IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661,p-ISSN: 2278-8727, Volume 18, Issue 4, Ver. I (Jul.-Aug. 2016), PP 25-36
www.iosrjournals.org

Framework of Security Mechanisms for Monitoring Adaptive Distributed Systems

Manjunath Kotari[1], Dr. Niranjan N. Chiplunkar[2], Dr. Nagesh H.R[3]
[1]Research Scholar , NMAM Institute of Technology, Nitte, India
[2]Principal , NMAM Institute of Technology, Nitte, India
[3]Professor & Head, MITE, Moodbidri, India

Abstract: *The distributed systems have a noteworthy role in today's information technology, whether it is governmental or nongovernmental organizations. The major concern of today's software industry is that dealing with security, scalability and dynamicity of distributed systems. Adaptive Distributed Systems (ADSs) are nodes in networked computers, which change itself according to changes in the environments. A monitoring system or tool is used to identify the changes in the distributed systems and all the activities of the entire network. This monitoring system may be compromised by the intruder while gathering the information from the distributed systems. This paper, framework of security mechanisms for monitoring ADSs is discussed by following tasks. Firstly, investigation of existing monitoring tools to find the impacts of monitoring activities in ADSs. When the monitoring tool collects security critical information, there is a high risk of information disclosure to unauthorized users. Secondly, a secure communication channel to monitor confidential information has been implemented by using RSA algorithm. Thirdly, a secure, customized network monitoring tool has been implemented by providing the necessary security for each parameter in the system. Security metrics are used to measure the security levels of each monitored parameter.*

Keywords: *Adaptive Distributed Systems, Network Monitoring Tool, Security Mechanisms, System Under Study.*

I. Introduction

In the present internet state of affairs, distributed systems have a noteworthy impact on private and governmental organizations. From time to time depending on the circumstances, the generation of information increases or decreases. Consequently, the value of information criticality also changes dynamically. Distribution of various information and services of different websites is frequent nowadays[1]. It is very essential to monitor the distributed systems of increasingly complicated IT environments [2]. The monitoring activities involve performance monitoring and resource usages of heterogeneous distributed information systems. The typical distributed system consists of components which are deployed into different subsystems or called services and security policies. The services on different server systems run as independent processes. Most of the time an enterprise systems need to contact external services for business functionalities like suppliers and banking [3]. Monitoring is considered expensive for the various business activities of IT infrastructures. While accessing the information, users expect a high level of system quality and more secure environment. These issues need intrinsic adaptation of the distributed systems. This adaptation may be achieved with the help of Adaptive Distributed Systems(ADSs)[4]. An ADSs is a system that vigorously adjusts its performance based on changes in the surroundings. The Adaptation method allows distributed system to revise its own configurations and functions in response to changes in its atmosphere. Monitoring proves the best adaptive capability of the system to take proper action with respect to the changes in the environment. The security issues are main concerns irrespective of whether the system is adaptive or non-adaptive. The adaptive monitoring system changes its performance depending upon the criticality of information collected for the adaptation purpose[5]. The security system allows only encryption mechanism for the security critical information during monitoring of target systems. The distributed systems may be hacked by unauthorized users in several ways, either during transmission or at storage[6]. There is a chance of a significant security problem if the monitoring process is hacked by intruders during the time of information gathering. The monitored systems may be overtaken by intruders during this adaptive monitoring process; which means that the possibility of security breaches are more while the system is under monitoring. One of the solutions to address these security problems of distributed systems is framework of security mechanisms for monitoring adaptive distributed systems.

The distributed systems provide deliberate functionalities and services continuously with the suitable level of quality only by the constant monitoring. The following changes need to be monitored with the help of monitoring tool which include IP Address of nodes, Host ID, Network ID, Processes, Running Applications, Memory, Disk, Link Failures etc. There is a chance that monitoring system may be overtaken by unauthorized user. In such cases, if the monitoring process is restricted for the purpose of protection; it may lead to severe limitations on the capacity of the adaptive system. Achieving adaptation with minimal impact on security mechanism is a challenging task[7]. In this regard, various issues need to be investigated thoroughly, like what data can be monitored, how it can be monitored and what is the impact of monitoring over security. It has been recognized that, different security levels of monitoring parameters exist based on user requirements. These security levels are measured using security metrics. Security metrics are likely to measure the level of the security risk based on the attributes of data pertinent to security problems. The security metric attributes involve the criticality, detail, size and support for inference during its calculation[8].

The security aims such as confidentiality meets only through the security metrics. Based on the security metric values actions can be taken to improve the overall security program and also to identify the security risks. These security metrics indicate the efficiency of various mechanisms of a secure technique. Information sharing via network is a hurdle nowadays. It is very essential to take care of two issues during sharing of information: Quality of Service(QoS) and Security. Distributed Systems are one kind of system which provides the QoS during accessing information on remote sites. To improve the QoS, it is necessary to make the distributed systems as adaptive. This adaptation happens with the three different phases such as constant watching, identification of changes in the environment and execution of required changes. The constant monitoring[20] of systems may leads to security problems. Hence security mechanisms should be provided during monitoring of distributed systems. Security-measurability-enhancing mechanisms are crucial to the wider acceptance of security metrics, measurements, and associated tools and methods. These are the research issues that provided motivating factors to closely study the problem and to propose the framework of solution.

II. Related Works

Demissie B. Aredo et al. [7] highlighted about the two main problems in adaptive distributed systems. Firstly, system monitoring to collect data necessary for adaptation may cause security problems. Secondly, restricting the monitoring and gathering of information may constrain the capacity of the system to adapt to the changing environment and maintain the security mechanism. Hence, in making a critical distributed system adaptive to deal with security threats, there is a risk of compromising the whole security mechanism. The authors does not address about how to achieve an adaptation through the minimal impact on its security mechanism. Also, the authors do not discuss about, what kind of data that can be monitored and how to monitor without affecting the performance of the distributed monitoring architecture.

Sule Yildirim et al. [8] proposed a security metric for quantifying the impact of monitoring of the effectiveness of a security mechanism of the target systems. The metrics are a function of a set of attributes of data to be collected by monitoring the system, which are relevant to the security implementation. Sule Yildirim et. al. [7] [8] identified the following attributes as relevant to the security issues and the metrics are defined in terms of these attributes: level of Criticality, Detail, Size, and support for Inferences. The security metrics has been defined by the equation: $M = \alpha.C + \beta D + \lambda S + \eta I$ for some nonnegative coefficients α ,β ,λ and η . The values of these coefficients and their relationships has been determined using some analytical techniques. Symbolically, $SM \approx 1/M$, where SM is the effectiveness of the security mechanism of the target system. However, Sule Yildirim et al. [8] does not address about ways to minimize the monitoring while designing monitoring systems. The important security issues like authorization, authentication and encryption are also not discussed. The parameters of the security metrics need to be redefined, because the size of data is directly related to its level of detail, and detail of data is directly related to support for inference. In this research work, we are going to discuss these issues in detail.

Reijo M et al. [9] proposed a process for security metrics. Initially, need to carry out threat and vulnerability analysis. Carry out threat analysis of the system under investigation and its usage environment. Identify known as wells as suspected vulnerabilities. Then, need to define and prioritize security requirements in a holistic way based on threat analysis. The most critical security requirements should be paid the most attention. Pay attention to the coherence of requirements. Finally need to identify basic measurable components of the high-level requirements using a decomposition approach. The Genetic Message Oriented Secure Middleware(GEMOM)[9] is used to carry out monitoring and enforce adaptive security operations based on on-line security metrics. The monitoring system in GEMOM consists of a proactive part and a reactive part. The proactive part makes long-term decisions based on security, trust, dependability and reputation information, expressed in the form of security indicators and metrics. The reactive part monitors message traffic, faults and resilience actions. The monitoring is based on appropriate security indicators and metrics.

Yin Guohui[10] et al., proposed about how to capture data with Sharpcap on dot Net. Sharpcap is a development package of capturing the first floor network data on the dot Net platform. The basic principle is through establishing the network adapter as the monitor pattern, all data that flows through data linker layer is captured. Sharpcap is a frame for capturing data packet, which was specially designed for dot Net. It was sure that, it had made a Sharpcap capture original packet, rapidly, accurately, steadily and efficiently. Limitations of the raw socket highlight superiority of the Sharpcap in capturing of the data packet.

Shiping Chen et al.[11] presents a unified monitoring framework for distributed information system management. This framework utilizes web service and messaging queue technologies to collect log data for business process visualization. A prototype tool is implemented and presented to demonstrate the feasibility and usability of the proposed framework, and experiments are conducted to evaluate the performance overhead introduced by the monitoring. Tests with and without monitoring were performed under a range of client loads. The experimental results show that monitoring infrastructure does not change the throughput behaviour of the whole system significantly. Therefore, it can be used for performance monitoring and tuning in real life applications. However, the current framework is one-way only for passive monitoring.

Teemu Kanstrén et al.[12] presented a core set of requirements for building a secure, dependable and adaptive distributed monitoring framework and reference architecture for building other monitoring frameworks that need to address these types of requirements. The presented requirements provide a basis for understanding the different needs of monitoring and how they are related to the domains in which the reader is interested. Marco Comuzzi et al. [13] discussed the issue of customized business process monitoring infrastructures in the context of cross instance process monitoring. The design of this tool decouples the monitoring concern from the underlying process engine, making the approach generic, i.e. replicable using alternative process enactment technology. In this work authors focused on the flexibility of current process monitoring technology.

III. Problem Statement And Formulation

The major challenges of ADSs are security issues. This can be partially solved by using techniques such as cryptographic techniques, auditing mechanisms and access control. There has also been an approach to monitor the communication channels to obtain secure communication, but the monitoring of a distributed system may cause security problems. Information about the various networking activities of users, their communication patterns as well as the contents of the messages are collected by monitoring system which is normally external to the target system. It is definite that a monitoring system becoming more knowledgeable about the environment, it is functioning in, then changes in the distributed environment can be detected and the remedial actions can be taken for improving the quality of service[14]. The security threats that occur during monitoring may be avoided by restricting the monitoring. Restriction of monitoring system may leads to degradation of quality of service. Also, in the mean time access to the monitoring systems should be protected from the intruders.

In this research work; framework of security mechanisms for monitoring adaptive distributed system involves the following tasks. Firstly investigation of the existing monitoring tool in the distributed environment and finding its impacts. The monitoring system should collect detailed and security-critical data such as user ID and IP address of a site. In that case, there is a high risk of information disclosure to unauthorized intruders. Secondly, implementation of secure communication channel by using cryptographic functions in existing monitoring mechanisms. Thirdly the research work discusses a secure customized monitoring tool by providing necessary security for each parameter in the system and use of monitored parameters in the adaptive distributed systems.

IV. Frameworks And Objectives

The framework for Security Mechanisms in the context of Adaptive Distributed Systems architecture is proposed as shown in the Fig 1. In the context of adaptive distributed systems the security mechanisms are investigated with the help of existing monitoring tool and customized monitoring tool. In this research work, Wireshark tool[14] has been used for studying the existing monitoring tool. The Wireshark tool helps to find the security vulnerability over distributed systems. If the security vulnerability is exists, then encrypt all the parameters and monitor the distributed systems, otherwise monitor all the parameters directly without any encryption. In case of developed customized monitoring tool, each system in the distributed system sets security metric for each parameter. The security metrics is defined with criticality, detail and size of parameters. The confidential parameters are needed to be monitored with high value of security metric. Depending upon the security metric value, the customized monitoring tool decides about encryption. In other words, if the security vulnerability of particular parameter is high then encrypt that parameter and monitor it. Otherwise, directly monitor such parameter.

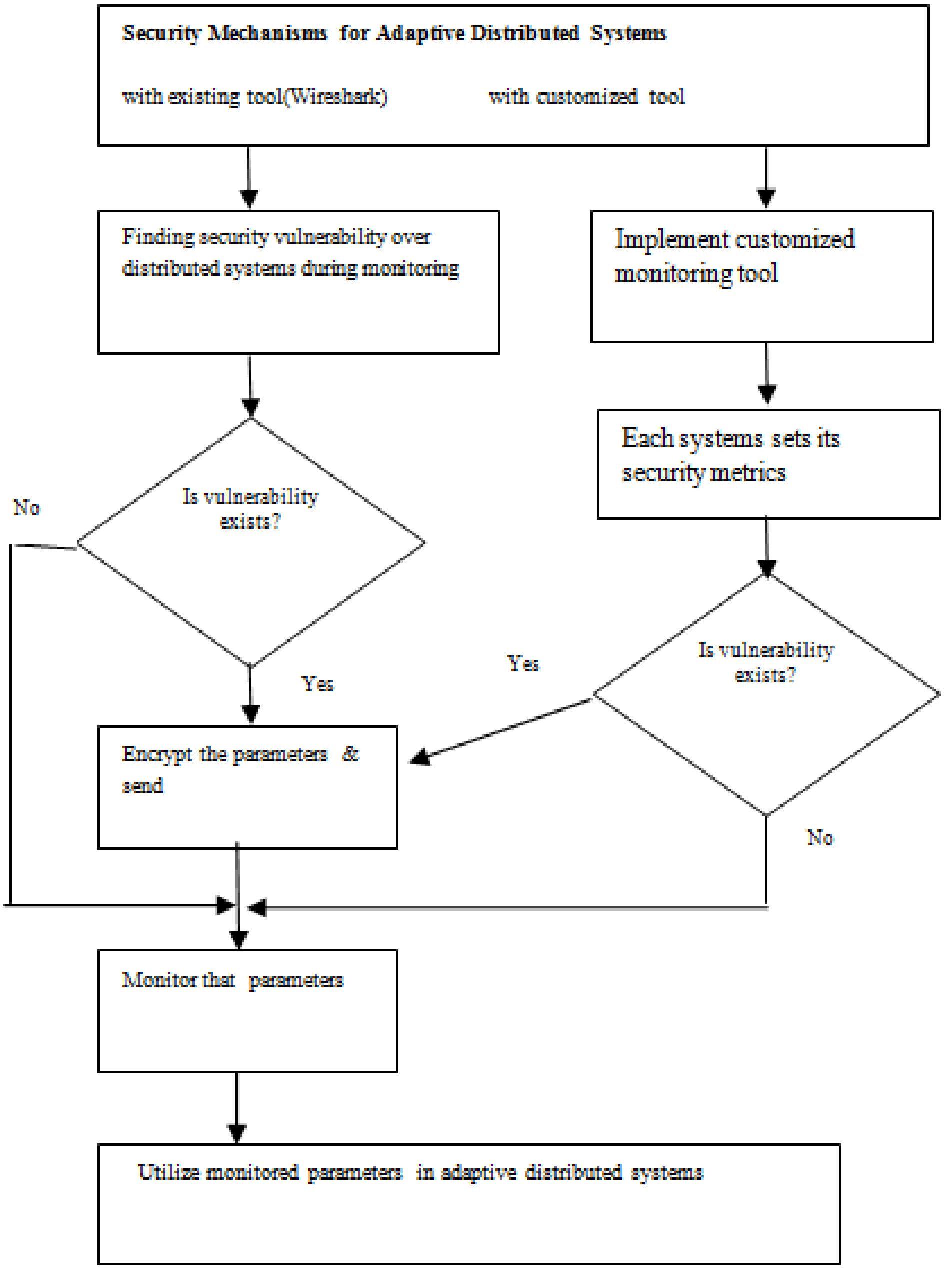
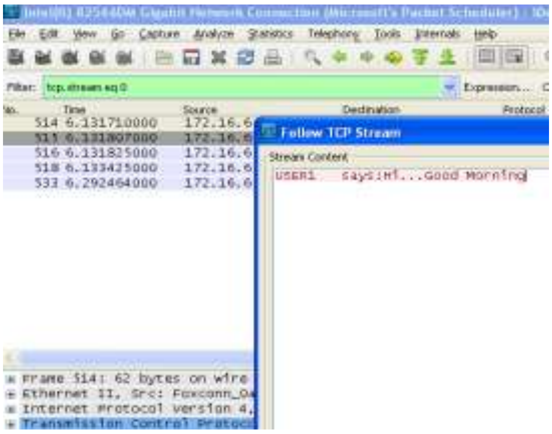

Security Mechanisms for Adaptive Distributed Systems
with existing tool(Wireshark)
with customized tool
Finding security vulnerability over distributed systems during monitoring
Implement customized monitoring tool
Each systems sets its security metrics
Is vulnerability exists?
No
Yes
Is vulnerability exists?
Yes
No
Encrypt the parameters & send
Monitor that parameters
Utilize monitored parameters in adaptive distributed systems

Figure 1. Framework architecture for Security Mechanisms

V. Implementation and Experimental Results

Investigation of the existing monitoring mechanisms

Initially, the proposed system involves monitoring of distributed systems and investigation of the existing monitoring tool. The impact of monitoring in distributed systems has been investigated with the help of existing monitoring tool called Wireshark. A message which is transferred between two users as well as information related to that message also can be monitored.


File Edit View Go Capture Analyze Statistics Telephony Tools Internals Help
Filter: tcp.stream eq 0
Expression...
Follow TCP Stream
Stream Content

```
No.  Time          Source        Destination     Protocol
 514 6.131710000   172.16.6
 515 6.131807000   172.16.6
 516 6.131825000   172.16.6
 518 6.133425000   172.16.6
 533 6.292464000   172.16.6

USER1   says:Hi...Good Morning

+ Frame 514: 62 bytes on wire
+ Ethernet II, Src: Foxconn_0a
+ Internet Protocol Version 4,
+ Transmission Control Protocol
```

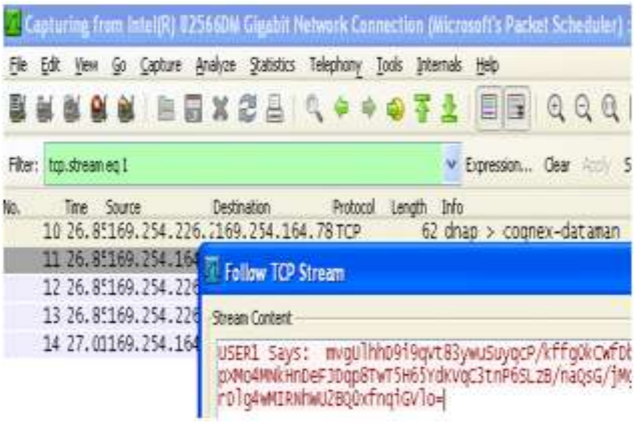
Figure 2. Monitoring Using Wireshark tool

The monitoring system monitors time and date of communication, source system IP address and received system IP Address(destination), Username and Exact exchanged message, etc. Also, intruder may misuse this information for other purposes.The Wireshark tool is a network packet analyzer and it is a widely used open source tool. A network packet analyzer captures the network packet and tries to display that packet data as detailed as possible[16]. The user selects the interface to capture the packets from capture menu. Packets in the network usually displayed in top with information like source IP Address, destination IP Address and protocol length. The stream of packets may be viewed by right clicking on any TCP packet and must choose Follow TCP stream.

The USER1, who present in the distributed system tries to send the message to USER2. The USER2 receives the message in the inbox and USER2 opens that message. In this experimental results, the existing monitoring tool Wireshark has been used to view the message by using follow TCP stream option. The Fig 4 shows that, by using Wireshark Monitoring tool one can access the information's of the others over the network. In the similar way, intruder who present in the distributed systems may misuse this Wireshark monitoring tool to see the information's.

Implementation of Secure Communication Channel

In the previous section it has been observed that, the intruder who may present inside the network or outside the network may try to access the information which is transferred between two users. So, security mechanism has needed to be used to protect the information transferred between the two users. In that case, the sender needed to encrypt the information by using RSA algorithm[17]. This encryption process happens upon sender clicks the send button. Upon the receiving message at receiver side, the message was decrypted and displayed on the inbox of receiver. Whenever the intruder tries to access the message through this secure communication channel, then only the encrypted message is displayed on the screen as shown in the Figure 3.


Capturing from Intel(R) 82566DM Gigabit Network Connection (Microsoft's Packet Scheduler)
File Edit View Go Capture Analyze Statistics Telephony Tools Internals Help
Filter: tcp.stream eq 1
Expression... Clear Apply
Follow TCP Stream
Stream Content

```
No.  Time  Source           Destination       Protocol  Length  Info
  10 26.8! 169.254.226.: 169.254.164.78  TCP        62  dnap > cognex-dataman
  11 26.8! 169.254.164
  12 26.8! 169.254.226
  13 26.8! 169.254.226
  14 27.0( 169.254.164

USER1 Says:  mvgUlhhD9i9qvt83ywuSuyqcP/kffg0kCwfDt
pXMo4MNkHnDeF3Dqp8TwT5H65YdKVqC3tnP6SLzB/naQsG/jM
rDlg4wMIRNhwU2BQ0xfnqiGVlo=
```

Figure 3. Monitoring using Wireshark tool

Implementation of Secure Customized Monitoring Tool

A simple client-server application has been established for communication in two directions. The remoting technique has been used in this application as shown in Fig 4. Inorder to build distributed applications, it is very essential to use remoting framework in networking services[22]. The client application should register with server application for the purpose of remote access. Here, the server and client application uses HTTP, SMTP and TCP channels for registration. The API System.Net.Sockets**,** provides a managed implementation of the Windows Sockets interface for developers that need to tightly control access to the network**.**

In this secure customized monitoring tool, systems are connected over a LAN/WAN, where one system acts as a Monitoring System known as Network Monitoring Tool (NMT) and other systems are monitored by NMT is called as System Under Study (SUS). The follwing parameters have been identified to find the level of security such as, IP Address, User Name, Host ID, Network ID, Port Number, File Name, File Content, Link Failure, Node Failure, Processor Response Time, Processor Utilization, Workload, Memory, CPU, and Disk.

Security Metrics has been developed for the reason of providing security to each parameter in the network. A customized tool is implemented in this regard. In particular, the aim of our customized tool is to investigate the criticality level of each parameter. The SUS system decides about the security to parameter based on the security levels. Security Metrics are defined based on these security levels for each parameter.

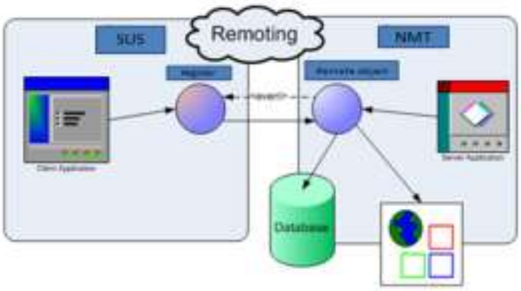


Figure 4. Remoting

The following attributes have been defined with respect to security metrics.

- **Level of criticality:** Finding the criticality level of each parameter out of 100 weight value can be defined in the following manner. The zero means Not Critical, the zero to 25 means Less Critical, 26 to 50 means Medium Critical, 51 to 75 means Critical and 76 to 100 means Extremely Critical. These assumed values are based on Mekonnen Feyissa et al., [21] the monitoring distributed systems for adaptive security. The more level of critical parameters needs to be sent in encrypted format. Table 1, Table 2 and Table 3defines assumed values for criticality of data, assumed values of detailed data and assumed values for sized data respectively.

Table 1 Assumed Values for Criticality of Data

Security Metric Range Value	Meaning
0	Not Critical
0-25	Less Critical
26-50	Medium Critical
51-75	Critical
76-100	Extremely Critical

- **Level of detail:** Finding the level of detail of each parameter out of 100 weight can be defined in the following manner. The zero to 25 means Less Detailed, 26 to 50 means Moderate Detailed and 51 to 100 means Full Detailed. The more detailed parameters describes the entire details of the information, it leads to more vulnerable to the system.

Table 2: Assumed Values for Detailed Data

Security Metric Range Value	Meaning
0-25	Less Detailed
26-50	Moderate Detailed
51-100	Full Detailed

- **Level of size:** Finding the level of size of each parameter out of weight 100 can be defined in the following manner. The 0 to 25 means less size, 26 to 50 means medium sized and 51 to 100 means full sized data.

Table 3: Assumed Values for Sized Data

Security Metric Range Value	Meaning
0-25	Less Sized
26-50	Moderate Sized
51-100	Full Sized

The Security Metric calculation is depending on the different parameter values as defined in Table I, Table II and Table III. To calculate the security metrics, the following equation (1) is defined.

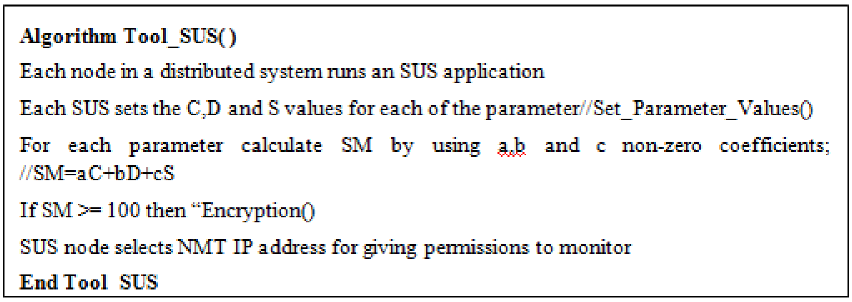
$$SecMet = a.C + b.D + c.S \qquad ---- \qquad (1)$$

Where, C indicates the level of Criticality, D indicates the level of detailed data and S defines size of the data. The parameter coefficients a, b, c is non-negative coefficients whose values can be determined using analytical techniques. After applying a, b, c values to the Equation (1) the SecMet values can be calculated from the parameters like IP Address, Network ID, Host ID, File Content, File Name, CPU Utilization, Disk Utilization, Bandwidth Utilization, Username and Memory Utilization. The parameter and corresponding SecMet values are depicted in Table 4.

Table 4: SecMet Values for various Parameters

Parameter	**SecMet Value**
IP Address	103.344
Network ID	86.674
Host ID	20.0015
File Content	133.35
Filename	45.002
CPU Utilization	78.339
Disk Utilization	44.9975
Bandwidth Utilization	78.3339
Username	86.674
Memory	78.339

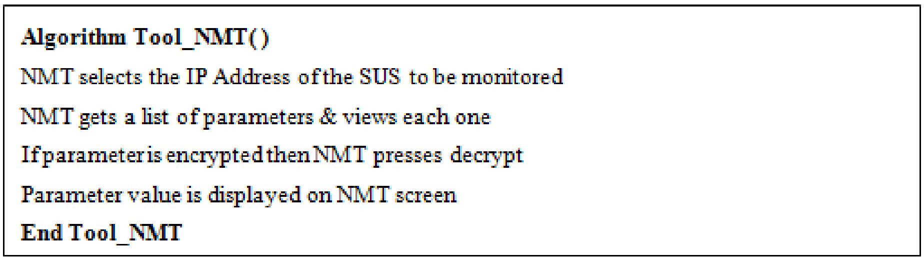
In the case of highly critical, more detailed and amount of the size of data is more, then the security metrics SecMet(SM) value reaches 133; means 100% security is required for this parameter. Therefore, it is required to encrypt the parameter and send it across the network (for example File Content and IP Address). In the case of less critical, less detailed and less sized parameter the SecMet value reaches approximately equal to zero.



```
Algorithm Tool_SUS( )
Each node in a distributed system runs an SUS application
Each SUS sets the C,D and S values for each of the parameter//Set_Parameter_Values()
For each parameter calculate SM by using a,b and c non-zero coefficients; //SM=aC+bD+cS
If SM >= 100 then "Encryption()
SUS node selects NMT IP address for giving permissions to monitor
End Tool  SUS
```

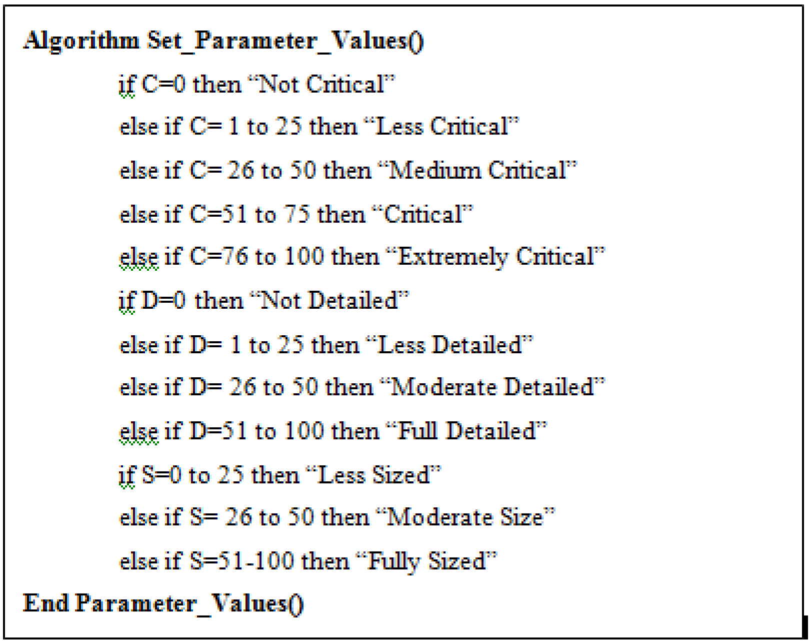
Figure 5: Algorithmic descriptions of SUS

Then monitoring node decides about whether to monitor the parameter with or without encryption. Depending upon the security requirement for the each parameter, that means, depending on SecMet values of each parameter, it is necessitate using with encryption or without encoding. The parameters are designated by the SUS system and the value of SecMet is calculated. As Grounded on the value of each parameter (level of criticality, level of detail and level of size), the parameter is monitored with or without encoding. The encryption technique used here is Rijndael Algorithm. The three modules have been used to describe the working procedure of the secure customized monitoring tool. The Algorithm Tool_SUS as depicted in Fig. 5 and it describes the working of SUS nodes. Here in which each distributed node runs an SUS application and sets the C, D and S values for each parameter depending on the attributes criticality, detail and size. The application then calculates SecMet values. If the Security Metric's value is greater than value 100, then it is need to encrypt the parameters otherwise no need to encrypt the parameter.



```
Algorithm Tool_NMT( )
NMT selects the IP Address of the SUS to be monitored
NMT gets a list of parameters & views each one
If parameter is encrypted then NMT presses decrypt
Parameter value is displayed on NMT screen
End Tool_NMT
```

Figure 6. Algorithmic description of NMT



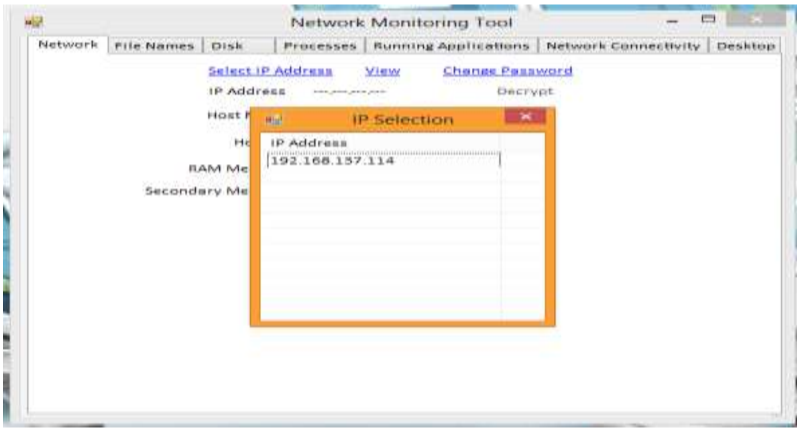
```
Algorithm Set_Parameter_Values()
        if C=0 then "Not Critical"
        else if C= 1 to 25 then "Less Critical"
        else if C= 26 to 50 then "Medium Critical"
        else if C=51 to 75 then "Critical"
        else if C=76 to 100 then "Extremely Critical"
        if D=0 then "Not Detailed"
        else if D= 1 to 25 then "Less Detailed"
        else if D= 26 to 50 then "Moderate Detailed"
        else if D=51 to 100 then "Full Detailed"
        if S=0 to 25 then "Less Sized"
        else if S= 26 to 50 then "Moderate Size"
        else if S=51-100 then "Fully Sized"
End Parameter_Values()
```

Figure 7. Algorithmic description of setting parameter values

The Algorithm Tool_NMT is described in Fig 6. Initially the NMT node selects the IP Address of the SUS node to be monitored. The NMT can view the list of parameters and can view values of each parameter by clicking on decrypt option. After decryption, the corresponding parameter value will be displayed on the NMT screen. The Fig 7 depicts that, how to set the values for each parameter of corresponding attributes in security metrics.

Results of System under Study and Network Monitoring Tool

The customized monitoring tool is running over the system and the values of each parameter are specified by the user or SUS node. In order to calculate the security metric values, the user or SUS node must provide the level criticality, detail and size of the parameters. Also, the SUS node must select the IP Address of the NMT for giving permission to monitor SUS by NMT.

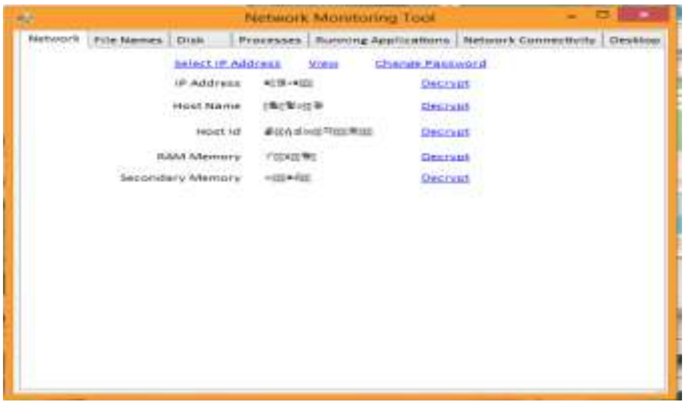


IP Address ---.---.---.---

IP Address
192.168.157.114

Figure 8. NMT Selects IP Address of SUS

The SUS node configures its security metrics values by setting the three values C, D and S for each parameters. Based on the values of C,D and S, the system calculates the SM value. The IP Address of the SUS to be monitored is selected and viewed by NMT is as shown in Fig 8. The NMT system selects the IP Address of different SUS system one at a time and presses view button to gather the information about SUS node. The NMT node selects the SUS IP Address over the network, alongside number of other systems are also exists.Based on the SM value of each parameter, the NMT node gathers information about SUS in the encrypted format as shown in Fig 9.



IP Address [illegible]

Host Name [illegible]

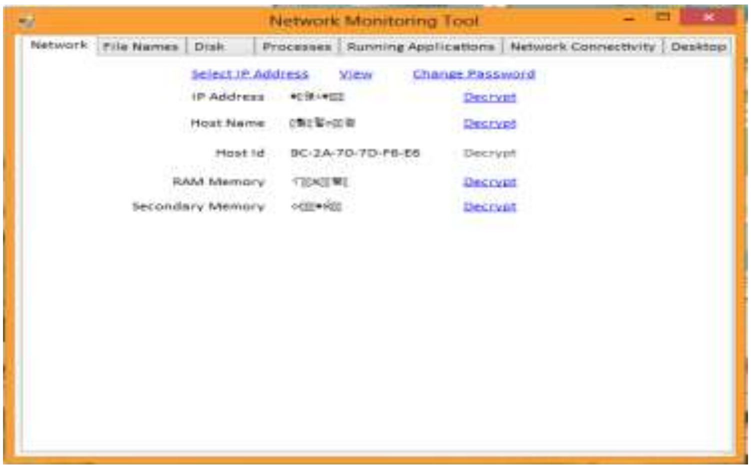
Host Id [illegible]

RAM Memory [illegible]

Secondary Memory [illegible]

Figure 9. Encrypted details of SUS viewed by NMT

The NMT node needs to click over the decrypt button and supply the key for getting the information about parameters of SUS. The NMT system supplies valid key for decryption process and decrypts value of Host-ID of SUS system. The actual Host-ID of SUS system is displayed on NMT screen as shown in Fig 10.



IP Address [illegible]

Host Name [illegible]

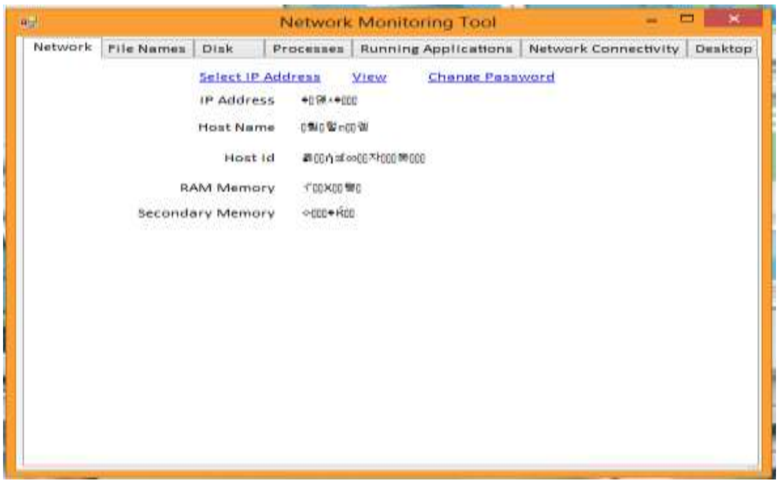
Host Id 9C-2A-70-7D-F6-E6

RAM Memory [illegible]

Secondary Memory [illegible]

Figure 10. Decrypted Host ID of SUS

Suppose if any SUS system not gives its permissions to monitor the system by NMT, then system can view only encrypted details of SUS as shown in Fig 11. The intruder node may tries to gather the information about SUS, during this time intruder also gets the same encrypted format of data.



IP Address [illegible]

Host Name [illegible]

Host Id [illegible]

RAM Memory [illegible]

Secondary Memory [illegible]

Figure 11. SUS not allowed to Monitoring by NMT

Likewise NMT may monitor IP Address, Host ID, Memory, HostName, File Name, Number of Disk, Disk spaces, Processes, Number of running applications and Network Connectivity of various SUS.

VI. analysis of results

The Table 5 shows the clear picture of advantages and drawbacks of security framework[18]. As per the results obtained, the different sizes of messages with encryption and without encryption are sent over the network. Primarily, result shows that with encryption of the messages takes more processing time.The RSA algorithm used here to encrypt the message. Whereas sending messages without encryption takes less processing time. On the other hand, these messages are not confidential during transmission.

Table 5: Comparison of Processing Time of Messages

Size of Message	With Encryption	Without Encryption
154	12.2	3.2
195	14.4	4.3
320	12.9	3.1
890	13.6	3.9

The Fig 12 depicts comparison of processing time of different sizes of messages which passes through the network. Before sending the message, if the message is encrypted and sent it to destination takes more time for processing. Whereas unencrypted messages are processed faster[19]. The secrecy of message is needed to be taken care during monitoring the distributed system. At the same time, the performance of the entire system also needs to be considered. The experimental results show that, if the user tries to protect the monitoring parameters with encryption then the performance of the system decreases.

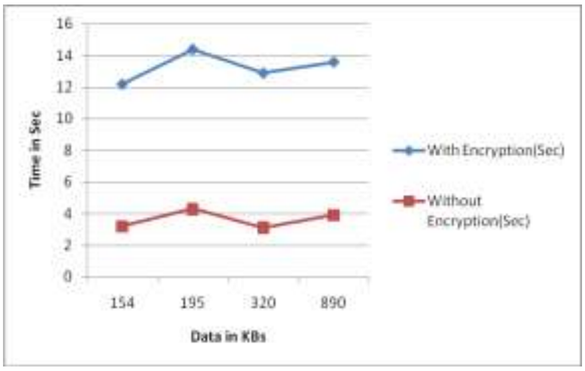


Figure 12. Comparison of Processing Times of Different Sizes of Messages

Security Metrics Calculation and Comparison

The Security Metrics function gives the SM value for various parameters used in monitoring of the networks. The Table 6 depicts the different security metrics values for various parameters of existing system security metrics (K)[8] and developed security metrics(SM).

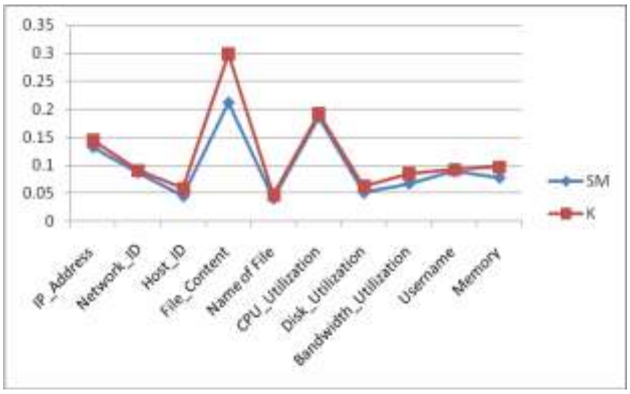


Figure 13. Comparison of existing system with developed system

The IP Address and File Content both exceeds the value 1.0, so these parameters are needed to be encrypted before sending it to the destination in both the cases. The parameter whose SM value exceeds the numerical value 1.0 is needed to be encrypted during the monitoring system. The security metric helps to avoid encryption process. The avoiding encryption by security metrics values reduces the overall burden of distributed system. The Fig 13 shows time complexity of security metrics calculation of both the technique, such as existing security metrics (K) and our developed security metrics (SM). For instance, the taken to calculate File_Content parameter security metric is comparatively lesser than existing system(K). This proves that, our developed system is less time consuming than the existing system. All the time values are taken in second(sec) time unit here.

Table 6: Comparison of Security Metrics Values SM with existing K

Parameters	Security Metrics(SM)	Existing Security Metrics (K)
IP Address	0.13234	0.14456
Network ID	0.08746	0.09023
Host ID	0.04523	0.05874
File Content	0.21135	0.29844
Name of File	0.04127	0.04675
CPU Utilization	0.18629	0.19145
Disk Utilization	0.05174	0.06187
Bandwidth Utilization	0.06783	0.08479
Username	0.08941	0.09179
Memory	0.07833	0.09654

VII. conclusion

The formulation framework for security mechanisms in the context of ADSs defines the security mechanisms during monitoring process. The frameworks initial component express about the study of existing monitoring tools and its drawback in terms of monitoring the distributed environments. Implementation of secure communication channel for monitoring is considered as the second component of the formulation of frameworks. The third component secure customized network monitoring tool defines the security metrics for each parameters of distributed systems. This customized monitoring involves two phases. In the first phase System Under Study(SUS) sets the security level of its parameters like IP Address, Host-ID etc. It provides a security to its parameters in such a way that only NMT node is permitted to monitor this SUS. If the security metrics value of particular parameter is high then it encrypts the parameter. The second phase involves secure monitoring, the Network Monitoring Tool(NMT), monitors the different parameters of SUS in encrypted format or unencrypted format depending on the security level set by SUS. The monitoring node decrypts its values by using valid key. The intruders have no chance to gather this information; since all the confidential parameters are in encrypted format. The security metric comparison gave that, developed customized monitoring tool is less time consuming than existing system. The future work involves the usage of gathered information in adaptive distributed systems functionalities like adaptive load balancing and adaptive network congestion.

References

[1] G. Couloris, J. Dollimore, and T. Kinberg, *Distributed Systems – Concepts and Design*, 4th Edition, Addison-Wesley, Pearson Education, UK, 2001.

[2] Jorma Jormakka , Jan Lucenius, Intruder Detection System Architecture for an SOA-based C^4I^2SR System Computation World: Future Computing, Service Computation, Cognitive, Adaptive, Content, Patterns, 2009

[3] A. Tanenbaum and M. Van Steen, *Distributed Systems: Principles and Paradigms*, Prentice Hall, Pearson Education, USA, 2002.

[4] R. D. Schlichting (1998). Designing and Implementing Adaptive Distributed Systems, available at http://www.cs.arizona.edu/adaptiveds/overview.html.

[5] Scarlet Schwiderski, Monitoring the Behaviour of Distributed Systems, Selwyn College University of Cambridge, A dissertation submitted for the degree of Doctor of Philosophy April-1996.

[6] F. M. Silva, Endler, and K. Fabio (2002). Dynamic adaptation of distributed systems, in the 16th European Conference on Object-Oriented Programming.

[7] Demissie B. Aredo, METRICS FOR QUANTIFYING THE IMPACTS OF MONITORING ON SECURITY OF ADAPTIVE DISTRIBUTED SYSTEMS. MASTER THESIS PROPOSAL – II, http://www.ifi.uio.no/~demissie (December 2005.)

[8] Aredo D. and Yildirim S., Security Issues in Adaptive Distributed Systems. Proceedings of the Fourteenth European Conference on Information Systems (ECIS 2006), Goteborg, Sweden.

[9] Reijo M. Savola, Habtamu Abie, Identification of Basic Measurable Security Components for a Distributed Messaging System, Third International Conference on Emerging Security Information, Systems and Technologies (IEEE,2009)

[10] Yin Guohui, Gong Wei, Application Design of Data Packet Capturing Based on Sharpcap, Fourth IEEE International Joint Conference on Computational Sciences and Optimization, , pp. 861-864, 2011.

[11] Shiping Chen, Surya Nepal, Suraj Pandey, A Unified Monitoring Framework for Distributed Information System Management,8th International Conference on Computing Technology and Information Management(ICCM), IEEE 2012, pp 259-264.

[12] Teemu Kanstrén, Reijo Savola, Sammy Haddad, Artur Hecker, An Adaptive and Dependable Distributed Monitoring Framework, International Journal on Advances in Security, vol 4 no 1 & 2, year 2011, http://www.iariajournals.org/security/

[13] Marco Comuzzi, Ruben Ivan Rafael Martinez, 2014 IEEE 8th International Symposium on Service Oriented System Engineering, 978-1-4799-3616-8/14 $31.00 © 2014 IEEE, DOI 10.1109/SOSE.2014.19. pp 122-127.

[14] Cesar Hernandez, Luis F. Pedraza, Camila Salgado, A proposal of traffic model that allows estimating Throughput mean values, 27th International Conference on Advanced Information Networking and Applications Workshops-2013.

[15] Mohamed Firdhous, Implementation of Security in Distributed Systems – A Comparative Study, International Journal of Computer Information Systems, Vol. 2, No. 2, 2011(ISSN 2229 5208).

[16] Kazi M Jahirul Islam*, Behrooz A. Shirazi*, Lonnie R. Welch+, Brett C. Tjaden+, Charles Cavanaugh*, Shafqat Anwar, Network Load Monitoring in Distributed Systems, Springer-Verlag Berlin Heidelberg 2000

[17] Shashi Mehrotra Seth, Rajan Mishra, Comparative Analysis Of Encryption Algorithms For Data Communication, International Journal of Computer Sci ence and Technology(IJCST) Vol. 2, Issue 2, June 2011(ISSN: 09768491-Online)

[18] Chun-Chieh Yang1, Ssu-Hsuan Lu1, Hsiao-Hsi Wang1, and Kuan-Ching Li2, On Design and Implementation of Adaptive Data Classification Scheme for DSM Systems, ISPA 2006, LNCS 4330, pp. 794 – 805, 2006.Springer-Verlag Berlin Heidelberg 2006.

[19] Deepak Jeswani, Maitreya Natu, R. K. Ghosh, Adaptive Monitoring: A Framework to Adapt Passive Monitoring using Probing, 8th International Conference on Network and Service Management (CNSM 2012), pp 350-356.

[20] Anandan, Sathya, Online Application Monitoring Tool, Master's Theses and Graduate Research,San Jose State University, 2010, http://scholarworks.sjsu.edu/etd_projects/.

[21] Mekonnen Feyissa, “Monitoring Distributed Systems For Adaptive Security”, Master Thesis, Department of Computer Science, School of Graduate Studies of Addis Ababa University, Addis Ababa, 2007.

[22] Richard Wiener, Remoting in C# and .NET,JOURNAL OF OBJECT TECHNOLOGY, Vol. 3, No. 1, January-February 2004, Published by ETH Zurich. Online at http://www.jot.fm . [last access January 2016].