JNNCE Journal of Engineering & Management, Volume 6, No.1, Jan – June 2022 ISSN 2582-0079
This paper is presented at National Workshop on Cryptology NWC 2021, held at JNN College of Engineering, Shimoga during 18-20 Nov 2021

Available online @ https://jjem.jnnce.ac.in
https:www.doi.org/10.37314/JJEM.2022.060118
Indexed in International Scientific Indexing (ISI)
Impact factor: 1.395 for 2021-22
Published on: 31July 2022

# Context Sensitive Privacy Algorithm Method for Users in Cloud

**H. Manoj T. Gadiyar[1]*, Thyagaraju G S[2], R H Goudar[3]**

[1*,2]SDM Institute of Technology, Ujire
[3]Visvesvaraya Technological University,

hmanojtgadiyar@gmail.com, profthyagu@gmail.com, rhgoudar.vtu@gmail.com

***Abstract***

*Cloud computing is one of the key computing platform and technology for sharing resources that may include infrastructure, software, applications, and business processes. Cloud computing incorporate within it data loss prevention, encryption, and authentication, as technologies aimed to support cloud environment. The main intention behind cloud computing is the work done on the client side that can be moved to some unseen cluster of resources over the internet. Context awareness is the process in which the system or system components gather information from its surroundings accordingly. It is responsible for collecting the data automatically and responds to the situation arising dynamically.*

*The focus of this paper is on developing a Context Sensitive Privacy Provision Algorithm such that the encryption and decryption of the data can be done only at the user end but not at the server end so as to preserve context privacy of an individual.*

**Keywords:** *Context, Privacy, Cloud, Encryption, Decryption, internet.*

## 1. Introduction

Cloud Service Provider maintains database and applications for the users on a remote server and provides independence of accessing them from any place through a network. There are three major cloud service categories are software-as-a-service (SaaS), platform-as-aservice (PaaS) and infrastructure-as-a-service (IaaS).

Context-awareness is an emerging area in the current technical world because it adapts mechanisms and interfaces of applications based on consumer preferences as well as environmental conditions.

Applications of IoT can be seen in the fields such as but not limited to agriculture, Industry, Medical and Healthcare, Smart Home Appliances, etc.

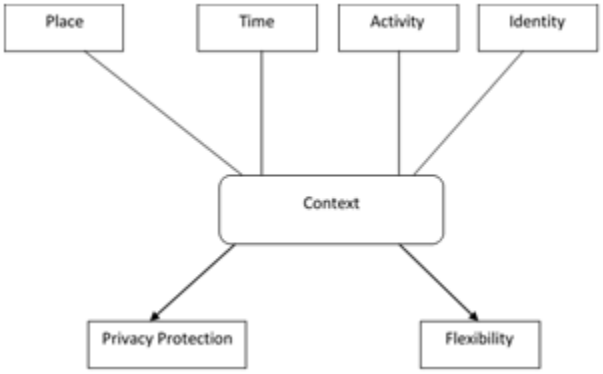



**Figure 1: Theoritical Model**

Lack of Privacy results in IoT devices being more vulnerable to the intruder and compromises the data integrity when it is transmitted over the network.

Privacy issues may affect sectors such as: Smart Cities (Eg: Electricity Management, Water Management, Traffic Control Management) Smart Transport (Eg: Car, Bus, High Speed Train) Smart Health (Eg: Remote Health Monitoring System, Bio Sensors) Smart

Industry (Eg: Sensors, Actuators, Lightning Control, Machine Control, Robotics)Smart Building (Eg: Smart Home, Smart Appliances, Actuators, Smart Meters, Lockers etc)

In context models the behavior of the application is influenced by contexts that are similar to location or user. Context aware systems incorporate components such as context sensor, context storage, context reasoner, context consumer.

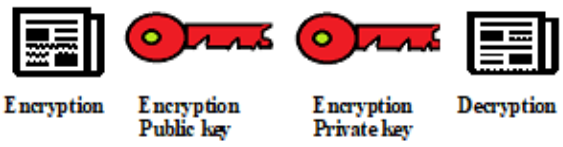
Let us take smart home as a scenario. There are many Internets of things when it comes to smart home which are well built and connected by one or the other smart home builders. Now, all the devices are in IoT will be connected to their server through network. Here, the privacy breach can take place either in the way information is shared with the server or though the network connected hand held devices. Cloud will include data loss prevention, encryption, Cloud has these capabilities. If a cyber-criminal can identify the provider whose vulnerabilities are the easier than a highly visible target.

If cloud providers do not supply sufficient security measures then these clouds can be high-priority targets for cyber criminals. By inheritant nature of the architecture, cloud offers a vulnerable opportunity for simultaneous attacks to number of websites, and without proper security, thousands of websites privacy and security could be compromised through even one malicious activity. Number of issues are included in cloud computing like multi tenancy, data loss and leakage, easy accessibility of cloud, identity management, unsafe API's, service level agreement inconsistencies, patch management, internal threats etc. Enforcing all security measures isn't that easy which can meet the security needs of all the cloud consumers, because several users may have different security demands based upon their objective of using the cloud services.
Cloud computing may suffer from a number vulnerabilities enabling attackers to either obtain computing services from cloud for free or snatch it from other cloud users. We know that in the digitized world privacy is a major concern and cloud computing is no exception to these issues. According to the major cloud user's opinion cloud is much like trusting the telephone company, post office communication medium where people frequently place confidential information into the hands of common carriers and other commercial enterprises.

Normally, users would not use the 'telephone' without taking security precautions beyond trusting the common carrier. As far as storing and accessing the contents in the cloud same thing applies as to never send anything but encrypted data to the cloud storage which provides maximum security by leveraging the capabilities of cryptography. Guidelines can be suggested to increase the security as well as the privacy of the data owner by transferring the process of encryption and decryption from the cloud to self. This is what we have tried to implement through our paper. For maximizing the security of data encryption of the data can be done using a secured co-processor.

As we discuss further on Symmetric and Asymmetric Encryption techniques we know that asymmetric encryption techniques are the more secure than symmetric-encryption techniques, while symmetric encryption techniques are faster than former. Based on the task-computation required, they are either deployed individually or with each other.

## 2. Existing Methods




### Symmetric Encryption:

Encryption prevents the senders' data from getting into the hands of intruder or any other non recipient. We can use single key both for

both sender and receiver. The same key can be used to encrypt the message and decrypt the same.

**Encryption:**

**How are you?(message)** **Qh4das18(Encrypted)**

**Decryption:**

**Qh4das18(Encrypted)** **How are you?(message)**

Symmetric encryption works tremendously fast. Here, the size of the cipher text is small.
It provides massive amount of data and provides more confidentiality.
Disadvantages of symmetric encryption are sharing a key between sender and receiver. A new key required for each pair of transmission.
Example: AES, 3DES, DES

**Asymmetric Encryption:**

Asymmetric encryption encrypts and decrypts the message using different keys. Asymmetric encryption is also called as public key encryption. It uses separate key for encryption and decryption. It uses two different keys private key and public key.

Asymmetric encryption is also called public key encryption which allows generating multiple public keys for agents to encrypt their information and private key decrypt the message from receiver side. This method to generate cryptographic algorithm using that algorithm generate a key pair. The key pair varies from one algorithm to another algorithm. The size of the cipher text is larger or same. Encryption process is too slow when compared to the symmetric encryption. In this method only small amount of data can be transferred. This encryption method can be used for securely exchanging data, because private key must always be hidden.

**AES Algorithm**

One of the popular encryption algorithm is AES encryption algorithm. It is six time fasterthan the triple DES.

*Operations of AES:*

Based on subsitution and permutation network the AES performs all its information on bytes rather than bits. It is uses 10 rounds to 128 key bits, 12 rounds to 192 key bits and 14 rounds to 256 key bits.

**The Process of Encryption:**
Each round compromises of 4 sub processes. The first process represented as below:

**Sub bytes** (Byte substitution): The16 bytes are substituted by observing a fixed table(s-box) certain design. The outcome is in a matrix of four rows and four columns.

**Shift-Rows:** Each of the four rows shifted to the left. First row is not shifted. Second row shifted one position to the left. Third row is shifted two positions to the left. The result is a new matrix consisting of the same 16 bytes but shifted to with respect to each other.

**Mix-Columns:** Here, each column of four bytes are transformed using a special mathematical function. It considers 4 bytes of one column and outputs 4 completely new bytes, which replace the original column. The result is another new matrix consisting of sixteen (16) new bytes.

**Add-Round-Key:** Here, the cipher key used for encryption 128 bits and are XORed to the 128 bits of round key. There will be 10 rounds and 11 keys are needed because one extra key is added to the initial state array before the rounds start.

**The Process of Decryption:**

The decryption process is similar to AES encryption process but in a reverse order. Each round consists of the four processes managed in reverse order.

- Adding Round Keys
- Mixing Column
- Shifting Rows
- Sub Bytes

## 3. Modified AES Algorithm Code Snippet

```
def prepareDataview(request):
totalFocusedTime=bytes([random.randint(0,10
)])
starTime=bytes([random.randint(0,10)])
endTime=bytes([random.randint(0,10)])
totalOnTime=bytes([random.randint(0,10)])

key=os.urandom(16)
iv=os.urandom(16)
userdata.objects.all().delete()
encryptedtotalfocus = aes.AES(key).encrypt_ct
r(b"+ totalFocusedTime, iv)
encryptedstarttime = aes.AES(key).encrypt_ctr
(b"+starTime, iv)
encryptedendtime = aes.AES(key).encrypt_ctr(
b"+endTime, iv)
encryptedtotalontime = aes.AES(key).encrypt_
ctr(b"+totalOnTime, iv)

devicedata=userdata(totalFocusedTime=encryp
tedtotalfocus,
starTime=encryptedstarttime,
endTime=encryptedendtime,
totalOnTime=encryptedtotalontime,key=key,iv
=iv)
devicedata.save()

return render(request,"userdata.html",{'totalfoc
uses':encryptedtotalfocus,'start':encryptedstartti
me,
'end':encryptedendtime,'totalon':encryptedtotal
ontime})
```

## 4. Realization of using Python

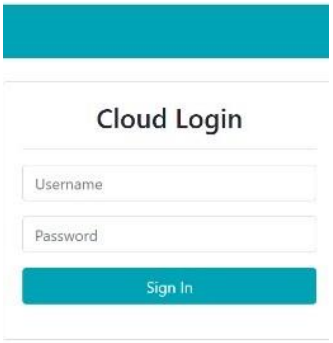



Figure 2: Cloud Login

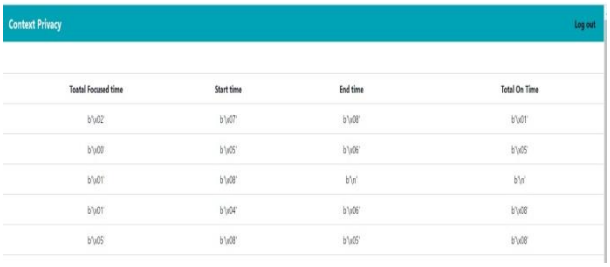



Figure 3 : Private Data Encrypted by the User

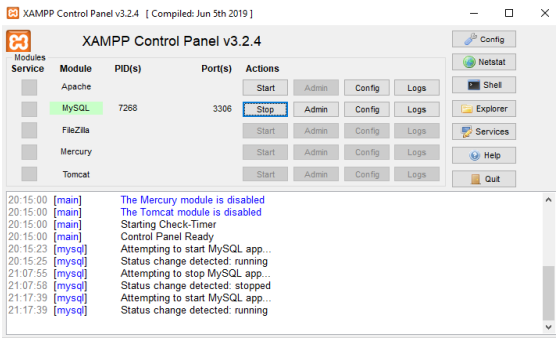



Figure 4: XAMPP Control Panel

## 5. Conclusion

End-to-end encryption is considered to be safer when data gets transmitted over the network because it reduces the number of people who might be trying to interfere or break the encryption.

This paper proposes algorithmic method where the information can be encrypted as well as decrypted by the user and the encryption

cannot be decrypted at the server side or any other intruder. The code snippet of the implementation is shown in the section 3 of this paper. Further, we are working on developing a hybrid algorithm with the combination of two privacy algorithms for providing context aware privacy for consumers in cloud.

## References

1. N. Su, Y. Zhang and M. Li, Research on Data Encryption Standard Based on AES Algorithm in Internet of Things Environment, 2019 IEEE 3rd Information Technology, Networking, Electronic and Automation Control Conference (ITNEC), 2019, pp. 2071-2075, doi: 10.1109/ITNEC.2019.8729488.

2. Abhishek Kumar Sinha , Jayaraj N, 2015, Performance Analysis of AES Cryptographic Algorithm, International Journal of Engineering Research & Technology (IJERT) NCRTS – 2015, Volume 3, Issue 27,2015.

3. Hayashi Eiji, Sauvik Das, ShahriyarAmini, Jason Hong, Ian Oakley, ―CASA: Context-Aware Scalable Authentication‖, Symposium on Usable Privacy and Security (SOUPS) 2013, Newcastle, UK.

4. A.S.M. Kayes, Jun Han and Alan Colman, ―OntCAAC: An Ontology-Based Approach to Context-Aware Access Control for Software Services, Computer Science Theory, Methods and Tools, The Computer Journal, Vol. 58 No. 11, 2015.

5. Schilit, B., Adams, N., and Want, R. Context-aware computing applications. in First workshop of Mobile Computing Systems and Applications, IEEE (1994), 85–90.

6. Mizouni, R., Matar, M.A., Al Mahmoud, Z., Alzahmi, S., and Salah, A. A framework for contextaware self-adaptive mobile applications SPL. Expert Systems with applications 41, 16 (2014), 7549–7564.

7. P. K. Das, S. Narayanan, N. K. Sharma, A. Joshi, K. Joshi and T. Finin, Context-Sensitive Policy Based Security in Internet of Things, 2016 IEEE International Conference on Smart Computing (SMARTCOMP), 2016, pp. 1-6,
doi: 10.1109/SMARTCOMP.2016.7501684.

8. B. Shebaro, O. Oluwatimi and E. Bertino, Context-Based Access Control Systems for Mobile Devices, in IEEE Transactions on Dependable and Secure Computing, vol. 12, no. 2, 1 March-April 2015, pp. 150-163.
doi: 10.1109/TDSC.2014.2320731.

9. Supriyo Chakraborty, Kasturi Rangan Raghavan, Matthew P. Johnson, and Mani B. Srivastava. 2013. A framework for context-aware privacy of sensor data on mobile systems. In Proceedings of the 14th Workshop on Mobile Computing Systems and Applications (HotMobile '13). Association for Computing Machinery, New York, NY, USA, Article 11, 1–6.
doi:https://doi.org/10.1145/2444776.2444791

10. PARATE, A., CHIU, M.-C., GANE-SAN, D., AND MARLIN, B. M. Leveraging graphical models to improve accuracy and reduce privacy risks of mobile sensing. In Pro-ceeding of the 11th Annual International Conference on Mobile Systems, Applications, and Services (2013), MobiSys ’13, pp. 83–96.