International Journal of Psychosocial Rehabilitation, Vol. 23, Issue 04, 2019
ISSN: 1475-7192

Intrusion Detection and Vulnerability Analysis with Temporal Relationalship

T. Krishnakaarthik, K.C. Rajavenkatesswaran,
Dr.S. Nandagopal and Dr.P. Saveetha

Abstract--- *The network attacks are discovered using the Intrusion Detection Systems (IDS). Anomaly, signature and compound attack detection schemes are employed to fetch malicious data traffic activities. The attack impact analysis operations are carried out to discover the malicious objects in the network. The system objects are contaminated with process injection or hijacking. The attack ramification model discovers the contaminated objects. The dependency networks are build to model the information flow over the objects in the network. The dependency network is a directed graph build to indicate the data communication over the objects. The attack ramification models are designed with intrusion root information. The attack ramifications are applied to identify the malicious objects and contaminated objects. The attack ramifications are discovered with the information flows from the attack sources. The Attack Ramification with Bayesian Network (ARBN) scheme discovers the attack impact without the knowledge of the intrusion root. The probabilistic reasoning approach is employed to analyze the object state for ramification process. The objects lifetime is divided into temporal slices to verify the object state changes. The system call traces and object slices are correlated to construct the Temporal Dependency Network (TDN). The Bayesian Network (BN) is constructed with the uncertain data communication activities extracted from the TDN. The attack impact is fetched with loopy belief propagation on the BN model. The network security system is build with attack impact analysis and recovery operations. Live traffic data analysis process is carried out with improved temporal slicing concepts. Attack Ramification and Recovery with Dynamic Bayesian Network (ARRDBN) is build to support attack impact analysis and recovery tasks. The unsupervised attack handling mechanism automatically discovers the feasible solution for the associated attacks.*

Keywords--- *Intrusion Detection, Attack Ramification, Vulnerability Analysis, Temporal Dependency Network and Dynamic Bayesian Network.*

I. INTRODUCTION

Network forensics is the extended phase of network security as the data for forensic analysis are collected from security products like firewalls and intrusion detection systems. The results of this data analysis are utilized for investigating the attacks. There may be certain crimes breach network security policies but may be legally prosecutable. These crimes can be handled only by network forensics. Network forensics can be used to analyze how the attack occurred, who was involved in that attack, duration of the exploit and the methodology used in the

T. Krishnakaarthik, Assistant Professor, Department of Information Technology, Nandha College of Technology, Erode, Tamil Nadu, India. E-mail: krishnakumarbtech@gmail.com
K.C. Rajavenkatesswaran, Assistant Professor, Department of Information Technology, Nandha College of Technology, Erode, Tamil Nadu, India. E-mail: rajavenkates@gmail.com
Dr.S. Nandagopal, Professor, Department of Computer Science and Engineering, Nandha College of Technology, Erode, Tamil Nadu, India. E-mail: asnandu@gmail.com
Dr.P. Saveetha, Professor, Department of Information Technology, Nandha College of Technology, Erode, Tamil Nadu, India. E-mail: saveepme@gmail.com

Received: 16 Sept 2019 | Revised: 18 Oct 2019 | Accepted: 15 Nov 2019

attack [11]. It also helps in characterizing zero-day attacks. In addition, network forensics can be used as a tool for monitoring user activity, business transaction analysis and pinpointing the source of intermittent performance issues.

Network security protects system against attack while network forensics focuses on recording evidence of the attack. Network security products are generalized and look for possible harmful behaviors. This monitoring is a continuous process and is performed all through the day. But, network forensics involves postmortem investigation of the attack and is initiated notitia criminis. It is case specific as each crime scenario is different and the process is time bound. Network forensics is the science that deals with capture, recording and analysis of network traffic. The network log data are collected from existing network security products, analyzed for attack characterization and investigated to trace back the perpetuators. This process can bring out deficiencies in security products which can be utilized to guide deployment and improvement of these tools.

Network forensics is a natural extension of computer forensics. Computer forensics was introduced by law enforcement and has many guiding principles from the investigative methodology of judicial system. Computer forensics involves preservation, identification, extraction, documentation and interpretation of computer data. Network forensics evolved as a response to the hacker community and involves capture, recording and analysis of network events in order to discover the source of attacks.

In computer forensics, investigator and the hacker being investigated are at two different levels with investigator at an advantage. In network forensics, network investigator and the attacker are at the same skill level. The hacker uses a set of tools to launch the attack and the network forensic specialist uses similar tools to investigate the attack. Network forensic investigator is further at disadvantage as investigation is one of the many jobs he is involved. The hacker has all the time at his disposal and will regularly enhance his skills, motivated by the millions of dollars in stake. The seriousness of involved makes network forensics an important research field.

The concept of network forensics deals with data found across a network connection mostly ingress and egress traffic from one host to another. Network forensics tries to analyze traffic data logged through firewalls or intrusion detection systems or at network devices like routers and switches.

Network forensics involves monitoring network traffic and determining if there is an anomaly in the traffic and ascertaining whether it indicates an attack. If an attack is detected, then the nature of the attack is also determined. Network forensic techniques enable investigators to track back the attackers. The ultimate goal is to provide sufficient evidence to allow the perpetrator to be prosecuted.

Network forensic systems are classified into two types each based on various characteristics like purpose, collection and nature: ‘General Network Forensics’ to enhance network security and ‘Strict Network Forensics’ to get evidence satisfying legal principles and requirements. ‘Catch-it-as-you-can’ systems where all packets passing through a particular traffic point is captured and analysis is subsequently done requiring large amounts of storage and ‘Stop-look-and-listen’ systems where each packet is analyzed in memory and certain information is saved for future analysis requiring a faster processor. Nature: The network forensic system is an appliance with hardware and pre-installed software or exclusively a software tool.

Received: 16 Sept 2019 | Revised: 18 Oct 2019 | Accepted: 15 Nov 2019

II. RELATED WORKS

Most of existing works for fault detection/diagnosis in CPSs request the availability of analytical models describing the physical process under investigation. In this direction, [2] proposes a decentralized method for detecting and isolating multiple sensor faults in large-scale systems based on observers monitoring subsets of sensors. The detection of faults by each observer is based on an adaptive threshold on the residual, i.e., the discrepancy between what predicted by the a-priori known model and the acquired measurements. There, multiple-fault detection is achieved thanks to an aggregation mechanism that processes decisions gathered by all the observers. The considered application is a robotic-manipulator system encompassing eight sensors. Similarly, a nonlinear model-based observer method for detection, isolation and identification of multiple faults affecting actuators and sensors is proposed in [3]. In this work a simulated waste water-treatment system endowed with six sensors and four actuators has been considered. A similar work is described in [4], and applied to a three-tank system with six sensors and two actuators.

A different approach is presented in [5] that rely on the analysis of the residuals for sensor fault detection and isolation. This method has been applied to a non-isothermal continuous stirred tank reactor and a ternary distillation column envisaging nine sensors. An analytical redundancy-based approach is proposed in [6] that also works on residuals for fault detection and isolation. The experimental campaign includes synthetic data derived from a three-tank system endowed with five sensors and two actuators. In [7], the authors designed a hybrid Kalman filter integrating a mathematical model of the system and a number of piecewise linear (PWL) models. Fault detection is achieved by interpolating the PWL models using a Bayesian approach. Their method is applied on a dataset coming from a simulated gas turbine engine with five sensors.

A contained literature for model-free fault detection/diagnosis is available [8]. These solutions generally rely on machine-learning or statistical mechanisms to infer a model for the system under inspection directly from data. For example, a method based on Auto-Regressive with eXogenous input (ARX) model to characterize time-invariant relationships between sensors is presented in [9]. There, fault detection relies on the analysis of residuals, while fault isolation employs a graph-based analysis to identify anomalous patterns. The considered dataset refers to a physical plant containing 1091 sensors. There, the case of heterogeneous sensors is not considered, and time invariance for the plant under inspection is assumed.

A data-driven method based on Principal Component Analysis (PCA) and Fisher discriminant analysis to diagnose multiple faults is presented in [10]. PCA is applied to raw data for detecting anomalies by checking residuals, while Fisher discriminant analysis isolates the faults. The application refers to an air-handling unit composed of 13 sensors. An adaptive monitoring method based on residuals coming from a sliding window is presented in [1]. The method is applied to a real air-compression process monitored by 8 sensors.

Neural networks have been often considered in model-free solutions to model the unknown physical process [13]. For example, considers Artificial Neural Networks to identify and isolate multiple faults in an industrial motor network composed of 6 sensors. There, the fault dictionary is assumed to be apriori available. Finally, a swarm intelligent-based approach for the diagnosis of multiple faults is proposed in [8]. The fault diagnosis problem is

Received: 16 Sept 2019 | Revised: 18 Oct 2019 | Accepted: 15 Nov 2019

modified so that the presence of a specific fault is associated to each vertex explored by an ant. The experimental framework considers an industrial remote monitoring of operating machines with 20 sensors.

III. PROBABILISTICALLY INFERRING ATTACK RAMIFICATIONS

Assessing and mitigating the effects of successful attacks against computing systems are the natural and essential next step once attacks are detected. The central task of assessment and mitigation is to identify the ramifications of an attack, which include both malicious objects residing in a compromised system and objects that are contaminated by an attack. Successfully carrying out this task is faced with significant challenges. Attacks are usually very sophisticated, leveraging various vulnerabilities to compromise target systems and employing advanced attack vectors such as process injection/hijacking to subsequently contaminate system objects [11]. Attacks' high complexity is further compounded by their increasing stealthiness, which commonly offers attackers a considerable amount of time before they are detected.

A generic strategy to solve these challenges is to monitor the information flow among objects in a computing system. For example, when a process reads from a file, a potential information flow is generated from the file to the process. Specifically, if the file contains malicious content such as exploits, a vulnerable process might be compromised. The dependency network serves as an effective method to model the information flow. A dependency network is a directed graph, where an edge $e(v_i, v_j)$ indicates a potential information flow from the object vi to another object v_j . The provenance propagation methods employ dependency networks to identify all objects that have malicious information flows from the intrusion root, i.e., the entry point of an attack.

While these methods partially satisfy the objective to reveal malicious and contaminated objects in a system, their practical effectiveness is fundamentally constrained. The vast majority of these methods assume that the intrusion root is a known priori or can be easily located. Unfortunately, revealing intrusion root itself is a challenging task in practice considering the high complexity of object interactions in a system, the stealthiness of the attacks and particularly the uncertainty caused by incomplete knowledge of all malicious actions. Therefore, such assumption is easily invalidated in practice, rendering these methods ineffective. These methods are also vulnerable to dependency explosion, when a large number of intertwined but irrelevant object interactions are recorded and used to build dependency networks. For example, methods assume all objects that interact with a suspicious object are infected and use them to construct dependency networks. All processes that have read malicious files could be falsely considered as infected only a few of them are actually contaminated by malicious content. Such approaches not only result in unnecessarily large graphs but also incur excessive false positives.

A few attempts have been proposed to mitigate dependency explosion by leveraging fine-grained logging or tracking. While they certainly lead to dependency networks that characterize the propagation of malicious information with higher fidelity, the performance cost could be prohibitively expensive. For example, BEEP divides a process to autonomous units and subsequently establish dependencies at unit-level. BEEP requires binary instrumentation of applications and mandates a priori analysis of applications. Therefore, it faces the difficulty in scaling to a large number of applications that typically run in modern systems. A more ambitious method has been proposed to perform byte-level dynamic taint tracking analysis, which incurs substantial run-time overhead.

Received: 16 Sept 2019 | Revised: 18 Oct 2019 | Accepted: 15 Nov 2019

The attack ramifications are identified with a light weight method. The method tackles the problem of undetermined intrusion root by leveraging dependency relationships of information flows between undetermined objects and a subset of objects with known security states. It overcomes dependency explosion by fusing evidence from both known infected objects and known legitimate objects. Specifically, it leverages observations that i) an object could be infected if it has malicious information interactions with other known infected objects and ii) information interactions that involve known legitimate objects might be attack-irrelevant, thus providing clues for the real malicious information flows. The method does not rely on fine-grained logging or tracking techniques. Instead, it uses only coarse-grained events, minimizes human efforts and incurs low run-time overhead. The method splits the lifetime of an object into consecutive time slices to profile how the security state of this object changes over time and considers explicit flows between different objects and implicit flows between different slices of an object.

The method consists of three phases. First, it constructs a temporal dependency network (TDN) to correlate object-slices (states of objects at different time slices) based on the inter object and intra-object information flows between them. Next, it builds a Bayesian network (BN) based infection model to characterize the infection propagation in TDN as a random process. Bayesian Network is used to take advantage of its capabilities of probabilistic inference over structured dependencies in TDN. Finally, it performs loopy belief propagation on the BN-based model to infer the security state of an object.

IV. PROBLEM STATEMENT

The attack ramifications are applied to identify the malicious objects and contaminated objects. The attack ramifications are discovered with the information flows from the attack sources. The Attack Ramification with Bayesian Network (ARBN) scheme discovers the attack impact without the knowledge of the intrusion root. The probabilistic reasoning approach is employed to analyze the object state for ramification process. The objects lifetime is divided into temporal slices to verify the object state changes. The system call traces and object slices are correlated to construct the Temporal Dependency Network (TDN). The Bayesian Network (BN) is constructed with the uncertain data communication activities extracted from the TDN. The attack impact is fetched with loopy belief propagation on the BN model. The following problems are identified from the current attack ramification techniques. Attack handling mechanism is not provided. Supervised vulnerability release model requires expert support. Undetermined intrusion roots and dependencies are not handled. Dynamic network traffic flows and their risks are not handled.

V. ATTACK IMPACT DISCOVERY AND RECOVERY WITH DYNAMIC BAYESIAN NETWORKS

The network security system is build with attack impact analysis and recovery operations. The attack impact discovery process is a postmortem analysis on the traffic log data values. The traffic log data and object vulnerabilities are analyzed in the model. The traffic data values are partitioned with the temporal information. The temporal slices are prepared with the time slots. The Temporal Dependency Network (TDN) is build with the information flows that are generated in an object. The temporal dependency information is passed to the Bayesian

Received: 16 Sept 2019 | Revised: 18 Oct 2019 | Accepted: 15 Nov 2019

Network (BN) construction process. The probabilistic inference analysis model is applied to discover the attack impacts on the objects. All the attack impact analysis operations are carried out on the static data values.

The live traffic data analysis process is carried out with improved temporal slicing concepts. The information flows are dynamically analyzed with on streaming data values. The Temporal Dependency Network (TDN) construction process is also tuned to handle the live data streams. The object states and its communication information are analyzed and passed to the inference analysis process. The object states are analyzed with infected and legitimate conditions. The infected object communications are monitored separately.

Attack Ramification and Recovery with Dynamic Bayesian Network (ARRDBN) is build to support attack impact analysis and recovery tasks. The system calls with service requests and its temporal relationships are analyzed for malicious activities. The operating system provides various built in functions for service execution support. The operating system functions are referred as system calls. The vulnerabilities are identified with the associated system call information. The unsupervised attack handling mechanism automatically discovers the feasible solution for the associated attacks.

The attack detection and recovery operations are performed on the network traffic log data values. The temporal intervals are applied for the data analysis process. The attack handler is adapted to suggest the solution to face the attack impacts. The system is divided into six major modules. They are traffic data observation, temporal dependency analysis, Bayesian network analysis, system call traces, attack ramification discovery and attack recovery process.

The network traffic traces are captured under the network traffic data observation process. The temporal dependency analysis is carried out to discover the load levels. The Bayesian network analysis performs the attack discovery process. The system call traces are analyzed to identify the system resource utilization levels. The attack impacts are identified under the attack ramification discovery process. The attack recovery process automatically handles the relevant attacks.

5.1. Traffic Data Observation

The network traffic traces are monitored and updated into the log files. The source, destination, service, protocol and time information are collected from the network. The session identification is assigned for each network request. The payload levels for each requires is also observed from the traffic data. The user session details shows the user session ID and request frequency levels. The user access sequences display the list of requests and data communications initiated by the selected user with time information.

5.2. Temporal Dependency Analysis

The complete traffic data analysis is a complex task. So the traffic data values are analyzed in the sliding indow concept. The time interval based data partitions are prepared. The traffic data values are partitioned with time information. The temporal slices are formed with the traffic data values. The slicing operations are carried out for an interval time period. The temporal dependency network is build with the traffic data association levels. The temporal dependency graph represents the payload level for the traffic data.

5.3. Bayesian Network Analysis

The Bayesian network is constructed with the temporal dependency network. The traffic monitoring process is carried out to discover the traffic level in each network environment. The Bayesian network is constructed with service, protocol and system information. The probabilistic inference model is applied to discover the suspected objects in the network. The traffic ranges in each sub network environment is analyzed. The traffic traces are summarized in the Bayesian network with its relationship levels.

5.4. System Call Traces

The network data communications are carried out with different protocols based transactions. All the data access and service requests are initiated with the system calls or library functions. The network services are executed with the support of the operating system functions. The operating system functions are referred as system calls. The system calls and associated service requests are maintained in the system call traces. The attack discovery process is carried out with the support of the system call traces.

5.5. Attack Ramification Discovery

The attack impacts are identified with Attack Ramification with Bayesian Network (ARBN) scheme. System calls, Bayesian network and payload levels are combined in the attack discovery process. The infected node and its communication flows are verified in the attack discovery process. The impact is discovered on shared services, directory access, file download and mail phishing activities. Each user request is verified with the payload and malicious data transfer activities. The attack root discovery is not required in the model. The data communication over the nodes is analyzed with the infected object and legitimate object data transfer details. The data communication over the malicious nodes is also suspected as infected object. The inference analysis is carried out to take the decision o the objects.

5.6. Attack Recovery Process

The attack recovery is an essential task in the network security systems. The attack recovery model supports the attack control operations. The recovery operation initiates necessary steps to handle the attacks. The attack detection and recovery operation are carried out using the Attack Ramification and Recovery with Dynamic Bayesian Network (ARRDBN) scheme. The Dynamic Bayesian Network (DBN) updates the associated changes the in the objects. The automatic attack recovery process is initiated with relevant attacks levels. The recovery model controls the network resource vulnerabilities.

VI. RESULT AND DISCUSSION

The security analysis on Internet traffic data is carried out to detect the attacks on the computer systems. The traffic data values are collected from the Internet and its load levels are analyzed to discover the attacks. The attacks and its impacts are analyzed using the ramification process. The temporal dependencies are verified in the analysis process. The temporal dependency network and Bayesian network schemes are employed for the attack impact discovery process. The Attack Ramification with Bayesian Network (ARBN) scheme and Attack Ramification and Recovery with Dynamic Bayesian Network (ARRDBN) scheme are used in the network security analysis. The

Received: 16 Sept 2019 | Revised: 18 Oct 2019 | Accepted: 15 Nov 2019

(ARBN) scheme is build to handle the attack impact discovery process. Supervised attack recovery measures are taken with reference to the attack impact levels. The (ARRDBN) scheme is constructed to discovery and recovers the attacks. The attack recovery operations are automatically called with reference to the attack type and impact levels.

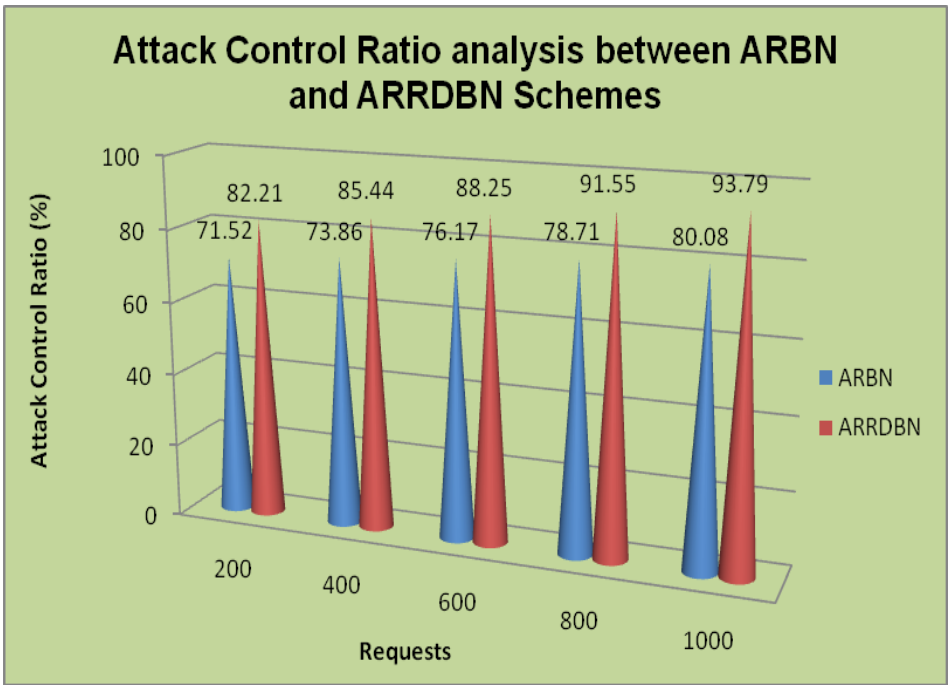

Figure 6.1: Attack Control Ratio analysis between ARBN and ARRDBN Schemes

The attack ramification and recovery scheme is tested with real network traces. The traces are collected from the active network environment. The source IP, destination IP, service name, protocol name, source bytes and time information are collected for the analysis. The system is tested with four performance measures. They are attack control ratio, delay, false positive rate and false negative rate measures. The attack control ratio analysis verifies the attack detection and prevention levels. The ratio between attack count and attack solution handle count is measured as attack control ratio. Figure 6.1. shows the attack control ratio analysis between the Attack Ramification with Bayesian Network (ARBN) and Attack Ramification and Recovery with Dynamic Bayesian Network (ARRDBN) schemes. The Attack Ramification and Recovery with Dynamic Bayesian Network (ARRDBN) scheme increases the attack control ratio 12% than the Attack Ramification with Bayesian Network (ARBN) scheme.

The delay analysis is carried out to measure the time period taken for the attack detection process. The difference between the detection process end time and start time is referred as delay time. Figure 6.2. shows the detection delay analysis between the Attack Ramification with Bayesian Network (ARBN) and Attack Ramification and Recovery with Dynamic Bayesian Network (ARRDBN) scheme. The Attack Ramification and Recovery with Dynamic Bayesian Network (ARRDBN) scheme reduces the attack detection delay 25% than the Attack Ramification with Bayesian Network (ARBN) scheme.

Received: 16 Sept 2019 | Revised: 18 Oct 2019 | Accepted: 15 Nov 2019

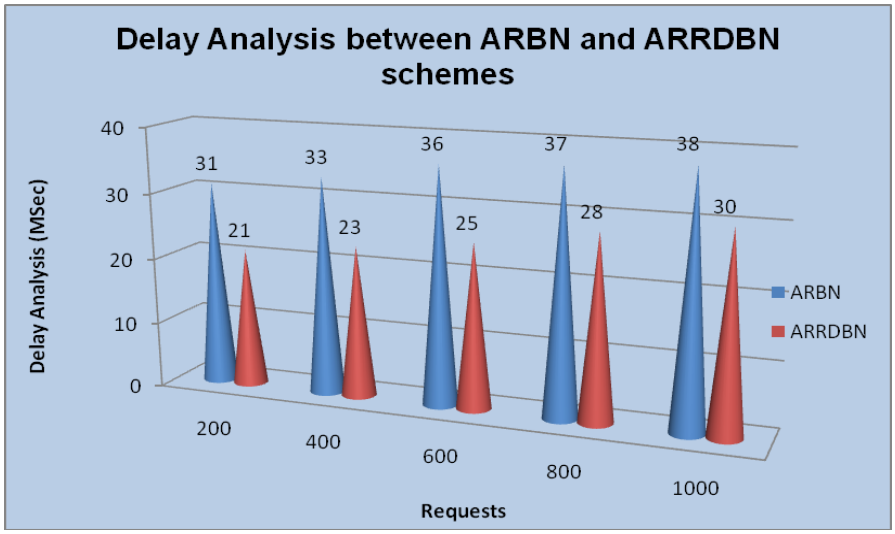

Figure 6.2: Delay Analysis between ARBN and ARRDBN Schemes

The false positive rate and false negative rate measures are used to identify the accuracy level of the attack detection process. The false positive rate is used to discover the ratio of falsely assigned false positive traffic flows. The false positive rate is calculated using the total traffic flow count and falsely assigned positive traffic flows. Figure 6.3. shows the false positive rate analysis between the Attack Ramification with Bayesian Network (ARBN) and Attack Ramification and Recovery with Dynamic Bayesian Network (ARRDBN) scheme reduces. The Attack Ramification and Recovery with Dynamic Bayesian Network (ARRDBN) scheme reduces the false positive rate 30% than the Attack Ramification with Bayesian Network (ARBN) scheme.

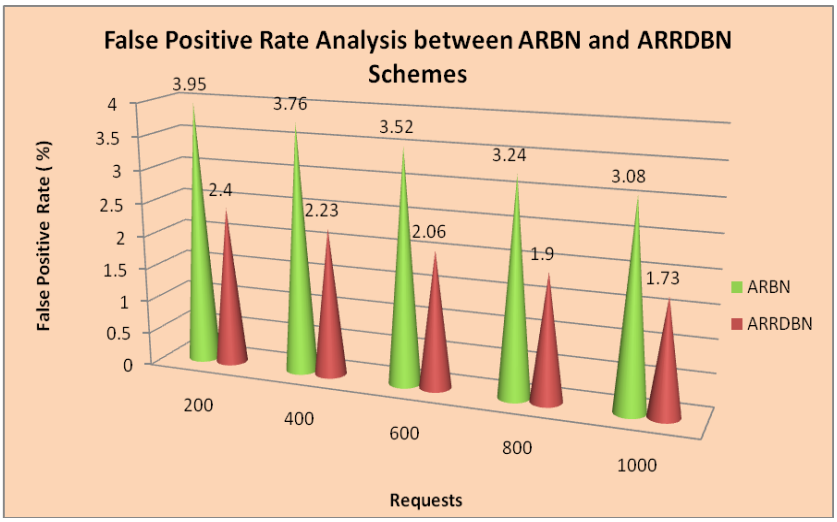

Figure 6.3: False Positive Rate Analysis between ARBN and ARRDBN Schemes

The false negative rate is used to discover the ratio of falsely assigned negative traffic flows. The false negative rate is calculated using the total traffic flow count and falsely assigned negative traffic flows. Figure 6.4. shows the false negative rate analysis between the Attack Ramification with Bayesian Network (ARBN) and Attack Ramification and Recovery with Dynamic Bayesian Network (ARRDBN) schemes. The Attack Ramification and

Recovery with Dynamic Bayesian Network (ARRDBN) scheme reduces the false negative rate 25% than the Attack Ramification with Bayesian Network (ARBN) scheme.

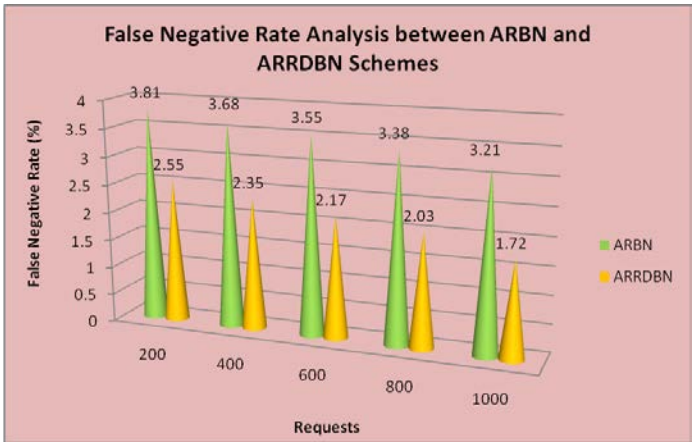

Figure 6.4: False Negative Rate Analysis between ARBN and ARRDBN Schemes

VII. CONCLUSION AND FUTURE ENHANCEMENT

The network security solutions are build with attack detection and recovery mechanisms. The Attack Ramification and Recovery with Dynamic Bayesian Network (ARRDBN) scheme is constructed to handle the attack impact discovery and recovery operations. The postmortem analysis is carried out on the network traffic data values to discover the vulnerability levels of the network resources. The automatic security solution suggestion mechanism supports the recovery process with suitable handler operations. The system can be enhanced with the following features. The attack impact detection and recovery process can be enhanced with cloud resource based data analysis mechanism. The attack discovery model can be enhanced to perform the traffic data analysis under distributed environment.

REFERENCES

[1] M.-W. L. DING-SOU CHEN and J. LIU, "Isolating multiple sensor faults based on self-contribution plots with adaptive monitoring," *China Steel Technical Report*, no. 24, pp. 64–73, 2011.

[2] V. Reppa, M. Polycarpou, and C. Panayiotou, "Decentralized isolation of multiple sensor faults in large-scale interconnected nonlinear systems,", *IEEE Transactions on Automatic Control*, vol. 60, no. 6, pp. 1582–1596, June 2015.

[3] D. Fragkoulis, G. Roux, and B. Dahhou, "A global scheme for multiple and simultaneous faults in system actuators and sensors," *in Systems, Signals and Devices, 2009. 6th International Multi-Conference on*, March 2009, pp. 1–6.

[4] L. Mhamdi, H. Dhouibi, N. Liouane, and Z. Simeu-Abazi, "Multiple fault diagnosis using mathematical models," *in Control Conference (ASCC), 2013 9th Asian*, June 2013, pp. 1–6.

[5] C.-C. Li and J.-C. Jeng, "Multiple sensor fault diagnosis for dynamic processes," *ISA Transactions*, vol. 49, no. 4, pp. 415 – 432, 2010.

[6] I. Issury and D. Henry, "A methodology for multiple and simultaneous fault isolation," *in Control Conference (ECC)*, 2009 European, Aug 2009.

[7] B. Pourbabaee, N. Meskin, and K. Khorasani, "Sensor fault detection, isolation, and identification using multiple-model-based hybrid kalman filter for gas turbine engines," *IEEE Transactions on Control Systems Technology*, vol. 24, no. 4, pp. 1184–1200, July 2016.

[8] P. Arpaia, C. Manna, and G. Montenero, "Ant-search strategy based on likelihood trail intensity modification for multiple-fault diagnosis in sensor networks," *Sensors, IEEE*, vol. 13, no. 1, pp. 148–158, Jan 2013.

[9] A. Sharma, H. Chen, M. Ding, K. Yoshihira, and G. Jiang, "Fault detection and localization in distributed systems using invariant relationships," *in Dependable Systems and Networks (DSN), 2013 43rd Annual IEEE/IFIP International Conference on*, June 2013, pp. 1–8.

[10] Z. Du and X. Jin, "Multiple faults diagnosis for sensors in air handling unit using fisher discriminant analysis," *Energy Conversion and Management*, vol. 49, no. 12, pp. 3654 – 3665, 2008.

[11] Yuan Yang, Zhongmin Cai, Chunyan Wang and Junjie Zhang, "Probabilistically Inferring Attack Ramifications Using Temporal Dependency Network", *IEEE Transactions on Information Forensics and Security*, Volume 13, Issue 11, 2018.

[12] Muhammad Ejaz Ahmed, Saeed Ullah and Hyoungshick Kim, "Statistical Application Fingerprinting for DDoS Attack Mitigation", *IEEE Transactions on Information Forensics and Security*, Volume: 14 , Issue: 6, June 2019.

[13] Alippi.C, S. Ntalampiras and M. Roveri, "Model-free fault detection and isolation in large-scale cyber-physical systems," *IEEE Trans. Emerg. Topics Comput. Intell.*, vol. 1, no. 1, pp. 61–71, Feb. 2017.

[14] Prakash, S. and Vijayakumar, M., "An effective network traffic data control using improved Apriori rule mining," *Circuits and Systems*, Issue 10, Vol. 07, pp. 3162-3173, June 2016.

[15] Sureshkumar V S, Chandrasekar A," Fuzzy-GA Optimized Multi-Cloud Multi-Task Scheduler For Cloud Storage And Service Applications" *International Journal of Scientific & Engineering Research*, Vol.04, Issue.3,pp-1-7, 2013.

[16] Preethi, B.C. and Vijayakumar, M. " A novel Cloud Integration Algorithm(CIA) for Energy Efficient High Performance Computing Applications in Big Data Multimedia Applications", *Romanian Journal of Information Science and Technology*, vol. 2, no.1, pp. 1-11, March 2018.

[17] Vijayakumar M, Prakash s, "An Improved Sensitive Association Rule Mining using Fuzzy Partition Algorithm", *Asian Journal of Research in Social Sciences and Humanities*, Vol.6,Issue.6, pp.969-981, 2016.

[18] Prakash S, Vijayakumar M, " Risk assessment in cancer treatment using association rule mining techniques", *Asian Journal of Research in Social Sciences and Humanities*,Vol.6,Issue.10, pp.1031-1037, 2016.

[19] Prabhakar E, " Enhanced adaboost algorithm with modified weighting scheme for imbalanced problems, *The SIJ transaction on Computer science & its application*,Vol.6,Issue.4, pp.22-26, 2018.

[20] Suresh kumar V S, Thiruvankatasamy S, Sudhakar R, "Optimized Multicloud Multitask Scheduler For Cloud Storage And Service By Genetic Algorithm And Rank Selection Method", Vol.3,Issue.2, pp.1-6, 2014.

[21] Prabhakar E, Santhosh M, Hari Krishnan A, Kumar T, Sudhakar R," Sentiment Analysis of US Airline Twitter Data using New Adaboost Approach", *International Journal of Engineering Research & Technology (IJERT)*, Vol.7, Issue.1, pp.1-6, 2019.

[22] Dhivyaa C R, Vijayakumar M," An effective detection mechanism for localizing macular region and grading maculopathy", *Journal of medical systems*, Vol.43, Issue.3, pp.53-, 2019.

[23] K Nithya, M Saranya, CR Dhivyaa, "Concept Based Labeling of Text Documents Using Support Vector Machine", *International Journal on Recent and Innovation Trends in Computing and Communication*, vol. 2, no. 3, pp. 541-544, (2014).

[24] Nandagopal S., Arunachalam V.P., Karthik S."A novel approach for inter-transaction association rule mining, *Journal of Applied Sciences Research* VOL, 8, Issue 7, 2012.

[25] Kannan R., Selvambikai M., Jeena Rajathy I., Ananthi S. Rasayan, A study on structural analysis of electroplated Nano crystalline nickel based thin films, *Journal of Chemistry*, Vol 10, issue 4, 2017.

[26] Arunvivek G.K., Maheswaran G., Senthil Kumar S., Senthilkumar M., Bragadeeswaran T. Experimental study on influence of recycled fresh concrete waste coarse aggregate on properties of concrete. *International Journal of Applied Engineering Research*, Vol 10, issue 11, 2015

[27] Krishna S.K., Sathya M. Usage of nanoparticle as adsorbent in adsorption process. *A review International Journal of Applied Chemistry*, vol 11, Issue 2, 2015.

[28] Sudha S., Manimegalai B., Thirumoorthy P. A study on routing approach for in-network aggregation in wireless sensor networks, *International Conference on Computer Communication and Informatics: Ushering in Technologies of Tomorrow, Today, ICCCI* 2014.

[29] Satheesh A., Jeyageetha V. Improving power system stability with facts controller using certain intelligent techniques, *International Journal of Applied Engineering Research*, Vol 9, no 23, 2014.

[30] Ashok V., Kumar N, Determination of blood glucose concentration by using wavelet transform and neural networks, *Iranian Journal of Medical Sciences*, Vol 38, Issue 1, 2013.

[31] Somasundaram K., Saritha S., Ramesh K, Enhancement of network lifetime by improving the leach protocol for large scale WSN, *Indian Journal of Science and Technology*, Vol 9, Issue 16, 2016.

[32] Jayavel S., Arumugam S., Singh B., Pandey P., Giri A., Sharma A. Use of Artificial Intelligence in automation of sequential steps of software development / production, *Journal of Theoretical and Applied Information Technology*, Vol 57, Issue 3, 2013.

[33] Ramesh Kumar K.A., Balamurugan K., Gnanaraj D., Ilangovan S, Investigations on the effect of flyash on the SiC reinforced aluminium metal matrix composites, *Advanced Composites Letters*, Vol 23, Issue 3, 2014.

[34] Suresh V.M., Karthikeswaran D., Sudha V.M., Murali Chandraseker D, Web server load balancing using SSL back-end forwarding method. *IEEE-International Conference on Advances in Engineering, Science and Management, ICAESM-2012*, 2012.

[35] Karthikeswaran D., Sudha V.M., Suresh V.M., Javed Sultan A, A pattern based framework for privacy preservation through association rule mining, *IEEE-International Conference on Advances in Engineering, Science and Management, ICAESM-2012*, 2012.

[36] Senthil J., Arumugam S., Shah P, Real time automatic code generation using generative programming paradigm, *European Journal of Scientific Research*, vol. 78, issue 4, 2012.

[37] Vijayakumar J., Arumugam S, Certain investigations on foot rot disease for betelvine plants using digital imaging technique, *Proceedings - 2013 International Conference on Emerging Trends in Communication, Control, Signal Processing and Computing Applications, IEEE-C2SPCA*", 2013.

[38] Vijayakumar J., Arumugam S. Odium piperis fungus identification for piper betel plants using digital image processing, *Journal of Theoretical and Applied Information Technology*, vol 60, issue 2, 2014.

[39] Manchula A., Arumugam S, Face and fingerprint biometric fusion: Multimodal feature template matching algorithm, *International Journal of Applied Engineering Research*, vol 9, issue 22, 2014.

[40] Ramesh Kumar K.A., Balamurugan K., Arungalai Vendan S., Bensam Raj J, Investigations on thermal properties, stress and deformation of Al/SiC metal matrix composite based on finite element method. Carbon - Science and Technology, Vol 6, Issue 3, 2014.

[41] Kanchana A., Arumugam S, Palm print texture recognition using connected-section morphological segmentation, *Asian Journal of Information Technology* Vol 6, Issue 3, 2014.

[42] Padmapriya R., Thangavelu P, Characterization of nearly open sets using fuzzy sets, *Global Journal of Pure and Applied Mathematics*, vol 11, issue 1, 2015.

[43] P.B. Narandiran, T. Bragadeeswaran, M. Kamalakannan, V. Aravind, Manufacture of Flyash Brick Using Steel Slag and Tapioca Powder. *Jour of Adv Research in Dynamical & Control Systems*, Vol. 10, No. 12, 2018, 527-532

[44] R. Girimurugan*, N. Senniangiri, K. Adithya, B. Velliyangiri, Mechanical Behaviour of Coconut Shell Powder Granule Reinforced Epoxy Resin Matrix Bio Composites, J*our of Adv Research in Dynamical & Control Systems*, Vol. 10, No. 12, 2018, 533-541.