I. J. Computer Network and Information Security, **2016, 6, 52-58**
Published Online June 2016 in MECS (http://www.mecs-press.org/)
DOI: 10.5815/ijcnis.2016.06.06

Application of Combinatory Mechanism on RTOS UFS ACM for Risk Optimization

Padma Lochan Pradhan
Dept. of CSE, Central Institute of Technology, CSVTU, Naya Raipur, CG, India.
E-mail: citrprcs@rediffmail.com

Abstract—At this fastest growing of information age, there is a rapid change of business, resources and technology, mean while increasing the requirement of electronic commerce for the sophisticated societies in around the globe. During this process increasing the uncertainty, un order, un safe and un setup due to un authorize users, hackers is a great issue for down time of communication system. Our objective to determine and resolve these uncertainty problems to develop this proposed combinatory ACM to optimize the accessibilities of resources and maximize quality of services for all the time and every time to co-op with pervasive, ubiquitous & autonomy system. The relation, function, operation, maintaince and services are the vital role for all aspect of multiples societies, business and technologies in around the cloud. Meanwhile, it will be more accountable for performance, fault tolerance, throughput, benchmarking on any computational services for all the time. We have to make more simplification, unification and step by step normalization by applying permutation & combinations on UFS ACM mechanism on distributed object oriented system on multi-dimensional work culture. This access control mechanism is preventing, detecting, correcting, verification and validation of the UFS ACM in background process of RTOS.

Index Terms—Access Control Mechanism, Unix File System, Real Time Operating System, Prevent, detect & correct(PDC), Risk Assessment.

I. Introduction

The complex operating system is a collection of hardware, software & application that manages system resources and provides common services for resources, program, application & users. The operating system is an essential component of the system software (shell, file & kernel) in computer system. The high level language (application programs) usually requires an operating system to function. The time-sharing operating systems schedule & reschedule tasks for efficient use of the internal utilities that may also include auditing system software for resource & cost allocation of processor and memory time, mass storage, printing and other resources [7], [15], [18], [22], [24].

There are various kinds of preventive control available and implemented on operating system to protect our IT assets for external & internal hacker. The PDC model & Mechanism traditionally prevent the core components of OS. The processor & memory is the core component of any type operating system. The processor and kernel is fully functional dependency on each other, but file and shell is the communication components of the RTOS. We can improve the performance of OS by updating the kernel time to time. Kernel is the Nucleus of the operating systems [7], [15], [22], [24].

II. Architecture of the Operating System

This access control mechanism is a prerequzise preventive control. The objectives of an access control system are often described in terms of protecting system resources against inappropriate or undesired user access. From a business perspective, this objective could just as well be described in terms of the optimal sharing of information. After all, the main objective of IT is to make information available to users and applications. A greater degree of sharing may get in the way of resource protection; in reality, a well-managed and effective access control system actually facilitates sharing. A sufficiently fine-grained access control mechanism can enable selective sharing of information where in its absence, sharing may be considered too risky altogether [7], [15], [18].

The access control mechanism is the process of mediating each and every request to system resources, application and data maintained by the real time operating system and determining whether the request should be created, approve, granted or denied as per top management policy. The AC mechanism, management and decision is enforced by implementing regulations established by a security policy [7], [15], [18].

The access control is concerned with determining the allowed activities of legitimate users, mediating every attempt by a user to access a resource in the system. The complex information technology (IT) infrastructure can implement access control systems in many places and at different levels. The real time operating systems use access control to protect files and directories. The database management systems (DBMS) apply access control to regulate access to tables and views. The most commercially available application systems implement access control, often independent of the operating systems programing and DBMSs on which they are installed [7], [15], [18].

The access control is the traditional centre of gravity of computer security. It is where security engineering meets computer engineering. This function is to preventive control which principals (persons, processes, machines, etc) have access to which resources in the system which files they can read, which programs they can execute, how they share data with other principals, and so on. The preventive access control works at a number of levels, as shown in Figure 1. and described in a systematic ways as follows[7], [15], [18]. The access control mechanisms, which the user apply at the RTOS & application level, may express a very rich and complex security policy. A modern online business could assign staff to one of dozens of different rights & roles, each of which could initiate some subset of several hundred possible transactions in the system [1-2], [10].

The real time operating system access controls will usually relay on hardware features provided by the processor, memory, and kernel or by associated memory management hardware. These system controls which memory addresses a given process can access [7], [15], [18].

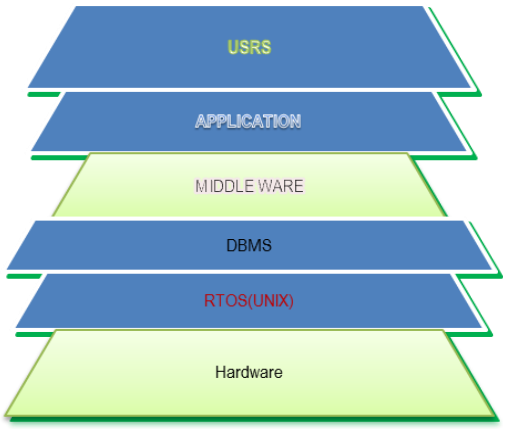


Fig.1. Multi-tire Architecture of Real time OS

III. Data Collection

There are many more preventive access control (PAC) mechanisms are define, design and developed as per requirement of the secure computing to achieve the highest level of business objective. There are some few mechanism to be developed based on Unix server and real time operating system programming. The Unix file system (UFS) have to be develop as per business requirement for all the time and every time in around the globe (pervasive & ubiquitous computing) [7], [15], [18].

Table 1. Sample of RTOS UFS data

SN	*Inode*	*Subjects*	*Link*	*Owner*	*Group*	*Byte*	*Date of File creation time stamp*	*Objects(UFS)*	*Remarks*	*RISK*
1	876345	drw-r--r--	1	e-com	Usr	1233	Jun 7 10:41	/etc/system	Directory file	High Risk
2	123450	drwx------	2	e-com	Usr	1234	July 6 12.23	/etc/host	Directory file	High Risk
3	098712	Dr-xr-xr--	1	sam	Staff	1024	Nov 2 00:10	/etc/ssh/sshd_config	Directory file	Low Risk
4	908761	drwx--	1	sam	SA	1024	Dec 2 00.10	/etc/service	Directory file	High Risk
5	012398	drwx--	1	USR	GRP	1024	July 6 12.23	/var/adm/message	Directory file	High Risk
6	908123	-rwxrwxrwx	3	e-com	Staff		Nov 2 00.10	Test.html	Ordinary file	High Risk
7	786540	Dr-x------	6	e-com	Usr	512	May 3 12:31	Public	Directory file	High Risk
8	765123	drwxr-xr---	1	sam	Staff	1024	Nov 2 00:10	Yourfile	Directory file	Medium Risk
9	451230	crw-------	1	root	System	0	Nov 2 00:10	/dev/rsd0a	Character Spl File	Medium Risk
10	564321	brw-rw----	1	root	System	0	Nov 2 00:10	/dev/sd0a	Block Spl File	High Risk
11	340999	lrwxrwxrwx	1	ram	Staff	8	May 3 12:31	Zn.dat->gold.dat	Link File	High Risk

IV. Problems Statement

There are many more issue regarding the resource allocation and distribution for USR, Business Owner over a multiple ROLE, RIGHT AND RESOURCES at various level of management (Top, Medium & Low).

As per data collection and analysis, the preventive control is not available of the current RT UFS ACM. The corrective action and reaction on file system, application & resources is a big concern on this current security age. The multiple Relation, Function, Operation and Services are happening over a multiple clients, business, application and resources on a complex heterogeneous IT

infrastructure on mobile, web and cloud computing. Therefore, resource conflicts are the biggest issue over a complex network, platform and user application. Therefore, there is no balance ratio among the Business, Technology & Resources. The PDC is applying through Unix access control mechanism over a UFS.

V. Research Methodology

This research work contributes to the define, design, development of an optimization and normalization model that aims and objective to determine the optimal cost, time and maximize the QoS to be implemented into the security model & mechanisms deciding on the measure components of UFS ACM based on proposed combinatory method as follows [7], [15], [18], [22], [24].

A. Define

The Top managements need to define, design & develop the policy, procedure to run the smooth business. The lower management need to operation and services for all the time and any time, but middle management have to co-ordinates and interact in between top & lower management.

This proposed combinatory ACM security controls for risk optimization can be design and develop to protect against given types of threats, unauthorized user & uncertainty. These combinatory ACM may range from simple to complex measures and usually involve with system architectures, engineering disciplines and security packages with a mix culture of hardware, software, application and firmware. All of these measures should work together to achieve the secure critical and sensitive data, information, and IT system functions. These security controls can be grouped into high, medium & low according to primary purpose the organization.

Table 2(a). Representation of Octal, Binary & fs Permission on UFS.

Octal digital	*Binary representation*	*Permission*	*Role*
0	000	None/Blank (-)	No body access (No Risk)
1	001	execute only(x)	Other-World (X) (No Risk)
2	010	write only(w)	Owner –USR(SU)(Risk)
3	011	write and execute(wx)	Owner –USR(Risk)
4	100	read only(r)	Owner –USR(Top Mgmt)(Less Risk)
5	101	read and execute (r x)	Owner –USR(Top Mgmt)(Less Risk)
6	110	read and write (r w)	Owner –USR(Developer) (Risk)
7	111	read, write, and execute (r w x)	Developer (Business USR) (G)(Risk)

Table 2(b). Interpreting Object Modes Pattern (Basic Architecture of Unix file system)

FIELD1										
	File Type	***Usr Access***			***Group Access***			***Other Access***		
Arttributes	-	**r**	**w**	**x**	**r**	**-**	**x**	**r**	**-**	**x**
Position	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
Read Access		•			•			•		
Write Access			•			•			•	
Execute Access				•			•			•
Full Access	**d**	**r**					•			•

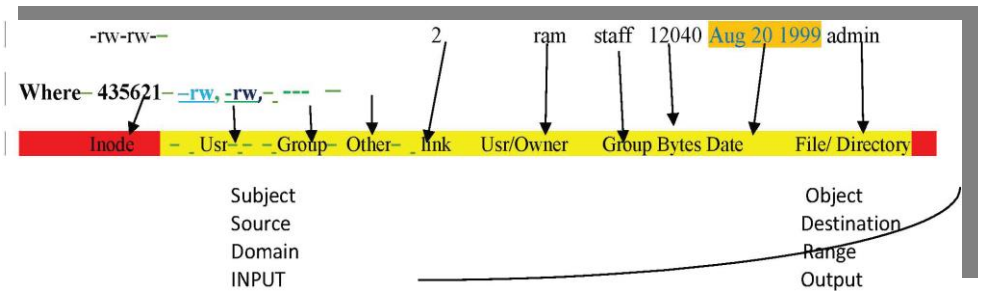


Fig.2. Internal UFS Structure

B. Design

We have to move forward to finding alternate solution and algorithm for Risk optimization based on combinatory theory. This scalable complex ACM definitely will be resolve our risk and security issue on complex real time system for multiple clients and location for all the times. We have to propose these eight objects is a set of elements as follows: {B, R, W, X, RW, RX, WX, RWX}. This eights objects ordering (Order & un order) apply to our designing methodology and as per top management decision and requirement of the current business & resources.

Prove that the set {0, 1, 2, 3, 4, 5, 6, 7} = {B, R, W, X, RW, RX, WX, RWX}. is a finite combinatory order& under composition. Whereas, S = { 0, 1, 2, 3, 4, 5, 6, 7} = {b, r, w, x, rw, rx, wx, rwx}.

Table 3. Allocation of UFS Attributes on ACM as per required level of Mgmt.

Blank	***X***	***W***	***WX***	***R***	***RX***	***RW***	***RWX***
000	001	010	011	100	101	110	111
0	1	2	3	4	5	6	7
Nobody	Any body	Developer	Developer	Top Mgmt	Top Mgmt	Developer	Developer

C. *Development*

Let us consider the set [b, r, w, x]. In how many ways (possibilities) can we select two of these letters (repetition is not allowed). There are two types of permutation & combinations as defined below [3], [11].

Orders (Certainty) & Unordered (Uncertainty)

Solution: Let us consider n is the number of object = 4 & r = 2, 3, 4 ways and hence the number of ways of selecting two letters from four letters is P(4,2) = 4!/(4-2)! = 4 x 3 x 2 x 1/ 2 x 1 = 12 ways/ possibilities are happening.

Order (Certainty)

Table 4. Order Mechanism

br	**rb**	**wb**	**xb**
bw	Rw	wr	xr
bx	Rx	wx	xw

Note: this mechanism is more secure but not business oriented

Unordered (Uncertainty)

An un order selections or arrangement of objects from a set of n objects is called a permutation (r-combination of n objects). It is denoted by C (n, r). If all the elements are distinct and non-repetition is not allowed, then by applying sum rule, it can be shown that:

C(n, r) = n (n-1) (n-2) ……. (n-r + 1) = n ! / r! (n-r) ! (un order-combination, Random)

We have to consider the set of eight objects {0, 1, 2, 3, 4, 5, 6, 7} = {0, X, W, WX, R, RX, RW, RWX)

Where n = 8, and r = 2, 3, 4, 5, 6, 7 and so on as shown in Tabular Form (Bernard, K. 2007).

Therefore we have to adjust our combination and permutations formula to optimize it by how many ways the objects (n=8) could be in order (because we are not interested in their order any more): $\mathbf{nCr} = \frac{n!}{r!(n-r)!}$, where n>=1. Where n is the number of things to choose from, and we choose r of them (No repetition, un order, order may be does not matter) [3], [11].

Table 5. Un Order Permutation (Relation & Function) $=$ **nCr** (n number of possibilities)

C(8,1) = 8 ! /1! (8-1)!= 8 x 7 x 6 x 5 x 4 x 3 x 2 x 1/1!(7x6x5x4 x 3 x 2 x 1) = 8 ways possibilities of ACM

C(8,2) = 8 ! /2! (8-2)! = 8 x 7 x 6 x 5 x 4 x 3 x 2 x 1/ 2! (6 x 5 x 4 x 3 x 2 x 1) = 28 ways possibilities of ACM

C(8,3) = 8 ! /3! (8-3)!= 8 x 7 x 6 x 5 x 4 x 3 x 2 x 1/3! (5 x 4 x 3 x 2 x 1) = 56 ways possibilities of ACM

C(8,4) = 8 ! /4! (8-4)!= 8 x 7 x 6 x 5 x 4 x 3 x 2 x 1/ 4!(4 x 3 x 2 x 1) = 70 ways possibilities of ACM

C(8,5) = 8 ! / 5!(8-5)!= 8 x 7 x 6 x 5 x 4 x 3 x 2 x 1/ 5! (3 x 2 x 1) = 56 ways possibilities of ACM

C(8,6) = 8 ! / 6!(8-6)!= 8 x 7 x 6 x 5 x 4 x 3 x 2 x 1/ 6!x2! = = 28 ways possibilities of ACM

C(8,7) = 8 ! / 7!(8-7)!= 8 x 7 x 6 x 5 x 4 x 3 x 2 x 1/ 7!1 = = 8 ways possibilities of ACM

Note: We have to make decision on better, faster and safer security and lower risk on distributed real time object oriented system.

P (4, 2) = 4!/ 2!(4-2)! = 6 ways/ possibilities

Table 6. High Risk

br	rw	wx
bw	rx	bx

Note: This mechanism is inefficient

Let us consider the set [b, r, w, x]. In how many ways (possibilities) can we select three of these letters (repetition is not allowed). P(4,3)= 4!/(4-3)!= 24

Table 7. Medium Risk (Alpha Form)

brw	brx	bwx	rxb	rbx	**xwr**
rwx	wxr	xrw	xrb	rxw	xwb
wxb	wrb	wxr	wbr	rbw	bwr
rwb	wrx	xbr	xbw	wbx	bxr

Note: this mechanism is more secure as well as business oriented for all the time & every time on RTOS UFS ACM. We have to make more simplification and unification for our secure business R=4, w=2, x=1, b=0

Table 8. Medium Risk (Octal Form)

6	5	3	5	5	7
7	7	7	5	7	3
3	6	7	6	6	6
6	7	5	3	3	5

NORMALIZED FORM FOR BUSINESS USR

Table 9. Medium Risk 1NF

SU	G	X	G	G	SU
SU	SU	SU	G	SU	X
X	X	SU	X	SU	SU
SU	SU	G	SU	X	G

Table 10. Medium Risk 2NF

SU	X	X	B	B	SU
SU	SU	SU	B	SU	X
X	X	SU	X	SU	SU
SU	SU	B	SU	X	B

Now we have to eliminate Group=G, only Super USR & End USR is required for our business

NORMALIZED FORM FOR BUSINESS USR

Table 11. Low Risk 3NF

X	X	X	X	X	X
X	X	X	X	X	X
X	X	X	X	X	X
X	X	X	X	X	X

D. Deployment of Combinatory UFS ACM

The top management have to audit and control this proposed ACM tools and parameters by help of system tools or manually. This order and un order ACM will be more reliable, scalable, high available, accountable & actionable for performance, fault tolerance, throughput, bench marking and risk optimization on any computational services for all the time. We have to make more simplification, unification and step by step normalization by applying combinatory UFS ACM mechanism based on distributed object oriented system on multi-dimensional work culture. We have to test and implement this combinatory ACM on RTOS Unix based platform for our secure, reliable, accountability and high availability for multiple function on multiple client, application, business and resources anytime and anywhere in around the globe. We have to audit the ACM, performance, benchmarking by the following method, when millions of users accessing the Web Portal in around the globe for all the time and any time [7], [15], [18].

E. Experimental Test Mechanism on Real time UNIX Machine (Verification)

We have to apply real time experiment on UFS ACM as follow

We can further apply the preventive access control (PAC) mechanism to optimize risk (Higher to Lower)

Action on ACM (Apply Preventive Control)

Remarks: USR or Business Owner could able to do READ, WRITE & EXECUTE, But Other Group & Other cannot do any things. That's why blank ----- is there on subject)

1NF: First Normal Form

pl@pl-HP-15-Notebook- 1NF
PC:~/log$ chmod 701 menu*.* (ACTION)
pl@pl-HP-15-Notebook-PC:~/log$ ls –
SUBJECT OBJECT
drwxr-xr-x 3 pl pl 4096 2014-11-11 12:26 kamal
-rwx-----x 1 pl pl 727 2014-11-08 16:02 menu1.sh
-rwx-----x 1 pl pl 461 2014-11-08 16:17 menu4.sh
-rwx-----x 1 pl pl 547 2014-11-08 16:37 menu5.sh
-rwx-----x 1 pl pl 505 2014-11-09 16:52 menu.sh

Action on UFS ACM
pl@pl-HP-15-Notebook-PC:~/log$ chmod 777 menu*.sh
pl@pl-HP-15-Notebook-PC:~/log$ ls –l

#chmod 777 menu*.sh
(ACTION) EVERY ONE R, W & X on UFS ACM
#Ls -l(Review the Reaction) HIGH RISK
-rwxrwxrwx 1 pl pl 727 2014-11-08 16:02 menu1.sh
-rwxrwxrwx 1 pl pl 461 2014-11-08 16:17 menu4.sh
-rwxrwxrwx l pl pl 547 2014-11-08 16:37 menu5.sh
-rwxrwxrwx 1 pl pl 505 2014-11-09 16:52 menu.sh

#chmod 666 menu*.sh
(ACTION) EVERY ONE R & W on UFS ACM
#Ls –l(Review the Reaction) HIGH RISK
-rw-rw-rw- 1 pl pl 727 2014-11-08 16:02 menu1.sh
-rw-rw-rw- 1 pl pl 461 2014-11-08 16:17 menu4.sh
-rw-rw-rw- 1 pl pl 547 2014-11-08 16:37 menu5.sh
-rw-rw-rw- 1 pl pl 505 2014-11-09 16:52 menu.sh

#chmod 555 menu*.sh
(ACTION) EVERY ONE R & X on UFS ACM
#Ls -l(Review the Reaction) MIDDUEM RISK
r-xr-xr-x 1 pl pl 727 2014-11-08 16:02 menu1.sh
-r-xr-xr-x 1 pl pl 461 2014-11-08 16:17 menu4.sh
-r-xr-xr-x 1 pl pl 547 2014-11-08 16:37 menu5.sh
-r-xr-xr-x 1 pl pl 505 2014-11-09 16:52 menu.sh

2NF: 2nd Normal Form

#chmod 444 menu*.sh (ACTION)
#Ls -l(Review the Reaction) MIDUM RISK
r--r--r-- 1 pl pl 727 2014-11-08 16:02 menu1.sh
-r--r--r-- 1 pl pl 461 2014-11-08 16:17 menu4.sh
-r--r--r-- 1 pl pl 547 2014-11-08 16:37 menu5.sh
-r--r--r-- 1 pl pl 505 2014-11-09 16:52 menu.sh

#chmod 333 menu*.sh (ACTION)
EVERY ONE W & X on UFS ACM
#Ls -l(Review the Reaction)) HIGH RISK
-wx-wx-wx 1 pl pl 727 2014-11-08 16:02 menu1.sh
--wx-wx-wx 1 pl pl 461 2014-11-08 16:17 menu4.sh
--wx-wx-wx 1 pl pl 547 2014-11-08 16:37 menu5.sh
--wx-wx-wx 1 pl pl 505 2014-11-09 16:52 menu.sh

#chmod 222 menu*.sh (ACTION)
ANY ONE WRITE THE UFS ACM- No Security
#Ls -l(Review the Reaction)) HIGH RISK
-w--w--w- 1 pl pl 727 2014-11-08 16:02 menu1.sh
--w--w--w- 1 pl pl 461 2014-11-08 16:17 menu4.sh
--w--w--w- 1 pl pl 547 2014-11-08 16:37 menu5.sh
--w--w--w- 1 pl pl 505 2014-11-09 16:52 menu.sh

Now Proved the 3NF: 3rd Normal Form
#chmod 111 menu*.sh
(ACTION) EVERY ONE EXECUTE THE UFS ACM (Pervasive, Umbiquintus & Autonomy system satisfying to all the time & every time.
#Ls -l(Review the Reaction) NO RISK & Secure
---x--x--x 1 pl pl 727 2014-11-08 16:02 menu1.sh
---x--x--x 1 pl pl 461 2014-11-08 16:17 menu4.sh
---x--x--x 1 pl pl 547 2014-11-08 16:37 menu5.sh
---x--x--x 1 pl pl 505 2014-11-09 16:52 menu.sh

#chmod 000 menu*.sh
(ACTION)NO ONE ACCESS THE UFS ACM- High Security
#Ls -l(Review the Reaction)
NO RISK & Highly Secure
---------- 1 pl pl 727 2014-11-08 16:02 menu1.sh
---------- 1 pl pl 461 2014-11-08 16:17 menu4.sh
---------- 1 pl pl 547 2014-11-08 16:37 menu5.sh
---------- 1 pl pl 505 2014-11-09 16:52 menu.sh

Remarks: No boby access on the Subject,
That's why blank ----------

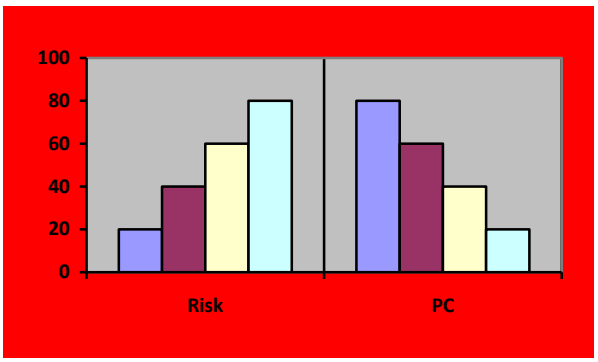
VI. Out Comes on ACM Test

Graphical Representation:

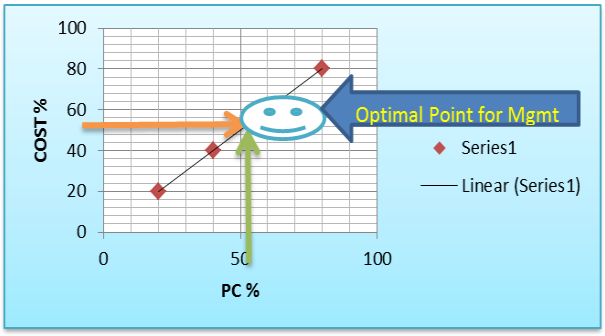
If Preventive Control is more Risk is less [UFS ACM =k.1/R]. As per FUZZ'S LAW=> ACM= PC, PC=k. Cost: Where C is a Cost.

Table 12. Risk Assessment & Review

PC %	*RISK%*	*CATEGORY(HML)*	*Mgmt Action*	*REMARKS*
20	80	High Risk	Top Mgmt	Low Cost
40	60	High Risk	Top Mgmt	Low Cost
60	40	Medium Risk	Middle Mgmt	Medium Cost (Optimum Level)
80	20	Low Risk	Operation & Services	High Cost



Graph 1. Risk Assessment Verses PC



Graph 2. PC Verses Cost

Therefore, the top management have to decide whether to accept expected losses or to invest into PC (technical security mechanisms=ACM) in order to optimize the risk. The preventive control is inversely proportional to the risk & meanwhile, the PC is directly proportional to the cost. This preventive control is directly proportional to risk mitigation & mitigation is directly proportional to TQM. Furthermore, this mechanism optimize the cost, time & resources is supposed to optimize the system risks.

Note: Preventive control is directly proportional to the cost

VII. Results

Our proposed Combinatory ACM method & mechanism is much more observable, accountable, measurable and actionable in a continuous manner for any time and all the time. This is not only Combinatory order ACM but also dynamic and distributed mechanism for pervasive and ubiquitous computing is much more self-adaptive, resilient and autonomy system. The Security (PC) is inversely proportional to the Risk. This order-un and order ACM will be more accountable for

performance, fault tolerance, throughput, bench marking and risk optimization on any computational services for all the time. We have to make more simplification, unification and step by step normalization by applying Combinatory UFS ACM mechanism based on distributed object oriented system on multi-dimensional environment. We have to maximize the protection at optimal cost all the time, everywhere & every time in around the globe.

VIII. Conclusion

This Combinatory order and unordered ACM will be great helpful for decision support system (DSS) under uncertainty, unordered for better, faster and secure management for any RTOS computational services. This proposed Combinatory ACM Method, Model, and Mechanism & Control providing the accountability for individuals who are accessing sensitive data information on application, system software, server and network. This accountability is accomplished through RTOS that requires identification, authentication, authorization, accountability, non-repudiation, availability, reliability & integrity are availability in the system security. This model keeping balance among business, society, technology and resources at optimal cost. This model help to top management to improve the high BCP, DRP, Quality of Service, scalability, reliability and availability for all the time.

That's why this security engineering is practically working as process of risk optimization and decision support system, when the sub system under uncertain, un ordered, and unsafe for all the time and every time.

References

[1] A.K. Gupta, Management Information System. New Delhi, India: S Chand Publishing, 2012.
[2] Adrian Waller, "Editorial: Special issue on Identity Protection and Management," Journal of information security and application, 19, 2014.
[3] Bernard Kolman, Discrete Mathematical Structures. New Delhi: Person Education India, 2007.
[4] Brendan Jennings and Rolf Stadler, "Resource Management in Clouds: Survey and Research Challenges, "Journal of Network System Management, Springer, Springer Science+Business Media, New York, February 2014.
[5] Diogo A. B. Fernandes, "Security issues in cloud environments: a survey, Intl. Journal of Info. Security, Springer. 13:113–170, 2014.
[6] Danny Bradbury, "Can Security and Privacy Co-exist?," Danny Bradbury Infosecurity. Volume 8, Issue 6, Pages 33–35, November–December 2011.
[7] Das Sumitabh, UNIX System V UNIX Concept & Application. Delhi, India: Tata McGraw Hill, 2009.
[8] Dario Forte, "Security audits in mixed environments," Network Security, Vol.3, No. 3, pp. 17-19, March 2009.
[9] Hwang Kai, Advance Computer Architecture. New Delhi, India: Tata McGraw Hill, 2008.
[10] John R Vacca, Computer and Information Security Handbook. Burlington, MA: Morgan Kaufmann, 2009.
[11] Joe. L Matt, Discrete Mathematics for Scientist and Mathematician. New Delhi: Person Education India, 2008.
[12] John B. Kramer, The CISA Prep Guide. New Delhi, India: Wiley Publishing Inc, 2003.
[13] Mathew Nicho Shafaq, "Identifying Vulnerabilities of Advanced Persistent Threats: An Organizational Perspective, "International Journal of Information Security and Privacy, 8(1), 1-18, January-March 2014.
[14] Nasir Abbas, Memory-Type Control Charts for Monitoring the Process Dispersion, Quality and Reliability Engineering International. Wiley, 30, 623–632, 2014.
[15] O' Reilly, Essential of System Administration. O' Reilly Media: USA, 1995.
[16] Shon Harrish, CISSP Exam Study Guide. New Delhi, India: Dreamtech, 2002.
[17] Shon Harrish, Security Management Practices. New Delhi, India: Wiley Publishing Inc, 2002.
[18] Sun-Microsystems, UNIX Sun Solaris System Administration. USA.
[19] Tong xin and Ban Xiaofang, "A Hierarchical Information System Risk Evaluation Method Based on Asset Dependence Chain," Intl. Jour. of Info & Network Security, 3, 3, 2014.
[20] Tanenbaum, Computer Network. New Delhi: Person Education India, 2009.
[21] Tim Thomas, "A Mandatory Access Control Mechanism for the Unix file system," Motorola Inc., Microcomputer Division, IEEE, 1988.
[22] Tanenbaum, Operating System Design and Implementation. New Delhi: Person Education India, 2010.
[23] William Stalling, Cryptography and Network Security. New Delhi: Person India, 2006.
[24] William Stalling, Operating System Internals & Design Principle. New Delhi, India: Person India, 2009.
[25] Weber Ron, Information System Control & Audit. New Delhi: Person Education India, 2002.

Authors' Profiles

Pradhan Padma Lochan received M.Sc. (Physics with Electronics) from Sambalpur University in 1983 and M Tech in Computer Science in 2012 from Berhampuer University, India. He is interested in system security, cryptography, Real time operating system, system programming & risk mgmt. In around 20 years in IT industries and 10 years in academic & research in various capacity in IBM, Sun Micro system, Thomson (ISI) in USA as well as Indian Telephone Industries etc. Now, working as an Associate Professor (Computer Science & Engineering Dept.) in Central Institute Raipur, CG, India. Apart from this, he is a PG DBA and certified UNIX SUN Solaris expert.

How to cite this paper: Padma Lochan Pradhan,"Application of Combinatory Mechanism on RTOS UFS ACM for Risk Optimization", International Journal of Computer Network and Information Security(IJCNIS), Vol.8, No.6, pp.52-58, 2016.DOI: 10.5815/ijcnis.2016.06.06

