Online at: https://researchtrendsjournal.com **ISSN No:** 2584-282X

INTERNATIONAL JOURNAL OF TRENDS IN EMERGING RESEARCH AND DEVELOPMENT

Volume 2; Issue 6; 2024; Page No. 153-159

Received: 02-08-2024
Accepted: 12-09-2024

Mitigating cybersecurity risks in cloud computing through cryptographic protocols and service architecture

[1]Abhale Babasaheb Annasaheb and [2]Dr. Rohita Yamaganti

[1]Research Scholar, P.K. University, Shivpuri, Madhya Pradesh, India
[2]Professor, Department of Computer Science Engineering, P.K. University, Shivpuri, Madhya Pradesh, India

DOI: https://doi.org/10.5281/zenodo.15239402

Corresponding Author: Abhale Babasaheb Annasaheb

Abstract

The capabilities of cloud computing have moved the IT industry forward, with large enterprises migrating their processing and storage to it. One mainstream service mode for public cloud computing is Infrastructure as a Service (IaaS) public cloud. Cloud computing has several advantages when it comes to cyber-security, but one of the main ones is the heightened security measures offered by reputable CSPs Cloud computing is continuously altering how organizations store, use, and share software, workloads, and data Cloud computing is a cutting-edge technological development with huge worldwide implications for the future. Businesses and consumers alike may get many benefits from it.

Keywords: Cybersecurity, Cloud, Computing, services and organizations

Introduction

The rise of cloud computing as a means to combat cyber threats in the internet age is another noteworthy development in this area. Cloud computing has revolutionized the delivery, access, and management of computer resources, offering significant advantages in security, scalability, and flexibility. Cloud computing, at its core, is the delivery of computing services over the Internet, enabling users to tap into a shared pool of resources including servers, storage, networking, and applications as needed. In contrast to conventional on-premises infrastructure, which requires organizations to acquire and manage their own hardware and software, cloud computing allows customers to pay for the infrastructure and services offered by cloud service providers (CSPs).

Cloud computing has several advantages when it comes to cyber-security, but one of the main ones is the heightened security measures offered by reputable CSPs. The best cloud providers invest heavily in state-of-the-art security tools, infrastructure, and personnel to protect their customers' data from cybercriminals. Strong encryption, IAM, network segmentation, intrusion detection systems (IDPS), and regular security audits and assessments are all part of these security measures. Another perk is that businesses may strengthen their cybersecurity posture without spending a fortune on new security gear and training by using the security capabilities offered by cloud providers. This is particularly helpful for SMBs, or small and medium-sized businesses, as they may not have the resources to implement comprehensive cybersecurity measures alone.

Cloud computing also has built-in flexibility and scalability, so businesses can easily increase or decrease their computer resources as needed. Because of this nimbleness, businesses can respond quickly to cyber-security concerns and demands, such as unforeseen spikes in traffic or the need to install more security measures in reaction to new threats. Security rules and configurations may be centrally managed and controlled across several environments using cloud computing. Through consolidated dashboards and management interfaces, organizations can get visibility into their cloud infrastructure, track security events in real-time, and implement uniform security policies and controls across their cloud footprint.

On top of that, businesses may take use of cutting-edge security services and technologies that would be too costly or complicated to implement in-house thanks to cloud

computing. Among them, you may find cloud-based backup and disaster recovery systems, automated security operations and remediation solutions, and machine learning algorithms for threat detection and response. While there are many cybersecurity advantages to cloud computing, there are also certain hazards and factors to keep in mind. Concerns about data privacy and compliance, regulatory requirements, and the shared accountability model for cloud security are just a few of the security implications that organizations must thoroughly handle before moving sensitive data and workloads to the cloud.

Literature Review

Raja, Vinayak. (2024) [1]. Security and privacy are becoming more important issues for enterprises moving to cloud computing. With a focus on privacy and security concerns, this study paper thoroughly investigates the many dangers and weaknesses of cloud computing. The research takes a close look at possible dangers, such as illegal access or data breaches, and how they may affect user confidence and data integrity in cloud infrastructure. Additionally, it delves into practical tactics and solutions that may be used to reduce security threats and protect user privacy in cloud computing. This study adds to the continuing conversation on cloud security and privacy and provides academics and practitioners with a helpful guide to understanding the ever-changing world of cloud-based services.

Mughaid, Ala *et al*. (2024) [2]. The relatively new concept of "cloud computing" provides services like processing, data exchange, and storage, and it may be useful for network users. Global investors are pouring a lot of money into cloud computing because of the benefits it offers. But this aside, cloud computing security remains a top concern for companies and individuals using cloud services. Cloud computing has some security issues that have been carried over from previous computer systems. However, the other problems were caused by the unique characteristics and architecture of cloud computing. New security features in the platform ensure that only authorized users may access sensitive information.

Subramanian, Nalini & Jeyaraj, Andrews. (2018) [3]. The concept of "the cloud" refers to a system that allows users to tap into a common pool of computer resources on an as-needed or pay-per-use basis. Organizations and individuals alike may reap the financial and operational advantages of cloud computing. Despite these advantages, there are still several problems that limit the use of cloud computing. Constantly taken into account is the need of security. There is personal, ethical, and economical suffering due to the negative effect of the computer typology caused by the absence of this crucial element. In this article, we'll take a closer look at the security issues that cloud organizations encounter. Cloud users, data owners, and cloud service providers are all part of this category.

Spencer, Katherine & Withana, Chandana. (2023) [4]. The public sector is starting to catch up with private companies when it comes to using cloud computing, but it's still got its work cut out for it when it comes to integrating the various public cloud services used by different departments and resolving the inherent security issues with both people and technology in order to achieve efficiency and compliance standards. The study's overarching goal is to offer light on some of the cultural and technological factors that pose threats to cyber security in multi-cloud settings.

Albshaier, Latifa *et al*. (2024) [5]. It is believed that a key component of the future internet, with growing use and acceptability, would be the combination of cloud computing with the Internet of Things (IoT), both of which are integral to our daily lives. Many applications are expected to be transformed by this combination, which will give We may encounter difficulties in integrating the IoT with the cloud. The ability of cloud computing to disperse data and resources across several sites has greatly improved the usefulness of the Internet of Things (IoT), allowing access from a variety of industrial settings. However, traditional computer security methods aren't always applicable to cloud-based systems, therefore the quick transition to the cloud has prompted security worries. Cloud computing and the Internet of Things (IoT) may work together to overcome these challenges.

Research Methodology

Cloud computing is continuously altering how organizations store, use, and share software, workloads, and data. As more and more people across the world use the cloud, the quantity of sensitive data that might be compromised is increasing. In order to meet the needs of corporations regarding cloud security, the CSA Enterprise Architecture creates a cohesive plan. In order to ensure the safety of cloud computing, the CSA has developed a framework they call the Cloud Controls Matrix (CCM).

Cloud security and compliance encompasses all the responsibilities of a security team, but in a cloud environment, which may seem simple the majority of the responsibility for security rests with the cloud provider, since users may only access and manage their own use of the application in the cloud and not alter its core functionality. Cloud service providers should be transparent about the security precautions they take internally and the services they provide to their customers so that consumers can make an informed decision.

Data Analysis

The decoupling of services in the SOA architecture makes it more adaptable to new types of computing environments, such as service-based cloud computing, where development agility is paramount. In this paradigm, service interfaces provide the functionality that is targeted. In most cases, services are offered over the SOAP or REST network protocols and documented using the Web Service Definition Language (WSDL) standards. This paradigm for software development has several benefits. A user needs zero knowledge to use the interface since its parts are not tightly coupled. Since the supplier and the client may not share a same language, communication may be less reliable.

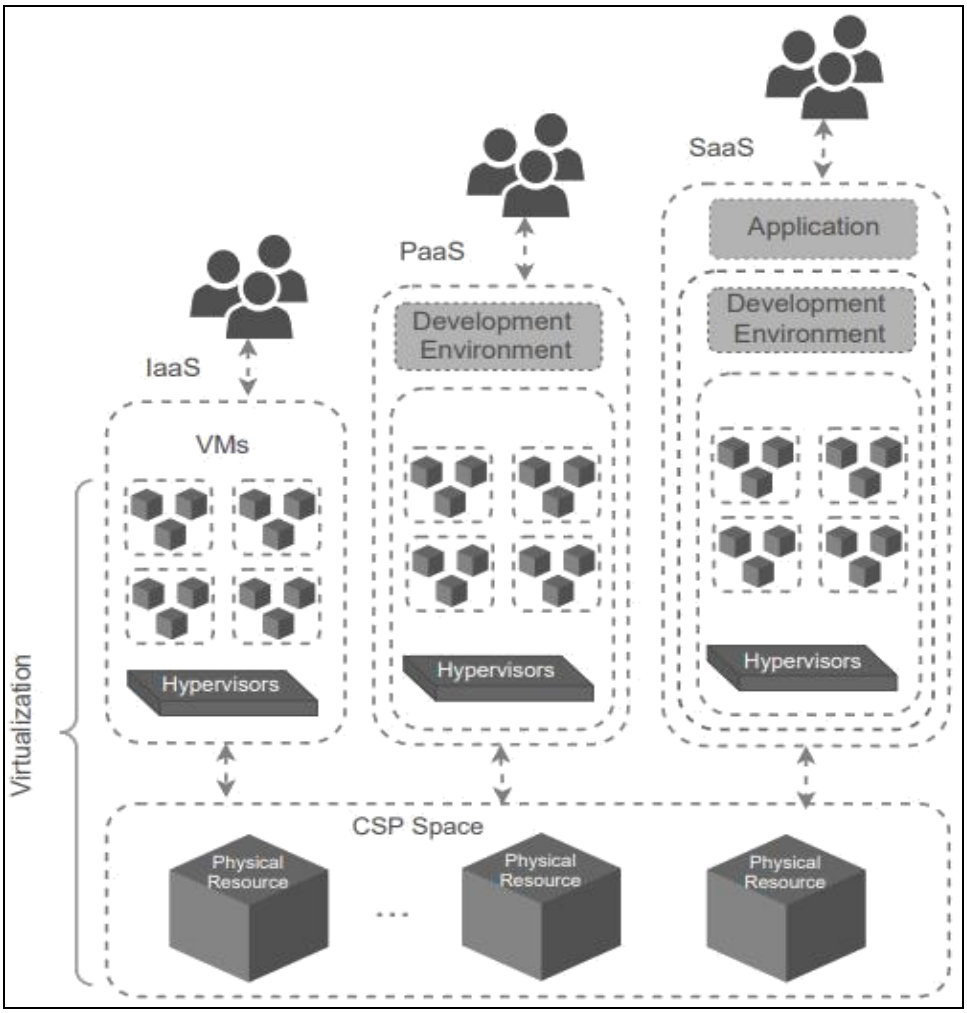

Fig 1: Multitenancy is abstracted in service-based cloud systems through virtualization

We will examine the usage of cloud computing and the behaviors that lead to the association reducing the event in different real-life scenarios. In each instance, we will swiftly establish the facts of the assault and then go into a discussion of those elements. We will also examine protocols for counteraction.

Criminals pose the threat of account hijacking when they get access to login credentials, which are highly sensitive data that may result in account breach.

There has been harassment on Instagram (Chatterbox) since May 2019. The American social media platform Instagram is owned by Facebook and allows users to post photos and videos. Display photos, followers, and contact information (email, phone number, etc.) stored in Instagram's database may be accessed without a password. While no financial data was compromised, personal details such as contact details and a location were made public, which may not have been the original intention. According to security expert Anurag Sen, an online database stores the data of 49 million Instagram users.

The graph demonstrates that malware injections increased faster in the years that followed. There were 165 million in 2013, and 702 in 2017, a significant growth. From this point on, the cases increased regularly until 2018.

The malware injections that occurred between 2009 and 2018, a ten-year span, are depicted in the following line graph:

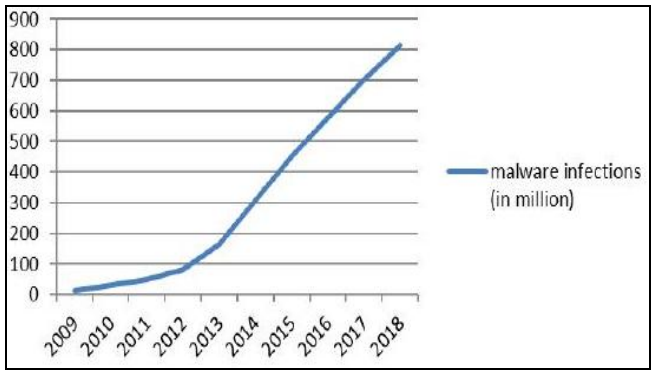

Fig 2: Total Malware Infections 10 years

In order to pull off a social engineering attack, one must have direct interaction with real people, testing various individuals to determine whether they can be manipulated to bypass standard security measures.

The CRT solution may be obtained by using the for Big Integer. Find Solution (Big Integer a, Big Integer function, where a and n are arrays of Big Integer integers. You can see the code for finding the CRT solution that works with Big Integers in Table 1. The function crypto Big. Inverse (Big Integer p, Big Integer g) is used to compute the modular multiplicative inverse in the find Solution function, as illustrated the inverse function's output is g-1 (mod p).

Table 1: The Code for Finding CRT Solution

```
import java.math.BigInteger;
public class CRTforBigInteger {
// Finding CRT solution Xr given Xr = a1 (mod n1), x = a2 (mod n2) ...
public static BigInteger findSolution(BigInteger[] a, BigInteger[] n)
      {
                  if(a==null || n ==null)
                     {
            System.out.println("a or n must have at least 2 elements");
                        return null;
                     }
                  if(a.length<2 || n.length<2)

                  {
            System.out.println("Length of a or n are less than 2");
                        return null;
                  }
                  if(a.length != n.length)
                  {
                  System.out.println("Length of a and n are not same");
                        return null;
                  }
```

```
                  BigInteger x = a[0];
                  BigInteger nInverse, y, z;
                  BigInteger ni = BigInteger.ONE;
                  int i = 0;
                  while(i< a.length-1)
                  {
                  ni=ni.multiply(n[i]);

//check if the n's are relatively prime
if(!BigInteger.ONE.equals(n[i+1].gcd(ni))){
 System.out.println("The n's are not relatively prime->"+n[i+1]+","+ni);
                              return null;
                                    }
                  // calculating the modular multiplicative inverse
                  nInverse = cryptoBig.inverse(n[i+1], ni);
                        if(nInverse.compareTo(BigInteger.ZERO) == -1 )
                        nInverse = nInverse.add(n[i+1]);
                         z = a[i+1].subtract(x);
                         z = z.multiply(nInverse);
                         y = z.remainder(n[i+1]);
                  if(y.compareTo(BigInteger.ZERO) == -1) // y < 0
                         y = y.add(n[i+1]);
                         x = x.add(ni.multiply(y));
                         i++;                }
                  return x; // CRT solution Xr
            }
      }
```

Table 2: The Code for Calculating the Modular Multiplicative Reverse

```
//returns g^-1 (mod p)
      public static BigInteger inverse (BigInteger p, BigInteger g)
            {
                  //g inverse = g ^ (p-2) (mod p)

BigInteger[] gInverse = ExtendedEuclid(p, g);//crypto.fastSq(p, g, p-2);
            if (gInverse[2].compareTo(BigInteger.ZERO) != -1)
                        //return p+gInverse[2];
                        return p.add(gInverse[2]);
                  return gInverse[2];


            }
public static BigInteger[] ExtendedEuclid(BigInteger a, BigInteger b)

/* This function will perform the EEA to find the GCD of a and b.  We
assume here that a and b are non-negative (and not both zero).  This
function also will return numbers j and k such that d = j*a + k*b where
d is the GCD of a and b.*/
                {
                    BigInteger[] ans = new BigInteger[3];
                    BigInteger q;
                     if (b.equals(BigInteger.ZERO)){
                        ans[0] = a;
                       ans[1] = BigInteger.ONE;
                       ans[2] = BigInteger.ZERO;
                    }
                    else
                  {    /*  Otherwise, make a recursive function call  */
                    q = a.divide(b);
                    ans = ExtendedEuclid (b, a.remainder(b));
                    BigInteger s = ans[2].multiply(q);
                    BigInteger temp = ans[1].subtract(s);// - ans[2]*q;
                     ans[1] = ans[2];
                     ans[2] = temp;
                  }
                    return ans;
                }
```

Returning the digest value in a byte array format, the function calculates the digest of the provided file according to the selected hash algorithm. The chosen algorithm determines the output digest length. For example, a digest length of 256 bits is produced using SHA-256. The next step is to encrypt the subsequent digest using the RSA encryption function for the purpose of verifying authenticity and integrity, the computer code is found in Table 3 and is utilized to compute the value, F Signature.

Table 3: The Code for Calculating the Integrity and Authenticity Proof

```
// Calculating the input file digest and data owner signature
String PrivKeyFile = "private1.key"; //Data owner private key

//Calculating the file digest
byte[] FileDigest = Hash.createFileHash(InputFile, "SHA-256");

// (Signature) Encrypting the file digest with data owner private key
byte[] FSignature = RSAforBytes.rsaEncrypt(FileDigest, PrivKeyFile);
```

The program is prepared to run when the parameters have been defined. First, as shown in up top, the execution determines the HMAC-SHA256 of the keywords. shows an example of how it encrypts the value Cr||Ks using each user's public key. By using the method CRT for Big Integer. Find Solution, the encrypted values that are generated with the customers' reasonably prime integers are used to ascertain the shared value Xr. As seen in, it took around 4.1 ms to compute the CRT solution for five users whose relative prime numbers were 1025 bits long and whose Cr||Ks value was 1024 bits short. The resulting Xr was 5122 bits.

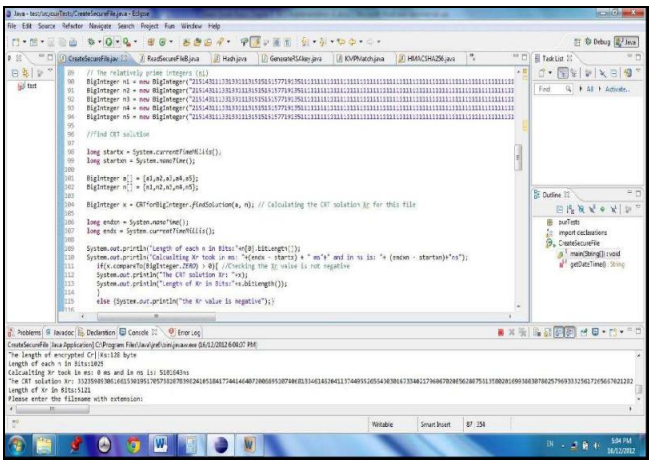

Fig 3: Snapshot of Calculating CRT Solution for Five Users

The DCS file includes the encrypted data file content as well as information such as the Xr value and the integrity and authenticity evidence. While the ReadSecureFileB.java class is active, the application will prompt the user to input the DCS file as input, followed by the output file's name and extension. See Table 4 for the source code that reads the DCS file as input and writes the encrypted data file as output.

Table 4: Code Description of Reading DCS file and Recreating the Original File

Code	Description
`RandomAccessFile f = new RandomAccessFile(file, "r");`	Creates a random access file stream to read from the file specified by the File argument (file) which is the input DCS file.
`RandomAccessFile out = new RandomAccessFile(Dest, "rw");`	Creates a random access file stream to write to the file specified by the File argument (Dest) which is the output file of the encrypted file content in the DCS file.
`int keyword_no = f.readInt();`	Reads the number of encrypted keywords attached. (first 4 bytes of the file)
`int l=keyword_no*32+4; f.seek(l);`	Calculates the length in bytes from the file beginning to the end of the encrypted key words. Then moves the file pointer to that end.
`String File_owner = f.readUTF();`	Reads the string of the data owner name. The file pointer moves to end of the bytes representing this string.
`long Cr = f.readLong();`	Reads the 8 bytes representing the C_r value in Long integer.
`int x1 = f.readInt();`	Reads the 4 bytes representing the number of bytes of the X_r in an Integer.
`byte[] xr = new byte[x1]; f.read(xr);`	Specifies the length of the X_r byte array from the previous step. Then reads the X_r according to its length.
`byte[] FSignature = new byte[128]; f.read(FSignature);`	Specifies the length of the byte array that represents the file signature (i.e. 128 bytes * 8 = 1024bit). Then reads file signature byte array according to the specified length.
`byte[] buf = new byte[1024]; int len; while ((len = f.read(buf)) > 0) { out.write(buf, 0, len); }`	This set of codes reads the content of the input DCS file (f) that represents the encrypted file content and writes it to the output file (out). This code resulted in creating the original encrypted file separated from its DCS file.

Table 4 shows the code that an authorized user or cloud provider may use to retrieve data from a DCS file. This code also includes fields for the data owner's name, Xr, Cr, and the encrypted data file's signed digest. The key Ks, which decrypts the encrypted file, may only be obtained by authorized users. Furthermore, the data file's authenticity and integrity may be checked by an authorized user.

An authorized user may confirm the encrypted data file's authenticity and integrity after reading the associated file signature and acquiring the file from a DCS. Table 5 displays the verification code for integrity and authenticity.

Table 5: Code Description of Verifying Integrity and Authenticity

Code	Description
`String PublicKeyFile = "publicOwner.key";`	Specifies the file containing the data owner public key.
`byte[] OldDigest = RSAforBytes.rsaDecrypt(FSignature, PublicKeyFile);`	Decrypts the attached encrypted (signed) file digest by using the data owner public key. The result of this operation represents the 256 bit of the original encrypted file digest in a byte array (i.e OldDigest).
`byte[] FileDigest = Hash.createFileHash(outfile, "SHA-256");`	Computes the recreated encrypted file (i.e. outfile) digest by hashing it with SHA-256 algorithm. The result of this operation represents the digest of the recreated file in a byte array (i.e. FileDigest).
`If(Arrays.equals(OldDigest,FileDige st)) System.out.println("OK, the integrity and authenticity proof are verified"); else System.out.println("Attention, the integrity and authenticity proof are NOT verified");`	Compares the attached digest (i.e OldDigest) with the new calculated digest (i.e. FileDigest). If they are equal, the encrypted file integrity is maintained and originated from its data owner. Otherwise, there is a concern about the file integrity.

The verifying process incurs a computational overhead and the amount of overhead is according to the file size. For example, verifying the integrity and authenticity of a file with a size of 40.7 MBs took about 720 milliseconds, as shown in Figure 4.

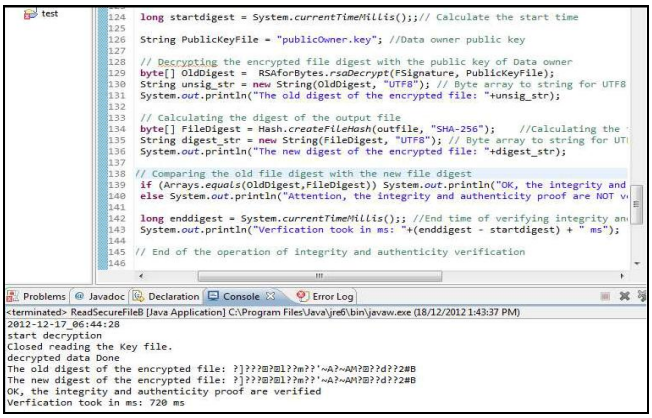

Fig 4: Snapshot of Verifying Integrity and Authenticity

Conclusion

Cloud computing is a cutting-edge technological development with huge worldwide implications for the future. Businesses and consumers alike may get many benefits from it. Cloud security and compliance encompasses all the responsibilities of a security team, but in a cloud environment, which may seem simple the majority of the responsibility for security rests with the cloud provider Cloud computing, at its core, is the delivery of computing services over the Internet, enabling users to tap into a shared pool of resources including servers, storage, networking, and applications as needed The capabilities of cloud computing have moved the IT industry forward, with large enterprises migrating their processing and storage to it.

References

1. Raja V. Exploring challenges and solutions in cloud computing: A review of data security and privacy concerns. Journal of Artificial Intelligence General Science (JAIGS). 2024;4:121–144. doi:10.60087/jaigs.v4i1.86
2. Mughaid A, Obeidat I, Abualigah L, Alzubi S, Daoud M, Migdady H. Intelligent cybersecurity approach for data protection in cloud computing-based Internet of Things. International Journal of Information Security. 2024;23:1–15. doi:10.1007/s10207-024-00832-0
3. Subramanian N, Jeyaraj A. Recent security challenges in cloud computing. Computers & Electrical Engineering. 2018;71:28–42. doi:10.1016/j.compeleceng.2018.06.006
4. Spencer K, Withana C. Exploring cyber security challenges of multi-cloud environments in the public sector. In: Lecture Notes in Computer Science (LNCS). 2023. doi:10.1007/978-3-031-29078-7_19
5. Albshaier L, Budokhi A, Aljughaiman A. A review of security issues when integrating IoT with cloud computing and blockchain. IEEE Access. 2024;PP:1–1. doi:10.1109/ACCESS.2024.3435845
6. Hao S. Data security issues in intelligent cloud computing systems. International Journal of Computer Science and Information Technology. 2024;4:101–106. doi:10.62051/ijcsit.v4n1.12
7. Singh S, Vadi V, Usmani A, Nayak P. Integration of cloud computing and deep learning for cybersecurity. International Multidisciplinary Research Journal Review. 2024;1:29–33. doi:10.17148/IMRJR.2024.010104
8. Alouffi B, Hassnain M, Alharbi A, Alosaimi W, Alyami H, Ayaz M. A systematic literature review on cloud computing security: Threats and mitigation strategies. IEEE Access. 2021. p. 1–1. doi:10.1109/ACCESS.2021.3073203
9. Mohamad N, Saidin N, Zaidi M. Data security and privacy issues in cloud computing: Challenges and solutions review. TechRxiv Preprint. 2023. doi:10.36227/techrxiv.170327865.59737799/v1
10. Patala N, Kadyamatimba A, Madzvamuse S. Adoption of cloud-cyber security: Challenges and perceptions within resource constrained higher education institutions. In: Proceedings of the 2018 Open Innovations Conference (OI). 2018. p. 313–318. doi:10.1109/OI.2018.8535939
11. Mamidi S. Deep learning applications in cloud security: Challenges and opportunities. Journal of Artificial Intelligence General Science (JAIGS). 2024;4:310–318. doi:10.60087/jaigs.v4i1.165
12. Singh A, Chatterjee K. Cloud security issues and challenges: A survey. Journal of Network and Computer Applications. 2016;79:88–115. doi:10.1016/j.jnca.2016.11.027
13. Walling S. A comprehensive review on cloud computing and cloud security issues. International Journal of Scientific Research in Computer Science, Engineering and Information Technology. 2020;483–490. doi:10.32628/CSEIT206489