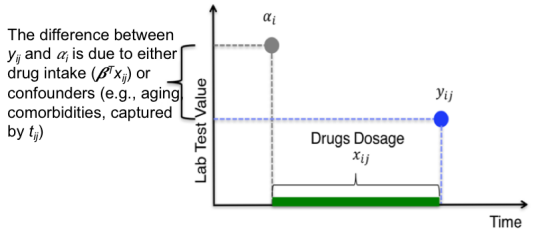

Fig. 11. The deviation of the measurement j from the baseline is due to either drug intake, or due to confounders such as aging and comorbidities, which will be accounted for in the model by t_{ij}. Reproduced from the open access source [46].

In addition, we observed that patients in EMRs have extremely diverse HbA1c level profiles (e.g. some people tend to have higher HbA1c level than the others because of their age, gender, and ethnicity). Thus, we imposed a parameter α_i which varies from patient to patient as different *healthy* patients may have different normal laboratory test values [46]. Figure 10 depicts the idea of our patient-specific baseline lab test result for each patient i. In other words, since there is a range of standard values for the laboratory test values, we cannot use the same value for all patients; therefore, the value α_i is patient-specific and learned from the data.

Furthermore, different HbA1c level measurements taken far apart in time might have very different values. For example, HbA1c levels on a *healthy* subject might change when the subject gets older, or some persistent blood sugar altering events (such as the diagnosis of diabetes) occur to a person. Thus, the change of HbA1c test results may not be because of the drug intake, but because of other confounders such as aging and comorbidities (see Figure 11). Therefore, we also included a time variant parameter t_{ij} in DELT that indicates the deviation of the measurement j of patient i from the baseline α_i to account for confounders [46].

Compared to prior art, our contributions in drug effect signal detection are as follows: (1) DELT looks at the joint exposure of multiple drugs at the same time (instead of marginal correlation). Therefore it is robust against confounders raised by co-medications. (2) DELT adds time-varying unobserved individual baseline parameters, and takes various other confounders into account (individual self-controlled design for gender, ethnicity, and time-varying baselines for aging, chronic comorbidity) implicitly. (3) DELT leverages the prior knowledge of drug therapeutic class and drug similarity network information into the SCCS model and achieves high accuracy in retrieving known effects of drugs. We evaluated the DELT algorithm on detecting drugs which lower HbA1c laboratory tests. Experiments show the evidence that DELT can be used to repurpose some unexpected drugs for diabetes [46].

VI. Related Work

In the area of trusted cloud platforms, Bessani et al. [8] present *TClouds*, a trusted cloud platform; our work is complementary to this work with an extension of container clusters. Jayaram et al present trustworthy geographically fenced clouds in [47]; this work is different from ours as it does not build a cloud for PHI data. Cloud platform providers like Amazon [48], Google [49] and Microsoft [50] offer HIPAA compliant services on their platforms to create and manage a compliant solution but not an architecture to create a compliant SaaS platform like discussed in this article. Intel SGX [51] and IBM SecureBlue++ [51], [52] support secrecy-preserving processing, which has applications to healthcare data processing.

In the area of health care platforms, Mohindra et al. [53] have presented a health cloud platform for the health care and life sciences industry. The design mentions a compliant cloud but does not exactly mention a trusted cloud platform. Authors in [54] present a scalable secure cloud architecture corresponding to IBMs Watson Health Cloud. The work however does not focus on the analytics environment and analytics use cases.

Data protection involving anonymization and leakage-free data integrity and authenticity verification has been proposed proposed by Kundu, Bertino and Atallah [27], [28], and later extended by [29], [30]. Yue et al. [55] proposed using blockchain for health care and privacy risk control. However, we are not aware of cloud-based systems that have used blockchain for security, data management and privacy of healthcare data.

VII. Conclusion

We have presented a cloud-based system for health care applications. Our system offers enhanced security and privacy over existing systems. We provide computational capabilities at clients to complement the processing taking place within cloud servers.

Acknowledgment

The authors would like to thank Isabelle Rouvellou and Ajay Mohindra for their valuable feedback and comments. The authors also would like to thank Mohamed Ghalwash, Fei Wang, Ying Li, and Jianying Hu for co-authoring bioinformatics application papers mentioned in Section V.

REFERENCES

[1] A. Iyengar, "Providing Enhanced Functionality for Data Store Clients," in *Proceedings of the IEEE 33rd International Conference on Data Engineering (ICDE 2017)*, April 2017.

[2] I. Drago, E. Bocchi, M. Mellia, H. Slatman, and A. Pras, "Benchmarking Personal Cloud Storage," in *Proceedings of IMC '13*, 2013, pp. 205–212.

[3] R. Gracia-Tinedo, M. Artigas, A. Moreno-Martinez, C. Cotes, and P. Garcia-Lopez, "Actively Measuring Personal Cloud Storage," in *Proceedings of the IEEE 6th International Conference on Cloud Computing*, 2013, pp. 301–308.

[4] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. H. Katz, A. Konwinski, G. Lee, D. A. Patterson, A. Rabkin, I. Stoica, and M. Zaharia, "Above the clouds: A berkeley view of cloud computing," EECS Department, University of California, Berkeley, Tech. Rep. UCB/EECS-2009-28, Feb 2009. [Online]. Available: http://www2.eecs.berkeley.edu/Pubs/TechRpts/2009/EECS-2009-28.html

[5] P. M. Mell and T. Grance, "Sp 800-145. the nist definition of cloud computing," Gaithersburg, MD, United States, Tech. Rep., 2011.

[6] "Information technology Trusted platform module," ISO/IEC, Standard 1654, August 2015. [Online]. Available: https://www.iso.org/standard/66510.html

[7] R. Yeluri and E. Castro-Leon, *Building the Infrastructure for Cloud Security: A Solutions View*, 1st ed. Berkely, CA, USA: Apress, 2014.

[8] A. Bessani, L. A. Cutillo, G. Ramunno, N. Schirmer, and P. Smiraglia, "The tclouds platform: Concept, architecture and instantiations," in *Proceedings of the 2nd International Workshop on Dependability Issues in Cloud Computing*, ser. DISCCO '13. New York, NY, USA: ACM, 2013, pp. 1:1–1:6. [Online]. Available: http://doi.acm.org/10.1145/2506155.2506156

[9] S. Berger, R. Cáceres, K. A. Goldman, R. Perez, R. Sailer, and L. van Doorn, "vtpm: Virtualizing the trusted platform module," in *USENIX Security Symposium*, 2006.

[10] D. C. E. Winn, *Cloud Foundry: The Cloud-Native Platform*, 1st ed. O'Reilly Media, Inc., 2016.

[11] I. P. Committee, "Ihe it infrastructure white paper: Health it standards for health information management practices," IHE International, Inc., Tech. Rep., September 2015. [Online]. Available: http://ihe.net/Technical_Frameworks/

[12] S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak, and Z. Ives, "DBpedia: A Nucleus for a Web of Open Data," in *Proceedings of the 6th International Semantic Web Conference (ISWC 2007) and 2nd Asian Semantic Web Conference (ASwC 2007)*, November 2007, pp. 722–735.

[13] D. Vrandecic and M. Krotsch, "Wikidata: A Free Collaborative Knowledge Base," *Communications of the ACM*, vol. 57, no. 10, pp. 78–85, October 2014.

[14] F. M. Suchanek, G. Kasneci, and G. Weikum, "Yago: A Core of Semantic Knowledge Unifying WordNet and Wikipedia," in *Proceedings of the 16th International World Wide Web Conference (WWW 2007)*, May 2007, pp. 697–706.

[15] J. Piñero, N. Queralt-Rosinach, À. Bravo, J. Deu-Pons, A. Bauer-Mehren, M. Baron, F. Sanz, and L. I. Furlong, "Disgenet: a discovery platform for the dynamical exploration of human diseases and their genes," *Database*, vol. 2015, 2015.

[16] E. E. Bolton, Y. Wang, P. A. Thiessen, and S. H. Bryant, "Pubchem: integrated platform of small molecules and biological activities," in *Annual reports in computational chemistry*. Elsevier, 2008, vol. 4, pp. 217–241.

[17] D. S. Wishart, C. Knox, A. C. Guo, S. Shrivastava, M. Hassanali, P. Stothard, Z. Chang, and J. Woolsey, "Drugbank: a comprehensive resource for in silico drug discovery and exploration," *Nucleic acids research*, vol. 34, no. suppl_1, pp. D668–D672, 2006.

[18] M. Kuhn, I. Letunic, L. J. Jensen, and P. Bork, "The sider database of drugs and side effects," *Nucleic acids research*, vol. 44, no. D1, pp. D1075–D1079, 2015.

[19] G. A. Miller, "Wordnet: a lexical database for english," *Communications of the ACM*, vol. 38, no. 11, pp. 39–41, 1995.

[20] A. Iyengar, "Supporting Data Analytics Applications Which Utilize Cognitive Services," in *Proceedings of the 37th IEEE International Conference on Distributed Computing Systems (ICDCS 2017)*, June 2017.

[21] F. Pérez and B. E. Granger, "IPython: a system for interactive scientific computing," *Computing in Science and Engineering*, vol. 9, no. 3, pp. 21–29, May 2007. [Online]. Available: http://ipython.org

[22] S. Chacon and B. Straub, *Pro Git*, 2nd ed. Berkely, CA, USA: Apress, 2014.

[23] S. Hosseinzadeh, S. Laurén, and V. Leppänen, "Security in container-based virtualization through vtpm," in *Proceedings of the 9th International Conference on Utility and Cloud Computing*, ser. UCC '16. New York, NY, USA: ACM, 2016, pp. 214–219. [Online]. Available: http://doi.acm.org/10.1145/2996890.3009903

[24] "Tcg pc client implementation specification for conventional bios," Trusted Computing Group, Incorporated., Tech. Rep., July 2005. [Online]. Available: https://trustedcomputinggroup.org/wp-content/uploads/PC-Client-Implementation-for-BIOS.pdf

[25] R. Sailer, X. Zhang, T. Jaeger, and L. van Doorn, "Design and implementation of a tcg-based integrity measurement architecture," in *Proceedings of the 13th Conference on USENIX Security Symposium - Volume 13*, ser. SSYM'04. Berkeley, CA, USA: USENIX Association, 2004, pp. 16–16. [Online]. Available: http://dl.acm.org/citation.cfm?id=1251375.1251391

[26] O. Goldreich, *Foundations of cryptography: volume 2, basic applications*. Cambridge university press, 2009.

[27] A. Kundu, M. J. Atallah, and E. Bertino, "Leakage-free redactable signatures," in *Proc. of ACM Conf. on Data and Application Security and Privacy (CODASPY)*, 2012, pp. 307–316.

[28] A. Kundu and E. Bertino, "Privacy-preserving authentication of trees and graphs," *International Journal of Information Security*, vol. 12, no. 6, pp. 467–494, Nov 2013. [Online]. Available: https://doi.org/10.1007/s10207-013-0198-5

[29] K. Samelin, H. C. Pöhls, A. Bilzhause, J. Posegga, and H. De Meer, "Redactable signatures for independent removal of structure and content," in *Information Security Practice and Experience*. Springer, 2012, pp. 17–33.

[30] M. U. Arshad, A. Kundu, E. Bertino, A. Ghafoor, and C. Kundu, "Efficient and scalable integrity verification of data and query results for graph databases," *IEEE Transactions on Knowledge and Data Engineering*, vol. PP, no. 99, pp. 1–1, 2017.

[31] S. Paul, D. Mytelka, C. Dunwiddie, C. Persinger, B. Munos, S. Lindborg, and A. Schacht, "How to improve rd productivity: the pharmaceutical industry's grand challenge," *Nature Reviews Drug Discovery*, vol. 9, no. 3, pp. 203–214, 2010.

[32] R. Berggren, M. Moller, R. Moss, P. Poda, and K. Smietana, "Outlook for the next 5 years in drug innovation," *Nature Reviews Drug Discovery*, vol. 11, no. 6, pp. 435–436, 2012.

[33] A. Chiang and A. Butte, "Systematic evaluation of drug-disease relationships to identify leads for novel drug uses," *Clinical Pharmacology Therapeutics*, vol. 86, no. 5, pp. 507–510, 2009.

[34] J. Dudley, M. Sirota, M. Shenoy, R. Pai, S. Roedder, A. Chiang, A. Morgan, M. Sarwal, P. Pasricha, and A. Butte, "Computational repositioning of the anticonvulsant topiramate for inflammatory bowel disease," *Science Translational Medicine*, vol. 3, no. 96, p. 96ra76, 2011.

[35] H. Luo, P. Zhang, X. Cao, D. Du, H. Ye, H. Huang, C. Li, S. Qin, C. Wan, L. Shi, L. He, and L. Yang, "Dpdr-cpi, a server that predicts drug positioning and drug repositioning via chemical-protein interactome," *Scientific Reports*, vol. 6, p. 35996, 2016.

[36] H. Ye, Q. Liu, and J. Wei, "Construction of drug network based on side effects and its application for drug repositioning," *PLoS ONE*, vol. 9, no. 2, p. e87864, 2014.

[37] Sirota, J. Dudley, J. Kim, A. Chiang, A. Morgan, A. Sweet-Cordero, J. Sage, and A. Butte, "Discovery and preclinical validation of drug indications using compendia of public gene expression data," *Science Translational Medicine*, vol. 3, no. 96, p. 96ra77, 2011.

[38] P. Zhang, F. Wang, and J. Hu, "Towards drug repositioning: a unified computational framework for integrating multiple aspects of drug similarity and disease similarity," in *AMIA Annual Symposium Proceedings*, vol. 2014. American Medical Informatics Association, 2014, pp. 1258–1267.

[39] Y. Koren, R. M. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *IEEE Computer*, vol. 42, no. 8, pp. 30–37, 2009. [Online]. Available: https://doi.org/10.1109/MC.2009.263

[40] A. Fokoue, M. Sadoghi, O. Hassanzadeh, and P. Zhang, "Predicting drug-drug interactions through large-scale similarity-based link prediction," in *Extended Semantic Web Conference Proceedings*, vol. 2016. Springer Nature, 2016, pp. 774–789.

[41] *Explorys SuperMart database*, 2016, https://www.ibm.com/watson/health/value-based-care/explorys-supermart/.

[42] *Truven Health MarketScan Research Databases*, 2016, https://marketscan.truvenhealth.com/.

[43] H. Xu, M. Aldrich, Q. Chen, H. Liu, N. Peterson, Q. Dai, M. Levy, A. Shah, X. Han, X. Ruan, M. Jiang, Y. Li, J. Julien, J. Warner, C. Friedman, D. Roden, and J. Denny, "Validating drug repurposing signals using electronic health records: a case study of metformin associated with reduced cancer mortality," *J Am Med Inform Assoc*, vol. 22, no. 1, pp. 179–191, 2015.

[44] R. Ruiter, L. Visser, M. van Herk-Sukel, J.-W. Coebergh, H. Haak, P. Geelhoed-Duijvestijn, S. Straus, R. Herings, and B. Stricker, "Lower risk of cancer in patients on metformin in comparison with those on sulfonylurea derivatives: results from a large population-based follow-up study," *Diabetes Care*, vol. 35, no. 1, pp. 119–124, 2012.

[45] S. Simpson, D. Madigan, I. Zorych, M. Schuemie, P. Ryan, and M. Suchard, "Multiple self controlled case series for large scale longitudinal observational databases," *Biometrics*, vol. 69, no. 4, pp. 893–902, 2013.

[46] M. Ghalwash, Y. Li, P. Zhang, and J. Hu, "Exploiting electronic health records to mine drug effects on laboratory test results," in *ACM International Conference on Information and Knowledge Management Proceedings*, vol. 2017. ACM, 2017, pp. 1837–1846.

[47] K. R. Jayaram, D. Safford, U. Sharma, V. Naik, D. Pendarakis, and S. Tao, "Trustworthy geographically fenced hybrid clouds," in *Proceedings of the 15th International Middleware Conference*, ser. Middleware '14. New York, NY, USA: ACM, 2014, pp. 37–48. [Online]. Available: http://doi.acm.org/10.1145/2663165.2666091

[48] "Architecting for hipaa security and compliance on amazon web services," Amazon Web Services, Inc., Tech. Rep., January 2018. [Online]. Available: https://d0.awsstatic.com/whitepapers/compliance/AWS_HIPAA_Compliance_Whitepaper.pdf

[49] "Hipaa compliance with g suite: Implementation guide," Google Cloud, Tech. Rep., October 2017. [Online]. Available: https://static.googleusercontent.com/media/gsuite.google.com/en//terms/2015/1/hipaa_implementation_guide.pdf

[50] "Microsoft azure hipaa/hitech act implementation guidance," Windows Azure, Tech. Rep., July 2017. [Online]. Available: https://gallery.technet.microsoft.com/Azure-HIPAAHITECH-Act-1d27efb0

[51] V. Costan and S. Devadas, "Intel sgx explained." *IACR Cryptology ePrint Archive*, vol. 2016, p. 86, 2016.

[52] R. Boivie and P. Williams, "Secureblue++: Cpu support for secure execution," *Technical report*, 2012.

[53] A. Mohindra, D. M. Dias, and H. Lei, "Health cloud: An enabler for healthcare transformation," in *2016 IEEE International Conference on Services Computing (SCC)*, June 2016, pp. 451–458.

[54] D. J. Dean, R. Ranchal, Y. Gu, A. Sailer, S. Khan, K. Beaty, S. Bakthavachalam, Y. Yu, Y. Ruan, and P. Bastide, "Engineering scalable, secure, multi-tenant cloud for healthcare data," in *2017 IEEE World Congress on Services (SERVICES)*, June 2017, pp. 21–29.

[55] X. Yue, H. Wang, D. Jin, M. Li, and W. Jiang, "Healthcare data gateways: found healthcare intelligence on blockchain with novel privacy risk control," *Journal of medical systems*, vol. 40, no. 10, p. 218, 2016.