ISSN(Online): 2320 - 9801
ISSN (Print) : 2320 - 9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 6, June 2016

Privacy Protection Using Noise Injection And Encryption

Ashwini Hanumant Gajare, Prof. Prashant M Mane, Prof. Nitin S Taware

Lecturer, Department of Computer Engineering, ZCOER, Savitribai Phule Pune University, Pune, India

Assistant Professor, Department of CO, ZCOER, Savitribai Phule Pune University, Pune, India

Assistant Professor, Department of CO, SVPMCOE, Savitribai Phule Pune University, Pune, India

ABSTRACT: Cloud computing is an open and virtualized environment where various malicious service providers may exists. Such service providers analyse the customer requests data set and deduce customer's privacy. So there is a need of taking certain technical actions at client side for privacy protection. Noise injection strategy generates noise requests and injects into real requests's so that malicious service providers will not able be to distinguish between them. CTIG is used for dynamic time interval generation and occurrence probability calculation. TPF is used for forecasting occurrence probabilities. According to the noise injection intensity we dynamically insert that number of noise requests into real real requests. Final requests queue is of combined real and noise requests. This final queue is encrypted using AES and ECC and sent to the service provider. At the service provider side, it decrypts and reads the data set of final service queue but not able to distinguish which requests are real ones and which noises are. So More privacy is provided as compare to existing TPNGS algorithm.

KEYWORDS: Cloud, Obfuscation, Noise, Time Interval, Forecasting, Occurrence Probabilities.

I. INTRODUCTION

Cloud is the platform for information infrastructures and resources as IT services. Because of some malicious service providers exist in cloud, customers have concerns about whether their privacy can be protected when facilitating IT services. Without proper privacy protection, customers may lose the confidence in cloud. So privacy protection is a critical issue in cloud computing. Large number of unknown and malicious service providers may exist in cloud. Such service providers easily get the data of customer's real requests. They observe the pattern or type of request, interest of user in that request and deduce the privacy without authorization. For example such malicious service providers can get the information about our light bill payment then date at which we pays , ATM used for that payment etc. and further they may do fraud. They may also get information or data set of customer's requests from travel agencies and analyses our interest from that data set. So the noise injection strategy generates some noise or dummy requests and injects into the customers real service requests so that malicious service provider will not able to distinguish between them when their occurrence will be same. Customers can control the noise injection function according to the privacy protection requirement and budget. As noise requests costs same as the real requests in pay-as-you-go cloud environment. So our aim is to reduce number of noise requests than random noise generation and provide same level of privacy. CTIG is used for time interval generation and occurrence probability calculation. TPF forecasts the future occurrence probabilities of real requests so that we can calculate noise generation probabilities. After that we calculate noise injection intensity that is number of noise request needs to be injected into real service queue at each time interval. Final queue of real and noise requests is encrypted and sent to the service provider. Existing noise generation strategies do not use encryption while sending final request to the service provider but we have provided more security by doing this. AES and ECC is used for encryption. At the service provider side, decrypts the queue and reads final request queue data set. But service provider would not able to analyse customers' requests as the noise are injected in that queue.

ISSN(Online): 2320 - 9801
ISSN (Print) : 2320 - 9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 6, June 2016

II. RELATED WORK AND LITERATURE SURVEY

In this section discussed about the work done by the researchers for privacy protection on cloud. In [1], existing noise generation strategies have not considered fluctuations of occurrence probabilities of real requests. In this case, the probability fluctuations could not be considered by existing noise generation approaches, and it is a serious risk for the customers privacy. To give a solution to this probability fluctuation risk, they developed a time series pattern based noise generation strategy for privacy protection on cloud. First, they analyzed this privacy risk and presented a cluster based algorithm to generate time intervals dynamically. Then, based on these time intervals, investigated corresponding probability fluctuations and proposed a time-series pattern based forecasting algorithm. Based on the forecasting algorithm, this novel noise generation strategy was presented to address the probability fluctuation privacy risk. This strategy has significantly improved the effectiveness of cloud privacy protection to withstand the probability fluctuation privacy risk.

In[2], Noise obfuscation is an effective approach for privacy protection. It generates the noise requests and injects into real customer service requests so that service providers could not be able to differentiate which requests are real ones if their occurrence probabilities are the same. However, existing noise generation strategies focus on the whole service usage period to achieve about the same final occurrence probabilities of service requests. In fact, such probabilities can fluctuate in a time interval such as three months and may significantly different than other time intervals. In this case, service providers might still be able to deduce the customers privacy from a specific time interval although unlikely from the overall period. That is to say, the existing noise generation strategies could fail to protect customers privacy for local time intervals. To address this problem, they developed a novel time-series pattern based noise generation strategy. They analyzed previous fluctuations in requests occurrences and developed a time-series patterns for predicting future fluctuated probabilities.

In [3], with the same level of disorientation, the number of noise or dummy requests should be kept as few as possible. Therefore in that paper they developed a novel historical probability based noise generation strategy. This strategy has generated noise requests based on their historical occurrence probability so that all requests including noise and real ones having same occurrence probability, and then service providers was not be able to distinguish in between. This strategy had significantly reduced the number of noise requests over the random strategy, by more than 90%.

In[4], proposes a client-side solution, which models the search privacy problem as an information inference problem and injects noise or dummy queries into user queries to minimize privacy breaches. Given the low amount of noise queries required by a privacy protection. Also provided the optimal protection given the number of noise queries. Also verified results with a special case where the number of noise request queries was equal to the number of user real request queries. The noise given bythat approach greatly reduced privacy gaps and outperforms random noise. This work was the first theoretical analyzed on user side noise injection for search privacy protection.

In[5], Obfuscation-based private web search (OBPWS) solutions allowed users to search information in the Internet while concealing their interests. The privacy concept in OB-PWS was the automatic generation of dummy queries that are sent to the search engine along with users real requests. These dummy or noise queries prevent the accurate inference of search profiles and provide query deniability. In this paper proposed an abstract model and an associated analysis framework evaluated the privacy protection offered by OBPWS systems.

In[6], propose the RASP data perturbation method to provide efficient, secure range query & k NN query services for the protected data in the cloud. The RASP data perturbation method combines order preserving encryption, dimensionality expansion and random noise injection, to provide strong resilience to attacks on the data and queries.

ISSN(Online): 2320 - 9801
ISSN (Print) : 2320 - 9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 6, June 2016

III. ABSTRACT MODEL OF NOISE INJECTION

To support the TPNGS noise generation this is basic concept needs to understand.

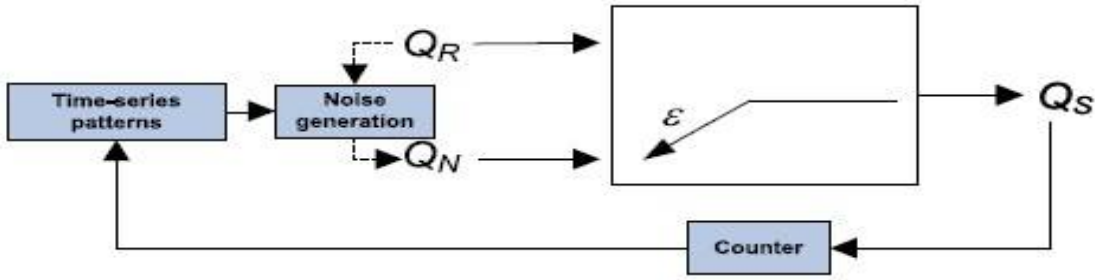


Figure 1. Noise Injection model

QR is the queue of real requests and QN is the queue of noise request needs to be injected into QR. ε is the noise injection intensity. QS is queue of final service requests combined with QR and QS. The function of noise generation is processed only if QR is not empty.

Q: a set of service requests, and Q = {q1,q2,…. ,qi, .. ,qn}. Every service request in QR, QS and QN is from this set. Hence, in the view of service providers, one request in the queue of final service request QS could be from real requests QR or noise requests QN.

IV. SYSTEM ARCHITECTURE

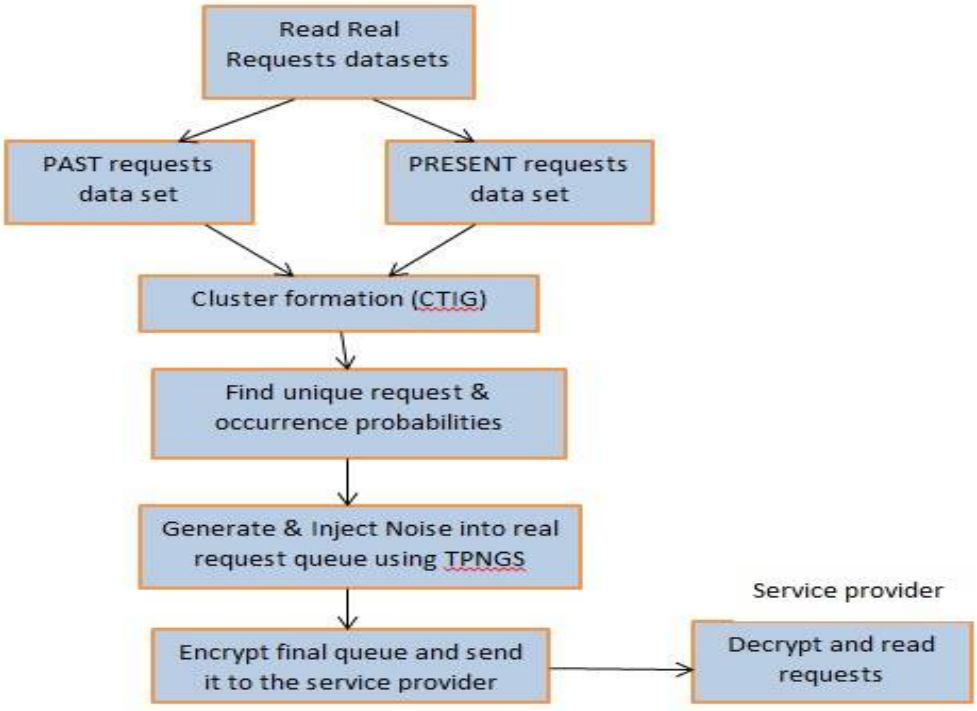


Figure 2. Architecture of system flow

ISSN(Online): 2320 - 9801
ISSN (Print) : 2320 - 9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 6, June 2016

In our system we first read the past one month data set of customers real requests and generates time intervals dynamically using CTIG , calculates occurrence probabilities, min and max occurrence probabilities of all requests in all the time intervals. After that system reads present month data set of real requests and performs same operations on that. TPF forecasts the future occurrence probabilities of requests by observing these two data sets and calculates noise injection intensity. According to that noise injection intensity, that number of noise requests is injected into real service requests and final queue is encrypted using AES as well as ECC and send it to the service provider. Service provider can decrypt the queue but would not able to distinguish between real and noise requests.

V. ALGORITHMS USED

A. Cluster Based Time Interval Generation Algorithm(CTIG)-

This algorithm's aim is to obtain time intervals dynamically to calculate or express occurrence probabilities in fluctuations of requests in the data set. Here we can use equal and unequal time intervals. First we initialize time intervals as time elements. Then we calculate unique requests in each time interval or cluster, calculates how many times that requests is occurred in that particular cluster. According to this min and max occurrence probability of each cluster is calculated. These time intervals with occurrence probabilities are used in TPF algorithm[1].

Title: Cluster based time interval generation.
Input: All past time elements
Output: All past occurrence probabilities at all time elements And Time Intervals.
Algorithm:
Step 1: Initializing time intervals.

```
for(i=1;i<T';i++)
  TI(i)=TE(i);
End
```

Step 2:

```
min_t= Obtain_min_t(for all I, P(QR=qi)(t));
max_t= Obtain_max_t(for all I, P(QR=qi)(t));
if(occurrence probability > distance boundry)
    stop the loop;
End
```

Step: Adjust time intervals to be equal.

B. Time series pattern based forecasting Algorithm(TPF)-

TPF use TSPG segmenting and pattern generation algorithm and PMF pattern matching & forecasting algorithm. TSPG divides entire time period with past occurrence probabilities and gets some time segments by checking validation on that. One pre-set parameter is used as boundary of variance in occurrence. Patterns are validated here.
PMF algorithm is used to match the current probabilities. CP is the current probabilities in queue. One output is MP that is matched pattern and another output is forecasting result FR. FR denotes future occurrence possibility of real request and it is decided by matched pattern. If cannot find out suitable pattern, the mean is used as the forecasting result FR to guide noise injection [1].

Title: Time series pattern based forecasting algorithm.

Input: All past occurrence probabilities P(QR=qi)(t). length of current probability queue is L.

Output: Group of future probabilities P(QR=qi)(t+1).

Algorithm:

Step 1: Execute forecasting process.

```
For(i=1;i<=n;i++)
TI[i]=TSPG(P(QR=qi)(t)),  tε[0,T]);
FR[i][t]=PMF(TI(i), P(QR=qi)(t),  t ε [T-L,T]);
End
```

Step 2: sum forecasted results.

ISSN(Online): 2320 - 9801
ISSN (Print) : 2320 - 9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 6, June 2016

For(i=1;i<=n;i++)

FRS=FRS+FR[i];

End

Step 3: Normalize forecasted results

P(QR=qi)(T+1)= FR[i]+FRS;

End

Combine the all the forecasted results and normalize into one group of "future" probabilities for noise generation. If the forecasting is inaccurate, noise generation will perform poorly with unnecessary cost.

C. TPNGS Algorithm-

In this section we introduce two key issues of noise generation. First are noise generation probabilities and second is noise injection intensity. In the process of noise generation, noise generation probabilities decide which kind of noise requests should be generated. And noise injection intensity decides how many noise requests should be generated [1].

Title: Time series pattern based noise generation strategy

Input: QR is the queue of real service requests

Output: QS is the queue of final service requests

Algorithm:

Step 1: Collect and record all occurrence probabilities in past time.

For all i, PE(QR=qi)(t),tε [1,T]

Step 2: Ulilize CTIG algorithm,

For all i, PE(QR=qi)(t), tε[1,T']= CTIG(for all i, PE(QR=qi)(t), tε[1,T]) ;

Step 3: Utilize TPF algorithm.

For all i,), P(QR=qi)(t)= TPFA (P(QR=qi)(t), t ε [1, T'])

Step 4: compute noise generation probabilities and noise injection intensity [1]:

Step 5: Execute noise injection process.

Inject QN into QR on the probability of ε to get QS final service request queue.

Update past probabilities.

D. Encryption Algorithms-

AES is an iterative, based on 'substitution–permutation,and number of rounds in AES is variable and depends on the length of the key. AES uses 10 rounds for 128-bit keys. AES performs byte substitution, shiftrows, mixcolumns and addRoundKey operations. [7]

ECC Create Faster, Smaller and more efficient keys as compared to other encryption algorithm. ECC is that much efficient that it provide level of security with 164 bit key that other system require a 1,024-bit key to achieve that security level i.e.it offers the maximum security with smaller bit sizes that is why it consumes less power.disadvantage of ECC is that it increases the size of encrypted text[7].

VI. RESULT AND DISCUSSION

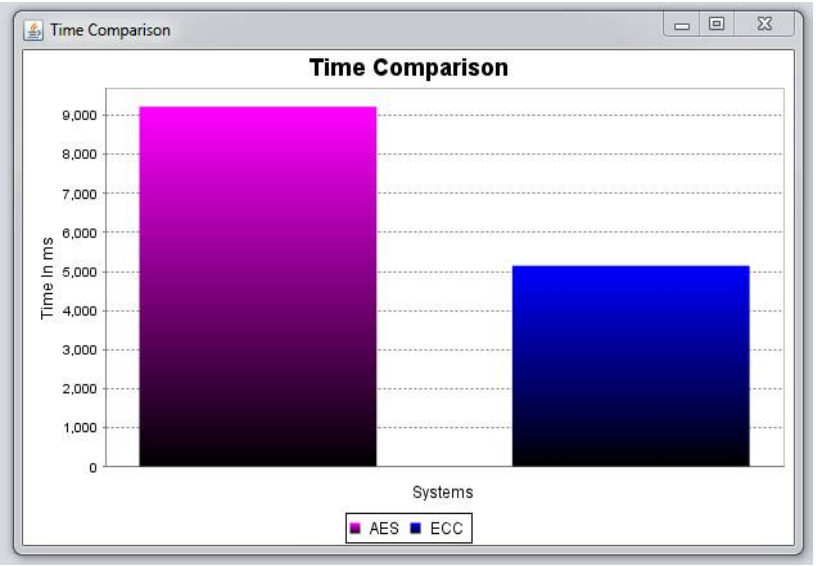
In the system execution we used dataset of this link : http://ita.ee.lbl.gov/html/contrib/NASA-HTTP.html it contain two months' worth of all HTTP requests to the NASA Kennedy Space Center WWW server in Florida.

ISSN(Online): 2320 - 9801
ISSN (Print) : 2320 - 9798

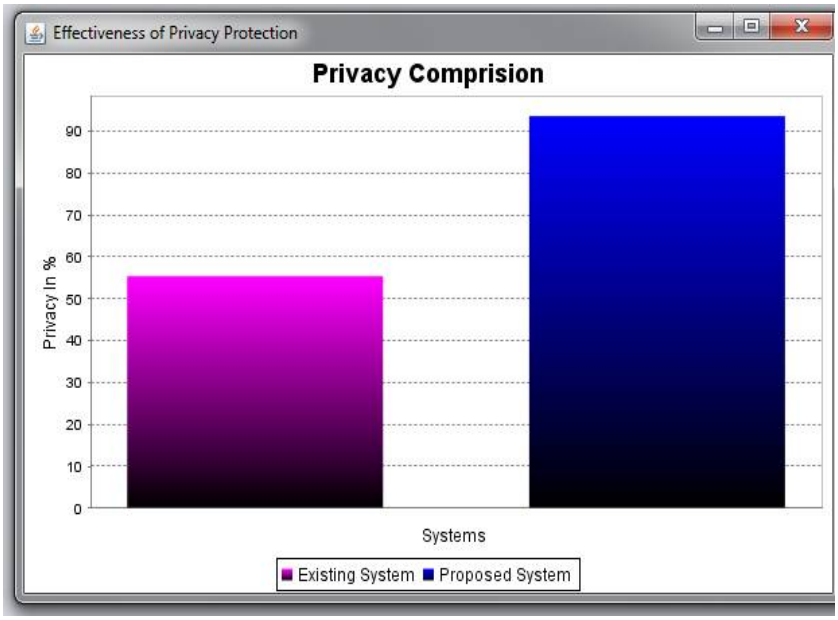
International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 6, June 2016



In the above result of time comparison graph shows that AES algorithms time complexity is more than ECC time complexity.



In the privacy comparison graph our aim is achieved that more privacy is provided as compared to existing noise generation strategies. Number of noise required are less using TPNGS[1], so cost of noise obfuscation is less as compared with random noise generation. And effectiveness of privacy protection is increased at service provider side cannot able to distinguish between real and dummy requests because of same occurrence probabilities.

VII. CONCLUSION

In this paper, introduce the system which reduces the cost of noise obfuscation as the number of noise requests are reduced as compared with random noise generation. Increases the effectiveness of privacy protection using dynamically noise injection.Encryption is used while sending final request queue to the service provider so more security is provided. Service provider not able to distinguish between real and noise requests so our aim of privacy protection is achieved.

ISSN(Online): 2320 - 9801
ISSN (Print) : 2320 - 9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 6, June 2016

ACKNOWLEDGMENT

The authors would like to thank the researchers as well as publishers for making their resources available and our HOD Prof. S.M.Sangave, PG coordinator Prof.P.M.Mane for their useful guidance. Also We are thankful to the authorities of Savitribai Phule University of Pune for their constant guidelines and support. Also we are thankful to staff members of ZCOER for their valuable suggestions. We also thank to the college authorities for providing the required Infrastructure and support. Also, we would like to extend a heartfelt gratitude to friends and family members.

REFERENCES

[1] Geofeng Zhang, Xiao Liu, and Yun Yang, "Time series pattern based effective noise generation for privacy protection on cloud," IEEE. Trans. Oncomputers, vol. 64, No- 5, May 2015.
[2]G. Zhang, Y. Yang, X. Liu, and J. Chen, "A Time-Series Pattern Based Noise Generation Strategy for Privacy Protection in Cloud Computing," Proc. 12th IEEE/ACM Int'l Symp. Cluster, Cloud and Grid Computing (CCGrid '12), pp. 458-465, May 2012.
[3] G. Zhang, Y. Yang, and J. Chen, "A Historical Probability Based Noise Generation Strategy for Privacy Protection in Cloud Computing," J. computer and System Sciences, vol. 78, no. 5, pp. 1374-1381, 2012.
[4] S. Ye, F. Wu, R. Pandey, and H. Chen, "Noise Injection for Search Privacy Protection," Proc. Int'l Conf. Computational Science and Eng. (CSE '09), pp. 1-8, Aug. 2009.
[5] E.Balsa, C. Troncoso, and C. Diaz, "OB-PWS: Obfuscation-Based Private Web Search," Proc. IEEE Symp. Security and Privacy, pp. 491-505, May 2012.
[6] H. Xu, S. Guo, and K. Chen, "Building Confidential and Efficient Query Services in the Cloud with RASP Data Perturbation," IEEE Trans. Knowledge and Data Eng., vol. 26, no. 2, pp. 322- 335, Feb. 2013.
[7] Rajdeep Bhanot, Rahul Hans, " A Review and Comparative Analysis of Various Encryption Algorithms", International Journal of Security and Its Applications, Vol. 9, No. 4 , pp. 289-306, 2015.
[8] X. Liu, Z. Ni, D. Yuan, Y. Jiang, Z. Wu, J. Chen, and Y. Yang, "A Novel Statistical Time-Series Pattern Based Interval Forecasting Strategy for Activity Durations in Workflow Systems," J. Systems and Software, vol. 84, no. 3, pp. 354-376, 2011.
[9] C.C. Aggarwal and P.S. Yu, "A Framework for Clustering Uncertain Data Streams," Proc. IEEE 24th Int'l Conf. Data Eng., pp. 150- 159, Apr. 2008.
[10] C.A. Ardagna, M. Cremonini, S. de Capitani di Vimercati, and P. Samarati, "An Obfuscation-Based Approach for Protecting Location Privacy," IEEE Trans. Dependable and Secure Computing, vol. 8, no. 1, pp. 13-27, Jan./Feb. 2011.

BIOGRAPHY

Ashwini H Gajare received the BE and ME degree in computer engineering from Pune University, India in 2013 and 2016 respectively. Area of interest is network security. She is currently working as assistant professor at zeal polytechnic pune.

Prashant Mane received the BE and ME degree in computer engineering from Pune University, Area of interest is network security and cloud computing. He is currently working as assistant professor at zeal college of engg & research pune.

Nitin S Taware received the BE and M Tech degree in computer engineering from Pune University and JNTU respectively, Area of interest is Database systems and Data mining. He is currently working as assistant professor at SVPM COE baramati,Pune from last 8 years.