ISSN 2278-3091

Volume 9, No.4, July – August 2020

International Journal of Advanced Trends in Computer Science and Engineering

Available Online at http://www.warse.org/IJATCSE/static/pdf/file/ijatcse25942020.pdf

https://doi.org/10.30534/ijatcse/2020/25942020

Two Fold Security on Cloud CRM using Hash Encryption and Role Based Security

Pawan Kumar[1], **Prof. Hariom Tyagi**[2]

[1]MTech. Student, Dept. of Computer Science & Engineering R.D. Engineering College at Duhai, Ghaziabad, India

[2]Professor & Head of Department, Dept. of Computer Science & Engineering R.D. Engineering College at Duhai, Ghaziabad, India

ABSTRACT

Virtually all companies today are existentially dependent on information and communication technology over the cloud computing. At the same time, the degree of networking between actors in business - companies, customers and government institutions - has risen steadily. This enables completely new, high-value-added applications, but at the same time poses new risks for the trouble-free functioning of a company's critical processes and the confidentiality of the stored, processed and transmitted data. In high-wage countries like India, companies are particularly forced to protect their own know-how and intellectual property well against competitors. Entrepreneurs who are concerned with knowing risks and evaluating measures to control them must therefore be able to address the challenges of effective information protection and be able to classify procedures that make an important contribution to this. The encryption of information is an effective technical procedure for this task. However, it is not an easy undertaking for a company to design an adequate overall concept for the use of encryption, since these methods can be used at very different levels of information technology and their use must be planned across processes. This scheme is intended to help overcome the obstacles to the efficient and effective use of encryption in the company using new enhanced hashing algorithm. It is designed to inform companies about how encryption helps to protect the confidentiality of information when using modern information and communication technology. It shows in which areas of application and against which attacks encryption is useful and how the various encryption methods can be meaningfully combined. This scheme is manufactured for managing directors, department heads and other decision-makers who want to familiarize themselves with the use of encryption technology in order to plan steps for designing an encryption architecture that makes sense for their company. Small and medium-sized companies in particular are in the spotlight, since they too are the focus of industrial espionage and they also have a higher risk that the loss of confidentiality of sensitive information will jeopardize their existence. We hereby introduce the new approach in the paradigm of cryptology (Role Based Security and Hashing Technique) in cloud computing for CRM solution with the name of H#ABE.

Key words: Cloud Computing, Data Security, Cryptography, Customer Relationship Management, Attribute Based Access control, Role Based Access Control, Secure Hash Algorithms

1. INTRODUCTION

Cloud computing technology is presented as an effort to allow access to resources and applications that can be accessed through the internet network. Cloud computing uses the concept of virtualization that can be accessed via the internet so as to reduce the cost of information technology and simplify the management of information technology services, and the services provided are multi-tenant so that computing resources can be used together and tailored to the needs of users [1]. In this research, a security analysis of cloud computing technology using Cloud Computing Security Standard Mapping [7] on CRM with the category of authentication and authorization and digital signature generation will be proposed. Within the Security Mapping Standard there is a security category for cloud computing. One of the categories is Authentication and Authorization, the selection of the category is due to problems that exist in CRM based on cloud in terms of authentication. Research conducted by [9] on cloud computing security techniques using attack centric methods, shows that cloud computing is currently inflexible.

1.1 ABAC (Attribute Base Access Control)

This approach allows the mechanism of access to digital resources to be more flexible and more complex in accordance with the needs of the interaction. Among the many models for Access Control Policy, such as Discretionary Access Control (DAC) [13], mandatory access control (MAC), Access Control List (ACL) [14], and Rule Based Access Control (RBAC)[15] and one of them is the ABAC (Attribute) model Based Access Control) [16]. This model is based on attribute verification and is believed to be an access control model that is adaptive to the access policy needs of various digital resources in the future. Meanwhile, for the purpose of implementing the access control policy from ABAC, the

Cipher text-Policy Attribute-Based Encryption [17] modeling language was developed. So far the application of ABAC as a system for access to digital resources is still very limited. A number of studies have been carried out including [18] and [19]. But in that research the model applied is the Next Generation Access Control (NGAC)[20] model with implementation in Cloud CRM along with Hashing algorithm as new proposed scheme and the next research model that is applied is the Role Based Access Control (RBAC) model with implementation in the Cipher text-Policy Attribute-Based Encryption (CP-ABE). For this reason, the exploration of Cipher text-Policy Attribute-Based Encryption (CP-ABE) and Hashing algorithm is designed for Cloud CRM[20] which is very important to study. This will contribute to the availability of resources and in the field of digital forensics in particular and cloud security in future. Given the importance of exploration of Cipher text-Policy Attribute-Based Encryption for access to digital resources, further research is needed to conduct further models to find out about how ABAC modeling of digital evidence resource access, implementation related to ABAC performance with Cipher text-Policy Attribute-Based Encryption compared to authorization, authentication, and verification approaches in general worn all this time. This approach is well suited to using the access control attribute or ABAC approach. Therefore, there has not been any study on how the application of ABAC with the implementation of Cipher text-Policy Attribute-Based Encryption and Hash Security on Cloud CRM.

1.2 Hashing Algorithm

Cryptography in digital signatures can overcome the problem of denial (repudiation). Cryptography not only provides tools for message security, but also a collection of useful techniques [21]. In cryptography there are two main processes, namely encryption and decryption. The process of encoding plaintext into cipher text is called encryption (encryption) or enciphering (standard names according to ISO 7498-2) [24]. While the process of returning the cipher text to the original plaintext is called decryption or deciphering (standard name according to ISO 7498-2) [25]; The following studies are related to digital signatures. Digital Signature here is not a digit digitized by a scanner, but a cryptographic value that depends on the message and the sender of the message. , if there is a party who wants to change the contents of the message, then he will track the contents of the message as long as the resulting hash value is the same [27]. Given the importance of this, a solution is needed to handle the data collisions that occur. Therefore in this study an analysis and implementation of the ElGamal algorithm and the application of the SHA-512 function to a digital signature are based on the studies described above. With the research to be carried out, it is expected that the application of the SHA-512 function can be used in conjunction with the ElGamal algorithm to handle collision problems on hash values that previously occurred in making digital signatures for cloud based CRM security incorporating Cipher text-Policy Attribute-Based Encryption.

1.3 Research Objectives

Based on the formulation of the problem that has been made then the purpose of this study can be taken as follows:
1. To find out the security of cloud computing based on Cloud Computing Security Standard Mapping Cloud Computing Standard Roadmap special publication 500-291.
2. Generate the design of the relevant attributes of access control digital resource for Cloud based CRM.
3. Generating RBACS policy design on access to digital resource control.
4. To find out the results of the analysis and implementation of the new technique this should be better than ElGamal algorithm & SHA-512 for handling data collisions on digital signatures over the cloud.

2. LITERATURE SURVEY

2.1 Cloud Computing

Cloud computing technology is presented as an effort to allow access to resources and applications from anywhere through the internet network. Cloud computing uses the concept of virtualization which can all be accessed through the internet so that it can reduce the cost of information technology and simplify the management of information technology services, and the services provided are multi-tenant so that computing resources can be used together and tailored to the needs of users [36].

2.1.1 Cloud Computing Service Models

There are three service models of cloud computing [40], namely:

1. *Cloud Software as a Service (SaaS).*

   The ability given to consumers to use provider applications can operate on cloud infrastructure. Applications can be accessed from various client devices through interfaces such as web browsers (for example, web-based email).

2. *Cloud Platform as a Service (PaaS).*

   The ability given to consumers to deploy applications created by consumers or obtained into cloud computing infrastructure using programming languages and equipment supported by providers.

3. *Cloud Infrastructure as a Service (IaaS).*

   The ability given to consumers to process, store, network, and other important computing resources; Where consumers can deploy and run software freely, which can include application operating systems.

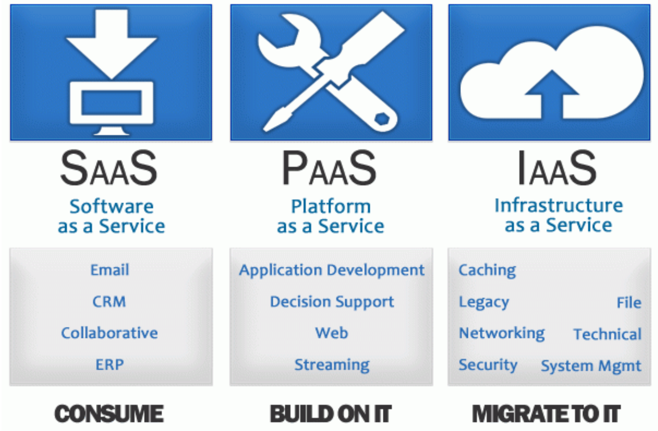


Figure 1: Cloud Computing Service Models

2.2 Cloud Computing Deployment Models

There are four cloud computing deployment models [41], namely:

1. *Private cloud:* Cloud infrastructure that is operated solely for an organization. This may be owned managed and run by an organization, third party or a combination of several parties and may exist on premise or off premise.
2. *Community cloud:* Cloud infrastructure is shared by several organizations and supports certain communities that have shared concerns. It may be managed by an organization or a third party and may exist on premise or off premise.
3. *Public cloud:* Cloud infrastructure is provided to the public or large industrial groups and is owned by an organization that sells cloud services.
4. *Hybrid cloud:* Cloud infrastructure is a composition of two or more clouds that are still entities but are bound together by standards or proprietary technologies that use data and application portability.

2.3 Information Security

The more information stored, managed and shared, the greater the risk of damage, loss or exposure of data to unwanted parties. Information security consists of protecting the following aspects:

1. *Confidentiality*: aspects that guarantee the confidentiality of data or information, ensure that information can only be accessed by authorized people and guarantee the confidentiality of data sent, received and saved.
2. *Integrity* (integrity): aspects that guarantee that data is not changed without the permission of the competent authority, maintaining the accuracy and integrity of information and the method of its process to guarantee aspects of integrity.
3. *Availability* (availability): aspects that guarantee that data will be available when needed and ensure that authorized users can use information and related devices.

2.4 Security Standard Mapping

The following is a standard roadmap in cloud computing security namely:

1. *Authentication and Authorization*
   Authentication is the process of ensuring that the plea is a true culprit. This process ensures that if there are other people who access it will be detected as another person not the perpetrator. In this authorization process, it will be determined what menus the user can run. Usually each user has been given certain rules in running applications that have been built.
2. *Integrity*
   Integrity is an aspect that guarantees that data cannot be changed without permission from the party concerned (authorized)..
3. *Availability*
   Aspects that guarantee that data is available when needed. and ensuring that authorized users can use information and related devices.
4. *Security Monitoring and Incident Response*
   Security Monitoring and incident response is a regulation that discusses security in computer networks and security in computer network security in the event of attacks in the network.
5. *Security Policy Management*
   Is a security infrastructure that must be owned by an organization or company or agency that wants to protect its most important information assets. With the policy, in addition to helping to secure important assets, it also avoids incidents or lawsuits caused by organizations, companies or agencies negligent in managing information assets or matters related to information governance in their environment.
6. *Availability*
   Aspects that guarantee that data is available when needed and ensuring that authorized users can use information and related devices.

2.5 Access Control

According to [13], access control is central to security computer. Furthermore according to [14], based on the main objective function of computer security itself is achieving three things, namely preventing users from unauthorized access to resources, preventing legitimate users from unauthorized access to resources, and to allow authorized users to officially access resources. Access control is in principle a mechanism to limit operations or action on a computer system only on legitimate users. Next according to [15, 16], there are 4 main issues in access control, namely identification, authentication, authorization and access decisions. The short explanation is as following:

1. Identification recognizes the party that will be responsible for the access request, can be tangible people or NPE (non person entity) as well as computers, or application.
2. Authentication is an effort to confirm the truth of a part of data or an entity. User authentication itself means confirming User data that has previously been saved.
3. Authorization is the process of determining what services are allowed to be used by users whose identity is clear (authenticated user).
4. Access Decision: based on a combination of the three aspects above then later decision is given whether the request is permitted or rejected by the system. In principle,

access control is a security feature that controls how users and systems communicate and interact with the system and other resources. Access control protects the system and a resource from that access is not entitled and generally determines the level of authorization after the authentication procedure completed successfully.

2.6 Elgamal Algorithm

The Elgamal algorithm is also a public-key cryptographic algorithm. This algorithm was originally used for digital signatures, but was later modified so that it could also be used for encryption and decryption. ElGamal algorithm is an algorithm in cryptography that is included in the category of asymmetric algorithms. The security of ElGamal's algorithm lies in the difficulty of calculating discrete logarithms on large prime modulo numbers so that the effort to solve this logarithmic problem becomes very difficult. The ElGamal algorithm has a public key in the form of three pairs of numbers and a secret key in the form of one number. ElGamal algorithm consists of three processes, namely the process of key formation, the encryption process and the decryption process. This algorithm is a block cipher, which performs the encryption process on plaintext blocks and produces cipher text blocks which are then decrypted and the results combined. Simply explained in figure below;

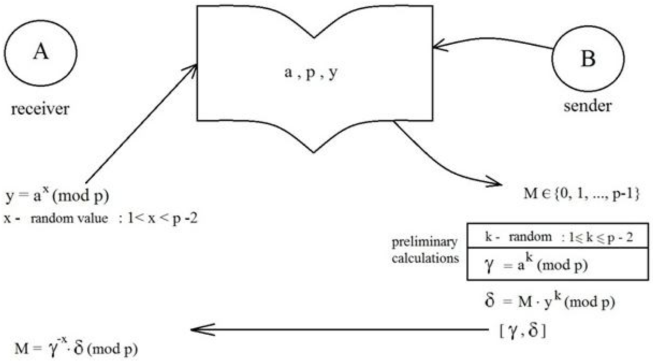

A
receiver
a , p , y
B
sender
$y = a^x \pmod p$
x - random value : $1 < x < p - 2$
$M \in \{0, 1, ..., p-1\}$
preliminary calculations
k - random : $1 \leq k \leq p - 2$
$\gamma = a^k \pmod p$
$\delta = M \cdot y^k \pmod p$
$[\gamma, \delta]$
$M = \gamma^{-x} \cdot \delta \pmod p$

Figure 2: Elgamal Scheme

Key Formation Process

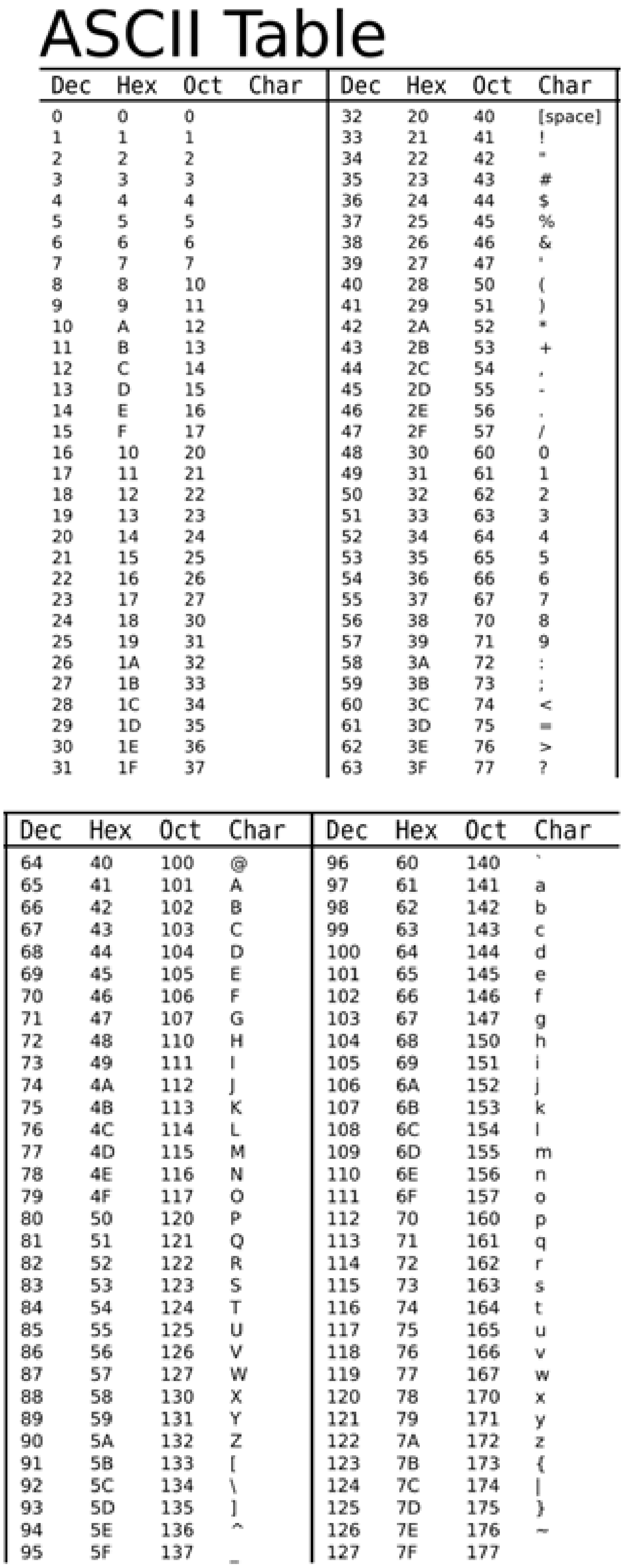
Input: Safe prime number p> 255

Output: Public key (p, α, β) and secret key a Process:

1. Select any prime number p> 255 (p can be published).
2. Choose two random numbers, g and x, with the condition that g <p and 0 ≤ x ≤ p – 2.
3. Calculate y = gx mod p. y is part of the public key, so the algorithm's public key ElGamal is a pair of 3 numbers, i.e. (y, g, p). While the secret key is the x number. The key generation process is carried out at the receiver side of the message, then after the public key is generated, the public key is sent to the sender message for the next encryption process.

Encryption Process

In this process the message is encrypted using public keys (y, g, p) and any random number k secret member {0,1, ..., p - 2}. Suppose m is the message to be sent. Furthermore, m is converted into blocks characters and each character is converted into ASCII code, such that obtained plaintext m1, m2, ..., mn with mi members {1, 2, ..., p - 1}, i = 1, 2, ..., n. The following is an ASCII table described in Figure 3.



ASCII Table

Dec	Hex	Oct	Char	Dec	Hex	Oct	Char
0	0	0		32	20	40	[space]
1	1	1		33	21	41	!
2	2	2		34	22	42	"
3	3	3		35	23	43	#
4	4	4		36	24	44	$
5	5	5		37	25	45	%
6	6	6		38	26	46	&
7	7	7		39	27	47	'
8	8	10		40	28	50	(
9	9	11		41	29	51	)
10	A	12		42	2A	52	*
11	B	13		43	2B	53	+
12	C	14		44	2C	54	,
13	D	15		45	2D	55	-
14	E	16		46	2E	56	.
15	F	17		47	2F	57	/
16	10	20		48	30	60	0
17	11	21		49	31	61	1
18	12	22		50	32	62	2
19	13	23		51	33	63	3
20	14	24		52	34	64	4
21	15	25		53	35	65	5
22	16	26		54	36	66	6
23	17	27		55	37	67	7
24	18	30		56	38	70	8
25	19	31		57	39	71	9
26	1A	32		58	3A	72	:
27	1B	33		59	3B	73	;
28	1C	34		60	3C	74	<
29	1D	35		61	3D	75	=
30	1E	36		62	3E	76	>
31	1F	37		63	3F	77	?

Dec	Hex	Oct	Char	Dec	Hex	Oct	Char
64	40	100	@	96	60	140	`
65	41	101	A	97	61	141	a
66	42	102	B	98	62	142	b
67	43	103	C	99	63	143	c
68	44	104	D	100	64	144	d
69	45	105	E	101	65	145	e
70	46	106	F	102	66	146	f
71	47	107	G	103	67	147	g
72	48	110	H	104	68	150	h
73	49	111	I	105	69	151	i
74	4A	112	J	106	6A	152	j
75	4B	113	K	107	6B	153	k
76	4C	114	L	108	6C	154	l
77	4D	115	M	109	6D	155	m
78	4E	116	N	110	6E	156	n
79	4F	117	O	111	6F	157	o
80	50	120	P	112	70	160	p
81	51	121	Q	113	71	161	q
82	52	122	R	114	72	162	r
83	53	123	S	115	73	163	s
84	54	124	T	116	74	164	t
85	55	125	U	117	75	165	u
86	56	126	V	118	76	166	v
87	57	127	W	119	77	167	w
88	58	130	X	120	78	170	x
89	59	131	Y	121	79	171	y
90	5A	132	Z	122	7A	172	z
91	5B	133	[	123	7B	173	{
92	5C	134	\	124	7C	174	|
93	5D	135	]	125	7D	175	}
94	5E	136	^	126	7E	176	~
95	5F	137	_	127	7F	177	

Figure 3: ASCII Chart

Input: Data string (customer id, customer balance) to be encrypted and public key (p, α, β).
Output: Cipher text (a_i, b_i), i = 1, 2, ..., n.
Process:

1. Arrange plaintext into blocks m1, m2, ... mn with each A block is a message character.
2. Convert each character into the ASCII code then obtained plaintext of n numbers, i.e. m1, m2,...M_n
3. For i from 1 to n, do:
    - Choose any secret random number ki ∈ {0,1, ..., p2}
    - Calculate AI = g^{ki} mod p
    - Calculate bi = y ki^{mi} mod^p
4. Cipher text results are (a_i , b_i), i = 1, 2, ..., n. So the size cipher text is twice the size of plain text .

The next scenario is the sender sends a message to receiver.

Decryption Process

After receiving the cipher text (a, b), the next process is decrypt cipher text using public key p and secret key x. It can be shown that plaintext m can be obtained from cipher text use the secret key x.

Input: Cipher text (a_i , b_i), i = 1, 2, ..., n public key p and secret key a.
Output: Original data
Process:

1. Cipher text (ai , bi), i = 1, 2, ..., n, public key p and key secret x.
2. For i from 1 to n, count m = b_i .a_i $^{p-1-x}$ mod p
3. Value of mi obtained in the form of ASCII then changed to be plaintext.
4. Arrange the plaintext in the order m_1, m_2, m_3, ..., m_n.

The results obtained from the decryption process in the form of an original message (plaintext).

2.7 Customer Relationship Management (CRM)

CRM or Customer Relationship Management is a core strategy in a business that integrates processes and functions internal with all external networks to be made properly realize value for profitable target customers. CRM Supported by quality consumer data and information technology. This definition is used in a company or context profit oriented organization. If the non-profit (non-profit) community can change the words 'business', 'consumer', and 'profit' with other appropriate terms, so the resolution must also be appropriate for their work relationship. Customers themselves are divided into three, namely:

a. *External Customers:* is a customer outside the company or organization?
b. *Internal Customers*: is a customer who is still a part of the organization or company?
c. *Customer Attributes:* is a characteristic of customers who grouped by demographic type (age, income, education, etc.), psychographics (culture, social class, hobbies, etc.) or company, type of company, hours work, etc.)

3.PROPOSED METHODOLOGY

This research scheme is carried out, so that it can provide details about the flow or steps that are made systematically as well can be used clearly in solving problems, making an analysis of research result. The research scheme can be seen in Figure 4.

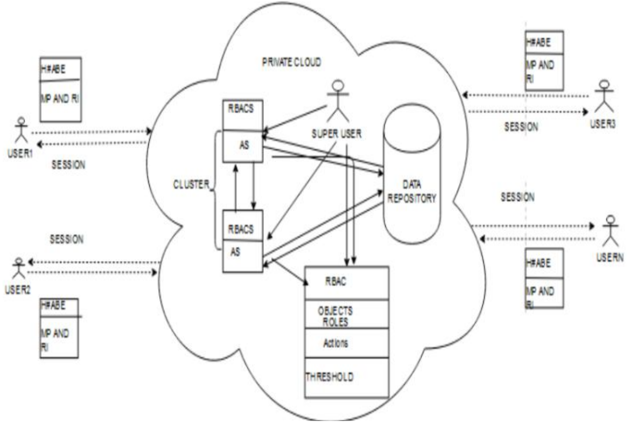


Figure 4: Proposed Workflow using RBACS and Hashing Algorithm for Cloud CRM

The proactive (H#ABE) and reactive (RBACS) mechanism are combined together to enhance the security of the private cloud in the proposed model. This model will be more secured than existing. Once the user gets login into the system, user gets access for authentication/authorization through HABE for encryption process and Role Based Access Control Security (RBACS). In the proposed architecture the data/message will pass through H#ABE. After the authentication, plaintext is changed into cipher text. The cipher text data is transmitted or transaction takes place in the private cloud. Thereafter, it passes through the Role Based Access Control Security (RBACS) module on the basis of credential passed to the system which evaluates the objects, Roles, Operations and the threshold. Subsequently, the data is stored in the data repository.

3.1 Role Based Access Control Structure

RBAC has received very broad attention in commercial applications. The RBAC model is defined in four component models: Core RBAC, Hierarchical RBAC, Static Separation of Duty Relations (SSD), and Dynamic Separation of Duty Relations (DSD). Core RBAC is defined as the minimum collection of RBAC elements in which the series of elements are fully correlated to obtain a system in RBAC. In the RBAC structure access rights will be assigned to roles rather than directly to the user, and those roles will be used by the user, then the user creates a session, and the session will get permission from the role obtained by the user and then will activate the role (roles). For the roles obtained by the user, it is adjusted to their management. Roles obtained by users are arranged according to their hierarchy where the roles obtained by seniors are more powerful and have more access rights than the roles obtained by juniors.

In an organization, roles are created to represent various job functions. The right of access to carry out certain operations / processes on the system is given to certain roles. Staff members (or other system users) get that role, and only through roles do they get access rights to perform system functions. Users are not given direct access rights to carry out certain operations, but only get through their roles, managing access rights for individual users is made easy by determining roles that are appropriate for users, this simplifies general operations, such as adding users, or changing organizational units the user. There are three main rules in the RBAC, namely:

1. *Giving roles:* A user or also called subject can run a transaction or process on the system only if the user is given a role that is permitted to run the transaction. Role can be considered as a user group, a person is allowed to carry out operations that are the rights of his group. Giving a role is done by stating that a user is a member of the group.
2. *Authorize roles:* Authorization for users to carry out transactions is done by activating the role that the user wants to use. In the condition that the user does nothing to the system, all roles are passive. The role activation is temporary during the session. When the session ends, the role is deactivated again. With rule number 1 above, we can be sure that the roles that are activated are only those roles that the user really has.
3. *Authorization of Transactions:* The user can execute the transaction only if the transaction has been authorized for the user's role that is activated. With rules number 1 and number 2, it is ensured that users can only make transactions for those who have been authorized. To define the RBAC model, the following conventions will be useful:
   a) S = Subject = A user or other application (automated agent)
   b) R = Role = Job or position function that determines the level of authority
   c) P = Permissions = Permission from access mode on resources
   d) SE = Session = Mapping involving S, R and / or P
   e) SA = Granting roles to subjects (subject assignment)
   f) PA = Granting permission for access to processes / transactions
   g) RH = Sequential role hierarchy partial (partially ordered role hierarchy). RH can also be written: ≥ (this notation: $x \geq y$ means x inherits the access rights y)
   h) Subject can have more than one role.
   i) Roles can have members of several subjects.
   j) Roles that can have more than one access right.
   k) Access permission can be granted to more than one role. Other restrictions can be made to realize binding rules, for example to prevent two opposite roles held by someone. For example the role to create a user account must be separated from the role to determine the access authorization. In the set notation, the following conditions apply:
      i. $PA \subseteq P \times R$, is a many-to-many relationship between access permits and roles b) $SA \subseteq S \times R$, is a many-to-many relationship between subjects and roles
      ii. $RH \subseteq R \times R$ All of the above conventions and their relationships can be seen in Figure 5 as under:-

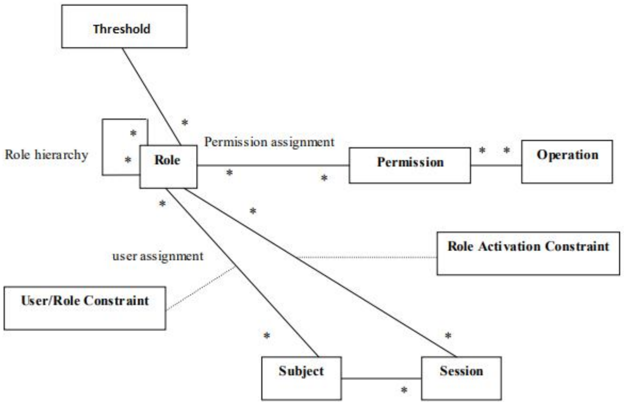


Figure 5: Role-Based Access Control (RBAC) Model

Algorithm for RBACS

Step 1:

SSD ⊆ (x N) is a group of pairs (rs, n) where each rs is a set of roles and n is number ≥ 2, with this property that users cannot be assigned to n or more roles than

Step 2:

set rs (rs, n) ϵ SSD. Various SSD policies can be applied differently, for example users can only run one role from a group of roles or the user cannot run all the roles of the predefined role

Step 3:

n (min) = threshold $| rs | t |$ where $\forall\forall\ (rs, n) \epsilon SSD, \forall t \subseteq rs: | t | \geq n \rightarrow \frac{\cap}{r \epsilon t}\ assignet_{users}(r) = \emptyset$ assign t_users (r) = SSDs that have been redefined with role hierarchy (RH) in are:

$\forall\ (rs, n)\ \epsilon\ SSD, \forall t \subseteq rs: | t | \geq n\ \forall \geq n \rightarrow \frac{\cup}{r \epsilon t}\ assign\ t_{users}(r) = \emptyset$

Step 4:

Thereafter $\forall rs \, \epsilon \, 2^{ROLES}$, $n \, \epsilon \, N$, $(rs,n) \, \epsilon \, DSD \Rightarrow n \geq 2. | rs | \geq n$, and

$\forall s \, \epsilon$ SESSIONS, $\forall rs \, \epsilon \, 2^{ROLES}$, $\forall role\_subset \, \epsilon \, 2^{ROLES}$, $\forall n \, \epsilon \, N$, $(rs,n) \, \epsilon$ DSD, $role\_subset \subseteq rs$, $role\_subset \subseteq session\_role(s) \Rightarrow | role\_subset | < n$.

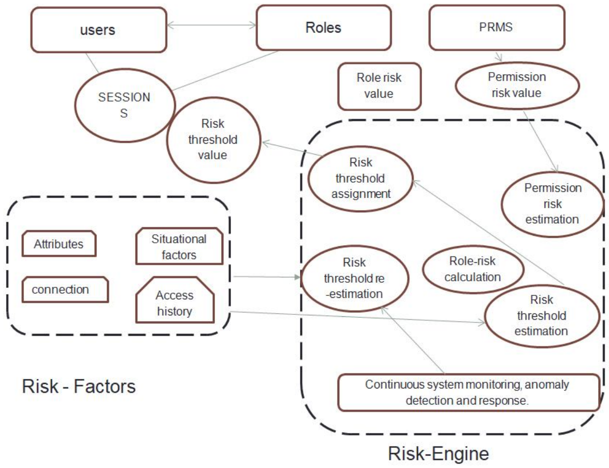


Figure 6: Role-Based Access Control Risk Engine and Risk Factors

3.2 Hashing Scheme

Hash function is an H function that maps random length bits of a string to fixed length length bits with attribute V which has the following properties:

1. Given a function and message M, then easily calculated the hash value of the message.
2. Given the hash value of the H function, very difficult to obtain an M message with H (M) = hash value given.
3. Given the message M and the H function, it is very difficult to find another message (M) with H (M') = H (M). A hash value is a value generated from processing a message with a hash function. Hash values are also often referred to by terms such as digest, hash code, or total hash (total hash).

Message Padding

Padding is the process of adding bits to an original message (represented in binary numbers). Therefore, the original message bit length is equal to m * n where m is the block size (number of bits in the block) used in the algorithm and n is the number of blocks. Attack on the Hash Function The attack is an attempt by a cryptanalyst to break a security service. An attack is said to be successful if it can defeat the security attributes possessed by a hash function.

Attacks can be divided into two namely:

a. Active attack is a attack carried out by enemies who are able to manipulate information from unsafe channels. Examples include dictionary attacks and algorithm-independent attacks.

b. Passive attacks are attacks carried out by enemies who are only able to read information from unsafe channels. Examples include ciphertext-only attacks. known-plaintext attack, chosen-plaintext attack, and chosen-ciphertext attack. Brute-force attack Brute force attack is an attempt to create a message with a predetermined message digest by trying every possible bit of input. If an intruder already knows the message in his message, he will be able to make a message related to the message digest using a brute force attack, experiments conducted with m length of output. Brute force attacks can be categorized in dictionary attacks. Birthday attack A birthday attack is an attempt to make a message pair (x.x ') have the same message digest (collision). To make random messages;

3.3 Proposed Hashing Algorithm

The process for generating HASH signature in this algorithm includes 5 steps:

Step 1: Message padding

Message input on the 512 bits based algorithm will be divided into blocks, each of which is 256 bits in length. As a result of this division, the last number of blocks will be smaller or equal to 512 bits. The last block will experience message padding. The steps of message padding are as follows:

a. Begins with the entry of the message input that has the American Standard Code for Information Interchange (ASCII) code and then is converted into a binary string of bits that will be calculated in length.

b. The series of bits is divided into blocks, each of which has a length of 512 bits. The result of division will cause the last number of blocks to be smaller or equal to 512 bits.

c. Perform additional padding in the last block of the message. The bits used as the fill bit is a bit followed by a number of ‘0’ bits as needed, with the following conditions

- If the bit length of the last message block is smaller than 448 bits, then the ‘1’ bit is added to the last bit position, followed by a few ‘0’ bits such that the total bit length after the process is 448 bits.
- If the bit length of the last message block is greater than or equal to 448 bits, then the ‘l’ bit is added to the last bit position, followed by several ‘0’ bits such that the total bit length after the process is 512 bits. Then 448 new bits are created with the contents ‘0’.
- If the bit length of the last message block is 512 bits, a new block must be created to accommodate the message padding process. The first bit of the new block is filled with bit ‘1’, while the next bits up to bit length 448 are filled with bit ‘0’. The total number of fill bits added is 448 bits.

For example, a message with a length (in bits) of 616 bits is denoted as M. After being divided into blocks of 512 bits, it becomes block1 = 512 bits and block2 = 104 bits. The last block (block2) is added with the filling bits. In accordance with the provisions of message padding, the last bit of block2 is added bit

- 1 ‘and followed by a few’ 0 ‘bits so that the total length of the block2 bits after the message padding process is 448 bits.

Step 2: Increase the bit length

After the message padding process, the number of bits in the last block is 448 bits. Represent M in binary numbers to get the last 64 bits, so that the total length of the last block is 512 bits.

a. The rightmost byte order of the message length representation value (M) is used as a low order.

b. Add the M representation in the last 448 bits, so the number the last block length is 512 bits.

In the example above, M = 616 bits and represented in 16-bit binary numbers are: 00000010 | 01101000

This value is made into 64 bits so the result is:

00000000 | 00000000 | 00000000 | 00000000 | 00000000 | 00000000 00000010 | 01101000

Because the rightmost byte order is a low order, the bit order is fixed. Add this M representation to the last block, so that the total length after this process is 512 bits.

Step 3 initialize the initial hash value

In SHA-256 to store initial initialization values and temporary output values, buffer, H1, H_0, H_1, H_2, H_3, H_4, H_5, H_6, H_7 buffers, on the other hand, temporarily store buffers a, b, c, d, e, f, g, h. The values of H_0, H_1, H_2, H_3, H_4, H_5, H_6, H_7 for initial initialization in hexadecimal notation:

H_0 = 6a09e667
H_1 = bb67ae85
H_2 = 3c6ef372
H_3 = a54ff53a
H_4 = 510e5271

H_5 = 9b05688c
H_6 = 1183d9ab
H_7 = 5be0cd19.

Step 4 Processing

Processing is a core part consisting of 1 round which has 64 operations. To process every single 512-bit message block requires 64 operations. Each message block, $M^{(1)}$,$M^{(2)}$,$M^{(N)}$ where N is the number of message blocks . For each message block $M^{(i)}$ will do step as under:-

a. Prepare message scheduling { w_t }

$$w_t = \begin{cases} M_{t^{(i)}} & 16 \le t \le 63 \\ \sigma_1(w_{t-2}) + w_{t-7} + \sigma_0(w_{t-15}) + w_{t-7} & 16 \le t \le 63 \end{cases}$$

With the functions $\sigma_0$and σ_1 are formulated as follows :-

$$\sigma_0(x) = ROTR^7(x) \oplus ROTR^{18}(x) \oplus SHR^3(x)$$
$$\tau_1(x) = ROTR^{17}(x) \oplus ROTR^{19}(x) \oplus SHR^{10}(x)$$

ROTR”(x) is a circular shift right-hand operation, where x is a message scheduling (w) and is an integer (0 ≤n <w), which can be defined ROTR” (x) = ((x >> n) v (x << w – n)) = $ROTC^{w-n}$. (x). In this case, SHR^n (x) is an operation to move x by n positions to the right.

b) Initialization of working variables a, b, c, d, e, f, g, and h, for $M^{(1)}$ with values. Initial hash:

$$a = H_{0^{(i-1)}}$$
$$b = H_{1^{(i-1)}}$$
$$c= H_{2^{(i-1)}}$$
$$d= H_{3^{(i-1)}}$$
$$e= H_{4^{(i-1)}}$$
$$f= H_{5^{(i-1)}}$$
$$g= H_{6^{(i-1)}}$$
$$h= H_{7^{(i-1)}}$$

c) For each schedule message W_t:

$$T_1 = h\sum 1\,(e) + Ch(e,f,g) \div K_1 \div W_1$$
$$T_2 = \sum 0\,(a) \div maj(a,b,c)$$

h = g
g = f
f = e
e = d + T_1
d = c
c = b
b = a
a = T_1 + T_2

With the functions ∑0, ∑1 Ch, Maj are formulated as follows:-

$$Ch(X,Y,Z) = (X\,\&Y) \oplus (\overline{X}\&\,Z)$$
$$Maj(X,Y,Z) = (X\,\&Y) \oplus (X\&\,Z) \oplus (Y\&\,Z)$$
$$\sum 0\,(x) = ROTR^7(x) \oplus ROTR^{13}(x) \oplus ROTR^{22}(x)$$
$$\sum 1\,(x) = ROTR^6(x) \oplus ROTR^{11}(x) \oplus ROTR^{25}(x)$$

d) An intermediate hash value for each message block:

$$H_0^{(i)} = a \div H_{0^{(i-1)}}$$
$$H_1^{(i)} = \mathrm{b} \div H_{1^{(i-1)}}$$
$$H_2^{(i)} = \mathrm{c} \div H_{2^{(i-1)}}$$
$$H_3^{(i)} = \mathrm{d} \div H_{3^{(i-1)}}$$
$$H_4^{(i)} = e \div H_{4^{(i-1)}}$$
$$H_5^{(i)} = f \div H_{5^{(i-1)}}$$
$$H_6^{(i)} = g \div H_{6^{(i-1)}}$$
$$H_7^{(i)} = h \div H_{7^{(i-1)}}$$

Output:

Output is obtained after all $M^{(N)}$ 512 bit blocks are processed. After all the processing steps are done a number of N times, you will get 256 bit message digest for M messages, namely:

$$H_0^{(N)}||H_1^{(N)}||\,H_2^{(N)}||H_3^{(N)}||H_4^{(N)}||H_6^{(N)}||H_7^{(N)}$$

4.RESULTS AND SIMULATION

4.1 Workflow of Cloud CRM with Proposed Scheme

The workflow model for proposed architecture of private cloud system can be visualized considering a particular example. In this case, a bill tracking system has been taken which would be implemented on proposed architecture of the private cloud for enterprise domains. The connectivity of the different roles assigned will be shown by various screenshots. The proposed scheme will precisely depict the scenario as proposed above. This proposed scheme defines the process flow for material, service bills and job requisitions. It defines the information to be taken in the workflow management through process owner and the participations.

4.1.1 Material related bills CRM Module (Bill Tracking System)

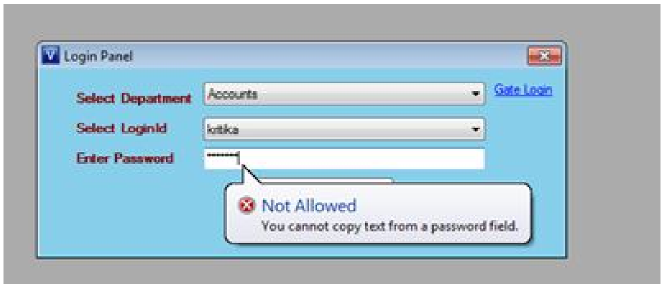
The process for material related bills where the store keeper mention IMR number on bills, will be the serial number of Item Requisition No.(IMR) register and further sends the bills to store immediately. In case of no PO is selected, system should escalate it to Purchase and that bill should appear in a separate menu to view all pending bills without PO. Store should send no PO bills to purchasing processing. If material is rejected by the user department, Bill Tracking Module BTM will have the option of rejection with necessary comments from store department .There will be no further routing for rejected items. In this architecture using RBACS and automation itself calculate the threshold value:

a) Accept bills
b) Change invoice Amount /Invoice amount
c) Concern person id(Employee id)
d) Invoice to Purchase/Accounts

4.2 Risk Evaluating Threshold

Threshold is termed as attribute values which hold the accelerated hierarchical structure dimensions in workflow model on which user will exercise or entertain its rights and execute actions vide a roles and objects delegated to it. However, the workflow reciprocates its approach and res-judicata. In this mechanism the said scheme will emphases on the Role Based Access Control Security (RBACS) to its maximum risk dimension. However, in this workflow, only three roles have been used i.e. Store, Purchase, Accounts. Four various roles assigned by system or super users are Store, Purchase and Accounts and Service generator. The windows for various threshold based on various roles to calculate the risk are implemented. The flow of different Serial IMR-NOs can be visualized through these windows. These windows are as follows:

1. Super User Login: as Administrator
2. Department Control Under The Super User
3. Supplier Control Under Super User
4. User Control & Role Control Under Super User
5. Role Based Login Using H#Abe and Session Management
6. Login as Store
7. Threshold Calculation for Store
8. Threshold Calculation for Stores To Purchase
9. Bill Rejected at The Stores
10. Purchase Login using H#ABE And RBACS
11. Invalid Session Credentials to Access Denied by RBACS Workflow for Purchase Whereas Impuissant or Delegation Model as per the RBACS enable masking and Descendent Control Access Based on Roles for Store Also Using Purchase Role.
12. Accepting Bill by Purchase and Pass By Store On Roles Respective of RBACS.
13. Acceptance of Bill At Purchase And Forwarded To Accounts While Calculating Thresholds.
14. Bill Rejected By Purchase.
15. Account Bill
16. Rejection In Account
17. Complete BTM Process In Accounts.
18. Simulation Module For BTM Process



Select Department: Accounts
Select LoginId: kritika
Enter Password: ********

Not Allowed
You cannot copy text from a password field.

Figure 7: Evaluating user credentials by H#ABE the proposed scheme.

Result Analysis

The maximum threshold value has been computed for different Serial IMR-NO for different roles i.e. (store, purchase and account). The required data has been obtained from the concerned workflow:-

Table 1: Workflow matrix depicting store role based on actions to calculate the threshold value

Sno	IMR_NO	Rejection	VIMR_NO	INV_NO	AMT	PO_NO	INV_TO_PUR	MAX. THRESHOLD
1	19-20-175	NO	1	2	3	4	0	10
2	19-20-174	NO	1	2	3	4	0	10
3	19-20-177	NO	1	2	3	4	0	10
4	19-20-172	NO	1	2	3	4	0	10
5	19-20-170	NO	1	2	3	4	0	10

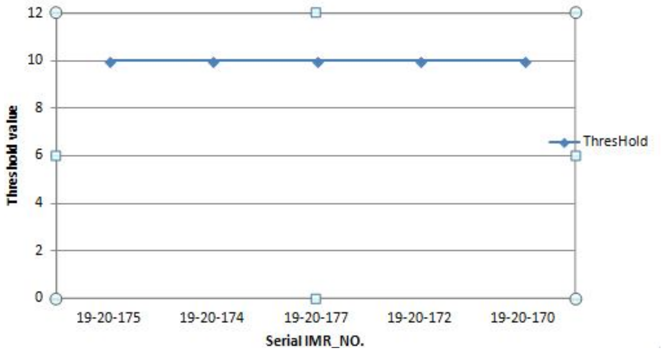


Figure 8: Relation between serial IMR NO. and threshold assignment. For Cloud Module (Stores) when No Rejections of Bill using Maximum Threshold

Table 2: Workflow matrix depicting store role based on actions to calculate the threshold value; In case of Rejection (store).

Sno.	IMR_NO	Rejection	VIMR_NO	INV_NO	AMOUNT	PO_NO	INV_TO_PUR	MAXTHRESHOLD
1	19-20-170	NO	1	2	3	4	0	10
2	19-20-172	YES	0	0	0	0	0	10
3	19-20-177	NO	1	2	3	4	0	10
4	19-20-174	NO	1	2	3	4	0	10
5	19-20-175	NO	1	2	3	4	0	10

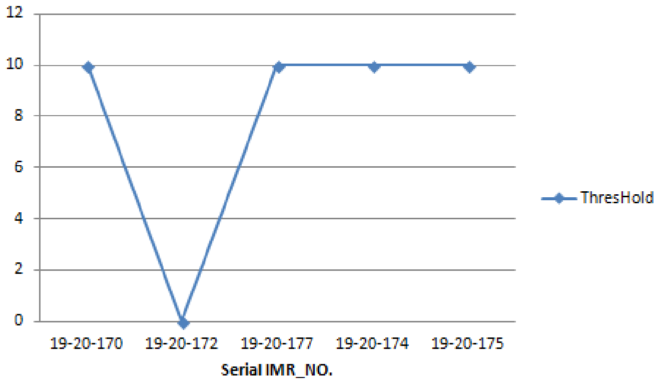


Figure 9: Relation between Serial IMR NO and Threshold value for module 1(store). In case of rejection (store)

Table 3: Workflow matrix depicting store role based on actions to calculate the Threshold value; In case of rejection and pending (Purchase).

Sno	IMR_NO	Rej.	A/C.B	AMT	PERSON_ID	INV_TO_ACC	MAX - THRESHOLD
1	19-20-170	NO	6	0	0	0	40
2	19-20-172	NO	0	0	0	0	40
3	19-20-177	NO	6	0	0	0	40
4	19-20-174	NO	0	0	0	0	40
5	19-20-175	NO	6	0	0	0	40

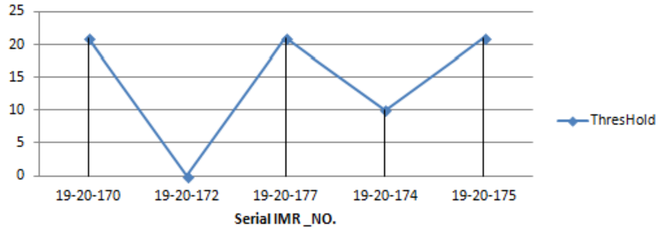


Figure 10: Relation between Serial IMR NO. and Threshold value for module 2 In case of rejection and pending (Purchase)

Table 4: Workflow matrix depicting store role based on actions to calculate the threshold value; In case of rejection and pending (Purchase).

Sno	IMR_NO	Rejection	A/B Bill	Amt	PERSON_ID	INV_TO_ACC	MAXTHRESHOLD
1	19-20-170	NO	6	7	8	9	40
2	19-20-172	NO	0	0	0	0	40
3	19-20-177	Yes	0	0	0	0	40
4	19-20-174	NO	0	0	0	0	40
5	19-20-175	NO	6	7	8	9	40

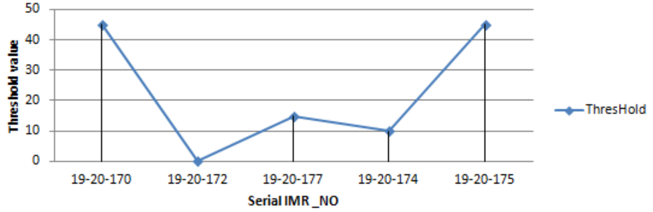


Figure 11: Relation between Serial IMR NO and Threshold value in case of rejection and pending (Purchase)

5.CONCLUSION

New algorithm H #ABE and RBACS has been proposed for implementation of security in the private cloud for authentication and authorization of the user. This model is more secured from the risk point of view based on CRM models over the cloud resources. The implementation of view proposed secured private model has been evaluated using simulation. The simulation results are obtained for different risk threshold values based on roles.

REFERENCES

1. M., Durairaj & Kannan, P.. (2014). **A Study On Virtualization Techniques And Challenges In Cloud Computing.** *International Journal Of Scientific & Technology Research* 2277-8616. 3. 147-151.
2. Komarek, Ales & Pavlik, Jakub & Sobeslav, Vladimir. (2017). **Performance Analysis of Cloud Computing Infrastructure.**
3. Barton, Thomas. (2020). **Cloud Computing/Anything as a Service (XaaS).** 10.3139/9783446464353.002.
4. Alam, Tanweer. (2020). **Cloud Computing and its role in the Information Technology.** *IAIC Transactions on Sustainable Digital Innovation (ITSDI).* 1. 108-115. 10.34306/itsdi.v1i2.103.
5. Parida, Sasmita & Nayak, Suvendu. (2013). **An algorithm that earning users' trust on cloud. 2013** *5th International Conference on Advanced Computing,* ICoAC 2013. 576-584. 10.1109/ICoAC.2013.6922015.
6. Zdraveski, Dejan & Janeska, Margarita & Taleska,. (2020). **EVALUATING CLOUD COMPUTING SERVICES.**
7. Brandão, Pedro. (2020). **Cloud Computing Security**.
8. Liu, Jing & Zhang, Liang-Jie & Hu, Bo & He, Keqing. (2012). CCRA: **Cloud Computing Reference Architecture. Proceedings -** *2012 IEEE 9th International Conference on Services Computing,* SCC 2012. 657-665. 10.1109/SCC.2012.110.
9. Sinchana, M. & Savithramma, R. (2020). **Survey on Cloud Computing Security.** 10.1007 / 978 – 981 – 15 – 2043 - 3_1.
10. Rodgers, Talia. (2018). **Digital Resources. Performance Research.** 23. 296-297. 10.1080 / 13528165.2018 1511035.
11. Chibesakunda, Mwelwa. (2020). **Public Key Cryptography: Digital Certificates**: Study on Attribute Certificates.
12. Gu, Dongxiao & Yang, Xuejie & Deng, Shuyuan & Wang, Xiaoyu & Wu, Jiao & Guo, Jingjing. (2019). **Tracking knowledge evolution in cloud healthcare literature: a cybermetrics study (Preprint).** 10.2196/preprints.15142.
13. Ahn, Gail-Joon. (2016). **Discretionary Access Control.** 10.1007/978-1-4899-7993-3_135-2.
14. Formisano, C. & Bonelli, L. & Balraj, K.R. & Shulman-Peleg, Alexandra. (2013). **Cloud access control mechanisms.** 94-108 10 4018 / 978- 1 – 4666 – 3934 - 8.ch007.
15. Afonin, Sergey. (2016). **Performance evaluation of a rule-based access control framework.** 1414-1418. 10.1109/MIPRO.2016.7522361.
16. Alkhorem, Azan. (2019). **Attribute-based access control (ABAC).**
17. Al-Dahhan, Ruqayah & Shi, Qi & Lee, & Kifayat,. (2019). **Survey on Revocation in Ciphertext-Policy Attribute-Based Encryption. Sensors.** 19. 1695. 10.3390/s19071695.
18. Zheng, Hua & Zhang, Xi & Yang, Qi. (2017). **An Improved Ciphertext-Policy Attribute-Based Encryption** Scheme. 400-411. 10.1007 / 978-3-319-52015-5_41.
19. Hohenberger, Susan & Waters, Brent. (2014). **Online/Offline Attribute-Based Encryption. Public Key Cryptography** (PKC'14),. 8383. 10.1007 / 978-3-642-54631-0_17.
20. Vanitha, M. & Kavitha, C.. (2015). **Client based security for cloud CRM application.** 10. 37414-37420.
21. O'Regan, Gerard. (2020). **Cryptography**. 10.1007/978-3-030-34209-8_10.
22. Hellwig, Daniel & Karlic, Goran & Huchzermeier, Arnd. (2020). **Blockchain Cryptography: Part 1**. 10.1007/978-3-030-40142-9_6.
23. Hellwig, Daniel & Karlic, Goran & Huchzermeier, Arnd. (2020). **Blockchain Cryptography: Part 2.** 10.1007/978-3-030-40142-9_7.
24. Stein, Scott. (1984). **The ISO connectionless transport standards. Computer Communication Review - CCR**. 14. 18-20. 10.1145/1024908.1024911.

25. Kantzavelou, Ioanna & Patel, Ahmed. (1995). **Issues of Attack in Distributed Systems - A Generic Attack Model.** 10.1007/978-0-387-34943-5_1.
26. Prasad, Ajay & Kaushik, Keshav. (2019**). Digital Signatures.**
27. Duc, Toan & Bui, Hong. (2017). **Building Background to the Elgamal Algorithm**. *International Journal of Mathematical Sciences and Computing.* 3. 39-49. 10.5815/ijmsc.2017.03.04.
28. Stallings, William. (2013). **Digital Signature Algorithms**. *Cryptologia*. 37. 10 . 1080 / 01611194 . 2013.797044.
29. Syed.Karimunnisa & Dr.Vijaya Sri Kompalli. (2019) **Cloud Computing: Review on Recent Research Progress and Issues.** *International Journal of Advanced Trends in Computer Science and Engineering.* Volume 8 No. 2 (2019). 216 **-** 223 10.30534/ijatcse/2019/36822019.
30. Dhananjaya. V & Dr. Balasubramani. R. (2020) **Design and Analysis of high security ECC based Cryptography by Holomorphic and data storage in Cloud.** *International Journal of Advanced Trends in Computer Science and Engineering.* Volume 8 No. 2 (2020). 1720 – 1728 10.30534/ijatcse/2020/124922020.
31. Remya Chandran and Dr.A.Sasi Kumar. (2020) **Efficient Cloud Authentication Scheme using Single Sign-On Nature in Hands with Branca Strategy.** *International Journal of Advanced Trends in Computer Science and Engineering.* Volume 9 No.2 (2020). 1800 – 1807 10.30534/ijatcse/2020/137922020.