Security Advanced Framework for the Robust Systems Monitoring

Parikshith Nayaka S. K., Dayanand Lal N., Brahmananda S.H., Neetha K. S.

Abstract: In the present communication technology the distributed networks have a vibrant job, Whether it's statutory or non-governmental organisations. The significant worry of the present IT industry is that stability, adaptability and the complexities of the distributed systems are handled. Robust distributed system (RDS) are hubs in organized PCs, which changes itself as indicated by changes to conditions. A study framework or tool is utilized to distinguish the adjustments in the distributed frameworks and many of the activities of the whole system. The intruder could compromise this monitoring system while gathering the data from the distributed framework. The following task will discuss this paper, the framework of security approaches for studying RDS. Initially, work into current studying tools to assess the impact of monitoring practices in RDSs. Once security-sensitive information is collected by the monitoring tool, the risk of information being leaked to unauthorized users is high, Secondly, a safe corresponding channel was introduced using the RSA algorithm to track confidential information. Thirdly, A stable, personalized network monitoring tool was introduced to provide the necessary protection for each parameter in the system. Protection metrics are used to calculate the security levels of each constraint to be monitored.

Keywords: Robust Distributed systems, Security approaches, Network Monitoring Tool.

I. INTRODUCTION

Distributed networks have a notable effect on private and public organizations in the current internet public of concerns. Depending on the circumstances, information production increases or decreases periodically. Consequently, the value of the criticality of information also dynamically changes. Distribution of specific website information and resources is widespread nowadays [1]. Monitoring the distributed networks of ever more complex IT environments is very important [2]. Monitoring activities involve monitoring the performance of heterogeneous distributed information systems and the use of resources. The standard distributed system consists of modules that are installed in various subsystems, or termed policies for amenities and security. The amenities run as autonomous processes on various computer systems. For market functionalities such as suppliers and finance, an enterprise system most often needs to contact external services [3]. Monitoring is seen as costly for the various IT infrastructure business activities. While accessing information users expect a high degree of system performance and a more secure environment. The distributed systems need an intrinsic adaptation of these problems. With support of Robust Distributed Systems (RDSs)[4], this adaptation can be accomplished. A RDSs is a device that changes its output vigorously built on surrounding variations. The method of Adaptation helps distributed systems to revisit their own settings and roles for rising environment. Observing demonstrates the best robust ability of the system to take appropriate action regarding changing conditions. Whether the system is robust or non-robust, security problems are the main alarms. The robust monitoring system changes its performance depending on the criticality of the data collected for the purpose of adaptation [5]. The safety system first enables the encryption process designed for critical safety info throughout the control of target structures. Unauthorized users can hack the distributed systems in numerous ways, whichever through transmission or at storage [6]. A significant security issue is likely if intruders hack the observing process during the time of gathering information. During this robust monitoring process, intruders can overtake the monitored systems; meaning the chances of security breaks is greater whereas the system is being monitored. One solution to addressing such safety glitches in distributed systems is the design of security mechanisms for managing Robust distributed systems.

Only through constant monitoring do the distributed systems have thoughtful functionalities and facilities incessantly with the correct level of excellence. The subsequent variations must be tracked using the monitoring tool that includes node IP address, host ID, network ID, processes, running programs, memory, disk, connection failures, etc. There is a risk that non-authorized users will bypass the monitoring system. In such situations, if the monitoring mechanism is limited for security purposes; this could result in serious constraints on the capability of the adaptive system. Adapting to the safety mechanism with minimal impact is a difficult task [7]. Several issues need to be thoroughly investigated in this regard, such as what information can be observed, how it can be checked, and What effect does detection have on safety? It has been predicted that, based on user requirements, different safety levels of observing constraints exist. Those levels of security are calculated by means of safety metrics.

Revised Manuscript Received on February 06, 2020.

***** Correspondence Author

Mr. Parikshith Nayaka S. K.*, Department of CSE, GITAM School of Technology, Bengaluru, India. Email: nayaka.parikshith@gitam.edu

Dr. Dayananda Lal N., Department of CSE, GITAM School of Technology, Bengaluru, India. Email: dayanandlal@gmail.com

Dr. Brahmananda S. H., Department of CSE, GITAM School of Technology, Bengaluru, India. Email: brahmananda.savadatti@gitam.edu

Mrs. Neetha K. S., Department of CSE, GITAM School of Technology, Bengaluru, India. Email: neeta.srinath@gitam.edu

Security measures are likely to assess the safety risk level on the basis of data characteristics related to security problems. During its calculation [8], The security attributes include criticality, precision, scale, and implication support.

Safety requirements, such as secrecy, are achieved only by safety measures. Action can be taken on the basis of the security metric values to progress the general safety System and also the safety risks found. Such protection metrics show the efficacy of a protected methodology with different methods. Sharing information via the network is nowadays a hurdle. Two issues during information sharing are very important to take care of Quality of Service (QoS) and Security. Distributed Systems is one type of system that offers QoS while accessing information from remote sites. To improve the QoS, the distributed systems need to be made as adaptive. Such transition begins with the three different phases, such as constant monitoring, detection of environmental changes and implementation of the required changes. Constant monitoring of systems [20] could lead to security issues. Thus, protection measures should be established during distributed network surveillance. Security-measurability-enhancing measures are critical for greater adoption of safety indicators, assessments, and related tools and processes. These are the research questions that gave motivating factors to analyze the problem closely and to propose the solution structure.

II. RELATED WORK

Demisie B. Ardo et al. [7] emphasized the two key issues in the distributed robust systems. Primarily, control of the network to gather information required for reworking can trigger safety issues. Second, limiting surveillance and information gathering can limit the system's ability to adjust to fluctuating environments and preserve the safety procedure. Therefore, there is a possibility of undermining the entire security process of rendering a sensitive distributed network adaptable to cope with security threats. The writers do not discuss how an implementation can be accomplished with negligible effect on its safety procedure. However, the writers do not address what type of data to track and how to observe without compromising the efficiency of the distributed monitoring system.

Sulee Yildrim et al. [8] suggested the protection measure for measuring the effects of tracking the efficacy of goal systems ' safety procedure. The measurements are based upon a set of data attributes to collect, which are important for enforcing the protection, by controlling the device. Yildirim Sulee et. al. [8] have the subsequent features listed as important for issues of protection, and metrical attributes have been defined: criticality level, information, size, and inferences support level. For certain nonnegative coefficient $\alpha\beta$ β, μ and μ the security metrics were defined in the equation $M= \alpha.C+ \beta D + \alpha + \beta$. Several computational methods have been used to evaluate certain coefficients ' meanings and relationships. Symbolically, the SM is 1/M where the efficiency of the target system's security mechanism is SM. Sulee Yildrim et al. [8], however, does not discuss behaviors to reduce monitoring during the construction of surveillance systems. There is also no mention of important security problems such as authorization, and encryption. The protection criterion criteria must be restated since the measurements of information are directly associated to their level of feature and information descriptions are unswervingly connected with inferencing assistance.

Reiijo M et al. [9] projected a safety metrics procedure. Threat and vulnerability analysis must first be performed. Conduct a hazard study of the analyzed device and its climate. Identify known vulnerabilities as suspects. Therefore, safety requirements need to be identified and goals focused on a systematic review of risks. Most attention should be paid to the most critical safety requirements. Be careful about the consistency of needs. Ultimately, a decomposition method will define essential observable components of high-level requirements. The Genetic Message Oriented Secure Middleware (GEMOM)[9] offers online protection measurements to track and execute adaptive security operations. A proactive component and a reactive portion are part of the monitoring system in GEMOM. The proactive party shall make long-term decisions in shape of safety indicators and parameters based on information on safety, trust, confidence, and repute. Sensitive component controls the flow, delays, and durability of messages. Appropriate health metrics and interventions are used to track.

Yiin Guohiui[10] et al. suggested in what way Sharpcap can be collected on the Net.Sharpcap is a progress set that catches network data on the first floor on dot network platform. The fundamental opinion is to collect all data, which flows through the device connector layer, by using the network adapter as the display template. Sharpcap is a device designed specifically for dot net to catch data packets. He was certain that he had easily, correctly, reliably and effectively designed an initial Sharpcap kit. The raw socket limitations illustrate Sharpcap's dominance in storing the data packet.

The centralized surveillance architecture for integrated information systems administration is laid out by Shiping Chen et coll. [11]. This system is used to capture the log data and simulate the business process using web service and Messaging Queue technologies. The functionality and efficiency of the suggested method are tested by a test device and tests are performed in direction to assess general performance of the monitoring structure. Tests were performed under different system loads with and without control. The experimental results indicate that the network control does not significantly change the system's overall performance. It can, therefore, be used in real-life applications for performance monitoring and tuning. The current framework is an only single way for proactive monitoring though.

The fundamental criteria for the design of secure, efficient and scalable shared monitoring system and reference architecture in the building of certain monitoring system to rectify such type of requirements were presented by Teemu Kanstrén et al. [12]. The requirements presented provide a framework to consider the diverse supervisory needs and how they relate to the fields of interest of the reader. Marco Comuzzi et al.

[13] addressed the issue of modifiable system observing business operations during multiple operation sytem monitoring. Such technique's architectural design distinguishes the monitoring issue from the generic, i.e. process structure, making the technique verifiable with alternative processing technologies. In this evaluation, the authors centered on the effectiveness of present process and computing.

III. DESCRIPTION PROBLEM

Security issues are key challenges for RDSs. This can partly be overcome by means of techniques like cryptography, audit systems, and regulation of access. There was also a method to monitoring message channels to get safe message, but monitoring a distributed network could trigger safety issues. Data about the many networking activities of users, their contact habits and the quality of the messages is gathered through a monitoring system that is usually outside the target structure. It is certainly a observing system that becomes supplementary informed about the world, it operates in, so improvements can be observed in the centralized setting and remedial steps can be taken to improve service quality [14]. Restricting monitoring can prevent security threats that occur during surveillance. Limiting the monitoring system may result in service quality being degraded. Furthermore, connections to the observing systems should be secured from intruders in the meantime.

IV. CONCEPTS AND GOALS

The Robust Distributed Systems Architecture Model for Security Mechanisms is suggested as shown in Fig 1. With the support of the current monitoring tool and new monitoring tool, the protection mechanisms are studied in the framework of Robust distributed systems. Wireshark tool [14] has been used in this research work for the analysis of the latest watching tool. The Wireshark tool helps in finding vulnerability to safety over distributed systems. If there is a safety vulnerability, otherwise encode all parameters and control centralized processes, otherwise, all parameters are checked explicitly without encryption. Each system in the distributed system sets a security metric for each parameter in case of a customized monitoring tool developed. The safety metrics are described with parameter criticality, description, and scale. The classified criteria need to be tracked with high-security numerical meaning. The custom testing method decisions on encryption according to the protection metric interest. In other terms, if a specific parameter's susceptibility to protection is strong, then encrypt and track the parameter. Otherwise, track certain parameters directly.

V. IMPLEMENTATION AND EVALUATION RESULTS

A. Inquiry into current reporting processes

The proposed system initially includes tracking distributed systems, as well as reviewing the current monitoring tool. The present watching tool called Wireshark is used to analyze the effects of monitoring in distributed systems. It is also possible to track a message that is transmitted among two users as well as data related to that communication.

The observing system tracks the period contact, the IP address of the source system and the destination network IP Address, Username and Exact message swapped, etc. An intruder may exploit this knowledge for other reasons. The Wireshark application is a packet analyzer in network and is an open-source tool that is commonly available. A packet analyzer collects the network message and aims to show as accurately as possible the packet data [16]. The consumer chooses the program from the Catch tab to catch the packets. Network packets are normally shown in the top with details such destination IP Address, protocol frequency and as source IP Address. The packet stream can be accessed by right-clicking on TCP file and must pick follow TCP stream.

The USER1, current in distributed system, is attempting to direct the message to USER2. The USER2 gets the notification in the mailbox and it opens the response to USER2. The current monitoring device Wireshark was used to display the message in these experimental results by using the option to follow the TCP path. Fig 4 demonstrates that one can control the other's details through the network by using the Wireshark Monitoring tool. Likewise, intruders exist in distributed systems can abuse this Wireshark monitoring tool to access the details

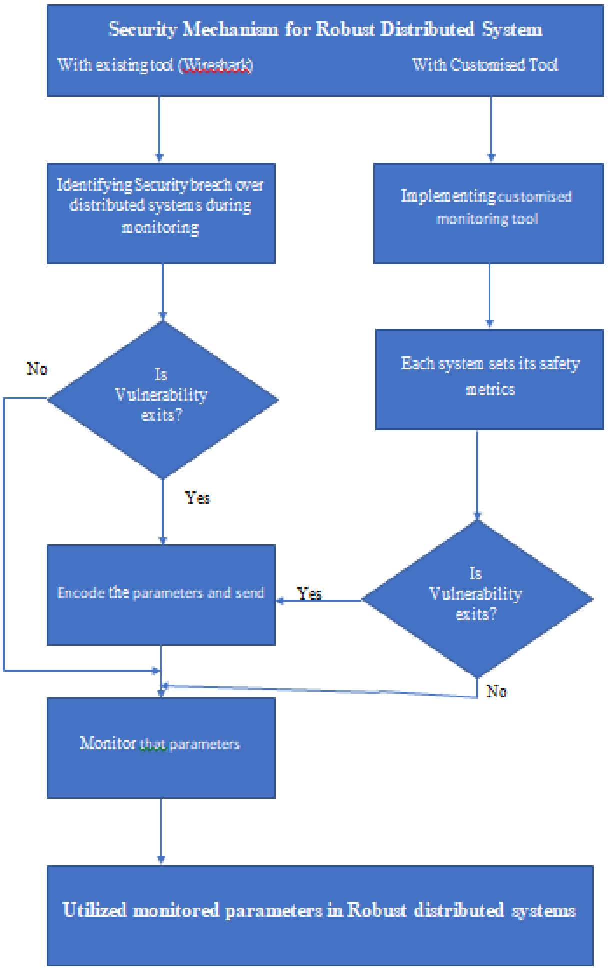
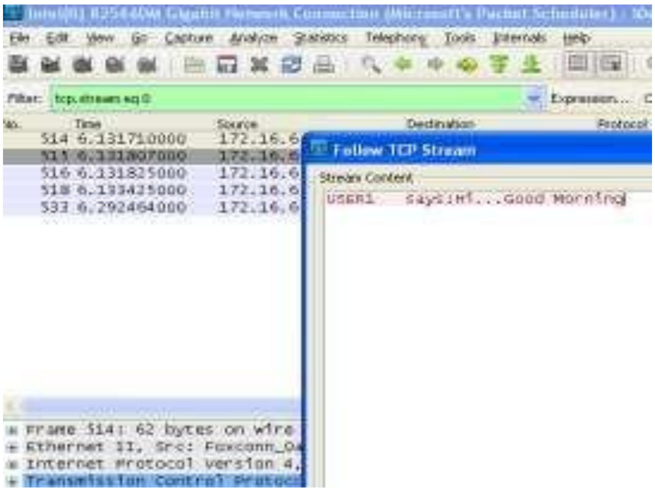

Security Mechanism for Robust Distributed System
With existing tool (Wireshark)
With Customised Tool
Identifying Security breach over distributed systems during monitoring
Implementing customised monitoring tool
No
Is Vulnerability exits?
Each system sets its safety metrics
Yes
Encode the parameters and send
Yes
Is Vulnerability exits?
No
Monitor that parameters
Utilized monitored parameters in Robust distributed systems

Fig 1: Safety Mechanism System Architecture

B. Secure Communication Channel Implementation

It has been observed in the previous section that the intruder that may present outside or inside the system may attempt to access the data that is transferred among two users. It was, therefore, appropriate to use the security mechanism to secure the information exchanged among the two users. The sender then had to encode the details using the RSA algorithm [17]. This procedure of encryption occurs after the sender clicks on the send button. The message was decrypted on the receiver side of the receiving message and displayed on the receiver inbox Whenever the attacker tries to view the message via this protected communication system, the device shows only the encrypted message as shown in Figure 3.

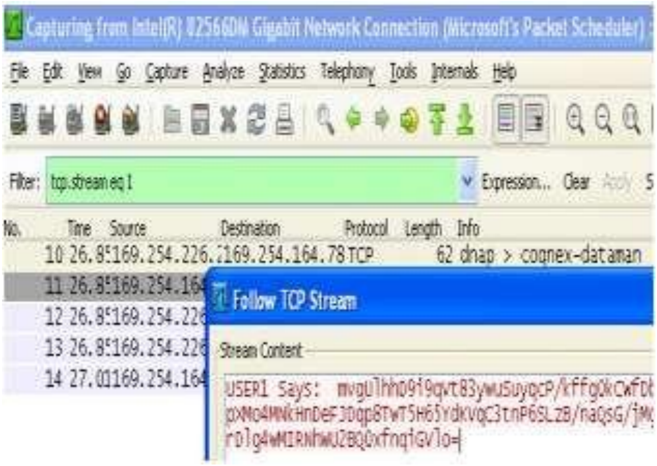

Intel(R) 82544DM Gigabit Network Connection (Microsoft's Packet Scheduler)
File Edit View Go Capture Analyze Statistics Telephony Tools Internals Help
Filter: tcp.stream eq 0
Expression...
Follow TCP Stream
Stream Content

```
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

+ Frame 514: 62 bytes on wire
+ Ethernet II, Src: Foxconn_0
+ Internet Protocol Version 4,
+ Transmission Control Protoco

USER1    says:Hi...Good Morning
```

Fig 2: Use Wireshark platform to track


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:  mvgUlhh09i9qvt83ywuSuyqcP/kffgOkCwfDt
pXMo4MNKHnDeFJDqp8TwT5H65YdKVqC3tnP6SLzB/naQsG/jM
rDlg4wMIRNhwU2BQ0xfnqiGVlo=
```

Fig 3: Use Wireshark platform to track

C. Secure Customized Monitoring Tool Implementation

For contact in two directions, a basic client-server program was developed. In this method, the removal strategy was used, as shown in Fig 4. In demand to build distributed applications, the use of remote platform in networking services is very important [22]. For remote access purposes, the client program will register with server service. The site and database program here utilize the authentication networks Ftp, SMTP, and TCP. The System. Net. Sockets API offers controlled implementation of the Windows Sockets protocol for developers requiring tight control of network access.

With this safe, customizable watching tool, devices are linked to a LAN / WAN where the system is named the System Under Study (SUS) and other systems are watched by the NMT. Security levels such as Host ID, User Name, IP Address, Server Title, Network ID, Port Number, File Information, Connection Error, Node Failure, Machine Response Time, Computer Usage, Workload, Ram, CPU, and Disk have been defined.

Security Metrics was established for the purpose that each parameter in the network is safe. In this relation, a personalized method is introduced. The aim of our specialized tool, in particular, is to investigate the degree of criticality of each parameter. The SUS program defines the parameter protection based on the level of vulnerability. Security metrics are specified for each parameter dependent on those security levels.

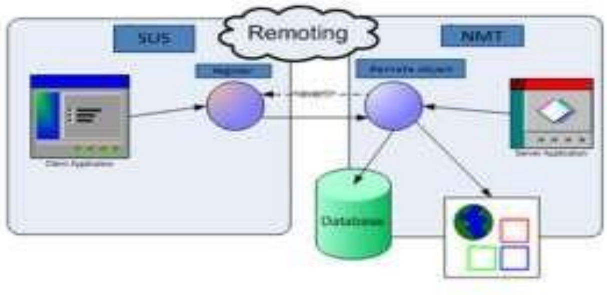


Fig 4: Availability

```
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
```

Fig 5: Definition of Tool_SUS algorithms.

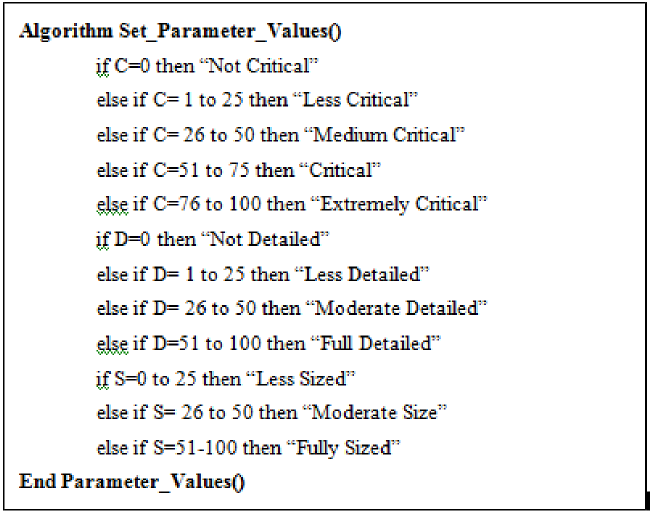
```
Algorithm Tool_NMT( )
NMT selects the IP Address of the SUS to be monitored
NMT gets a list of parameters & views each one
If parameter is encrypted then NMT presses decrypt
Parameter value is displayed on NMT screen
End Tool_NMT
```

Fig 6: Definition of Tool_NMT algorithms

The monitoring node then decides whether to observe the parameter with or without encoding. Depending on the security requirement for each parameter, that means using it with encryption or without encoding, depending on the SecMet values of each parameter. The parameters are defined by the SUS program, and the SecMet value is determined. The parameter is monitored with or without encoding, as it is based on the value of each parameter (level of criticality, level of detail and level of size). The method used for encryption here is Rijndael Algorithm. The three modules were used to define the Safe Customized Monitoring System operating process. The Tool_SUS algorithm, as shown in Fig. 5 And it explains how SUS nodes operate. Here, each distributed node is executed by a SUS application and sets the values of C, D, and S according to the criticality, information, and size of each parameter.

The SecMet values are then determined by the program. If the value of the Security Metric is greater than 100, then the parameters must be encrypted otherwise no need to encrypt the parameter.

The Tool NMT algorithm is defined in Fig 6. The NMT node initially selects the IP address of the monitoring SUS node. The NMT may view the parameters list by clicking the decryption option and viewing the values of each parameter. The relevant parameter value will then be shown on the NMT panel upon decryption. Figure 7 demonstrates that, in protection parameters, how to set values for each parameter with corresponding attributes.



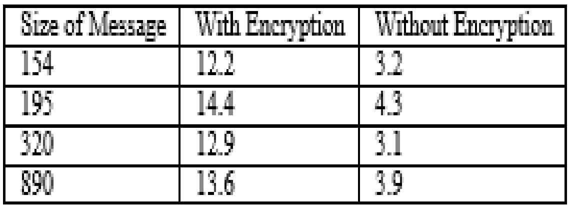
```
Algorithm Set_Parameter_Values()
        if C=0 then "Not Critical"
        else if C= 1 to 25 then "Less Critical"
        else if C= 26 to 50 then "Medium Critical"
        else if C=51 to 75 then "Critical"
        else if C=76 to 100 then "Extremely Critical"
        if D=0 then "Not Detailed"
        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()
```

Fig 7: Setting parameter values algorithmic overview.

VI. RESULT AND DISCUSSION

Table 1 shows a clear view of the defense framework's benefits and drawbacks [18]. The various sizes of messages are sent over the network, with encryption and without encryption, according to the results obtained. The result shows primarily that it takes more time to process the messages with encryption. The RSA process used here to encode the message. Whilst transfer messages without encoding takes fewer time to handle. On the other side, certain communications during transmission are not private.

Table 1: Comparison of message processing time



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

Figure 8 shows a difference of the processing time of dissimilar message sizes that pass over the network. Before transfering the message, it takes more time to process the message when encrypted and sent to the destination. Unencrypted messages are handled more quickly [19]. When watching the distributed system, the secrecy of the message must be taken into account. The performance of the whole system must also be taken into account at the same time. The experimental results indicate that the device efficiency decreases if the user attempts to safe the monitoring parameters with encryption.

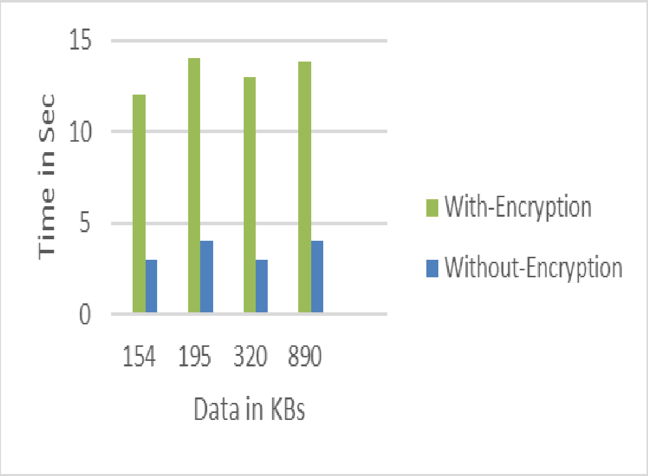


Fig 8: Evaluation of response times of messages of dissimilar sizes

Calculation and analysis of security metrics

The function Protection Metrics provide the SM value for different parameters used in network monitoring. Table 2 shows the different values of protection metrics used for different parameters of existing system security metrics (K) [8] and established safety metrics (SM).

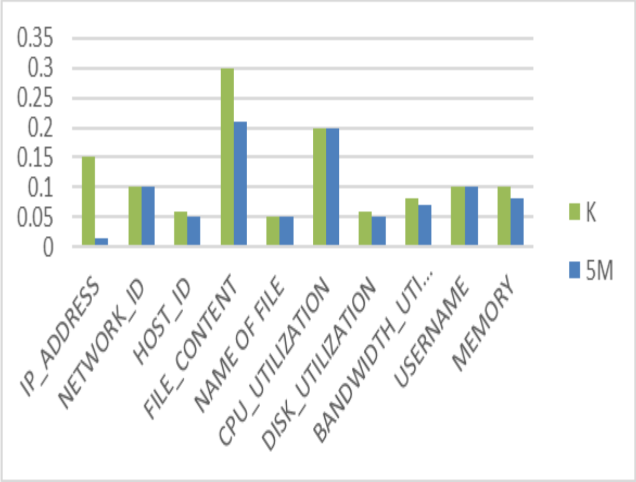
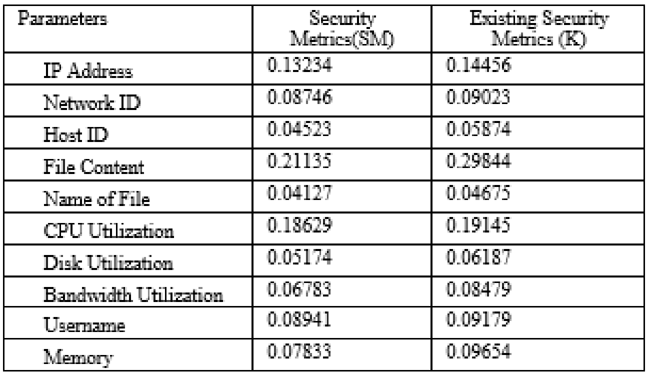


Fig 9: Comparison of the system existing with the system established

The File Content and IP Address both exceed the 1.0 value, so in both cases, these parameters need to be encoded before it is sent to the destination. During the monitoring system, the parameter whose SM value exceeds the numerical value 1.0 is required to be encoded. The security parameter helps prevent the cycle of encryption. The avoidance of encryption by the values of security metrics reduces the overall distributed network burden. Figure 9 shows the time complexity of both the technique's calculation of security metrics, such as existing security metrics (K) and our developed safety metrics (SM). For eg, the security metric used to quantify the File Content parameter is comparatively smaller than the current system(K). This demonstrates that our system developed takes less time than the existing system. Both time values are taken here in the time unit second(sec).

Table 2: Safety Metrics compared SM values with current K values



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

In the case of RDSs, the design structure for security mechanisms describes the security mechanisms during the monitoring phase. The initial aspect of the systems reflects the analysis of current monitoring tools and their disadvantage with regard to tracking the dispersed ecosystems. Implementation of a secure channel of communication for observing is measured the second module of frameworks formulation. The third protected aspect, personalized network monitoring tool determines the protection metrics for each distributed device parameter. This personalized surveillance requires more the one levels. Sets the safety level of its parameters such as, Host-ID, IP Address etc. In the first step of System Under Study (SUS). It provides safety for its parameters in such a way that it is only allowed to monitor this SUS by NMT node. If the parameter protection metrics rating is strong then the function is encrypted. The second phase relates to secure tracking the Network Monitoring Tool (NMT), which monitors the various SUS parameters in an encrypted or unencrypted format, depending on the level of security set by SUS. By using the correct key, the control node decrypts the values. Such information cannot be accessed by the intruders; all sensitive criteria are encrypted. The comparison of the security metrics showed that the customized monitoring tool developed is less time consuming than the existing system. The future work includes the use of collected information in functionalities of adaptive distributed systems such as adaptive charge balance and adaptive grid 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 C4I2SR System Computation World:
3. Future Computing, Service Computation, Cognitive, Adaptive, Content, Patterns, 2009
4. A. Tanenbaum and M. van Steen, Distributed Systems: Principles and Paradigms, Prentice-Hall, Pearson Education, USA, 2002.
5. R. D. Schlichting (1998). Designing and Implementing Adaptive Distributed Systems, available at http://www.cs.arizona.edu/adaptiveds/overview.html.
6. 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.
7. F. M. Silva, Endler, and K. Fabio (2002). Dynamic adaptation of distributed systems, in the 16th European Conference on Object-Oriented Programming.
8. 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.)
9. Aredo D. and Yildirim S., Security Issues in Adaptive Distributed Systems. Proceedings of the Fourteenth European Conference on Information Systems (ECIS 2006), Goteborg, Sweden.
10. 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)
11. 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.
12. 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.
13. Teemu Kanstrén, Reijo Savola, Sammy Haddad, Artur Hecker, An Adaptive and Dependable Distributed Monitoring Framework,
14. International Journal on Advances in Security, vol 4 no 1 & 2, the year 2011, http://www.iariajournals.org/security/
15. 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.
16. 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.
17. 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).
18. 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
19. Shashi Mehrotra Seth, Rajan Mishra, Comparative Analysis Of Encryption Algorithms For Data Communication, International Journal of Computer Science and Technology (IJCST) Vol. 2, Issue 2, June 2011(ISSN: 09768491-Online)
20. 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.
21. 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.
22. Anandan, Sathya, Online Application Monitoring Tool, Master's Theses, and Graduate Research, San Jose State University, 2010, http://scholarworks.sjsu.edu/etd_projects/.
23. 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.
24. 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]
25. S Kuragod, P Nayak, M Kotari – 2016, "Implementation of IBE with outsourced revocation technique in cloud computing", International Journal of Innovative Research in Electrical, Electronics, Instrument and Control Engineering (IJIREEICE) Vol. 4, Issue 5, PP. 190-193 DOI 10.17148/IJIREEICE.2016.4548
26. Sujatha Manni1, Parikshith Nayak -2015, "Apriori Based Muti-Keyword Search Over Encrypted Cloud Data", International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering (IJIREEICE) Vol. 3, Special Issue 1, PP. 234-236 DOI 10.17148/IJIREEICE

27. Manjunath Kotari, Dr. Niranjan N. Chiplunkar, Dr. Nagesh H.R, "Framework of Security Mechanisms for Monitoring Adaptive Disturbed Systems", IOSR Journal of Computer Engineering (IOSR-JCE) Volume 18, Issue 4, Ver. I (Jul.-Aug. 2016), PP 25-36.
28. Vikhyath K. B, Dr. Brahmanand S. H, "wireless sensor network security issues and challenges: A survey", "International Journal of Engineering & Technology" (IJET), (2018) PP. 89-94
29. [27] Dayanand Lal.N, Dr.G.Saravana Kumar, Dr. S.Ravi3, Dr.Anand, "Porting presentation layer to ensure network security in mobile devices" "International Journal of Applied Engineering Research", ISSN 0973-4562 Volume 10, Number 16 (2015) pp 37255-37260

AUTHORS PROFILE

Mr. Parikshith Nayaka S. K., is working as an Assistant Professor in the Department of Computer Science & Engineering, GITAM School of Technology, Bengaluru, Karnataka, India. He received his BE in CSE from the Visvesvaraya Technological University in 2010 and his MTech in CSE from the Visvesvaraya Technological University in 2012. His area of interest is Cryptography, Web Technologies, Networks, Wireless Sensor Networks, Blockchain Technologies. He has published around 8 papers in the referred international journals and conferences. He is also a life member of ISTE.

Dr. Dayanand Lal. N., Assistant Professor in CSE Department, GITAM School of Technology, Bengaluru campus. He has published three research papers "Porting presentation Layer to ensure network security in mobile devices", "configuring a secure wireless network using GNS3" and "Protective and Efficacious cloud evaluating Schema" in Scopus journals. The Research interest area is Cyber Security.

Dr. Brahmanand S. H., Professor and Head of Dept. CSE GITAM School of Technology, Bengaluru, Karnataka, India. I have also published article under reputed journals and some of them are lies under Scopus which have been identified by Elsevier Scopus among them "Review of Resource allocation in fog computing", "A Survey in IOT cyber-attacks and deep learning assistance that can be used to detect the attacks". And so on. The Research interest area is Cyber Security.

Mrs. Neetha K. S., is working as an Assistant Professor in the Department of Computer Science & Engineering, GITAM School of Technology, Bengaluru, Karnataka, India. Currently Perusing PhD in the area of Machine Learning and AI. She also published around two papers in reputed Journals and Conference.