

The Bucharest University of Economic Studies

The Faculty of Cybernetics, Statistics and Economic Informatics

Economic Informatics Specialization

Secure Blockchain-Based Application for Electronic Health Records

Bachelor Thesis

Advisor:

Prof. univ. dr. Răzvan Daniel Zota

Graduate:

Lupșan Sabrina

Bucharest 2021

Contents

[1. Introduction - 1 -](#_Toc66282924)

[a. What is Blockchain - 2 -](#_Toc66282926)

[b. Types of Blockchain: a comparison - 2 -](#_Toc66282927)

[c. Why private blockchain - 3 -](#_Toc66282928)

[2. Cybersecurity - 3 -](#_Toc66282929)

[a. The rise of threats - 3 -](#_Toc66282930)

[b. The CIA and AAA models - 3 -](#_Toc66282931)

[c. Encryption - 4 -](#_Toc66282932)

[a. Integrity with hashing - 6 -](#_Toc66282933)

[b. SQL Injection - 6 -](#_Toc66282934)

[*c.* Proof of work and DOS - 7 -](#_Toc66282935)

[3. Windows Application - 8 -](#_Toc66282936)

[a. .NET Framework, Windows Forms platform and C# Programming Language - 8 -](#_Toc66282937)

[b. Microsoft Azure with SQL - 8 -](#_Toc66282938)

[4. Using Blockchain to create, view and manage EHRs - 9 -](#_Toc66282939)

[5. Conclusions and future work - 9 -](#_Toc66282940)

[a. Conclusions - 9 -](#_Toc66282941)

[b. Future work - 10 -](#_Toc66282942)

[6. Bibliography - 10 -](#_Toc66282943)

[7. Annexes - 10 -](#_Toc66282944)

[References - 10 -](#_Toc66282945)

## Introduction

On the 10th of September 2020, the first casualty of a cyber attack was recorded. A ransomware attack hit The University Hospital of Düsseldorf and temporarily affected thirty servers. (CERT-RO, 2020) As a consequence, a patient had to be transferred to another hospital but died in transit. This happened during the COVID-19 pandemic, when hospitals were used at full capacity, and raised the alarm that cyber-attacks should be taken more seriously.

A study conducted by the Romanian National Computer Security Incident Response Team in 2020 uncovered 7670 vulnerabilities in hospitals and clinics across Romania. Out of those, 1337 were considered to pose a high risk, 4241 were deemed medium-risk and 2092 were low-risk. (CERT-RO, 2020). These vulnerabilities may range from computers with old and unpatched operating systems or operating systems that haven’t been updated in a long time, missing hardware, the use of common or simple passwords and others. The systemic lack of investments and awareness is now brought to light by the COVID-19 pandemic as more and more attackers find the medical system an area of interest and come up with more ways to break in.

Health records contains sensitive information about a patient and can become very dangerous vectors of attacks when fallen into the wrong hands. For example, a spear-fishing attack (meaning, an attack directed to a specific person with the means of tricking them into paying an amount of money, downloading a document or doing a certain action favorable to a hacker) can be very convincing if the attacker would posess key information such as the name of a medicine a patient urgently needs to buy. Such phishing attacks can be very dangerous, especially for elders, who are not aware of the cybernetic attacks that are being conducted nowadays. Deception schemes are more and more creative and effective and should be stopped from the root, before the attacker even obtains access to confidential informations.

The scope of this paper is to present a secure solution for the management of electronic health records, which does not allow the modification or deletion of records or permit unauthorized persons to access personal data. Several cybersecurity principles such as AAA (Authentication, Authorization and Accounting) and CIA (Confidentiality, Integrity and Availability) were implemented in order to efficiently implement a Blockchain-based application that has a user-friendly interface but modern tehnologies in the background.

## Blockchain

## What is Blockchain

TODO: define difficulty of blockchain (it is cited in chapter 4)

## Types of Blockchain: a comparison

There are two types of Blockchain: public and private. Both types serve different needs and bring different measures of security. There is a big debate on which blockchain is safer and, after analyzing the characteristics of both, I draw some conclusions.

Firstly, the problem of decentralization of data: in a public blockchain, the data is not stored on a server, in a single place, but rather everywhere: anyone that follows the rules imposed for the blockchain they want to contribute to can step in and verify, add and read the data on the blockchain. This is the case for Bitcoin. On the other hand, in a private blockchain, decentralization cannot be kept. However, only trusted entities can participate in the manipulation of the blockchain. This capability of the private blockchain increases access control, because not anyone can participate in the actions specific to a blockchain, but introduces the vulnerability of being hacked.

Secondly, a public blockchain’s security increases as the number of peers that take part in it increases. This is a problem of scalability since the transactions can only occur at a slow pace. This happens because every time an entity wants to contribute with a block, it must be approved by everyone in the blockchain. This can severely impact performance, which is not the case for a private blockchain. In the permissioned one, the number of authorized nodes is much smaller, and therefore the data is processed much faster. (Sharma)

Lastly, a private blockchain is more prone to hacking. If a hacker would get access to a trusted entity’s credentials, the vulnerability created would be critical. This raises many security problems that are not to be found in the unpermissioned version. The latter one can only be attacked if over 51% of the participants in the transactions would have the same ill-intention, all blocks would be tampered with and the proof of work would be completely redone for the entire blockchain. (Blockchain Tutorial for Beginners: Learn Blockchain Technology)

## Why private blockchain

After considering the advantages and disadvantages of both permissioned and unpermissioned blockchains, I have decided that the private blockchain would be more fit for the application presented in this paper.

One capability I considered is the invitation-only aspect of the private variant, which allows the trusted entities to keep their data private in only one institution. This type of technology permits an organization to only include its employees, suppliers and clients and to keep the data off of the internet. Otherwise, in a public environment, the transactions would not be kept private. Acknowledging this aspect, a private blockchain is a secure tool to manage sensitive data in an enterprise.

There is also the possibility of implementing different levels of access and a different set of transactions for entrants. For example, in a company, an employee would have different rights and needs than a client. (Heath, 2018) This can be achieved by creating different profiles and assigning them correctly to participants.

Another decisional factor in differentiating the two blockchains is the cost of the technology. In an unpermissioned implementation, validation and proof of work are essential. Validating transactions is time-costly because every entrant is considered untrusted. In the private blockchain, this is exactly the opposite, as the contributors to transactions are already trusted nodes, thus speeding up the process of accessing the blocks.

The biggest drawback of the private blockchain is the fact that it is credentials-based, which means that anyone that has access to credentials of a trusted person (that can be obtained through cybersecurity attacks like social engineering, phishing and others) or gains the trust of the already-existing participants can read and add to the blocks.

Taking into account the risks and the benefits of this type of blockchains, many companies have adopted this technology. Walmart, Spotify, DHL are just a few examples of the giants that acquired a permissioned blockchain. (Euromoney)

## Cybersecurity

## The rise of threats

## The CIA and AAA models

After analyzing the constant rise of threats and the numerous types of complex attacks that have been developed, cybersecurity provides two models to guide professionals. The CIA model represents the principle that an IT component should provide the following characteristics: Confidentiality, Integrity and Availability. The AAA model provides the means to achieve the goals of cybersecurity described in the CIA model. The triple-A concept refers to Authentication, Authorization and Accounting. (Nweke, 2017)

**CIA Model**

The first goal in the CIA model expresses the need for privacy of the data. Protecting data over the internet is crucial and the first step in a hacker’s malicious plans includes gathering sensitive information about their victim. Therefore, an attack can be stopped from the reconnaissance stage if an attacker cannot gain confidential information. This element of the model can be assured using data encryption and access control, among others.

Integrity assures that the original data is not tampered with by a third party. This plays the part in not allowing someone unauthorized to alter or delete information. Integrity can be achieved through hashing, which is the process of introducing data in a hash function that produces a unique output for every input.

The last principle in the CIA model is Availability. This component assumes that the owner of their data can access it and the needed resources whenever they want. DoS or DDoS attacks are built to bring down a system and block the users from accessing the needed resources. Availability can be provided when a system maintains redundancy, fault tolerance, access lists, etc. (CYBER EDU)

**AAA Model**

The first letter in the triple-A model stands for Authentication and described the way a user can be identified. Authentication serves to uniquely identify a user on the internet through the use of credentials. Experts strongly advise using multi-factor authentication in order to prevent unwanted individuals from accessing someone’s account just by guessing the password. Multifactor authentication is implemented by using something you know (along the lines of a password or PIN), something you have (like a key), or something you are (this represents biometrics, such as fingerprints). Multifactor authentication can be achieved by combining at least two categories and is more secure than simple authentication.

The second A in the mentioned model is Authorization. Authorization can be realized by implementing access control and limiting the resources a user can access based on their role in the organization. To achieve the best result, users should only be given the necessary permissions. This rule prevents leakage of sensitive data or limits the impact of malicious activity carried out by authorized personnel.

The last element of the AAA model is Accounting. Keeping records of what every individual does not only holds them accountable in case a suspicious action or a cybersecurity incident occurs but also discourages users from doing anything they desire in the organization. Accounting can be realized by logging the activities of individuals that can later be accessed for forensics. (Nweke, 2017)

By implementing the above-mentioned principles, identity and data theft can be limited.

## Encryption

Encryption is the process of altering input data in order to make it unreadable and only allow authorized access to the data. It is a very used tool that prevents unwanted access to sensitive data.

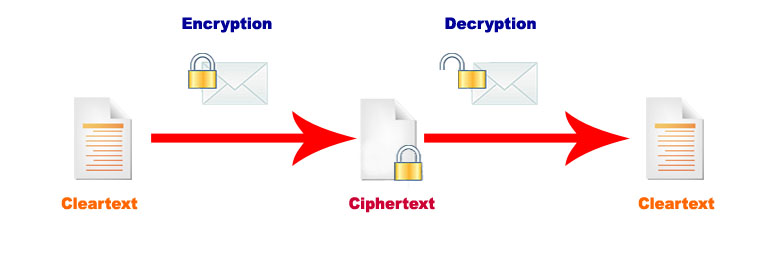


Figure 1 (Any Difference Between)

There are several categories of encryption. Two main types are symmetric and asymmetric encryptions. The first one involves a key that is used both for encryption and decryption and focuses more on safely managing the existing key. It is less costly because it doesn’t involve changing information back-and-forth. However, this is not the case with asymmetric encryption. In the latter one, there are two keys: one public and one private. Usually, the public one is used to encrypt the information and the private one is used to decrypt it. The private key is available only for authorized users. This variation is more costly, however, but is more effective.

3DES (Triple Data Encryption Standard) is a symmetric encryption algorithm that consists of a variation of the original DES algorithm, which is no longer secure because of its small key size. 3DES is improved because it encrypts the data three times (hence the name) with a key length of 192 bits. It encrypts data in blocks. The weakness of this algorithm is that its keys should be renewed often.

Another widely used algorithm is RSA, named after its creators (Ron Rives, Adi Shamir, Leonard Adelman). It is an asymmetric algorithm considered secure because it is very expensive to decrypt it. The key uses the factorization of a product of two large prime numbers with lengths of 100-200 digits. The public keys are exchanged using the Diffie-Hellman algorithm, which is used by secure protocols such as SSL, TLS, SSH and IPSec.

AES (Advanced Encryption Standard) is a private key algorithm with a fixed block size of 128 bits, which provides good protection. The key can have a size of 128, 192 and, the most recommended, 256 bits, hence why this algorithm is considered secure. An advantage is that AES is faster than DES and 3DES, being used for the protection of sensitive data and classified government information. (Cisco Systems, Inc.)

Elliptic Curve Cryptography (ECC) is a new and advanced take on public-key algorithms. ECC uses number theory and elliptic curves to encrypt data and brings robust security with smaller and more efficient keys. As a comparison, an RSA key of 15,360 bits provides the same level of security as an ECC key of 512 bits. (Cisco Systems, Inc., 2020)

Another type of encryption is hashing. It is an irreversible algorithm also called a digital fingerprint.

## Integrity with hashing

A cryptographic hash function is a mathematical algorithm that takes a variable-sized input and generates a fixed-size output that represents the hash of that data. There are a few characteristics that enable this function to be used for integrity checks.

This function is irreversible, which means that by only knowing the output of the operation, the input cannot be discovered. Another characteristic is that even if only one bit from the original input is changed, the entire result is significantly modified. Using this property, it is very easy to detect any alteration in the existing data. Furthermore, the hash function will always generate the same result for the same input (that means that the hash value will not change unless the data is changed). This mathematical algorithm is also collision-resistant, meaning that no two values lead to the same hash value. (Synopsys Editorial Team, 2015)

Knowing these properties, integrity over the internet can be achieved using a cryptographic hash function. This algorithm is only vulnerable to brute-force attacks. This type of attack represents repeated attempts to guess the input by comparing its hash value to the hash value of every attempted guess. A brute-force attack required big computing power and a lot of time. The cryptographic hash algorithms have improved over time in order to make it computationally infeasible to try to guess passwords.

Moreover, an enhancement named “salting” can be used to further increase the time needed to guess a value. The process of salting involves combining another string, like the username or the email of an account, in the initial input of the function. This is effective because it prevents the hacking of the most common passwords, like “123456” or „password”. (Picheta, 2019) If the user “John” would log in with the password “123”, the input data for the hash function could be “Jo123hn”. This is significantly harder to guess, thus increasing the protection from brute-force attacks. (M., 2016)

The most common hashing algorithms are MD5, SHA-1, SHA-2 and LANMAN. MD5 (Message Digest, version 5) was developed by Ron Rivest. It is a one-way hashing function that generates outputs of 128 bits; however, it was compromised in 2012 by The Flame malware and it is no longer considered. (Rountree, 2011) The SHA (Secure Hash Algorithm) family was created by the U.S. National Institute of Standards and Technology (NIST) and includes SHA-0, SHA-1 and SHA-2, with SHA-2 being a suite of functions. SHA-1 produces a result with 160 bits but is now replaced with SHA-2, which is considered safer. Other next-generation algorithms are SHA-386 and SHA-512. (Eastlake & Jones, 2001)

The irreversability of the hash function allows software developers to use it for safe authentication. In order for the application to protect its users’ personal credentials, like passwords, PIN codes, safe words etc, the hash of the corresponding credential can be stored rather than the plain-text data. This prevents data theft in case the application’s database is breached. Moreover, enhancing the hash with „salt” makes it very difficult for ill-intended people to guess or find the credentials. Using this technique both external and internal threats can be mitigated.

## SQL Injection

SQL Injection is a common web hacking technique that exploits input requests by writing malicious SQL code with the purpose to unknowingly run SQL statements on a database. Such efforts to inject code can modify or delete database data, can read sensitive information, or even shut down a DBMS. The ill-intended code can be designed to change the purpose of the affected SQL query, cause an error or delay a command.

A way of dealing with this type of vulnerability is placing validation policies on the input received from users. This can mean not accepting meta characters, limiting the amount of data accepted, or limiting the user to a set of allowed values. However, at some point intruders will find a way around that; it is very difficult to think of all the possible vulnerabilities that a piece of code can cause and there are many ways an SQL statement can be exploited.

A more versatile and secure way of protecting your application from SQL Injection is the use of Prepared Statements (or parameterized statements). A prepared statement is a stored procedure that doesn’t concatenate the query string and compiles it, but rather keeps the command compiled and executes the statement every time. (SQL Injection, n.d.) This is also efficient because the statement is only compiled once, so the overhead decreases. (Prepared statements and stored procedures, n.d.) It is more resilient as well to SQL injection because it uses placeholders in the query string and every parameter is checked if it is correct and if its type corresponds to the database column type first. (SQL Injection, n.d.)

## Proof of work and DOS

DoS or Denial of Service attack is a type of cybersecurity attack designed to shut down a machine or a network by repeatedly sending traffic or requests. This type of attack deprives users of accessing resources and is very used in the banking industry, as well as commerce, social media, and government organizations. (Paloalto Networks)

A way to protect a blockchain application from DoS or DDoS attacks is to implement a Proof of Work algorithm to slow down the process of adding records (blocks) to the chain. Proof of Work (PoW) is a consensus mechanism enabled for the entire blockchain that operates as a set of rules which must be met to allow a user to contribute to a blockchain. This usually requires the contributor machine to do some computer processing work, slowing down the pace at which blocks are added (therefore discouraging DoS attacks). PoW can be implemented in various forms, but one example is setting rules for the hash of each block; the algorithm can demand that every hash begins with a certain number of zeros, forcing the user to recompute every hash many times until they satisfy that rule. The difficulty is established by the number of zeros required. Other rules can also be put in place to increase the difficulty. (Karaivanov, 2019)

Other alternatives to the Proof of Work mechanism are Proof of State (PoS), Proof of Elapsed Time (PoET), Proof of Authority (PoA), Proof of Reputation (PoR), etc. PoS differs from PoW in the way that not every entity on the network can participate in the blockchain operations, but they have to be validated by the existing validators by executing a special type of transaction. PoET is a consensus algorithm that does not require high computational power but requires participating nodes to wait a period of time chosen at random before they are permitted to contribute with another block. PoA and PoR are two fairly similar algorithms. The basic idea is that users are only approved to contribute to the blockchain if they become validators first. A node can become a validator if they accumulate a high score that leads to a good reputation. The reputation is calculated using predefined formulas. (Zhang, Xue, & Liu, 2019)

## Windows Application

## .NET Framework, Windows Forms platform and C# Programming Language

.NET is a platform built for developing different types of applications. It is open-source, cross-platform, free, and can be used with different editors in different languages. It was released by Microsoft in 2002 and has reached over 3700 companies and 60.000 developers. .NET applications can be written in C#, F# or Visual Basic. Compiled code is stored in files which are called assemblies and are files with .dll or .exe extensions. A widely used tool for developing .NET applications is Visual Studio.

.NET Framework is used for building and running applications on Windows. It is a part of the .NET platform and is its original implementation. Besides desktop apps, it supports other services and websites. Two components make up the .NET Framework: the Common Language Runtime (CLR) and the Class Library.

The CLR is the engine that executes and handles the running applications. It delivers many useful services, including exception handling, garbage collection, thread management, and others. The Class Library is a set of APIs for writing and reading files, drawing, connecting to databases, etc. (.NET Framework documentation, n.d.)

Windows Forms is a User Interface framework designed to build Windows desktop applications. With this technology, the graphical construction of the application is easy to deploy and update, can be worked on while online or offline and brings many functionalities like drag and drops and print previews. Windows Forms carries many different controls, from the most used ones like buttons, textboxes and date pickers to drop-down boxes, contextual menus and error providers. The flexibility of this platform also allows developers to create and design their own controls as well as drawing pie charts, histograms, etc. (Desktop Guide (Windows Forms .NET), 2020).

C# is an object-oriented programming language rooted in the C family of languages. It is a modern programming language that provides garbage collection, exception handling, lambda expressions, asynchronous operations and many more. (A tour of the C# language, 2020)

## Microsoft Azure with SQL

Azure SQL Database is a Microsoft Azure service that provides relational database storage for applications. It is a cloud Platform as a Service (PaaS) engine with a scalable architecture and enterprise-needed features such as advanced security, monitoring and alerting, elastic pools and different price tiers to serve different needs. (Microsoft, 2020)

The security capabilities integrated in the Azure SQL Database service provide a layered defense using TLS (Transport Layer Security), advanced threat protection, server firewall and others.

TLS is a widely used protocol that assures encryption, authentication and data integrity. (Cloudflare) Data travelling to and from the Azure SQL Database is always encrypted using TLS.

The Server Firewall allows the database administrator to set IP rules. A rule can have a name, a starting and an ending IP. This allows the admin to set a rule for a single host, thus specifying the same IP address for both start and end IP fields, or to allow multiple devices to access the SQL server by delimiting the subnet. The minimum TLS version can also be set from the firewall page, rejecting any non-complient user.

Using SQL Authentication, the administrator has to set a username and a password when creating a database, later being allowed to access SQL Database only with those credentials. Microsoft Azure also allows the creation of other users with custom privileges, giving authorization based on the granted rights.

Another security feature that Microsoft Azure provides for this service is Advanced Thread Protection, a capability that monitors user’s actions and is able to detect abnormal activities such as DoS, brute-force, privilege escalations and others. Alerts can be viewed based on activity monitoring.

Other security features can be easily enabled from the Microsoft Azure Portal. Vulnerability assessment daa discovery and classification, compliance (a functionality that allows the database to participate in regular audits) and others are just a few of the utilities supplied by Microsoft Azure to help deploy a secure application that uses Microsoft SQL Database. (Microsoft, 2020)

## Using Blockchain to create, view and manage EHRs

The solution presented in this paper is a user-friendly application meant to store sensitive data in a secure environment and assure data confidentialy. It has a blockchain-based storage which does not allow the alteration of the medical records nor does it permit unauthorized access. The scope of this application is to provide a feasible solution for managing electronic health records for a hospital and could be the beginning of a national project designed to mitigate attacks and fix cyber security vulnerabilities present in hospitals.

The application has three types of users: the patients, the doctors (along with nurses) and the administrator.

The patients have to first register in the presence of the application administrator. This is done for security purposes in order to avoid impersonation. This could have scalability issues but I considered acceptable because every patient and every doctor only register once. The patient has to input their data: name, surname, patient ID (social security number), birthdate, password, confirm password and email. After this is accomplished, the user receives an email with their PIN code, which changes once every 30 days to prevent a brute-force threat. This is a simple form of Two Factor Authentication which adds another layer of security. After logging in, they can see a list of their records along with the doctor names and print them.

The doctors register in a similar way but also have their specialization included. Their possible actions are adding a new patient to their list (by knowing their patient ID), adding a new record (by knowing their PIN code), viewing patient’s records and printing them.

The administrator can, besides registering the users, see statistics on the database and perform an encrypted, local database backup. This can only be done after the validity of the blockchain is assured with a function that calculated all the hashes of the blocks from the beginning. The administrator can also overwrite the database in case of blockchain failure. However, this is a very critical measure and should only be performed in order to replace false data that was added to the database in the case of an attack. In order for this to be effective, the local backup should be done once a day at least.

It is important to be noted that every action on the application is logged. Every time the user wrongly inputs a password or a PIN code 5 times consecutively, they are locked out and the event is logged in a file. This happens every time the doctor adds a new patient, adds a new record, the administrator performs a local backup etc. Logging the actions of users is required in order to sustain the Accounting function of the AAA principle mentioned in chapter 2.b.

Regarding the blockchain aspect of the application, every patient’s record is constructed as a block which then is attached to the hospital chain. At every insert operation, the entire blockchain’s validity is checked. This measure is put in place in order to make sure that no records are added on top of false records. Instead of inserting, the doctor is informed that an error occured, the event is logged and the administrator is immediately emailed. If the blockchain is valid, the new block is added to the database. The hash of the previous block is also stored in the new blockchain, along with its own hash. If any block were to be altered, the corresponding hash would have to be recalculated, along with all the hashes of the previous blocks. This would require very high computational power and speed, factors which can be increased by setting the difficulty of the hash as high as possible (without affecting the application performance). More on the difficulty of a hash can be read in the chapter 1.a.

The defined classes are: Patient, Doctor, Block, Email, Hash and SqlBuilder. All of the classes contain private attributes and public constructors, getters and setters (except the SqlBuilder and Email classes, which are implemented using the Design Pattern Singleton).

The Patient class contains the following attributes, along with their types: lastName (string), firstName (string), patientID (long), PINCode (byte[]), hashedPassword (byte[]), birthdate (DateTime), emailAddress (string), doctors (ArrayList<Doctor>). The default constructor is required in order to connect using SqlConnection and different constructors with parameters are implemented (for different use cases).

The Doctor class contains the following attributes, along with their types: lastName (string), firstName (string), doctorID (long), PINcode (byte[]), hashedPassword (byte[]), birthdate (DateTime), emailAddress (string), patients (ArrayList<Patient>), specialisation (string). The implemented constructors are the default one and others with a mix of paramters depending on the case (for example, not in all situations a doctor needs a list of patients).

The Block class represents the medical records and contains the following attributes: title (string), description (string), doctorID (long), patientID (long), date (DateTime), timestamp (Timestamp), nounce (int), index (int), hashOfPrevBlock (string), hashOfCurrBlock (string). The date attribute represents the actual date of the appointment, while the timestamp field represents the date the block was added to the database. Nounce is a dummy variable incremented with 1 each time the hash of the block is calculated. This variable assures the hash changes everytime it doesn’t meet the difficutly required. The index of the block represents the index in the database and the last two attributes contain the hashes of the previous block, respectively the current block.

The Email class is a Singleton.

## Conclusions and future work

## Conclusions

To sum up everything that has been stated so far,

## Future work

Add Xrays, add other design patterns (like Builder)

## Bibliography

## Annexes

# References

*.NET Framework documentation*. (n.d.). Retrieved from Microsoft: https://docs.microsoft.com/en-gb/dotnet/framework/

*A tour of the C# language*. (2020, 08 06). Retrieved from Microsoft: https://docs.microsoft.com/en-us/dotnet/csharp/tour-of-csharp/

Any Difference Between. (n.d.). *Difference Between Encryption and Decryption.* Retrieved from Any Difference Between: https://anydifferencebetween.com/difference-between-encryption-and-decryption/

*Blockchain Tutorial for Beginners: Learn Blockchain Technology.* (n.d.). Retrieved from Guru99: https://www.guru99.com/blockchain-tutorial.html

CERT-RO. (2020, November 8). *Informare referitoare la vulnerabilități și atacuri.* Retrieved from CERT-RO: https://cert.ro/vezi/document/prezentare-sesiune-online-spitale-octombrie-2020

CERT-RO. (2020). *Informare referitoare la vulnerabilități și atacuri cibernetice privind spitale și clinici din România.* Bucharest: CERT-RO.

Cisco Systems, Inc. (2020, October). *Next Generation Cryptography.* Retrieved from Cisco: https://tools.cisco.com/security/center/resources/next\_generation\_cryptography

Cisco Systems, Inc. (n.d.). *What Is Encryption?* Retrieved from Cisco: https://www.cisco.com/c/en/us/products/security/encryption-explained.html

Cloudflare. (n.d.). *What is TLS (Transport Layer Security)?* Retrieved from Cloudflare: https://www.cloudflare.com/en-gb/learning/ssl/transport-layer-security-tls/

CYBER EDU. (n.d.). *What is the CIA Triad?* Retrieved from Forcepoint: https://www.forcepoint.com/cyber-edu/cia-triad

*Desktop Guide (Windows Forms .NET)*. (2020, 10 26). Retrieved from Microsoft: https://docs.microsoft.com/en-us/dotnet/desktop/winforms/overview/?view=netdesktop-5.0

Eastlake, D., & Jones, P. (2001). *US Secure Hash Algorithm 1 (SHA1).* RFC Editor. Retrieved from https://tools.ietf.org/html/rfc3174

Euromoney. (n.d.). *The rise of private blockchains.* Retrieved from Euromoney: https://www.euromoney.com/learning/blockchain-explained/the-rise-of-private-blockchains

Heath, N. (2018, September 05). *What’s the difference between a private and public blockchain?* Retrieved from In The Black: https://www.intheblack.com/articles/2018/09/05/difference-between-private-public-blockchain

Karaivanov, D. (2019, October 27). *Proof of Work Explained in Simple Terms.* Retrieved from The Chain Bulletin: https://chainbulletin.com/proof-of-work-explained-in-simple-terms/

M., I. (2016, October 27). *Three random words or #thinkrandom.* Retrieved from NCSC UK: https://www.ncsc.gov.uk/blog-post/three-random-words-or-thinkrandom-0

Microsoft. (2020, October 26). *An overview of Azure SQL Database and SQL Managed Instance security capabilities.* Retrieved from Microsoft: https://docs.microsoft.com/en-us/azure/azure-sql/database/security-overview

Microsoft. (2020, September 21). *What is Azure SQL Database?* Retrieved from Microsoft: https://docs.microsoft.com/en-us/azure/azure-sql/database/sql-database-paas-overview

Nweke, L. O. (2017). Using the CIA and AAA Models to Explain Cybersecurity Activities. *PM World Journal*, 3.

Paloalto Networks. (n.d.). *What is a denial of service attack (DoS) ?* Retrieved from Paloalto Networks: https://www.paloaltonetworks.com/cyberpedia/what-is-a-denial-of-service-attack-dos

Picheta, R. (2019, April 23). *How hackable is your password?* Retrieved from CNN: https://edition.cnn.com/2019/04/22/uk/most-common-passwords-scli-gbr-intl/index.html

*Prepared statements and stored procedures*. (n.d.). Retrieved from php.net: https://www.php.net/manual/en/pdo.prepared-statements.php

Rountree, D. (2011). *Hashing Algorithm.* Retrieved from Science Direct: https://www.sciencedirect.com/topics/computer-science/hashing-algorithm/pdf

Sharma, T. K. (n.d.). *PUBLIC VS. PRIVATE BLOCKCHAIN : A COMPREHENSIVE COMPARISON.* Retrieved from Blockchain Council: https://www.blockchain-council.org/blockchain/public-vs-private-blockchain-a-comprehensive-comparison/

*SQL Injection*. (n.d.). Retrieved from Owasp: https://owasp.org/www-community/attacks/SQL\_Injection

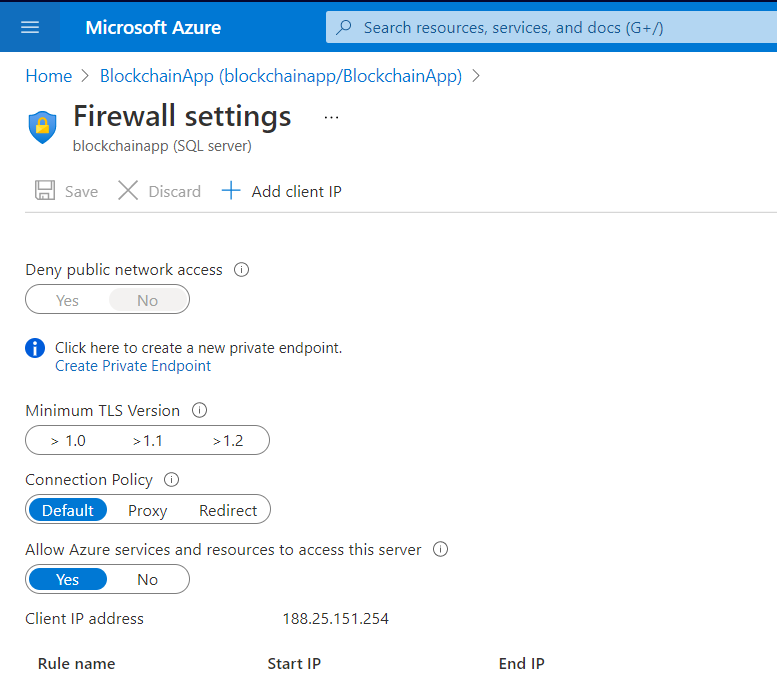
*SQL Injection*. (n.d.). Retrieved from W3schools.com: https://www.w3schools.com/sql/sql\_injection.asp

Synopsys Editorial Team. (2015, December 10). *What are cryptographic hash functions?* Retrieved from Synopsys: https://www.synopsys.com/blogs/software-security/cryptographic-hash-functions/

Zhang, R., Xue, R., & Liu, L. (2019). Security and Privacy on Blockchain. *ACM Computing Surveys*, 35.

[Figure 1 (Any Difference Between) - 5 -](#_Toc57114865)

Annexes

****