PROPOSED AES JAVA CODING DYNAMICALLY OPTIMIZING THE RISK ON OPERATING SYSTEM-I

1Dr. Prashant Kumar Patra & **2**Padma Lochan Pradhan
1.Dept. of CSE, College of Engineering & Technology, BPUT, Bhubaneswar-751003
2.Dept. of CSE, Sikha 'O' Anusandhan University. Bhubaneswar, Orissa, India

ABSTRACT:

The increased of the use of computer & communications system by IT industries has increased the risk of theft of proprietary information. Advanced Encryption Standard (Encryption) is a primary method of protecting system resources. AES is inversely proportional to the Risk(C=K/R) & mean while control is directly proportional to the quality of standard(S). AES Control will be optimize the risk as well as improve the IS standard. Control is directly proportional to mitigation & mitigation is directly proportional to standard. Encryption Key length(AES-E) is inversely proportional to the Risk. This paper contributes to the development of an optimization method that aims to determine the optimal cost to be invested into security method, model & mechanisms deciding on the measure component of operating system resources(i.e. Processor, Memory, Encryption). Further more, the method & mechanism optimize the cost, time & resources is supposed to reduce the system risks. Java based codeing would be update the value of Processor, Memory & Encryption key dynamically as per business requirement and availability of technology & resources. Proposed Java based program is going to be optimizing risk and maximizing the performance.

KEYWORD: PC-Preventive Control, DC-Corrective Control, CC-Corrective Control, CKM-Cryptographic Key Management. AES: Advanced Encryption Standard. CPU: Central Processing Unit.ROI-Return on Investment, SSH – Secure Shell, TCO-Total Cost Ownership, RM-Risk Mitigation.

Introduction: [3],[7],[8]

The machine is consists of millions of chips, each capable of testing a million keys per second, such machine could be test 2^56 key in 20 hours. It is easy to design a machine with a million parallel processors, each working independent of the others. The encryption key length size is depends Memory, Control, Arithmetic unit, Processor etc. to perform the functionality of the operating system. The operating system control is the process to address security weaknesses in operation systems by implementing the latest OS patches, hot fixes and updates and the procedures and policies to reduce attacks and system down time mean while increase the throughput of the system. Preventive control of the operating systems is the first step towards safeguarding systems from intrusion, workstations, applications, network and servers typically arrive from the vendor, installed with a multitude of development tools and utilities, which although beneficial to the user, also provide potential back-door access to the systems. Control of an operating system involves the removal of all non essential tools, utilities and other systems programmer options, any of which could be used to ease a hacker's path to our systems. The greatest difficulty in getting millions of routers and computers to work on a brute-force attack is convincing millions of computer owners to participate around the globe. We could ask politely, but that's time consuming and they might say no. We could try breaking into their machines, but

that's even more time consuming and we might get arrested, we could also use a computer virus and other hacking tools & scripts to spread the cracking program more efficiently over many computers as possible at a time. There fore , we need pre-planed prevention to safe guard the critical IT Infrastructure. The primary purpose of security policy is to inform those responsible for protecting assets such as hardware, software, and data of their obligations. Management establishes a security policy based on the risks it is willing to tolerate. The policy itself does not set goals, but serves as a bridge between management's goals and the technical implementation.
Security is a process, not a result. It is a process which is difficult to adopt under normal conditions; the problem is compounded when it spans several job and application running simultaneously under complex based web infrastructure which using million of user accessing the same piece of devices and information or data on the around the clock (24 x 7 x 52). All the system level security in the world is rendered useless by insecure web-applications. The converse is also true, programming best practices, such as always verifying user inputs are useless when the code is running on a server which hasn't been properly system programmed. Control is directly proportional to hardening or vice versa. Securing forward facing free BSD, GNU, SystemV based unix and Linux web servers can seem like a counting task, but it can be made much easier by breaking the process into manageable portions.[3],[7],[8].

Literature Survey: [4], [5], [16] & [17]

The technical literature survey in IS Security area is very critical & tedious task to collect the actual data and evidence in the real life. It is one of the on going process in a continuous manner. It is a very time consuming to investigate & judge the information. There are many text book & reference books help to us to find out the real issue. The reference books like: Applied Cryptography by Bruce Schneier and Cryptography & Network Security by William Stalling is very much help full to expand our idea. The object oriented java programming is very help to make programming for cryptographic key management issue. The Sun Micro-system UNIX sun solaris system administration guide: Vol 1 & Vole 2. & O' Reilly, Essential of System Administration is helpful to collect the basic data in real life environment.

No, where develop the following method in Graphically as well as Mathematically. We have to develop the following issue for betterment of the IS organization. There are many things not develop till now like : system Characterization, Risk Identification, Risk Analysis, Vulnerability Identification, Risk Mitigation Model & method, Control Analysis, Impact Analysis, Risk Determination, Control Recommendations & Results. We have to develop Risk Identification, Risk Analysis, Vulnerability Identification, Risk Mitigation in both analytical & graphical way. There many documents are available in general sense of risk identifications, risk analysis, risk mitigation, but operating system level, system software, application software, database, network and middleware level, the classification & categorization of risk is not available on today itself.

Existing Encryption Control: System Programmed(Automated control)[13],[18]

We have to develop this method based on our past experience of the hardening as well as controlling of operating system and network issue. Part of our assessment of the controls will necessitate an evaluation of the use of automated or programmed controls to mitigate risk as

opposed to manual controls that also may be implemented to perform similar functions. The common wisdom is that programmatic controls will work without fail because the machine-driven control does not have an opportunity to ignore its programming as human process might. However automated controls come at price. They must be tested, coded, monitored and maintained to be effective. When circumstances or risks changes, these controls must be reconfigured and go through another rigorous development and testing cycle. They will not work just as automatically when systems are not set up correctly as they would work effectively when the implementation of the control is done correctly. Automated systems do not think or recognize bad instructions in most cases. However, they are much more reliable than manual controls and can be assumed to be working in an unattended fashion, with only minor monitoring to ensure continued effectiveness once way are up & running. When drawing conclusions on the overall effectiveness and cost of automated controls, the building, maintaining, and monitoring costs must be offset by the potential losses to best understand the cost effectiveness of the controls. Additionally, the reliability or net effectiveness of the controls, which are assumed higher in automated and programmatic implementation, also must be factored in. Where loss due to risk cannot be left to change, automated controls should be recommended. Because we will, no doubt, be focusing on the high risk situations as we triage our work and provide risk-based solutions to our clients these will be our recommendations more often than not. Proper research, investigation, survey, implementation and routine monitoring are a prerequisite. Encryption control is a major action plan for control system which prevent the major components of IT infrastructure. We have to followup the following file system to develop the CKM issue.

Table:1

1.	etc/ssh/sshd_config Cryptiographic Control based on AES	Cryptography enable through ssh implementation AES: 256 bits chipper. chipper blowfish-CBC,aes256-CBC, aes256-chr.ssh-key gen -b 1024 -f /etc/ssh_host_key -n " chmod - - - /etc/ssh/ssh_config	Preventative control n=1024, 2048, 4096 chimed r w x (i. e. 4 2 1) – blank is nothing [H, M, L]

DATA COLLECTION BASED ON EXISTING CONTROL: (BASIC DATA) [13],[18],[19]

There are number of hardening and control methods developed as per requirement of the secure computing to achieve the highest level of business objective. There is a few method developed based on unix server and operating system programming.Unix file system have to be develop as per business requirement.

Table: 2

SN	SYSTEM FILES	ACTION PLAN	REMARKS
1	/etc/system	Can be update the kernel & n-bit processor	Can be improve the system performance

2	/etc/hosts	Develop the scripts: allow/disallow as per policy, chimed 000= /etc/mark disallow	Preventative control [H, M, L]
3	/etc/services	Disable the third parties services. Remove the ftp, http, telnet, port no, printer, IP services. Those services are not required.	Preventative control [H, M, L]
4	/use/bin/rash, etc/pam.conf	Disable all remote services: chmod 000 /usr/bin/rsh, rksh,rcp, ruser,rlogin, uptime.	Preventative control
5	/vary/dam/message	Date & time stamp (DC) [event mgmt]	Internal audit purpose [H, M, L]
6	/etc/rc.conf script	Run level script Run level script have to develop as per requirement. /etc/init.conf,rc2.d example:httpd_flags="NO"	Preventative control [H, M, L]
7	/etc/init.conf	OS services, run level	Preventative control

Problem in existing Control: [13] [18] [19]

- The IS security is a process which is difficult to adopt under normal conditions; the problem is compounded when it spans several jobs and applications running simultaneously under complex based web infrastructure which using million of user accessing the same piece of devices and information or data on the around the clock (24 x 7 x 52). The ssh key 1024 & AES key 256 is limited shorter key size. OS Hungering & control as well as high utilized of CPU Times: system throughput became slow down. Slow down the network resources, loss of communication system. There is no balance ratio among the Processor, Memory & Time slot of the high end OS.

- When too many packet are present in the subnet, performance degrade and in this situation data/packet congestion is happening. In this way transmission error is happening. At the high end traffic, performance collapses completely and almost no packet are delivered. If there is insufficient memory to hold all of them, packet will be lost. If slow the processor can also cause the congestion. Similarly, the low bandwidth can also cause congestion. Therefore, the OS became hungering & highly utilizing of CPU Times, system throughput became slow down, also slow down the network resources & loss of communication system. There is no protection, detection & automatic correction on the Shell, File & Kernel. There is no balance ratio among the Kernel, Processor, Memory, File System [Encryption Key] & Time slot of the high end OS. The high level decision process is required to implement resources like kernel, processor and instruction level parallelism (SISD, SIMD, MISD, MIMD) & high memory & encryption key sizes for high end business. The high end technology would be match with high volume business.

Why we use stronger (larger) key sizes?

Problem in existing control : due to shorter keys size

That's why we are using stronger key: AES 512, 1024, 2048, 4096 & so on.

Increasing the business (large volume of data, Information, data warehousing & data mining.

- Increasing the million of users.
- Increasing the hackers as well as experiences hackers.
- Increasing the hardware & software capabilities (n-th bits processor & no of CPU, Memory).

Our proposed cryptographic key management(encryption) control will be help to :

- Business continuity planning & disaster planning (BCP/DRP)
- Internal & external system audit.
- Keep the system balance among devices, sub-systems, resources & users need.
- Improve the throughput, interoperability, CPU utilization, total cost ownership (TCO) & Return on Investment (ROI, ROA)
- The least cost & best fit approach.

PROPOSED DYNAMIC AES METHOD:
(ANALYTICAL & GRAPHICAL METHOD)

The Cryptographic Key Management (CKM) keys must be securely managed when cryptographic functions are implemented in various other controls. Cryptographic key management includes key generation, distribution, storage and maintenance purpose. This is a measure preventive control in security world around the globe. As a brief, in the operational security domain, preventive controls are designed to achieve two things: to lower the amount and impact of unintentional errors their are entering the system and to prevent unauthorized intruders from internally or externally accessing the system. An example of these controls might be renumbered forms or a data validation and review procedure to prevent duplications. How the operating system maintaining ratio & proportion among various sub systems like server key, encryption key, processor & memory capability, availability and efficienciency. This issue is high lighted in our action plan. We have to find out some method, to make the more efficient, secure, high available & reliable the robust high end operating system. The SSH key in the existing Unix based operating system support only up to 1024 etc/ssh/sshd_config (Cryptography enable through ssh implementation AES: 256, bits chipper, chipper blowfish-CBC,aes256-CBC, aes256-chr.). The existing system supporting only 1024 in SSH key & 256 key size in AES Level. These AES-256 and SSH-1024 is not sufficient for high end processor, CPU, Memory, instruction pipe lines (SIMD, MISD, MIMD) . But, the propose control will be facilitate and resolve the various problems when it spans several jobs and applications are running simultaneously under heterogeneous complex infrastructure & mobile computing environment, which using million of user accessing the same piece of data around the clock(24 x 7 x 52). [Internet & Intrnet].

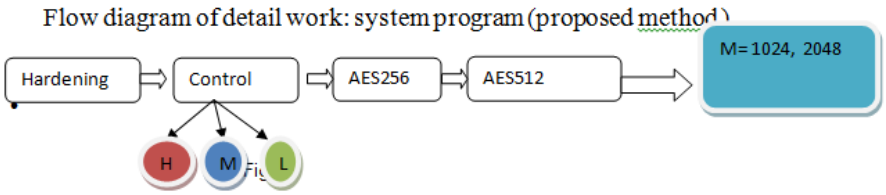

Fig:1

Flowchart of our proposal: [C α M, M α S, C α1/R]

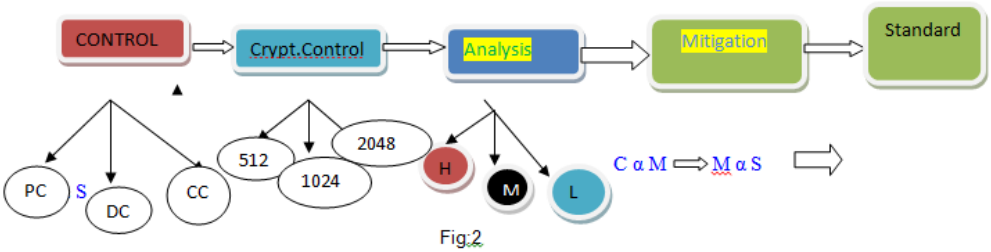

Fig:2

PREVENTIVE CONTROL TOOLS & TECHNIQUE:

Technical security controls for risk mitigation can be configured to protect against given types of threats. These controls may range from simple to complex measures and usually involve system architectures; engineering disciplines; and security packages with a mix of hardware, software, and firmware. All of these measures should work together to secure critical and sensitive data, information, and IT system functions. Technical controls can be grouped into the following: critical, major, minor (H, M, L) categories, according to primary purpose: Supporting controls are generic and underline most IT security capabilities. These controls must be in place in order to implement other controls which is deals with the system & sub system, devices dependencies. Preventive controls focus on preventing security breaches from occurring in the first place. Preventive control is the cryptographic control. These controls focus on detecting (DC)and recovering (CC) from a security breach.We can define Mathematically and Graphically as follows: C=K= (PC+DC+CC), Cα1/R, C α S.

Control flow chart:

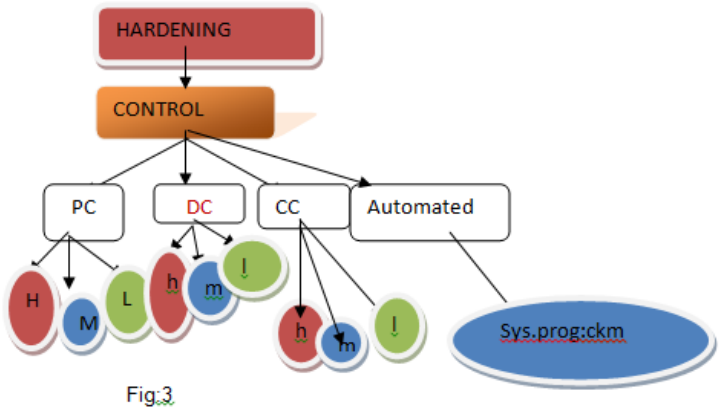

Fig:3

Our Action Plan:

Aim and Objective of the Thesis:

a).To investigate the preventive control and technique to safe guard our operating system that should be free from risk.

Our objective is that, to find out the maximum policy, procedure, tools (CKM), commands & technique how to safe guard our assets from internal and external hacker as well as misuse of the critical infrastructure components like (N-tire architecture) network components, operating system , server, data and file, database, middleware, application, scanner, printer and storage devices. Therefore, the following sub systems have to protect as per business requirement. Prevention is the first step of the risk assessment of the critical IT infrastructure. We are going to develop many more hard core prevention methods to protect the multi-tire architecture for web based application, which is facilitating million of customers . To maintain the integrity & privacy of the sub system for multi-core operating system as well as current virtualization technology. There are four level of prevention control methods are require to minimize the risk of any web based multi-tire infrastructure.

- Operating system control (Sun Solaris, AIX, LINUX, HPUX, NT, Microsoft OS)
- Network control (Load balancing, Clustering, DNS, Firewall, Proxy, Squid, SSH, SSL, FTP, LDAP)
- Database control
- Application Control (cryptography, mod-ssl, mod-security, mod_autho, vintila)

b). Minimize the Risk: (PC=K/R)

Meanwhile we have to investigate and audit the system security for detection, prevention & corrective action for minimize the various level of risk : like : High, Medium & Low. Risk is never eliminated, only we can able to minimize into lowest level. We can minimize the risk on IT infrastructure in the three ways of doing continues ways process of protection, detection and risk assessment method. But, risk never be eliminated. Risk can be transferable and minimize. We should always optimize the risk at the minimum level to protect our critical IT infrastructure and assets as well as sub systems (OS : processor, multi-plexier, memory, cpu and other related components).

In this paper, we are focusing only **operating system level** : (Advanced Cryptography Encryption control)

Control is directly proportional to AES-M. (PC=keys-M) [Where M=512, 1024, ……..] & AES & CKM key size inversely proportional to the risk. PC=K/R.

where k is the constant factor, then automatically reduced the risk. But, there is some limitation of architecture of processor & memory (instruction pipe lines: SIMD, MIMD, MISD) capability of the operating system for large based key generation of AES & CKM.

PROPOSED AES ALGORITHM: Advanced key expansion algorithm:

The advanced key expansion algorithm scheme takes as input a 4-word (N), (16-byte) key and produces a linear array of 44 words (176 bytes). This is insufficient to provide a 4-word round key for the initial add round key stage and each f 10 rounds of the chipper. The following pseudo code describes the expansion. Let us consider N = 4, 8, 16, 32……

Advanced key expansion (byte key {16 }, word w [44])

{

word temp

for (i = 0; i < N; i++) w[i] = (key [N*i], key [N*i+1]) , key [N*i+ 2] , key [N*i+ 3]);

for (i = N; I < 44; i++)

{

Temp = w [i - 1];

if (i mod N = 0) temp = SubWord (RotWord (temp)) Rcon [i/N]⊕

w[i] = w [i - N] = temp

}

}

The key is copied into the first four words of the expanded key. The reminder of the expanded key is filled in four words at a time. Each added word w{i} depends on the immediately proceeding word, w[i – 1], and the word four positions back, w[i – N]. In three out of four cases, a simple XOR is used. For a word whose position in the w array is a multiple of 4 (i.e N), a more complex function is used.

Based on the above algorithm, we can decide the programme(JAVA/C++) , then we can find out the /var/adm/message (action & reaction) of the JVM issue as well as space utilization like RAM & Cache etc (space & time complexity). Mean while we can use the various system commands and scripts for further review and analysis of the unix based server. How is behaving the server and its sub components, when we are running on the different processor & CPU of the same cryptographic algorithm program AES512, AES1024)? How far is system (hardware,

software, application, network bandwidth & related devices) maintaining risk level, we can only review practically based on theoretical idea. We can review the system behavior with space & time complexity. How is system is behaving when million of users accessing the same piece of devices?

PROPOSED DYNAMIC AES METHOD: DERIVED DYNAMIC DATA [D^3]FROM TABLE : 1

This research paper contributes to the development of an optimization method that aims and objective to determine the optimal cost, quality & time to be invested into security method & mechanisms deciding on the measure component of operating system resources(i.e. Processor, Memory, I/O devices). Further more, the method & mechanism optimize the cost, time & resources is supposed to reduce the system risks. We have to optimize the technology & resource cost and maximize the business (throughput).

The VALIDATION & VARIFICATION OF HARDWARE & SOFTWARE BASED ON ENCRYPTION TECHNOLOGY. The hardware & software validation for the high performance computing to manage E-Commerce, E-Payment and product like B2B, B2C, P2P & G2G. These system validation, verification & benchmarking can be define in Table :1, 3& 4. We have to maintain the risk free environments on the hardware, software & application level on basis of the following data.

PREVENTION TABLE: TABLE:3

(DERIVED DATA): ENCRYPTION CONTROL MATRIX

E	128	256	512	1024	2048	A=2^n	AES	HA
S	512	1024	2048	4096	8192	S=2^n	SSH	HA
P	32	64	128	256	512	P=2^n	Processor	HA
M	16	32	64	128	256	M=2^n	Memory(GB)	HA
K	L	L	M	H	H	K=2^n	Control	HA

(L- LOW RISK, M-Medium RISK, H-HIGH RISK) (PC+DC+CC=K) [CKM=k.1/R] Fuzz's Law]

We have to maintain the sequence as follows: 1st we have take care of the Processor, 2nd RAM, then AES & SSH key. These four parameters should be satisfied according to our table data for better performance & high security.

Processor is directly proportional to the Memory, Advance Encryption Standard is directly proportional to the Memory. Hardening is directly proportional to control, mean while control is proportional to the Mitigation. The set of parameter [{P, M, E, K, A} € RM], Where A is the

High Availability, RM is Risk Mitigation. [K=PC+DC+CC].We can optimize the risk factor by help of these five elements. All these five elements depends on each others . Availabilities is the main concern among the all of them.

As per Rijndael-AES very well suited for restricted space management where either encryption and decryption is implemented as per table data. It is very high memory requirement will increase the both encryption & decryption are implemented simultaneously. For 192, 256,512 &1024 bits key sizes throughput falls in standard, That's why we need fully pipelined implementations (SIMD,MIMD), the RAM increases, but the throughput is unaffected. Our prevention Table:3 data will be helpful to potential for Processor & Instruction level parallelism for high end computing.

PROPOSED AES JAVA BASED PROGRAM: [As per Algorithm & Table:3]

```
import javaxnet.*
 import java.security.*;
 import javax.crypto.*;
 import javax.crypto.spec.*;
 import java.io.*;
 import java.util.regex.*;
 import javax.net.ssl.*;
import javax.net.ssl.*;
 /**
 * This program generates a AES1024 key, retrieves its raw bytes, and
 * then reinstantiates a AES1024 key from the key bytes.
 * The reinstantiated key is used to initialize a AES1024 cipher for
 * encryption and decryption( AES-ADS ).
 */
 public class AES1024 {
 /**
 * Turns array of bytes into string
 *
 * @param buf Array of bytes to convert to hex string
 * @return Generated hex string
 */
 public static String asHex (byte buf[]) {
 StringBuffer strbuf = new StringBuffer(buf.length * 2);
 int i;
 for (i = 0; i < buf.length; i++) {
 if (((int) buf[i] & 0xff) < 0x10)
 strbuf.append("0");
 strbuf.append(Long.toString((int) buf[i] & 0xff, 16));
 }
 return strbuf.toString();
 }
 public static void main(String[] args) throws Exception {
 String message="This is just an example----AES1024";
```

```
// Get the KeyGenerator
 KeyGenerator kgen = KeyGenerator.getInstance("AES1024");
 kgen.init(1024); // 128,192,256, 512, 1024, 2048 and 4096 bits may not be available as per
system specification

 // Generate the secret key specs.
 SecretKey skey = kgen.generateKey();
 byte[] raw = skey.getEncoded();
 SecretKeySpec skeySpec = new SecretKeySpec(raw, "AES1024");

 // Instantiate the cipher block
 Cipher cipher = Cipher.getInstance("AES");
 cipher.init(Cipher.ENCRYPT_MODE, skeySpec);
 byte[] encrypted =
 cipher.doFinal((args.length == 0 ?
 "This is just an example" : args[0]).getBytes());
 System.out.println("encrypted string: " + asHex(encrypted));
 cipher.init(Cipher.DECRYPT_MODE, skeySpec);
 byte[] original =
 cipher.doFinal(encrypted);
 String originalString = new String(original);
 System.out.println("Original string: " +
originalString + " " + asHex(original));
 }
 }
```

We have to compile the programm : javac AES1024.java and then run #java AES1024. (step 1).

We have to review how is behaving the system. Mean while we can review the various parameter through(/var/adm/message >> /var/temp.txt). we have already highlighted the issue in existing control. (step 2).

There are various issue is bring up to the notice of the real reseacher (space, time, processor) for further advancement of this article. Control is directly proportional to AES512. (C=kAES512) where k is the constant factor, then automatically reduced the risk. But, there is some limitation of processor capability of the operating system (N-bit processor, Memory). (Step:3)

HOW TO INVESTIGATE PROPOSED DYNAMIC AES METHOD RESULT ON UNIX OS:

- A**. RISK IDENTIFICATION (Detective Control):** Intrusion Detection System : In unix operating system, the risk can be identified by the following dynamic OS logs. (date and time stamp)

  /var/adm/syslog : syslog system logs

  /var/adm/sulog : super user log

  /var/adm/loginlog : user login log

/var/adm/message >> Temp

using last| head or tail -f /var/adm/syslog, tail –f /var/adm/message .

These logs are very helpful to the internal as well as external audit purpose. The incident management department analysis these logs to mitigate the risk factor.

- B. **RISK ANALYSIS:** (above system logs:DC) Meanwhile we have to investigate and audit the system security for detection, prevention & corrective action for minimize the various level of risk : like : High, Medium & Low. Risk is never eliminated, only we can able to minimize into lowest level. High level preventive control will be take care of minimize the lowest level of risk. (Prevention is inversely proportional to Risk) PC=k/R and PC=k.S (k is the proportionality constant) or CKM =kS

REVIEW THE INTERNAL OS by applying the following commands and scripts in key boards in super user mode.

Table:4

SN	SCRIPT S & COMMANDS	DESCRIPTIONS	REMARKS
01	iostat	Input /output statistics	CPU & Device Utilization PRIMARY RISK ASSESSMENT
02	pmstat	Processors statistics	Global Statistics among all the processors & users: Risk Analysis
03	vmstat	virtual memory statistics [MEMORY ACTIVITIES]	Statistics of all the processor runable, block, swap, free buffer, input/output block devices, CPU detail, system, user, idle, waiting stage. Risk Analysis
04	sar	system activities	Activities report on: paging & swapping of OS detail. Risk Analysis
05	ps –ef | grep	ACTIVITIES OF PROCESSOR	The suspious processor or orphan/dead one. [space & time complexity issue]
06	lsof l more	FILE SYSTEM ACTIVITIES	List of open files system which is very high risk. Risk Analysis
07	/etc/system	KERNEL SYSTEM ACTIVITIES	Can be update the kernel PRIMARY RISK ASSESSMENT

08	who –a	current user login on the system	Identified the specific user: Risk Analysis
09	lastlogin	last login on the system	Who is on the system: Risk Analysis
10	/etc/.profile	USER PROFILE INCLUDING SHELL	Profile file PRIMARY RISK ASSESSMENT
11	/var/adm/message	System mesg	Date & time stamp: Risk Analysis

Analysis: Reaction

- What can going wrong? (past, present)
- What is the likelihood that it would go wrong?
- What are the consequences and alternate?

Synthesis: Action

- What can be done (future plan)?
- What options are available and what are their associated tradeoffs in terms of all costs, time, benefits and risks? (TCO & ROI)
- What are the impacts of current management decisions on future plan (BCP/DRP)

C.RISK MITIGATION (Preventive Control):

Risk mitigation is the important method to be adopt to avoid risk. The following method is most well come on unix platform for high security, reliable, scalable and high available (HA).

- Disable last history on the command prompt #.
- Disable keyboard.
- Disable shell prompt on environment (Environment, system & user profile)
- Lock the suspicious user. (password, shadow)

Unix file system have to be develop as per business requirement: (DERIVED DATA)

BENEFITS: DECISION MAKING DATA FOR RISK ASSESSMENT.

a. The top management have to decide the optimal cost of HW/SW.(Processor, Memory, CPU, IO Units & Multiplexier).
b. Capacity planning of operating system (P, M & E)
c. Business continuety planning & Desaster recovering planning (BCP/DRP)
d. Risk Assesment and decision mgmg.
e. Internal & External of operating system audit.
f. The top management have to decide the encryptions standard (AES, SSH, CKM) key sizes as per business requiremet.

RESULT:

This dynamic AES Method helps to the any organization the compete more effectively in local, national and international level at any time and any place around the clock as follows:

- Minimizing costs & risk.
- Maximize profits, ROI, ROA , TCO & TQM.
- Improving functionality (P, M, E), quality & decison
- Maximizing Productivity at optimal cost.
- Optimizing time & Maximize the utilization of (Resources) Man,Machine, Material, Market, Money & Method(M^5).
- Solving multiple problems at right time with optimal cost.
- Utilizing overall resources more effectively at right time and right place.

Summary:

- The cryptographic control provide accountability for individuals who are accessing sensitive information on application, system software, server and network. This accountability is accomplished through access encryption control mechanisms that require identification, authentication, authorization, accountability, non-repudiation, availability, reliability & integrity through the audit function (/var/adm/message). By the help of these model we can keep the balance ratio among the Processor, Memory, Encryption key & Time slot of the high end OS as per business requirement and availability of the resources around the globe in basis of 24 x7 x 52 pattern .

- The encryption control is a one of the efficient & best control for risk management. Alternatively we can say preventive & automated control which is help to reduce the risk of the IT infrastructure. This control is mostly acceptable on heterogeneous hardware like: INTEL, MAC & MOROLA.

- Risk can be mitigate by ongoing process of various action plan of control of OS, Network, Database, Application and devices as well as relevant resources of the IT infrastructure.

- To minimize the risk, operating system hardening, preventive, detective and corrective action is the most well advanced action plan for the long term business activities of the every organization. Therefore, contingency plan is the most effective & efficient plan for safe guard of the organizational assets. Therefore operating system control, anti-virus solution, periodically security system programming (SSH,AES) & patches updation are the most preventive, detective and corrective action plan of the any organization to survive. In summary, the risk assessment process is about making decisions to minimize the risk. The impact of a successful attack and the level of acceptable risk for any given situation is a fundamental policy decision. Likewise, vulnerabilities are design issues and must be addressed during the design, development & implementation of information resources.

- A sound information security policy identifies prevention, detection, and response measures. The appendix provides more details on risk assessment tools and practices that may be used to

improve information security programs. Preventive measures may include regularly using vulnerability assessment tools and conducting periodic penetration analyses. Intrusion detection tools can be effective in detecting potential intrusions or system misuse. Institutions should also develop a response.

- By reading through this paper and utilizing the checklist for preventive, detective and corrective action plan of that organization, an OS and Security programmer & administrator now has a base knowledge of security, server hardening, intrusion detection, auditing, and security tools. This knowledge can be directly applied to their servers and many vulnerable holes will now be filled. Bear in mind that many holes that exist have yet to be discovered. Therefore, it is critical that every Unix security-minded programmer maintains their knowledge of security by researching and referring to the Internet resources that have been mentioned. If there is ever a question about updation & implementation of any of the suggested features, refer to the OS server Security manuals that were designated with the specified feature (all features have been research, design, developed & documented).

References:

[1]] Bernard Colman, Discrete Mathematical Structures Year 2007 5th Edn, PHI Chap 1, pp-15

[2] Bocholt, J. L. (1989) "Implementing Security and Integrity in Micro-Mainframe Networks," MIS Quarterly (13) 2, pp. 135-144.

[3] Bruce Schneier, Applied Cryptography, Wiley 2nd Edition 1996 Chap. pp-155

[4] CISA Review Mannual 2003, Chap 4, pp226-230

[5] Coriolis , CISSP Exam cram, dramatic year 2002 (Chap 4 p 61-77)

[6] Edgar G, Discrete Mathematics with Graph Theory(2007), 3rd Edn pp57-58

[7] International Journal of Computer communication Technology(IJCCT Vole 1, Issue 1 Nov 2009 pp71

[8] International Journal of Computer Science & Tech(IJCST Vole 2, Issue 4 Oct-Dec 2011 pp22-27

[9] Joe. L Matt. Discrete Mathematics for Scientist and Mathematician(2008), PHI 2nd Edn Ch. P179.

[10] John B. Kramer, The CISA Prep Guide, Wiley Publishing Inc.Year; 2003 Chap 7 pp420-450

[11] Mcl.ean, Kevin & Len watts (1996) Risk Analysis Methodology " IS audit & Control Journal III 32-36

[12] NIST special publication 800-30 Risk management guide for IS July 2002, page 8

[13] O' Reilly, Essentail of System Administration 1995 (Chap 10, P467- 485) & Chap 6(p201-243), Chap11

[14] Pressman, Software Engg 5th Edn, year 2001 MGH,Chap 6 (P 145- 162)

[15] Pichnarczyk, Karen, Weber, Steve & Feingold, Richard. "Unix Incident Guide: How to Detect an Intrusion CIAC-2305 R.1". C I A C Department of Energy. December,1994.

[16] Shon Harrish, CISSP Exam study guide, Dreamtech year 2002 DRP/BCP (Chap 9 P 591-603)

[17] Shon Harrish, Security Mgmt Practices, Wiley, Dreamtech CISSP Exam Year 2002 study guide 2003 (Chap 4 P 57-92)

[18] Sumitabh Das UNIX System V UNIX Concept & Application Chap 4-8.

[19] Sun-Microsystem UNIX sun solaris system administration: Vole 1 & Vole 2.

[20] William Stalling, Cryptography and Network Security(2006) 4th Edn Ch6.3 pp192 Ch9.2 pp-609-614

[21] Weber Ron , Information System audit & control PHI 2002(Chap 7 P- 243-285)