A framework on Group Data Sharing Based Agreement Implementation in Cloud Computing

Ms. Sohilara Kazi[1], Dr. A. S. Kapse[2]

[1]CSE Department, Student, Anuradha Engineering College, Chikhli, INDIA
[1]CSE Department, Asst. Prof., Anuradha Engineering College, Chikhli, INDIA

Abstract

Cloud computing and cloud storage have become hot topics in recent decades. Both are changing the way we live and greatly improving production efficiency in some areas. At present, due to limited storage resources and the requirement for convenient access, we prefer to store all types of data in cloud servers, which is also a good option for companies and organizations to avoid the overhead of deploying and maintaining equipment when data are stored locally. The cloud server provides an open and convenient storage platform for individuals and organizations, but it also introduces security problems. Due to the particularity of some industries, some enterprises have extremely high requirements on the security of data. If the data is lost or destroyed, the enterprise will bear serious consequences. The second problem is reliability. Only by ensuring that the data records are accurate and reliable can the data transmission be more secure and the enterprise win a higher reputation. In addition, the issue of cloud computing regulation is also very important. Only through efficient and orderly regulation can enterprises operate healthily and normally. At present, many small and medium sized enterprises need to spend huge costs on data storage, which will continue to increase with the expansion of data volume. However, the emergence of cloud storage services has brought the dawn for these enterprises, they will need to store the data transmission to the cloud, through the cloud storage service provider to solve this problem, so that they can save the cost of their own data center, use low price to achieve the purpose of data storage.

Keywords: Computing, Storage, Security, Data, Transmission

1. INTRODUCTION

The main process includes inputting the dongle after the computer is running and then verifying the media. If the conditions are met, the media will be further loaded, and if the media verification is not passed, the program will be forbidden. At this time, the entire system cannot continue to run, and the judgment efficiency is high. After the loading medium is passed, the next step is to verify the certificate file and then verify the user' name and password. After these subroutines are verified, the program can be successfully opened, and if any intermediate link fails to pass the verification before starting the program, then the safety protection system will all recognize it as a failure and return to the prohibition of loading the program again. It can be seen that the authentication of login authority plays a vital role in the security protection of the entire computer system, and the security protection level and overall operation efficiency are relatively high.

A. Process Security Protection

Process security protection belongs to the application layer of the computer multimedia security protection system. Its existence can control the process of the computer multimedia security protection system to start, terminate, verify external information, exclude untrusted processes, and add trusted process. In the actual running process, when the process security protection module is opened, if it is judged that it is a safe condition at this time, the process will be further created, and if the process security protection module fails to open or the process creation fails, the program will be directly terminated. After the program is created successfully, the computer will further execute the underlying driver and verify the underlying process information after success, open the process after passing it, send the program to the application layer after failure, and use the query open program mode to operate the entire running process. Only the program that trusts the current operation is executed in the open program, and the untrusted operation is added to the blacklist. After the trusted program is successfully run, it can be further added to the trusted program to speed up the efficiency of the next operation. During the entire running process of the security protection module, the entire running program will end whether it is the failure of the underlying driver or the failure of the program to be sent to the program.

B. Document Security Protection

Document security protection is also subordinate to the application layer of the computer multimedia security protection system. Its existence can ensure the integrity and confidentiality of data files. During the operation process, it plays a role in starting the program, loading the driver, reading the document, and controlling the document application effect. Since the current computer system generally uses the drive system provided by Windows officially, although this drive system is stronger than the application programming interface in protection and safety, it has greater difficulties in specific technical implementation.

C. Mobile Media Security Protection

The mobile media security protection also belongs to the application layer of the computer multimedia security protection system, and its existence can realize the functions of computer reading and writing control and information transmission. In the process of daily work, life, and study, mobile media are ubiquitous, and the security protection of such mobile media with communication and connection attributes is considered to be a module that needs to be strengthened. From the perspective of past mobile media security protection strategies, the overall structure should be mainly carried out from the steps of monitoring the system before use, preventing during use, and eliminating hidden dangers after use

D. Objectives

- To share data securely and efficiently between many participants in cloud environment, with help of key generation. To achieve the same key agreement protocols is used along with the encryption and decryption technique. Same key is shared with the participants and to avoid malicious attack the fault tolerance property is used to deliver secure data.
- To provide security to all these components and interaction of these components with each other needs to be addressed.
- To improve the quality of service delivered on the network is another concern.
- To control loss over physical, Logical of system, and alternative control to client’s assets, mismanagement of assets Are some additional concerns.

2. LITERATURE SURVEY

Jian Shen et al. proposed an efficient and secure block design-based key agreement protocol by extending the structure of the SBIBD to support multiple participants, which enables multiple data owners to freely share the outsourced data with high security and efficiency. Note that the SBIBD is constructed as the group data sharing model to support group data sharing in cloud computing. Moreover, the protocol can provide authentication services and a fault tolerance property. [1]

Mehdi Bahrami et al. proposed fields to establish a virtual IT department via the Internet. The cloud computing offers different virtual services like traditional IT department, such as storage, stream server and database server. The cloud provides a cost effective model through pay-per-use that allows each individual or businesses in healthcare start a cloud based service with minimum investment.[2]

Tessema Mengistu at al. proposed that The current Cloud Computing services are based on the “data center” approach, where hundreds of thousands of dedicated servers are setup to give the services. Setting up the data center for cloud is expensive and running the infrastructure needs expertise as well as a lot of resources such as high power for cooling, redundant power for assured availability, etc. For example, 45% of the data center cost goes to the acquisition of servers, 25% goes to specialized infrastructure for fault tolerance, redundant power, cooling systems, and backup batteries, while electrical cost consumed by the machines accounts for 15% of the amortized total cost. In addition to the vast number of servers used in data centers, there are billions of Personal Computers (PCs) owned by individuals and organizations worldwide. [3]

Zhao Tianhai et al. proposed that Cloud computing is the new research and cooperation pattern for science computing. Keahey, et al. proposed one of the first cloud-based infrastructures for computational science, Science Cloud [1]. Software-as-a-service [2] is at the top end of the cloud computing stack, which is seen as a replacement to traditional software. With SAAS, the cloud operator provides end users with an integrated service which comprising hardware, software and development platform. The resource management is the key of application software service in science cloud computing. The optimization decomposition approach to solve cloud resource allocation for satisfying the cloud user’s needs and the profits of the cloud providers [3].A new resource management framework presented provides efficient green enhancements within a scalable cloud computing architecture [4].

Wang Xiaoyu et al. proposed that Cloud computing has many definitions, but they are all pretty much the same. So called cloud computing, is composed of distributed processing, parallel processing and grid computing, development of a dynamic, good extension, virtualization technology is used to establish a unified infrastructure, services, applications and information resources pool, with distributed technology to the resource pool of infrastructure to effectively organize and run a computing model. Cloud computing as a new IT resources service mode, is composed of a large number of computer resources Shared IT resources pool, users have to use the data will be uploaded to the data resources pool, we can see IT as a huge data centres, again after the data integration to provide users with computing and storage services, to a great extent, improve the efficiency of resource use. After the cloud users upload their data to the cloud, the management of the data will no longer be controlled by the cloud users. At this time, the user data is in an uncontrollable domain, which makes the users lack sufficient trust in the cloud service provider, and of course, they do not want their data to be seen by others. [5]

Yanhong Shang1 et al.states that the outbreak of various information data leakage incidents have made computer network users increasingly demanding personal information security. The construction of a computer network information security environment can not only effectively ensure information security but also achieve greater effective protection of information security. This requires that the traditional information security protection measures be used as the basis to continuously increase the construction of information security. [6]

3. MULTIDIMENSIONAL DATA SECURITY PROTECTION SYSTEM

A. Cloud Computing Services

Cloud services are infrastructure, platforms, or software that are hosted by third-party providers and made available to users through the internet. Cloud services facilitate the flow of user data from front-end clients (e.g., users' servers, tablets, desktops, laptops—anything on the users' ends), through the internet, to the provider's systems, and back. Cloud services promote the building of cloud-native applications and the flexibility of working in the cloud. Users can access cloud services with nothing more than a computer, operating system, and internet connectivity. Cloud computing has risen massively in terms of popularity in recent times. This is due to the way it reduces on-premise infrastructure cost and improves efficiency. Primarily, the cloud model has been divided into three major service categories:

- **Platform as a Service** (**PaaS**)

Platform as a service refers to the cloud computing platform as a service, which provides the application running environment for users to meet their needs for the development environment. PaaS can integrate various resources, calculate the resource requirements according to the business requirements of customers, invoke the corresponding hardware resources through the interface provided by IasS, monitor the utilization of resources, and provide services to SaaS through the relevant interface. When using the service, users only need to use the API provided by the platform to easily develop and deploy applications without the need for cloud infrastructure management and control.

- **Software as a Service** (**SaaS**)

SaaS is a mode in which application software is uniformly installed on various cloud platforms and users order application software services through the Internet. Users can use the whole software or only some functions of a certain software. The cloud service provider is responsible for the hardware facilities, management and maintenance of the software, so that users can use the software conveniently and quickly at any time and place through the Internet. Under the SaaS service mode, customers only need to pay for the corresponding services, which greatly reduces the operating cost of the enterprise, which is also the most efficient operation mode of network applications.

- **Infrastructure as a Service (IaaS)**

Infrastructure as a Service often provides the infrastructure such as servers, virtual machines, networks, operating system, storage, and much more on a pay-as-you-use basis. IaaS providers offer VM from small to extra-large machines. Infrastructure-as-a-Service, commonly referred to as simply "IaaS," is a form of cloud computing that delivers fundamental compute, network, and storage resources to consumers on-demand, over the internet, and on a pay-as-you-go basis. IaaS enables end users to scale and shrink resources on an as-needed basis, reducing the need for high, up-front capital expenditures or unnecessary "owned" infrastructure, especially in the case of "spiky" workloads.

B. Hash Tree Technology and HDFS Encryption Algorithm

The system gives a binary hash tree (tiger hash tree commonly used two binary hash tree, but also the form). It is often used in some distributed hash tree in distributed storage system or the anti-entropy mechanism (Antientropy), also called to entropy[11-13]. These applications include Amazon Dynamo and Apache Cassandra database, through to the entropy can do each synchronization of different nodes, each node keeps abreast of the latest information is. The characteristics of the hash tree is very clear: the leaf node is stored in the data file, and the nonleaf node is stored in the hash value of its child nodes (called Message Digest) of the non-leaf nodes of the Hash is called the path hash value, Hash value of leaf nodes is the real data.

4. PROPOSED SYSTEM

- Through block design-based key agreement protocol that supports multiple participants, which could flexibly extend the quantity of participants in associate extremely cloud setting in step with the structure of the block vogue.
- Supported the projected cluster info sharing model, we've got an inclination to gift general formulas for generating the common conference key K for multiple participants. Note that by creating the foremost of the (v; k + 1; 1)-block vogue, the procedure quality of the projected protocol linearly can increase with the quantity of participants and thus the communication quality is greatly reduced. In addition, the fault tolerance property of our protocol permits the cluster info sharing in cloud computing to set about to all totally different key attacks.
- A key agreement protocol is used to return up with a customary conference key for multiple participants to create positive the security of their later communications, and this protocol is applied in cloud computing to support secure and economical info sharing.

A. Architecture including HDFS Client

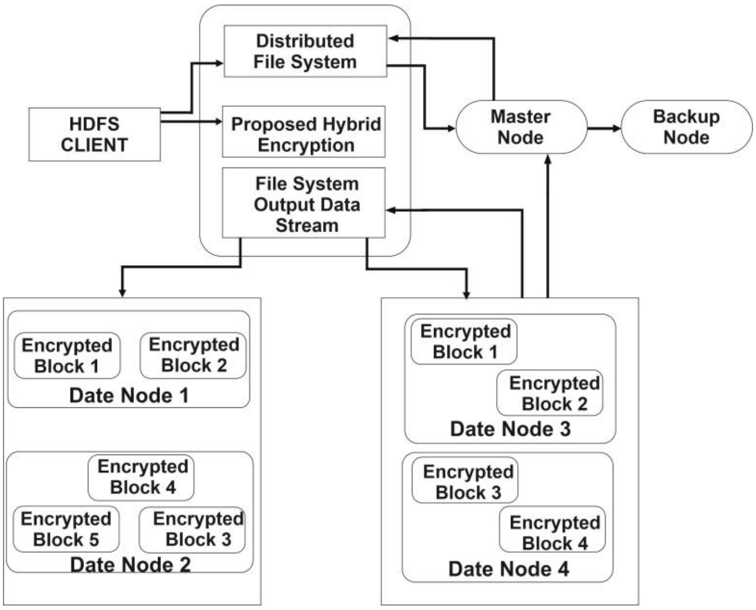
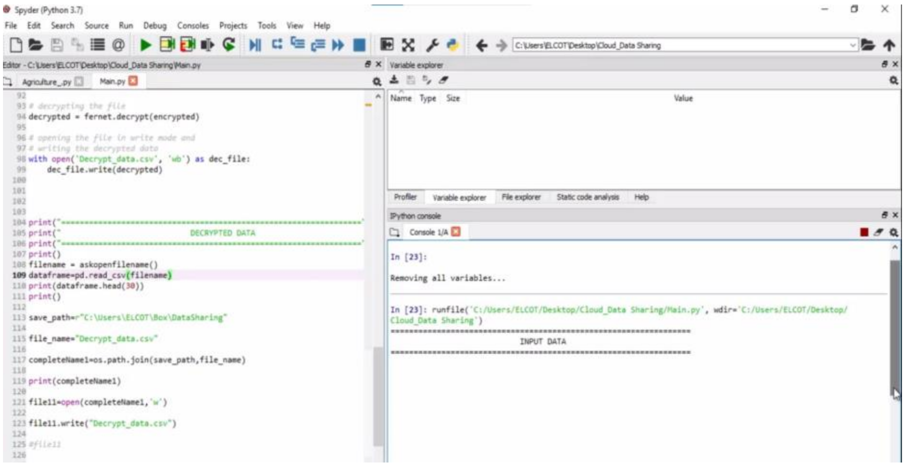


Fig. 1 Architecture of HDFS with algorithm process

HDFS is an open source, distributed and extensible file system. It is one of the core subprojects of Hadoop project and is written and implemented in pure java language. Because of its lack of POSIX compatibility and its inability to mount, HDFS provides shell commands and Java API interfaces similar to other file systems. The Hadoop cluster nominally has a single data node and a data node cluster, although due to its criticality, redundancy options are available for name node. Each data node can provide data blocks over the Internet and using HDFS specific block protocols. The HDFS file system uses TCP / IP sockets for communication clients use remote procedure calls (RPCs) to communicate with each other. Figure 4.1 shows the overall architecture of HDFS. It can be seen that HDFS is an extensible distributed file system with master-slave structure. Many computers or servers that make up HDFS are called clusters. In the HDFS cluster, there is a name node node, which is the main server responsible for maintaining the file system namespace and regulating and controlling the client's access to the file system. In addition, there are many data node nodes, which are usually each node in the HDFS cluster, responsible for managing the data storage attached to the nodes they run. The architecture provides a namespace without reservation. Users can use the namespace to store their own private or non-confidential information in the file. All the attribute information related to the file is maintained by name node. The data stored in HDFS is not the original complete single file, but needs to take some measures to cut these large data into one block, or many blocks; then the small block with certain original information is stored in the specified data node or multiple slave nodes. The main function of name node is to execute a series of namespace related instructions, such as file open, file close and file rename. In addition, it can determine the mapping relationship between each small data block and the corresponding data node. The main function of the corresponding large number of data nodes is to receive the requests from the client to read and write data. At the same time, it can also perform the operations related to the data block according to the instructions given by the name node.

5. IMPLEMENTATION AND RESULTS

A Screenshot

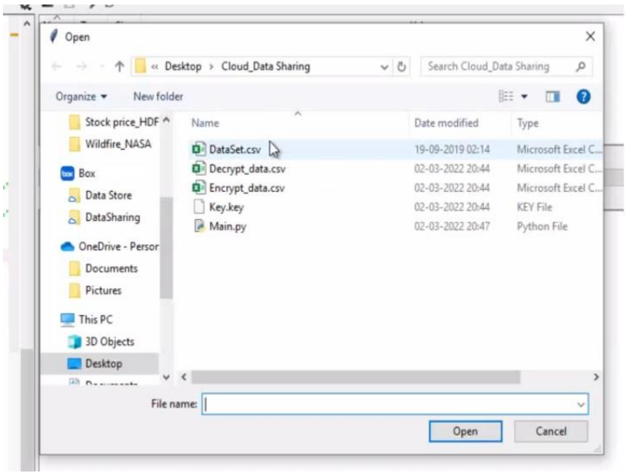


Spyder (Python 3.7)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\ELCOT\Desktop\Cloud_Data Sharing
Editor - C:\Users\ELCOT\Desktop\Cloud_Data Sharing\Main.py
Agriculture_.py
Main.py
Variable explorer
Name Type Size Value
Profiler Variable explorer File explorer Static code analysis Help
IPython console
Console 1/A

```
# decrypting the file
decrypted = fernet.decrypt(encrypted)

# opening the file in write mode and
# writing the decrypted data
with open('Decrypt_data.csv', 'wb') as dec_file:
    dec_file.write(decrypted)




print("==================================================================")
print("                          DECRYPTED DATA")
print("==================================================================")
print()
filename = askopenfilename()
dataframe=pd.read_csv(filename)
print(dataframe.head(30))
print()

save_path=r"C:\Users\ELCOT\Box\DataSharing"

file_name="Decrypt_data.csv"

completeName1=os.path.join(save_path,file_name)

print(completeName1)

file11=open(completeName1,'w')

file11.write("Decrypt_data.csv")

#file11
```

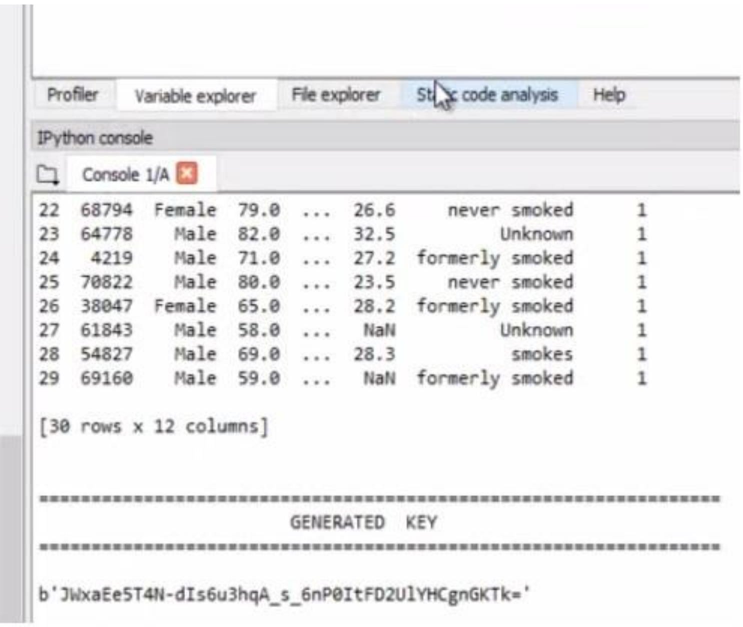
```
In [23]:

Removing all variables...

In [23]: runfile('C:/Users/ELCOT/Desktop/Cloud_Data Sharing/Main.py', wdir='C:/Users/ELCOT/Desktop/
Cloud_Data Sharing')
==================================================================
                          INPUT DATA
==================================================================
```

Fig. 2 Editor Window



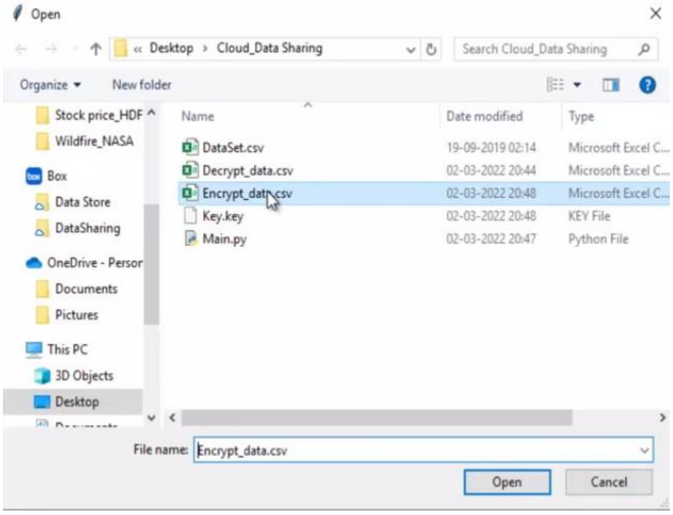
```
Name              Date modified       Type
DataSet.csv       19-09-2019 02:14    Microsoft Excel C...
Decrypt_data.csv  02-03-2022 20:44    Microsoft Excel C...
Encrypt_data.csv  02-03-2022 20:44    Microsoft Excel C...
Key.key           02-03-2022 20:44    KEY File
Main.py           02-03-2022 20:47    Python File
```

Fig. 3 Input selections



```
22  68794  Female  79.0  ...  26.6     never smoked     1
23  64778    Male  82.0  ...  32.5          Unknown     1
24   4219    Male  71.0  ...  27.2  formerly smoked     1
25  70822    Male  80.0  ...  23.5     never smoked     1
26  38047  Female  65.0  ...  28.2  formerly smoked     1
27  61843    Male  58.0  ...   NaN          Unknown     1
28  54827    Male  69.0  ...  28.3           smokes     1
29  69160    Male  59.0  ...   NaN  formerly smoked     1

[30 rows x 12 columns]


==============================================================
                      GENERATED  KEY
==============================================================

b'JWxaEe5T4N-dIs6u3hqA_s_6nP0ItFD2UlYHCgnGKTk='
```

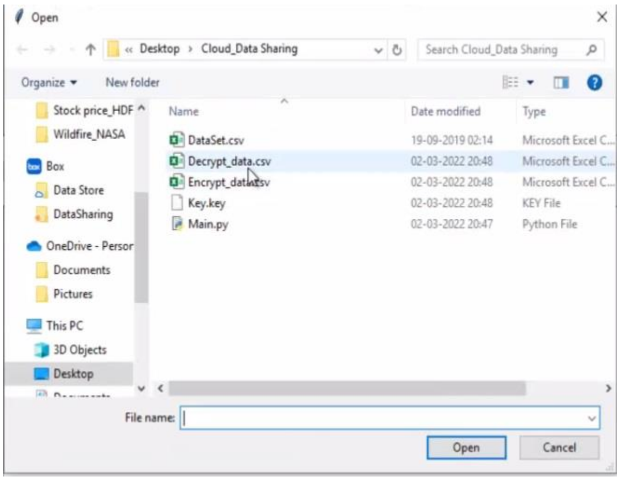
Fig. 4 Data Selection and generation of key



Name	Date modified	Type
DataSet.csv	19-09-2019 02:14	Microsoft Excel C...
Decrypt_data.csv	02-03-2022 20:44	Microsoft Excel C...
Encrypt_dat.csv	02-03-2022 20:48	Microsoft Excel C...
Key.key	02-03-2022 20:48	KEY File
Main.py	02-03-2022 20:47	Python File

File name: Encrypt_data.csv

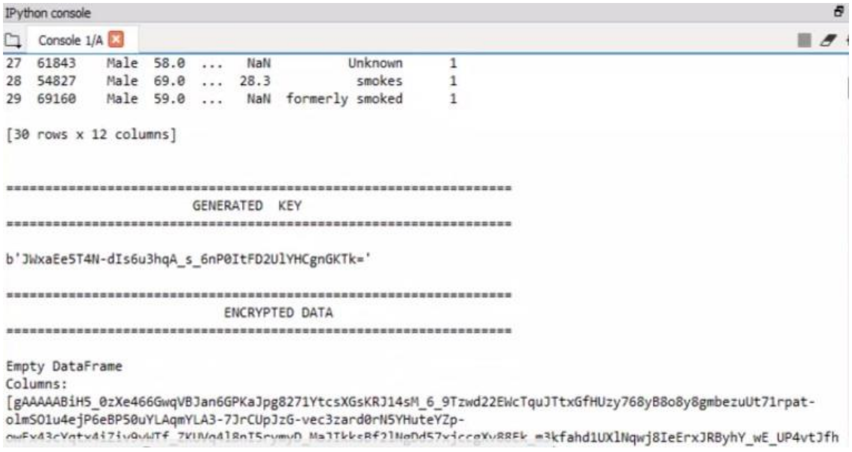
Fig.5 Encryption Data selection



Name	Date modified	Type
DataSet.csv	19-09-2019 02:14	Microsoft Excel C...
Decrypt_data.csv	02-03-2022 20:48	Microsoft Excel C...
Encrypt_data.csv	02-03-2022 20:48	Microsoft Excel C...
Key.key	02-03-2022 20:48	KEY File
Main.py	02-03-2022 20:47	Python File

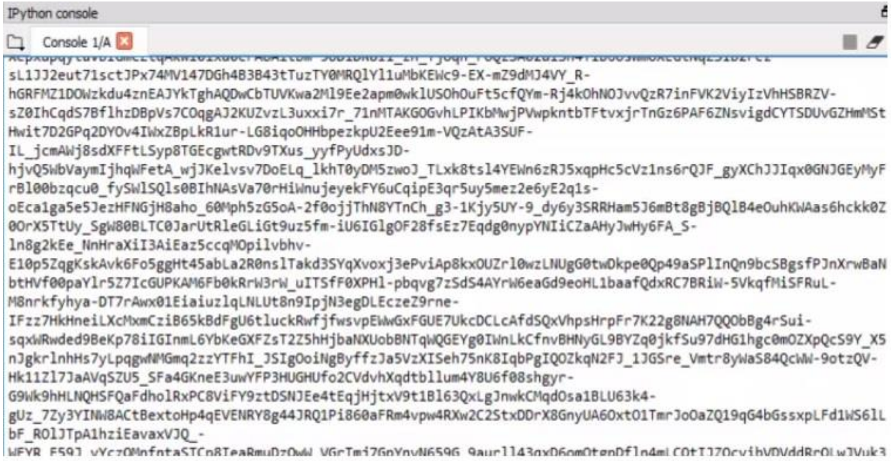
File name:

Fig 6 Decryption data selection



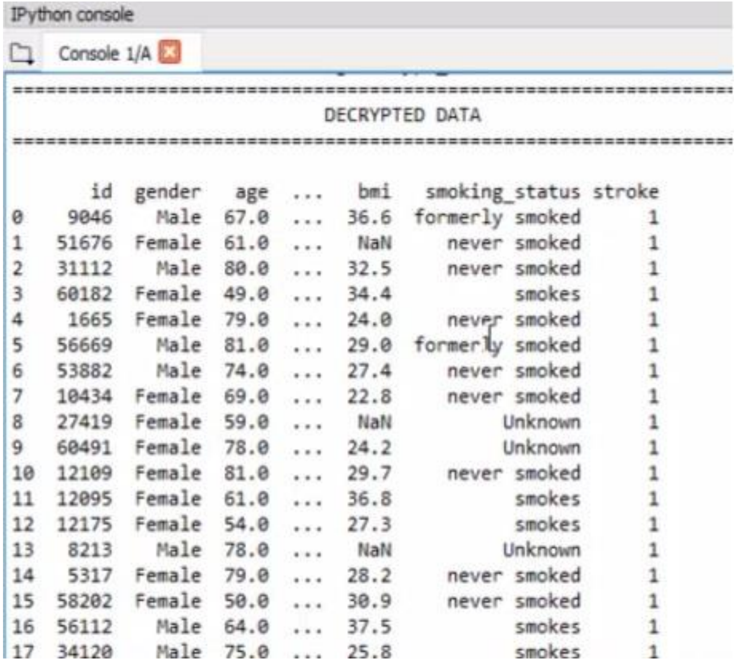
```
27  61843    Male  58.0  ...    NaN          Unknown      1
28  54827    Male  69.0  ...   28.3           smokes      1
29  69160    Male  59.0  ...    NaN  formerly smoked      1

[30 rows x 12 columns]


=====================================================================
                        GENERATED  KEY
=====================================================================

b'JWxaEe5T4N-dIs6u3hqA_s_6nP0ItFD2UlYHCgnGKTk='

=====================================================================
                            ENCRYPTED DATA
=====================================================================

Empty DataFrame
Columns:
[gAAAAABiH5_0zXe466GwqVBJan6GPKaJpg8271YtcsXGsKRJ14sM_6_9Tzwd22EWcTquJTtxGfHUzy768yB8o8y8gmbezuUt71rpat-
olmSO1u4ejP6eBP50uYLAqmYLA3-7JrCUpJzG-vec3zard0rN5YHuteYZp-
owFx43cYqtx4iZiv9vWTf_ZKUVq4l8nI5rymvD_MaJIkksBf2lNgDd57xjccgXv88Fk_m3kfahd1UXlNqwj8IeErxJRByhY_wE_UP4vtJfh
```

Fig 7 Encrypted data generation



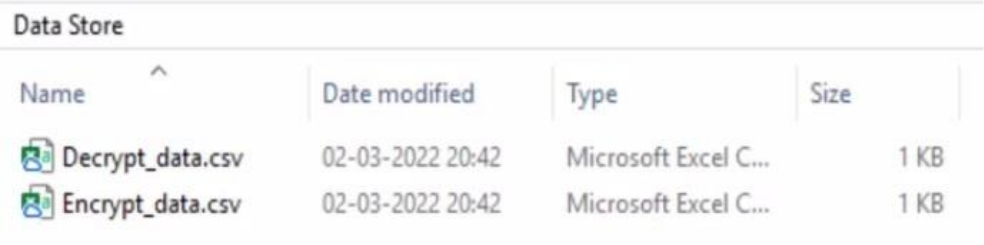
```
sL1JJ2eut71sctJPx74MV147DGh4B3B43tTuzTY0MRQlYl1uMbKEWc9-EX-mZ9dMJ4VY_R-
hGRFMZ1DOWzkdu4znEAJYkTghAQDwCbTUVKwa2Ml9Ee2apm0wklUSOhOuFt5cfQYm-Rj4kOhNOJvvQzR7inFVK2ViyIzVhHSBRZV-
sZ0IhCqdS7BflhzDBpVs7COqgAJ2KUZvzL3uxxi7r_71nMTAKGOGvhLPIKbMwjPVwpkntbTFtvxjrTnGz6PAF6ZNsvigdCYTSDUvGZHmMSt
Hwit7D2GPq2DYOv4IWxZBpLkR1ur-LG8iqoOHHbpezkpU2Eee91m-VQzAtA3SUF-
IL_jcmAWj8sdXFFtLSyp8TGEcgwtRDv9TXus_yyfPyUdxsJD-
hjvQ5WbVaymIjhqWFetA_wjJKelvsv7DoELq_lkhT0yDM5zwoJ_TLxk8tsl4YEWn6zRJ5xqpHc5cVz1ns6rQJF_gyXChJJIqx0GNJGEyMyF
rBl00bzqcu0_fySWlSQls0BIhNAsVa70rHiWnujeyekFY6uCqipE3qr5uy5mez2e6yE2q1s-
oEca1ga5e5JezHFNGjH8aho_60Mph5zG5oA-2f0ojjThN8YTnCh_g3-1Kjy5UY-9_dy6y3SRRHam5J6mBt8gBjBQlB4eOuhKWAas6hckk0Z
0OrX5TtUy_SgW80BLTC0JarUtRleGLiGt9uz5fm-iU6IGlgOF28fsEz7Eqdg0nypYNIiCZaAHyJwHy6FA_S-
ln8g2kEe_NnHraXiI3AiEaz5ccqMOpilvbhv-
E10p5ZqgKskAvk6Fo5ggHt45abLa2R0nslTakd3SYqXvoxj3ePviAp8kxOUZrl0wzLNUgG0twOkpe0Qp49aSPlInQn9bcSBgsfPJnXrwBaN
btHVf00paYlr5Z7IcGUPKAM6Fb0kRrW3rW_uITSfF0XPHl-pbqvg7zSdS4AYrW6eaGd9eoHL1baafQdxRC7BRiW-5VkqfMiSFRuL-
M8nrkfyhya-DT7rAwx01EiaiuzlqLNLUt8n9IpjN3egDLEczeZ9rne-
IFzz7HkHneiLXcMxmCziB65kBdFgU6tluckRwfjfwsvpEWwGxFGUE7UkcDCLcAfdSQxVhpsHrpFr7K22g8NAH7QQObBg4rSui-
sqxWRwded9BeKp78iIGInmL6YbKeGXFZsT2Z5hHjbaNXUobBNTqWQGEYg0IWnLkCfnvBHNyGL9BYZq0jkfSu97dHG1hgc0mOZXpQcS9Y_X5
nJgkrlnhHs7yLpqgwNMGmq2zzYTFhI_JSIgOoiNgByffzJa5VzXISeh75nK8IqbPgIQOZkqN2FJ_1JGSre_Vmtr8yWaS84QcWW-9otzQV-
Hk11Zl7JaAVqSZU5_SFa4GKneE3uwYFP3HUGHUfo2CVdvhXqdtbllum4Y8U6f08shgyr-
G9Wk9hHLNQHSFQaFdholRxPC8ViFY9ztDSNJEe4tEqjHjtxV9t1Bl63QxLgJnwkCMqdOsa1BLU63k4-
gUz_7Zy3YINW8ACtBextoHp4qEVENRY8g44JRQ1Pi860aFRm4vpw4RXw2C2StxDDrX8GnyUA6OxtO1TmrJoOaZQ19qG4bGssxpLFd1WS6lL
bF_ROlJTpA1hziEavaxVJQ_-
WFYR_F59J_vYczOMnfntaSICn8TeaRmuDzOwW_VGrTmi7GnYnvN659G_9aur1l43qxD6omOtgnDfln4mLCOtIJZOcyibVDVddRrOLwJVuk3
```

Fig.8 Encrypted data generation continuation



```
=====================================================================
                        DECRYPTED DATA
=====================================================================

       id  gender   age  ...   bmi   smoking_status  stroke
0    9046    Male  67.0  ...  36.6  formerly smoked       1
1   51676  Female  61.0  ...   NaN     never smoked       1
2   31112    Male  80.0  ...  32.5     never smoked       1
3   60182  Female  49.0  ...  34.4           smokes       1
4    1665  Female  79.0  ...  24.0     never smoked       1
5   56669    Male  81.0  ...  29.0  formerly smoked       1
6   53882    Male  74.0  ...  27.4     never smoked       1
7   10434  Female  69.0  ...  22.8     never smoked       1
8   27419  Female  59.0  ...   NaN          Unknown       1
9   60491  Female  78.0  ...  24.2          Unknown       1
10  12109  Female  81.0  ...  29.7     never smoked       1
11  12095  Female  61.0  ...  36.8           smokes       1
12  12175  Female  54.0  ...  27.3           smokes       1
13   8213    Male  78.0  ...   NaN          Unknown       1
14   5317  Female  79.0  ...  28.2     never smoked       1
15  58202  Female  50.0  ...  30.9     never smoked       1
16  56112    Male  64.0  ...  37.5           smokes       1
17  34120    Male  75.0  ...  25.8           smokes       1
```

Fig.9 Decrypted data generation from encrypted data and key



Data Store

Name	Date modified	Type	Size
Decrypt_data.csv	02-03-2022 20:42	Microsoft Excel C...	1 KB
Encrypt_data.csv	02-03-2022 20:42	Microsoft Excel C...	1 KB

Fig. 10 Encrypted and decrypted file storage

6. CONCLUSION AND FUTURE SCOPE

A. Conclusion

In this way the data sharing in cloud computing enables multiple participants to freely share the group data, which improves the efficiency of work in cooperative environments and has widespread potential applications. It also tries to ensure the security of data sharing within a group and how to efficiently share the outsourced data in a group manner are formidable challenges. Note that key agreement protocols have played a very important role in secure and efficient group data sharing in cloud computing. The main advantage of the symmetric balanced incomplete block design (SBIBD), by presenting a novel block design-based key agreement protocol that supports multiple participants, which can flexibly extend the number of participants in a cloud environment according to the structure of the block design was apparently design and help to provide security more than the previous system. The proposed group data sharing model, we present general formulas for generating the common conference key K for multiple participants. And the protocol linearly increases with the number of participants and the communication complexity is greatly reduced. In addition, the fault tolerance property of our protocol enables the group data sharing in cloud computing to withstand different key attacks.

B. Future work

Aiming at the security problems of the internal personnel of cloud service providers that are easy to be ignored, this paper puts forward the identity authentication and role-based access control strategies based on account and certificate, analyzes and studies the cloud security standards and legal maintenance, and puts forward the cloud security assessment system and relevant legal suggestions and measures to provide a strong guarantee for data security protection.

REFERENCES

1. Jian Shen, Member, IEEE, Tianqi Zhou, Debiao He, Yuexin Zhang, Xingming Sun, Senior Member, IEEE, and Yang Xiang, Senior Member, IEEE, “Block Design-based Key Agreement for Group Data Sharing in Cloud Computing”, 1545-5971 (c) 2017 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information
2. Mehdi Bahrami 1 and Mukesh Singhal 2, “A Dynamic Cloud Computing Platform for eHealth Systems”, 2015 IEEE 17th International Conference on e-Health Networking, Applications and Services (Healthcom)
3. Tessema Mengistu∗, Abdulrahman Alahmadi∗, Abdullah Albuali, Yousef Alsenani, and Dunren Che, “A "No Data Center" Solution to Cloud Computing”, 2017 IEEE 10th International Conference on Cloud Computing

4. Zhao Tianhai," The Key of Application Software Service in Science Cloud Computing", 2021 IEEE 6th International Conference on Cloud Computing and Big Data Analytics
5. Wang Xiaoyu, Gao Zhengming , "Research and Development of Data Security Multidimensional Protection System in Cloud Computing Environment" , 2020 International Conference on Advance in Ambient Computing and Intelligence (ICAACI)
6. Yanhong Shang1 and Jing Zhang," Computer Multimedia Security Protection System Based on the Network Security Active Defense Model", Hindawi, Advances in Multimedia, Volume 2021, Article ID 8792105, 9 pages, https://doi.org/10.1155/2021/8792105
7. L. Zhou, V. Varadharajan, and M. Hitchens, "Cryptographic rolebased access control for secure cloud data storage systems," Information Forensics and Security IEEE Transactions on, vol. 10, no. 11, pp. 2381–2395, 2015.
8. F. Chen, T. Xiang, Y. Yang, and S. S. M. Chow, "Secure cloud storage meets with secure network coding," in IEEE INFOCOM, 2014, pp. 673–681.
9. D. He, S. Zeadally, and L. Wu, "Certificateless public auditing scheme for cloud-assisted wireless body area networks," IEEE Systems Journal, pp. 1–10, 2015.
10. W. Diffie and M. E. Hellman, "New directions in cryptography," IEEE Transactions on Information Theory, vol. 22, no. 6, pp. 644–654, 1976.
11. J. Shen, H. Tan, S. Moh, I. Chung, and J. Wang, "An efficient rfid authentication protocol providing strong privacy and security," Journal of Internet Technology, vol. 17, no. 3, p. 2, 2016.
12. L. Law, A. Menezes, M. Qu, J. Solinas, and S. Vanstone, "An efficient protocol for authenticated key agreement," Designs Codes and Cryptography, vol. 28, no. 2, pp. 119–134, 2010.
13. X. Yi, "Identity-based fault-tolerant conference key agreement," IEEE Transactions on Dependable and Secure Computing, vol. 1, no. 3, pp. 170–178, 2004.
14. R. Barua, R. Dutta, and P. Sarkar, "Extending joux's protocol to multi party key agreement (extended abstract)." Lecture Notes in Computer Science, vol. 2003, pp. 205–217, 2003.
15. J. Shen, S. Moh, and I. Chung, "Identity-based key agreement protocol employing a symmetric balanced incomplete block design," Journal of Communications and Networks, vol. 14, no. 6, pp. 682–691, 2012.
16. B. Dan and M. Franklin, "Identity-based encryption from the weil pairing," Siam Journal on Computing, vol. 32, no. 3, pp. 213–229, 2003.
17. S. Blakewilson, D. Johnson, and A. Menezes, "Key agreement protocols and their security analysis," in IMA International Conference on Cryptography and Coding, 1997, pp. 30–45.
18. I. Chung and Y. Bae, "The design of an efficient load balancing algorithm employing block design," Journal of Applied Mathematics and Computing, vol. 14, no. 1, pp. 343–351, 2004.
19. O. Lee, S. Yoo, B. Park, and I. Chung, "The design and analysis of an efficient load balancing algorithm employing the symmetric balanced incomplete block design." Information Sciences, vol. 176, no. 15, pp. 2148–2160, 2006.
20. R. Curtmola, J. Garay, S. Kamara, and R. Ostrovsky, "Searchable symmetric encryption: Improved definitions and efficient constructions," Journal of Computer Security, vol. 19, no. 5, pp. 79–88, 2011.
21. N. Cao, C. Wang, M. Li, K. Ren, and W. Lou, "Privacy-preserving multi-keyword ranked search over encrypted cloud data," IEEE Transactions on Parallel and Distributed Systems, vol. 25, no. 1, pp. 222–233, 2014.

22. J. Yu, K. Ren, C. Wang, and V. Varadharajan, “Enabling cloud storage auditing with key-exposure resistance,” IEEE Transactions on Information Forensics and Security, vol. 10, no. 6, pp. 1–1, 2015.
23. J. Yu, K. Ren, and C. Wang, “Enabling cloud storage auditing with verifiable outsourcing of key updates,” IEEE Transactions on Information Forensics and Security, vol. 11, no. 6, pp. 1–1, 2016.
24. S. D. C. D. Vimercati, S. Foresti, S. Jajodia, S. Paraboschi, and P. Samarati, “Encryption policies for regulating access to outsourced data,” Acm Transactions on Database Systems, vol. 35, no. 2, pp. 78–78, 2010.
25. H. Guo, Z. Li, Y. Mu, and X. Zhang, “Cryptanalysis of simple three-party key exchange protocol,” Computers and Security, vol. 27, no. 1-2, pp. 16–21, 2008.
26. Z. Tan, “An enhanced three-party authentication key exchange protocol for mobile commerce environments,” Journal of Communications, vol. 5, no. 5, pp. 436–443, 2010.
27. Y. M. Tseng, “An efficient two-party identity-based key exchange protocol.” Informatica, vol. 18, no. 1, pp. 125–136, 2007.
28. A. Shamir, “Identity-based cryptosystems and signature schemes,” Lecture Notes in Computer Science, vol. 21, no. 2, pp. 47–53, 1985.
29. E. Bresson, O. Chevassut, D. Pointcheval, and J. J. Quisquater, “Provably authenticated group diffie-hellman key exchange,” Acm Transactions on Information and System Security, vol. 10, no. 3, pp. 89–92, 2001.