A Testing On Secure Key Policy Attribute-Based Encryption Policy For Data Sharing Among Dynamic Groups In The Cloud

Soniya Thomas[1], Mr.J.Santhosh MSc.,MCA.,Mphil.,ME.,(Ph.D)[2]

[1]M.Phil Scholar Department Of Computer Science Sree Narayana Guru College Kg Chavady, Coimbatore 641105 Tamil Nadu

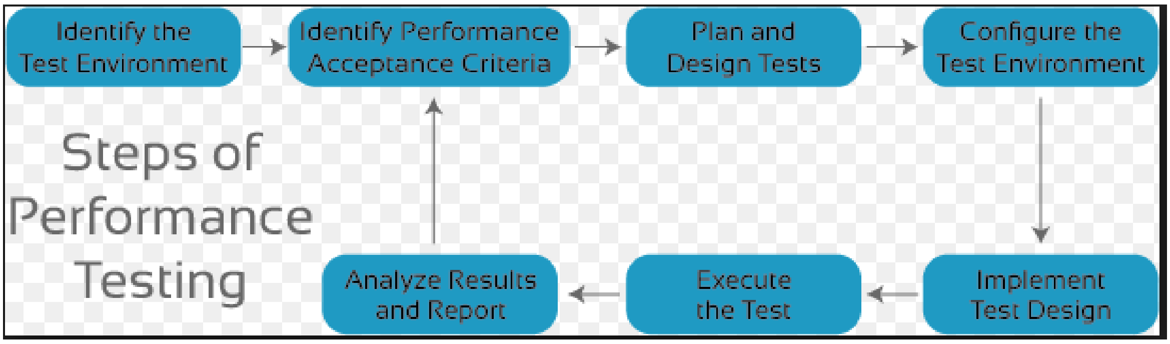
[2]Asst. Professor Department Of Computer Science Sree Narayana Guru College Kg Chavady, Coimbatore 641105 Tamil Nadu

Corresponding Author: Soniya Thomas

ABSTRACT: *Performance testing is based on the assertion that language and culture study are best brought together when the teacher is effective in the affective as well as cognitive and skills domains, teaching strategies and activities are presented that combine specific teaching techniques and ideas with other human and cultural resources in and out of the classroom. Chapters discuss: when and in what language culture should be taught; what "culture" refers to and its connection with language; approaches to understanding how societies evolve different ways to satisfy their needs; seven goals of cultural instruction; performance objectives for each of the seven goals; developing effective learning activities for each of the performance objectives; two techniques to sensitize students to the miscommunication accompanying interaction with those of another culture (empathetic literature and minidramas); three techniques for teaching cultural concepts (culture assimilators, culture capsules, culture clusters); approaches to helping students ask significant questions; measuring shifts in attitude toward the target culture; stresses and challenges of biculturalism; and suggestions for implementing a curriculum fostering intercultural communication.*

Date of Submission: 18-11-2018 Date of acceptance: 04-12-2018

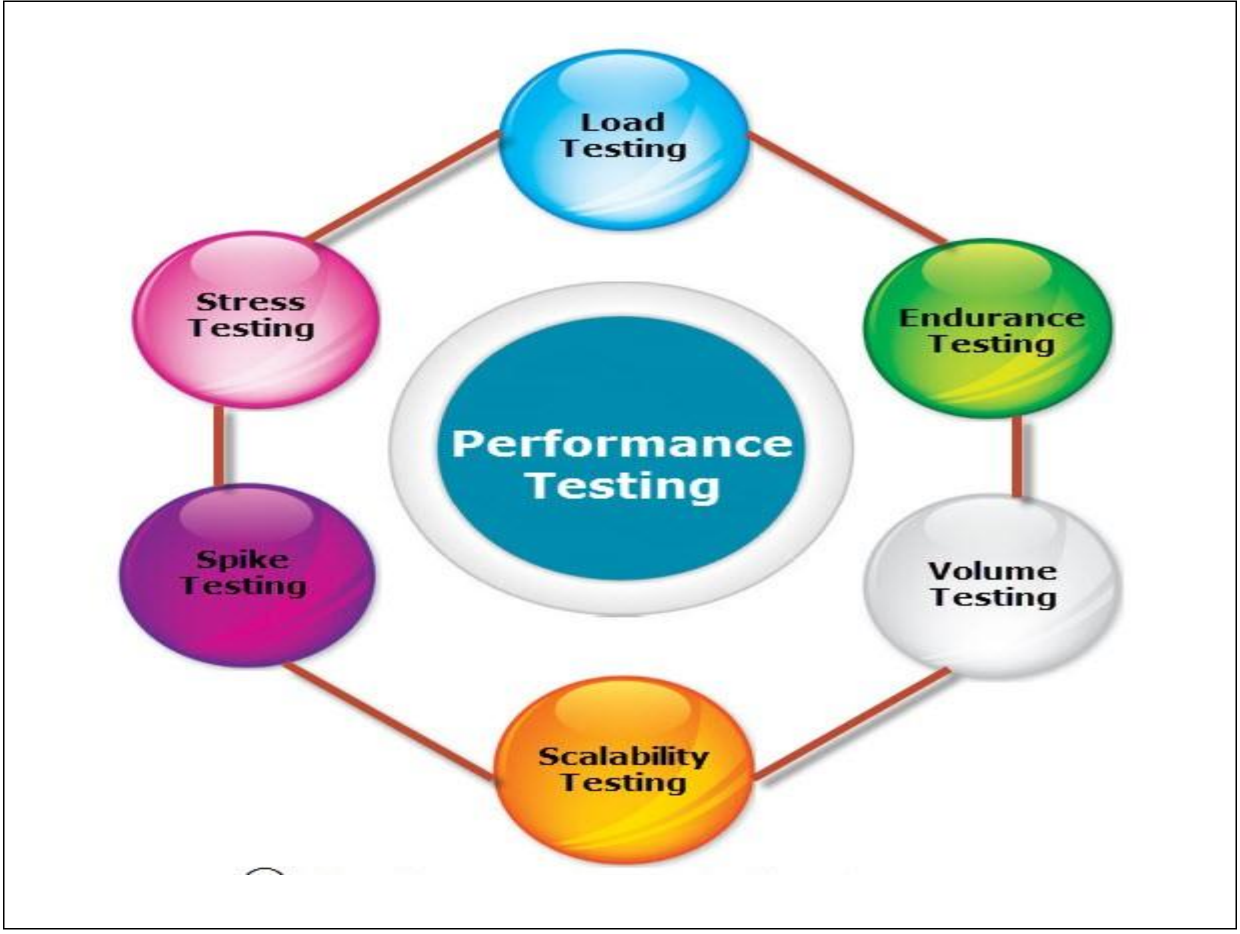
I. Steps Of Performance Testing


Identify the Test Environment
Identify Performance Acceptance Criteria
Plan and Design Tests
Configure the Test Environment
Steps of Performance Testing
Analyze Results and Report
Execute the Test
Implement Test Design

Why do performance testing?

Before going live in the market, the software system should be tested against the Speed, Stability and scalability under variety of load conditions. If system goes live without doing performance testing may cause the issues like running system slow while simultaneously accessing system by several users, poor usability which likely to gain the bad reputation and it affects the expected sales goal directly. Performance testing encompasses a range of different tests which enable analysis of various aspects of the system. The Performance testing is tells about what needs to fix before going live (mainly the issues faced under the variety of load conditions).

Types of Performance Testing:



1) Load Testing:

Load Testing is type of performance testing to check system with constantly increasing the load on the system until the time load is reaches to its threshold value. Here Increasing load means increasing number of concurrent users, transactions & check the behavior of application under test. It is normally carried out underneath controlled environment in order to distinguish between two different systems. It is also called as "**Endurance testing**" and "**Volume testing**". The main purpose of load testing is to monitor the response time and staying power of application when system is performing well under heavy load. Load testing comes under the **Non Functional Testing** & it is designed to test the non-functional requirements of a software application.

Load testing is perform to make sure that what amount of load can be withstand the application under test. The successfully executed load testing is only if the specified test cases are executed without any error in allocated time.

Simple examples of load testing:

- Testing printer by sending large job.
- Editing a very large document for testing of word processor
- Continuously reading and writing data into hard disk.
- Running multiple applications simultaneously on server.
- Testing of mail server by accessing thousands of mailboxes
- In case of zero-volume testing & system fed with zero load.

2) Stress Testing:

Stress Testing is performance testing type to check the stability of software when hardware resources are not sufficient like CPU, memory, disk space etc.

"**To determine or validate an application's behavior when it is pushed beyond normal or peak load conditions.**"

Stress testing is Negative testing where we load the software with large number of concurrent users/processes which cannot be handled by the systems hardware resources. This testing is also known as **Fatigue testing**, this testing should capture the stability of the application by testing it beyond its bandwidth capacity.

The main idea behind stress testing is to determine the failure of system and to keep an eye on how the system gracefully get recover back, this quality is known as recoverability. Stress testing comes under the Non Functional Testing & it is designed to test the non-functional requirements of a software application. This testing is to be carried out under controlled environment before launch, so that we can accurately capture the system behavior under most erratic scenarios

3) Spike testing:

Spike testing is subset of Stress Testing. A spike test is carried out to validate the performance characteristics when the system under test subjected to workload models and load volumes that repeatedly increase beyond anticipated production operations for short periods of time.

4) **Endurance testing**:

Endurance testing is a non functional type of testing. Endurance testing involves testing a system with a expected amount of load over a long period of time to find the behavior of system. Let's take a example where system is designed to work for 3 hrs of time but same system endure for 6 hrs of time to check the staying power of system. Most commonly test cases are executed to check the behavior of system like memory leaks or system fails or random behavior. Sometimes endurance testing is also referred as Soak testing.

5) **Scalability Testing**:

Scalability Testing is type of non-functional tests and it is the testing of a software application for determine its capability to scale up in terms of any of its non-functional capability like the user load supported, the number of transactions, the data volume etc. The main aim if this testing is to understand at what peak the system prevent more scaling.

6) **Volume testing**:

Volume testing is non-functional testing which refers to testing a software application with a large amount of data to be processed to check the efficiency of the application. The main goal of this testing is to monitor the performance of application under varying database volumes.

II. Introduction

There is an acceleration of adoption of cloud computing among enterprises. However, moving the infrastructure and sensitive data from trusted domain of the data owner to public cloud will pose severe security and privacy risks. Attribute-based encryption (ABE) is a new cryptographic primitive which provides a promising tool for addressing the problem of secure and fine-grained data sharing and decentralized access control. Key-policy attribute-based encryption (KP-ABE) is an important type of ABE, which enables senders to encrypt messages under a set of attributes and private keys are associated with access structures that specify which ciphertexts the key holder will be allowed to decrypt. In most existing KP-ABE scheme, the ciphertext size grows linearly with the number of attributes embedded in ciphertext. In this paper, we propose a new KP-ABE construction with constant ciphertext size. In our construction, the access policy can be expressed as any monotone access structure. Meanwhile, the ciphertext size is independent of the number of ciphertext attributes, and the number of bilinear pairing evaluations is reduced to a constant. We prove that our scheme is semantically secure in the selective-set model based on the general Diffie-Hellman exponent assumption.

Cloud computing is a model for enabling ubiquitous, convenient, and on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction [1]. There are two main categories of cloud infrastructure: public cloud and private cloud. To take advantage of public clouds, data owners must upload their data to commercial cloud service providers which are usually considered to be semitrusted, that is, honest but curious [2]. That means the cloud service providers will try to find out as much secret information in the users' outsourced data as possible, but they will honestly follow the protocol in general.

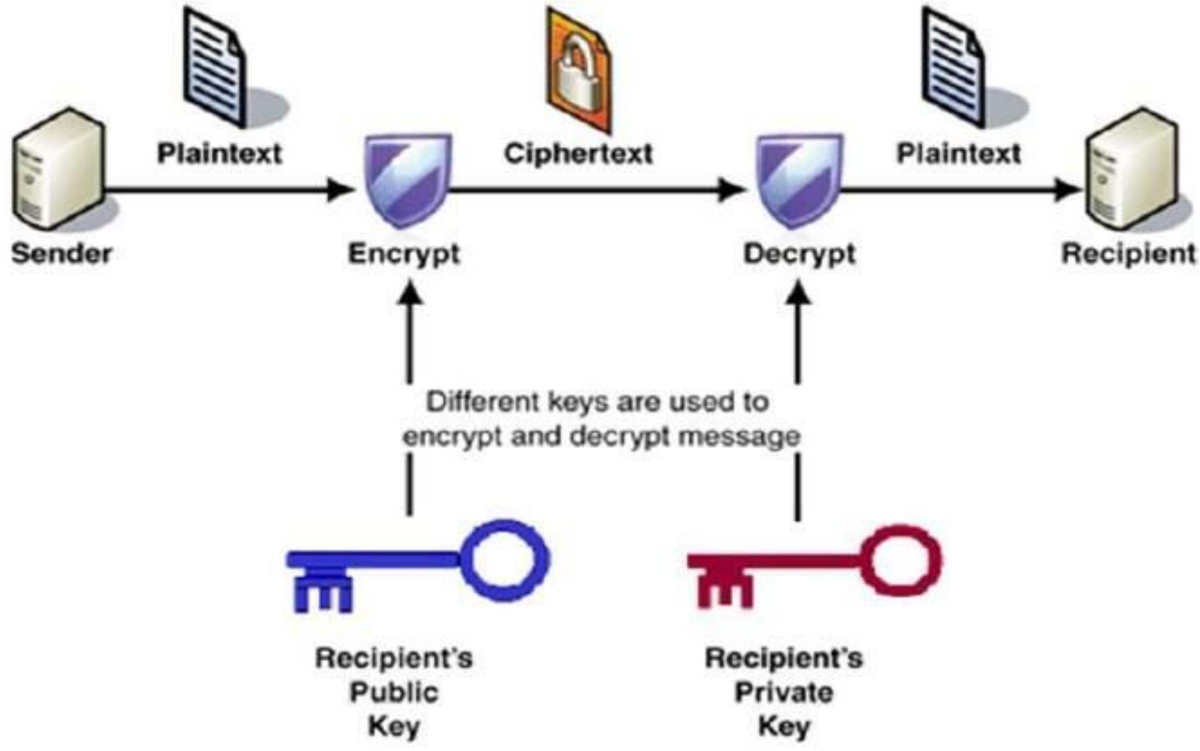
Traditional access control techniques are based on the assumption that the server is in the trusted domain of the data owner, and therefore an omniscient reference monitor can be used to enforce access policies against authenticated users. However, in the cloud computing paradigm this assumption usually does not hold, and therefore these solutions are not applicable. There is a need for a decentralized, scalable, and flexible way to control access to cloud data without fully relying on the cloud service providers.

III. Advanced Technologies Used

1. Encryption And Cryptography

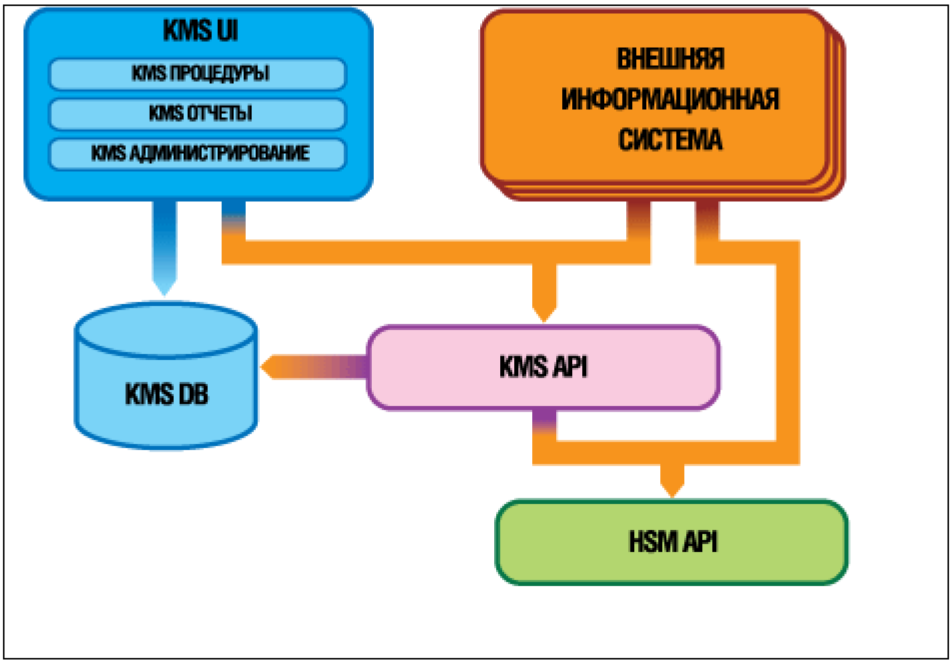
Generic architecture for implementing the advanced encryption standard (AES) encryption algorithm in silicon is proposed. This allows the instantiation of a wide range of chip specifications, with these taking the form of semiconductor intellectual property (IP) cores. Cores implemented from this architecture can perform both encryption and decryption and support four modes of operation: (i) electronic codebook mode; (ii) output feedback mode; (iii) cipher block chaining mode; and (iv) ciphertext feedback mode. Chip designs can also be generated to cover all three AES key lengths, namely 128 bits, 192 bits and 256 bits.

Based on the assertion that language and culture study are best brought together when the teacher is effective in the affective as well as cognitive and skills domains, teaching strategies and activities are presented that combine specific teaching techniques and ideas with other human and cultural resources in and out of the classroom. Chapters discuss: when and in what language culture should be taught; what "culture" refers to and its connection with language; approaches to understanding how societies evolve different ways to satisfy their needs; seven goals of cultural instruction; performance objectives for each of the seven goals; developing effective learning activities for each of the performance objectives; two techniques to sensitize students to the miscommunication accompanying interaction with those of another culture (empathetic literature and minidramas); three techniques for teaching cultural concepts (culture assimilators, culture capsules, culture clusters); approaches to helping students ask significant questions; measuring shifts in attitude toward the target culture; stresses and challenges of biculturalism; and suggestions for implementing a curriculum fostering intercultural communication.



2. Cryptographic Key Management System

AWS Key Management Service (KMS) is a managed service that makes it easy for you to create and control the encryption keys used to encrypt your data, and uses FIPS 140-2 validated hardware security modules to protect the security of your keys. AWS Key Management Service is integrated with most other AWS services to help you protect the data you store with these services. AWS Key Management Service is also integrated with AWS cloudtrail to provide you with logs of all key usage to help meet your regulatory and compliance needs.



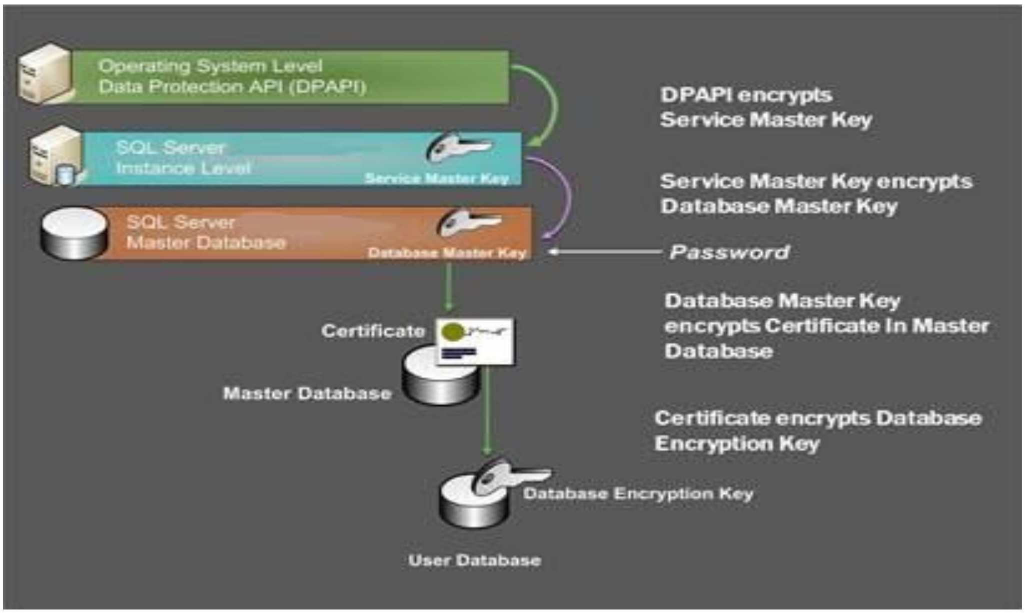
3. Stay Safe And Secure With Encryption

Users can keep their email or social media accounts secure even if their usernames and passwords have been compromised by using two-factor authentication. It works by adding an additional authentication process over the existing password. If anyone tries to break into your account using another device, they will be asked to furnish the two factor authentication number or fingerprint.

Google's Advanced Protection Program takes it one step further with Security key, which is basically a small USB device used to authenticate users at the time of login. Users have to plug in the USB device in the PC and type the pin number to login to their Google account, Drive or Gmail. The Advanced Protection Program automatically restricts all third-party apps from accessing anything on Gmail or Drive.

A system and method is provided here that allow computer user to create a temporary guest running space for application without switching user environment. This unique method allows user to run trusted applications in regular running space while keeping a separate working space for applications that uses or visit non trusted data sources.

Proposed method provides a safe execution environment while application running in guest space can't temper or alter data information stored in regular running space. A set of policy rules dictates how information will be exchanged between applications running in two separate working spaces transparently.



4. Data Encryption Methods

In AES is a symmetric key encryption, you must share the key with other individuals for them to access the encrypted data. Furthermore, if you don't have a secure way to share that key and unauthorized individuals gain access to it, they can decrypt everything encrypted with that specific key.

3DES

Triple Data Encryption Standard, or 3DES, is a current standard, and it is a block cipher. It's similar to the older method of encryption, Data Encryption Standard, which uses 56-bit keys. However, 3DES is a symmetric-key encryption that uses three individual 56-bit keys. It encrypts data three times, meaning your 56-bit key becomes a 168-bit key.

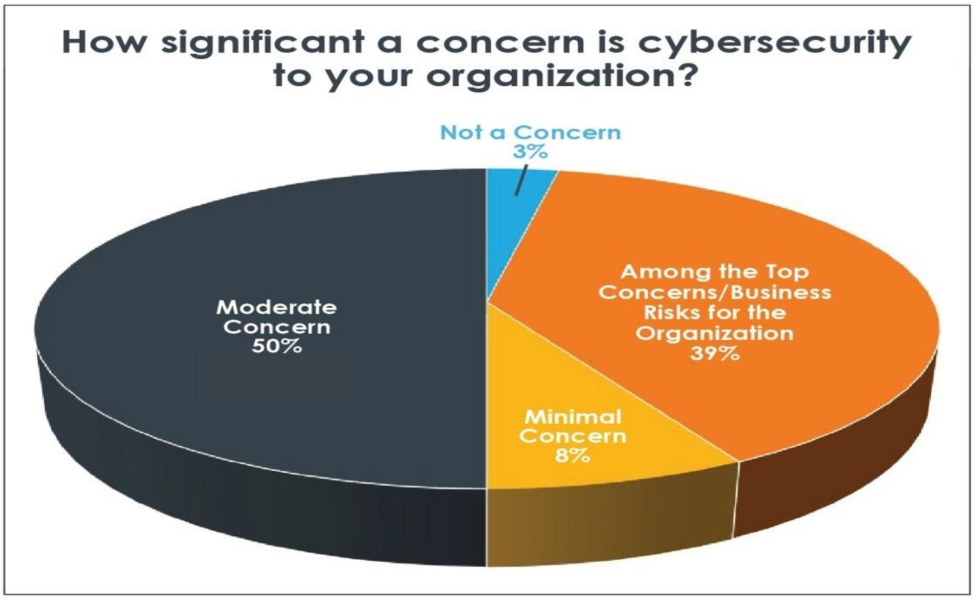
Unfortunately, since it encrypts data three times, this method is much slower than others. Also, because 3DES uses shorter block lengths, it is easier to decrypt and leak data. However, many financial institutions and businesses in numerous other industries use this encryption method to keep information secure. As more robust encryption methods emerge, this one is being slowly phased out.

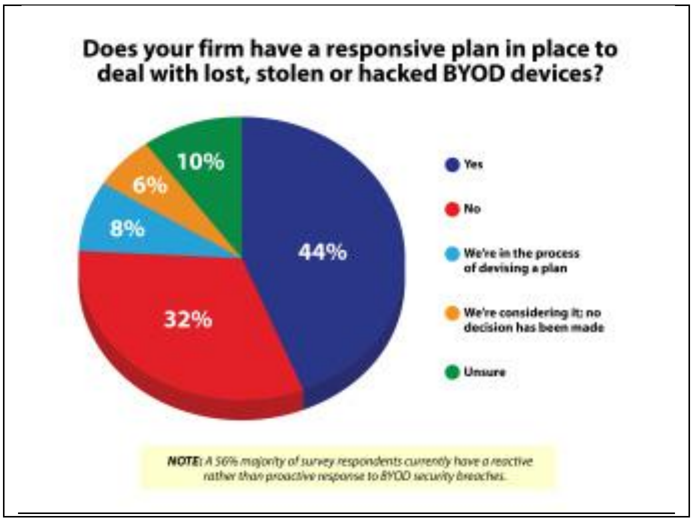
Two methodologies for efficient audiovisual data encryption are studied in this research, i.e. selective encryption and MHT (multiple-Huffman-table). We first discuss benefits and limitations of selective encryption, and propose a selective encryption scheme for ITU G.723.1 speech coding. A review of all cryptanalysis work of selective encryption algorithms in the literature is given, our cryptanalysis for the JPEG/MPEG sign-bit encryption scheme is presented, and two rules to determine whether selective encryption for a compression system is suitable are concluded. Then, we investigate another encryption methodology via the manipulation of Huffman coding tables of multimedia coding systems. The MHT scheme chooses several different Huffman tables from a vast number of possible candidates, and uses them alternatively to encode multimedia data. The choice of Huffman tables and the order that they are used are kept secret as the key. This method requires very little computational overhead, and can be applied to the encryption of MPEG audio, MPEG video and JPEG/JPEG2000 images.

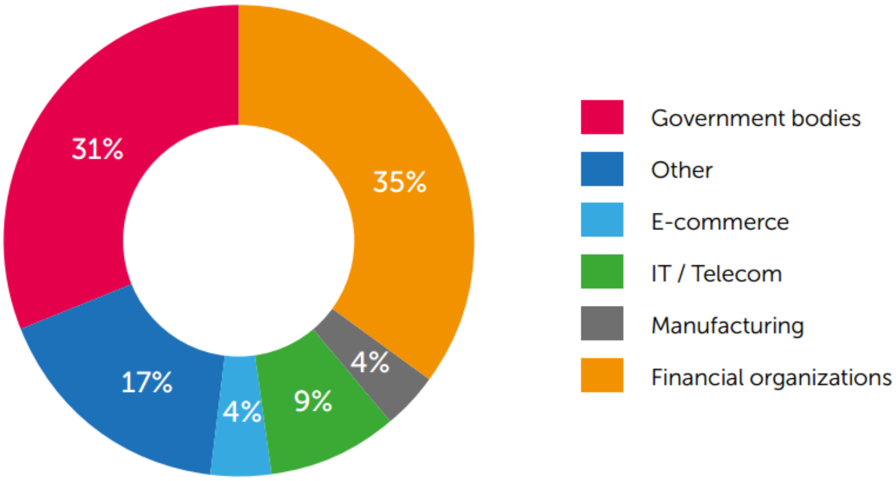
IV. Results And Discussion

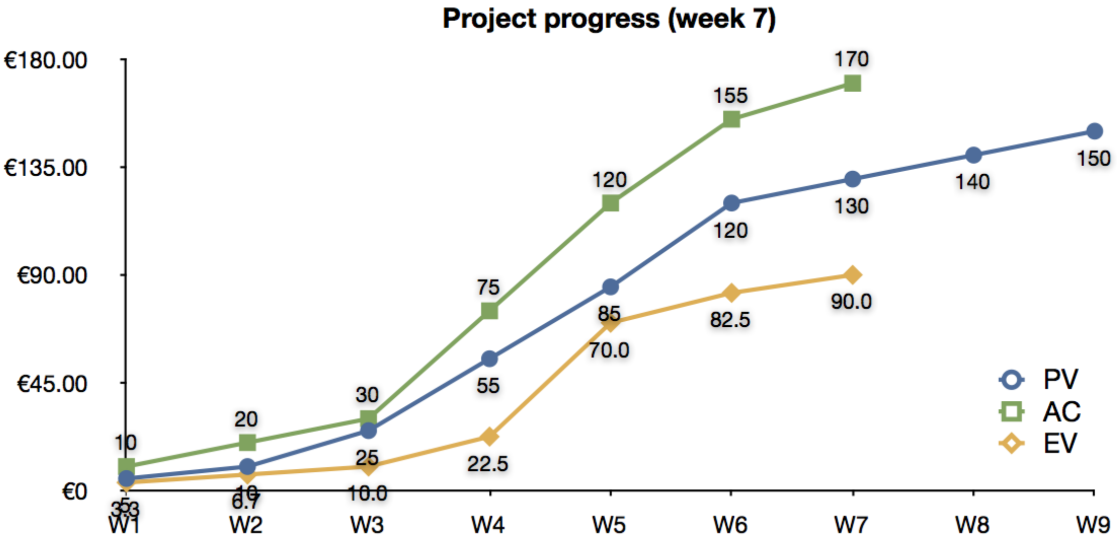
Attribute-based encryption (ABE), introduced by Sahai and Waters, is a promising cryptographic primitive, which has been widely applied to implement fine-grained access control system for encrypted data. In its key-policy flavor, attribute sets are used to annotate ciphertexts and secret keys are associated with access structures that specify which ciphertexts a user is entitled to decrypt. In most existing key-policy attribute-based encryption (KP-ABE) constructions, the size of the ciphertext is proportional to the number of attributes associated with it and the decryption cost is proportional to the number of attributes used during decryption. In this paper, we present a new construction of KP-ABE. Our proposed construction is the first KP-ABE scheme, which has the following features simultaneously: expressive (i.e., supporting arbitrary monotonic access structures); fully secure in the standard model; constant-size ciphertexts and fast decryption. The downside of our construction is that secret keys have quadratic size in the number of attributes.

The comparative analysis of time and memory are analyzed by the existing method.The graphical representation of the technique is shown below.




Does your firm have a responsive plan in place to deal with lost, stolen or hacked BYOD devices?
44%
32%
8%
6%
10%
Yes
No
We're in the process of devising a plan
We're considering it; no decision has been made
Unsure
NOTE: A 56% majority of survey respondents currently have a reactive rather than proactive response to BYOD security breaches.


31%
35%
17%
4%
9%
4%
Government bodies
Other
E-commerce
IT / Telecom
Manufacturing
Financial organizations


Project progress (week 7)
€180.00
€135.00
€90.00
€45.00
€0
W1
W2
W3
W4
W5
W6
W7
W8
W9
10
20
30
75
120
155
170
5
10
25
55
85
120
130
140
150
3.3
6.7
10.0
22.5
70.0
82.5
90.0
PV
AC
EV

V. Conclusion

My software is very reliable and useful to all web developers and web developing companies.It is easy to install,administrator can control all over the computers in one hand   Web application security is a stack of attack surfaces and defensive mitigating solutions. It is not enough to protect web applications with only one technique, or at only one layer of the stack. Vulnerabilities in the platform, or in protocols, such as TCP or HTTP, are just as devastating to the security and availability of applications as attacks against the application itself. A full stack of mitigating solutions is necessary to realise a positive web application security posture. It is important to note that a comprehensive approach requires collaboration across network, security, operations and development teams, as each has a role to play in protecting applications and their critical data.

Symantec has just presented the main conclusions of its annual report on Internet security threats, which analyses all the threats identified in 2018.

In general, we see that cyber attackers have made a change in tactics: They infiltrate networks and escape detection by diverting the infrastructure of large enterprises and using it against the companies themselves. In particular, they catch companies out by making them self-infect through Trojans during standard software updates. They then wait patiently for their targets to download these infected updates, giving them free access to the company's network.

2018 was also a record year for zero-day vulnerabilities, with software companies taking an average of 59 days to create and deploy patches. The attackers took advantage of this delay and, in the case of Heartbleed for example, were very responsive in exploiting the vulnerability in the four hours that followed. A total of 24 zero-day vulnerabilities were discovered in 2018, leaving the field open for attackers to exploit known security flaws before they were corrected.

References

[1]. M. Abdalla and L. Reyzin. A new forward-secure digital signature scheme. *Asiacrypt'00*, LNCS vol. 1976, pp. 116–129, Springer-Verlag, 2000.

[2]. R. Anderson. Two remarks on public key cryptology. Invited Lecture, ACM-CCS'97.

[3]. D. Beaver and S. Haber. Cryptographic protocols provably secure against dynamic adversaries. In *Eurocrypt' 92*, LNCS vol. 658, pp. 307–323, Springer-Verlag, 1992.

[4]. D. Beaver, Plug and play encryption, *Advances in Cryptology — Crypto' 97*, LNCS vol. 1294, pp. 75–89, Springer-Verlag, 1997.

[5]. M. Bellare and S. K. Miner. A forward-secure digital signature scheme. *Advances in Cryptology — Crypto' 99*, LNCS vol. 1666, pp. 431–448, Springer-Verlag, 1999 1.

[6]. Rebollo, O., Mellado, D.: Systematic Review of Information Security Governance Frameworks in the Cloud Computing. Journal of Universal Computer Sc. 18(6), 798–815 (2012) 3.

[7]. Rittinghouse, J., Ransome, J.: Security in the Cloud: Cloud Computing. Implementation, Management, and Security, 1st edn. CRC Press (2009) 4.

[8]. Hannay, J.E., Sjøberg, D.I.K.: A Systematic Review of Theory Use in Software Engineering Experiments. Journal of IEEE Transaction on Software Engineering 33(2), 87–107 (2007)

[9]. Summers, G. (2004). Data and databases. In: Koehne, H Developing Databases with Access: Nelson Australia Pty Limited. p4-5.

[10]. **Jump up^** Knowing Your Data to Protect Your Data Archived 2017-09-28 at the Wayback Machine.

[11]. **Jump up^** *Waksman, Adam; Sethumadhavan, Simha (2011), "Silencing Hardware Backdoors" (PDF), Proceedings of the IEEE Symposium on Security and Privacy, Oakland, California, archived (PDF) from the original on 2013-09-28*

[12]. **Jump up^** https://www.staysmartonline.gov.au/Protect-yourself/Doing-things-safely/backups

[13]. **Jump up^** *"Data Masking Definition". Archived from the original on 2017-02-27. Retrieved 1 March 2016.*

[14]. **Jump up^** *"data masking". Archived from the original on 5 January 2018. Retrieved 29 July 2016.*

[15]. **Jump up^** *Michael Wei; Laura M. Grupp; Frederick E. Spada; Steven Swanson. (February 2011). "Reliably Erasing Data From Flash-Based Solid State Drives" (PDF). FAST '11: 9th USENIX Conference on File and Storage Technologies.*

[16]. **Jump up^** *"data protection act". Archived from the original on 13 April 2016. Retrieved 29 July 2016.*

[17]. **Jump up^** *Peter Fleischer, Jane Horvath, Shuman Ghosemajumder(2008). "Celebrating data privacy". Google Blog. Archivedfrom the original on 20 May 2011. Retrieved 12 August 2011.*

[18]. **Jump up^** https://www.itgovernance.co.uk/dpa-and-gdpr-penalties

[19]. **Jump up^** *"Detect and Protect for Digital Transformation". Informatica. Informatica. Retrieved 27 April 2018.*

[20]. **Jump up^** *"PCI DSS Definition". Archived from the original on 2 March 2016. Retrieved 1 March 2016*

Soniya Thomas. " A Testing On Secure Key Policy Attribute-Based Encryption Policy For Data Sharing Among Dynamic Groups In The Cloud." IOSR Journal of Computer Engineering (IOSR-JCE) 20.6 (2018): 52-59.