CIT. Journal of Computing and Information Technology, Vol. 27, No. 2, June 2019, 29–39
doi: 10.20532/cit.2019.1004706

CK-RAID: Collaborative Knowledge Repository for Intrusion Detection System

Saidat Adebukola Onashoga, Adio Taofiki Akinwale, Opeyemi L. Amusa and Gboyega Adebayo

Federal University of Agriculture, Abeokuta, Nigeria

Intrusion Detection Systems (IDSs) are an integral part of an organization's infrastructure. Without an IDS facility in place to monitor network and host activities, attempted and successful intrusion attempts may go unnoticed. This study proposed a Collaborative Knowledge Repository Architecture for Intrusion Detection (CK-RAID). It is based on a distributed network of computer nodes, each with their individual IDS with a centralized knowledge repository system, and firewall acting as a defence. When an unfamiliar attack hits any node, the first step the intrusion monitor takes is to request from Knowledge Repository Server the most effective intrusion response. To improve performance, Intrusion Update module collaborates with IDSs sensor and log by updating their expert rule and intrusion information respectively and removing the old intrusion signature from the knowledge base with the aid of Intrusion Detector Pruning. To ensure security of information exchange, RSA encryption and Digital Signature were used to encode information during transit. The result showed that CK-RAID had a detection rate of 97.2%, compared with Medoid Clustering, Y-means, FCM and K-means that have an accuracy of 96.38%, 87.15%, 82.13% and 77.25% respectively. Therefore, CK-RAID can be deployed for efficient detection of all categories of intrusion detection and response.

ACM CCS (2012) Classification: Security and privacy → Intrusion/anomaly detection and malware mitigation → Intrusion detection systems

Keywords: intrusion, knowledge repository, network, security, digital signature

1. Introduction

Intrusion detection system (IDS) is an authorized way of identifying illegitimate users, attacks and vulnerabilities that could affect the proper functioning of computer systems, see Onashoga *et al*. [1]. The need for IDS can be summed up by a simple principle of network security defence in-depth, see Farooqi and Khan [2].

Typically, heavy reliance is placed on protection and prevention using controls such as routers, firewalls, public key infrastructures, virtual private networks, and virus scanners. In contrast, critical detection and response functions such as those provided by intrusion detection systems are often overlooked. As such, there are no mechanisms to detect and respond to intrusion attempts that evade the first lines of defence, see Uddin and Rahman [3]. Intrusion detection is another type of security tool that was designed to protect and secure the information resources in the system. It complements firewalls by allowing a higher level of analysis of traffic on a network, and monitors the behaviour of the sessions on the servers, see Sharma and Singh 2016 [4]. Tiwari and Gour [5] added that current attacks cannot be thwarted by just blocking ports 80 (HTTP) and 443 (HTTPS). An intrusion detection system (IDS) is needed to detect and respond effectively whenever the confidentiality, integrity, and availability of computer resources are under attack, see Sandhu *et al.* [6].

A distributed intrusion detection system (DIDS), consists of multiple intrusion detection systems (IDS) placed over a large network communicating with each other, or with a central server that facilitates advanced network monitoring, incident analysis, and instant attack data. These co-operative systems distributed across a network, enable incident analysts, network operators and security personnel to get a broader view of what is occurring on their network as a whole or node. The current state of IDS technology is not yet fully reliable, which makes the area of IDS an attractive and still open research field. A major problem with current IDS is their inability to guarantee intrusion detection (low accuracy): the current IDS technology is not accurate enough to provide reliable detection, see Sharma and Singh [4]. The Knowledge-based IDSs are considered good but their completeness is not based on the fact that they detect all possible attacks but by regular update of knowledge about the attacks, see Tiwari *et al.* [7]. They have been identified as possible solution to resolving some major anomalies of other variants of IDS if made, see Uddin and Rahman [3], Tiwari *et al.* [7].

In this study, a collaborative knowledge repository architecture for intrusion detection is proposed with a secure knowledge base and robust inference engine with a view to optimise attack detection and update rate of known and newly emerging attacks so as to update the knowledge and the inference rules having classified the new alerts as attacks, timely and regularly.

2. Related Work

It is not surprising that a lot of research work has been put into developing techniques to mitigate intrusion detection. In this section, a review of related work on collaborative intrusion detection systems was conducted. Yang *et al.* [8] proposed a scheme called Coordinated Attack Response and Detection Systems (CARDS), that uses a signature-based model for resolving issues. It consists of signature manager, monitor and directory services. The system collects data in a flexible, distributed manner and the detection processes are decentralized among various monitors and event-driven. CARDS generate and distribute detection tasks among monitors to cooperatively detect attacks. Detection tasks are parts of an attack signature, which the authors refer to as predefined queries.

A collaborative framework for intrusion detection networks (CIDNs) that uses a Bayesian approach for feedback aggregation was proposed by Fung *et al.* [9]. The approach was designed to solve the problem of traditional intrusion detection systems such as host-based intrusion detection system and network-based intrusion detection system, which can easily be compromised by new or unknown attacks. Collaboration among IDSs enables each IDS to use collective information and experience from other IDSs to achieve more accurate intrusion detections. CIDN employs an overlay network that connects IDSs to exchange information with each other, which helps in minimizing the combined cost of missed detection and false alarm. However, CIDN is a passive IDS with a decentralized architecture. In another related work by Fung *et al*. [10], bayesian learning was used to select and maintain a list of collaborators which they can consult about intrusions.

Reputation-based collaborative intrusion detection network has also been used to lessen the impact of malicious alarms. Pérez *et al.* [11] designed a collaborative intrusion detection network (CIDN) that is capable of building and sharing collective knowledge about isolated alarms in order to efficiently and accurately detect distributed attacks. The authors' model is structured in a modular way by clustering the closest autonomous IDSs in groups. These groups are available to share and combine their collective knowledge of each other inside the same administrative domain, while a small group of them are available to share high-level information among the administrative domains involved in the collaborative detection process. In order to avoid the spread of false alert within the system, a Wise Committee (WC) is saddled with the responsibility of disseminating information with the rest of internal members. However, this research did not address the issue of data privacy.

Sharma and Singh [4] proposed an approach to enhance the collaborative decision making by conducting polls between registered intrusion detection systems in a network. Intrusion activity for new packets and false positives is decided based on all opinions gathered from registered intrusion detection systems.

In the work of Ranjan and Sahoo [12], an anomaly intrusion detection which uses K-medoids method of clustering (data mining approach) for detecting possible intrusion and attacks was proposed. KDD Cup99 was used to test the algorithm with the result of 96.38% of accuracy and 3.20% of false alert. The weakness of this work is the user to root attack, and lack of data privacy which was not addressed.

A hybrid architecture for distributed intrusion detection system in wireless network was proposed by Rashida [13]. The author built an IDS that uses agents as their lowest-level element for data collection and analysis and employs a structure to allow for scalability using both misuse (signature-based) detection and anomaly (behaviour-based) detection types of intrusion detection. The proposed IDS architecture consists of seven modules – Tracker, Anomaly Detection Module, Misuse Detection Module, Monitor, Signature Generator, Inference Detection Module and Countermeasure Module combining the results of the three detection modules. The proposed architecture has several shortcomings; detection of intrusions at the Inference Module is delayed until all the necessary information gets there from the agents. This is a problem common to distributed IDSs.

3. Methodology

The general architecture of Collaborative Knowledge Repository Architecture for Intrusion Detection (CK-RAID) is based on a distributed network of computer nodes, each with their individual IDS with a centralized knowledge repository system, and firewall as the first line of defense against attacks.

CK-RAID architecture can be viewed from two perspectives: operational structure and logical structure. The operational structure is based on the physical layout of the CK-RAID. The CK-RAID is protected by a firewall as its first line of defence, which first analyses both incoming and outgoing packets for access authorization. However, the IDSs basically detect and report network intrusions with improved performance in collaboration with the Knowledge Repository Server (KRS).

The logical structure is based on the action taken by the Knowledge Repository Server (KRS) to ascertain the existence of an attack, which in turn aids the subsequent decision taken on a re-occurrence of such an attack at any other node. The diagram of the logical structure is shown in Figure 1.

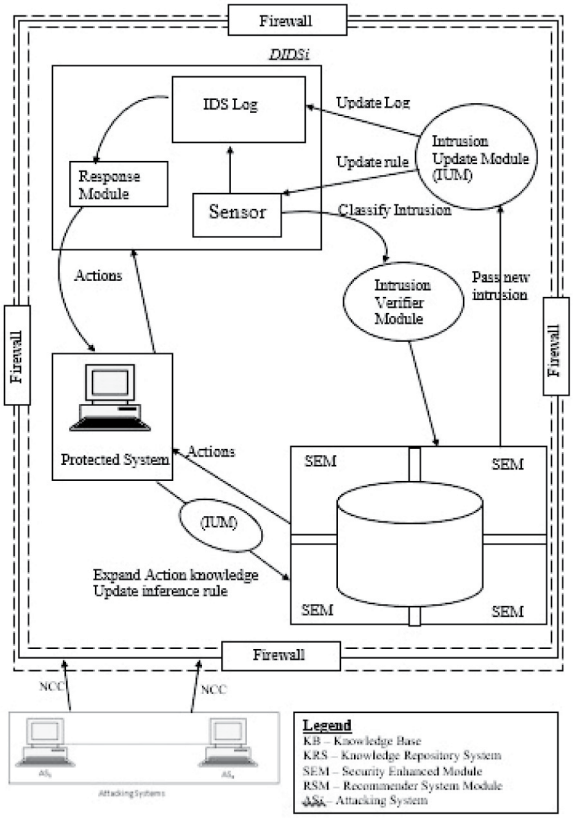

Figure 1. CK-RAID Logical Architecture.

The CK-RAID comprises of three major components, namely Intrusion Detection System (IDS), Knowledge Repository Server (KRS) and External Modular Components.

3.1. Intrusion Detection System (IDS)

The IDS comprises of IDS log, sensor and response module.

3.1.1. The IDS Log

The IDS log enables an administrator to review any suspicious network traffic, though logs cannot be solely depended upon when deploying Network Intrusion Detection System (NIDS)

because a determined hacker may easily flood the network to the extent that the log reaches its capacity and fails. Depending upon the operating system of the IDS, a hacker may also compromise the IDS and easily delete information in the log. On the other hand, all false positives will also be present in the log. If there is a large number of false positives, this can be quite annoying to the person who has to review the log. However, it is important to use the IDS log to search for any possible threats. If the IDS detects a match between current network activity and an attack in the signatures database, it will document the attempted attack in a log.

3.1.2. Sensor

The sensor of the IDS is located in a particular host to monitor system-level behaviour and acts as a sniffer of network traffic in promiscuous mode. The console is the point of central management for an IDS system. By using the console, an administrator may take notice of any current attack alerts. In many cases, the console may be used to customize certain preferences for the IDS. IDS sensor will also send an alert to the console regarding the attack.

3.1.3. Response Module

This is the component module of the IDS in the CK-RAID layout, whose duty is to return action from an intrusion event which could either be false positive or false negative. It identifies intrusion actions sniffed by the sensor on the protected system on the network and also gets details of old intrusion from the IDS log.

3.2. Knowledge Repository Server (KRS)

The knowledge repository server (KRS) consists of the following components:

3.2.1. Knowledge Base

The knowledge base handles the building and sharing of a collective knowledge about isolated alerts that have been detected individually by autonomous IDSs using a schema system. This system provides symbolic structure for encoding, representing and storing intrusion signature data into the knowledge base in structured form. The main goal of these cooperative systems is to extend the capability of individual IDSs to detect and respond to alerts beyond its individual experience. Also, KB uses a knowledge filtering technology known as the recommender module to mine and recommend intrusion responses to intrusion upon request by any node when they are faced with any new threat.

3.2.2. Recommender Module

The recommender module takes in intrusion parameters such as Intruder Port, Intruder IP, Target Port, and Target services as input, and computes the closest intrusion response based on the previous responses. The intrusion response set is built by getting all the intrusion responses where intrusion parameter(s) matches any of the input intrusion parameters, using the Hunt's Algorithm that grows a decision tree (as shown in Algorithm 1) in a recursive fashion by partitioning the dataset into successively purer subsets, using the following if-then procedures.

Algorithm 1. Hunt's Algorithm for Decision Tree Classification.

Input: Intrusion Dataset D
Output: Decision tree t
Process:
Induce(D):
if all tuples t in D have label + **then**
    Return +
if all tuples t in D have label – **then**
    Return –
for all split criteria C:
$D1$, $C = \{t$ in $D \mid t$ satisfies $C\}$
$D2$, $C = D - D1$
measureQuantity ($D1$, $D2$)
Let C be the split
Continue while $D \neq \{D1 + D2 + \ldots Dm\}$ where

$$n = |D| = \sum_{i=0}^{n-1} Ti$$

Let Dt be the dataset of a node t, the general recursive procedure is defined as:

- if Dt contains records that belong to the same class yt, then t is a leaf node labeled as yt,

- if *Dt* is an empty set, then *t* is a leaf node labeled by the default class, *yd*,
- if *Dt* contains records that belong to more than one class, use an attribute test to split the data into smaller subsets.

It recursively applies the procedure to each subset until all the records in the subset belong to the same class.

3.2.3. Security Enhanced Module (SEM)

This consists of Advanced Encryption Standard (AES) and RSA cryptosystem for information encryption, decryption and role verification, which involves key generation, encryption, decryption, signature and verification.

There are four components of SEM:

1. Node Registration Module which assigns an identifier and security parameters to every node that joins the intrusion knowledge community.
2. The Node Request Generation Module which takes the intrusion parameters and the session key of the requesting nodes to produce the cipher text, which the server uses to compare with the node id for verification purpose before authentication.
3. Knowledge Server Response Generation Module decrypts the cipher text to get the intrusion parameters to check if there are similar attack patterns in the database of the knowledge server. Once found, it recommends the best suggestion for the intrusion.
4. Node Response Retrieval Module takes server's signature parameters, server public key, encrypted session key of server and private key of the node as input to produce the server plaintext session key used for symmetric key encryption. It then uses the cipher text of the server response and session key to produce the plaintext response which is the actual response to the intrusion parameters.

3.3. CK-RAID External Modular Components

The external modular components of CK-RAID include Intrusion Update Module (IUM) and Intrusion Verifier Module (IVM).

(i) The IVM collaborates with the IDS sensor in verifying the existence of any suspicious/sniffed intrusions in the knowledge repository server. This module hints a particular node on the intrusion experiences of another systems node on the CK-RAID. Algorithm 2 describes this process.

Algorithm 2. IVM Algorithm.

```
subProcedure veriftyTreat(Alert lt, Repository L_n,
Location loc){
    counter ← 0
    found ← false
    z ← sizeOf (L_n)
    while (counter < z){
    forEach(t in L_n){
        if (exist(lt.getType(),loc) && ( lt.getType()
        == t.getType())){
    IR_0 ← getRule(t)
    applyRuleTo(lt, IR_0);
    c ← treat (lt, L_n);
    updateLog(lt, L_n, IR_0);
    if (c.responseType()=='Intrusion')(
    found ← true;
    }
    }
    ) // endforeach
    counter ← counter + 1;
    } //endwhile}
    return found;
    } // end subProcedure
```

(ii) The Intrusion Update Module (IUM) collaborates with IDS sensor and log by updating their expert rule and intrusion information respectively and by removing the old intrusion signature from the knowledge base with the aid of the Intrusion Detector Pruning (IDP) process. Algorithm 3 describes this process.

However, The IDP serves two purposes in this framework, first to reduce redundancy in the local and global knowledge, and also to increase

the search and decision-making time when a new threat arrives at any node. As adopted by Onashoga *et al.* [14], the Detector for Ageing Technology for Adaptive and Collaborative SMS Spam Filtering was employed to remove aged intrusion signature from the knowledge base and also to reduce computation time when accessing knowledge at the global scope. Algorithm 4 shows the pseudo-code description of the intrusion detection pruning process.

Algorithm 3. IUM Algorithm.

```
subProcedure
buildUpdateMod(Alert lt, IDS_ Log TL,
KB_Dataset Dm){
n ← sizeOf(TL);
m ← sizeOf(Dm);
IsMatch = Null
Feedback = false;
Params[] = extractParam(lt);
DR[] = getInferenceRule (Dm);
DRk+1 ← DR[] + Rec(lt, Params);
c ← treat(lt, DRk+1);
if (c, responseType() == 'Intrusion'){
tn+1 = lt ;
foreach(t in Dm){
if (tn+1.getClass (c) == t.getSigClass()){
IsMatch = t;
}
}
If(IsMatch! = Null){
tn+1.maptoClass(IsMatch);
      }
else {
createNewSigClass (gn+1);
assignToClass(tn+1, gn+1)
}
IRk+1 = add(tn, IRk+1)
//update IDS inference Rule
TLn = add(tn, n + 1)
//update IDS log
DRk+1 = add(tn, DRk+1)
// update knowledge inference Rule
Dm = add(tn, m + 1)
// update knowledge base dataset
feedback = true;
            }
return feedback;
}
```

Algorithm 4. Intrusion detection pruning algorithm

```
Intrusion Ageing threshold t, current system daytime c,
knowledge base intrusion set T, intrusion age to , index i.
    i ← 0;
    while (i < sizeOf (T))
    (       to = Difference (c,
    LastEntryDateTime (T[i]))
          if (to ≥ t) {

                Delete(T[i]);
          )

          else{

    LastEntryDatetime(T[i]) = currentSystime();
                Retain(T[i]);
          } endif
    } endwhile
```

4. Implementation

To show the performance of CK-RAID, a virtual computing device per host was set up on a Linux server with the hardware configuration of Intel(R) core i7 @ 2.90 GHz, 4GB RAM, and 500GB HDD running on Snort IDS, which is an open source IDS that relies on packet dumping application. WinPcap was used to detect network intrusion.

4.1. CK-RAID Implementation Interface

It consists of five fundamental phases which are: Node Registration Phase, Node Intrusion Monitor Phase, Node Intrusion Detection and Intrusion Response Request Phase, Knowledge Server Intrusion Response Computation Phase, Node Intrusion Response Retriever and Application Phase.

4.1.1. Node Registration Phase

This phase requires every new node that joins the intrusion knowledge community to register the node to the community. This phase is required to assign an identifier to the node and also to generate a security parameter for the node administrator.

4.1.2. Node Intrusion Monitor Phase

This phase scans the snort log for possible intrusion signature and raises alert whenever an intrusion hits any host on the node as shown in Figure 2.

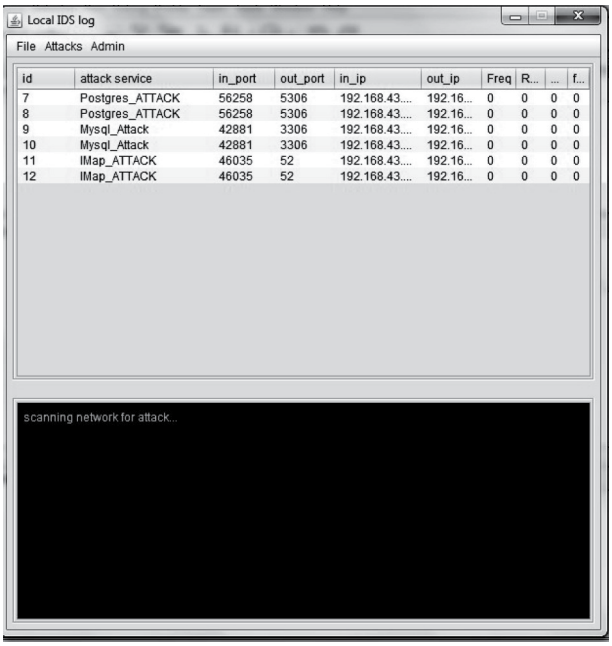

Figure 2. Intrusion monitor interface.

4.1.3. Node Intrusion Detection and Response Request Phase

The intrusion monitor scans the IDS log for possible intrusion on any host. Each node can be attacked with either known attacks or unknown attacks. A known attack is an attack that the local intrusion monitor has the knowledge of and can easily respond to with the previous successful response. This type of attack may not request the attention of an administrator, but will log the intrusion and the response for the administrator to review later. Figure 3 shows the interface of response to known attacks.

When an unfamiliar attack hits any node, the first step taken by the intrusion monitor is to request from the Knowledge Repository Server (KRS) the most effective intrusion response to the intrusion. This is done with the following steps:

1. Encrypt the intrusion parameters with Symmetric Encryption AES in this case,
2. Encrypt the Symmetric Key with Asymmetric Encryption RSA in this case,
3. Send the cypher text for the KRS to decrypt and send the intrusion response back to the node in signed encrypted format,
4. Verify signature on the response,
5. Decrypt the response and apply the response to the intrusion.

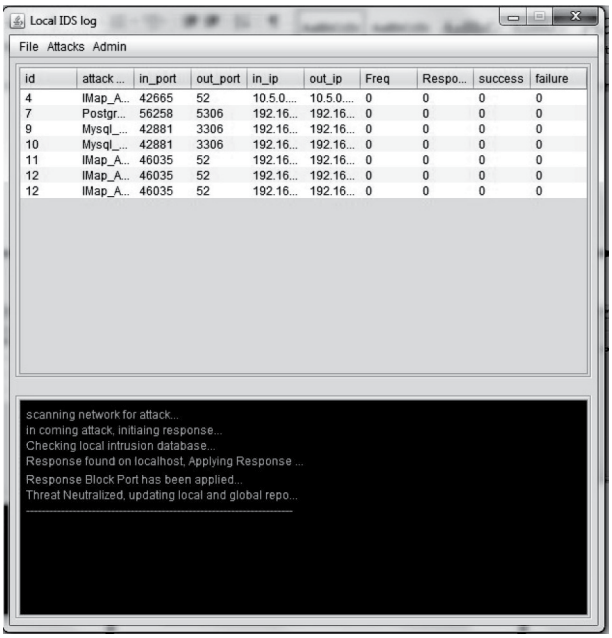

Figure 3. Response to known attacks.

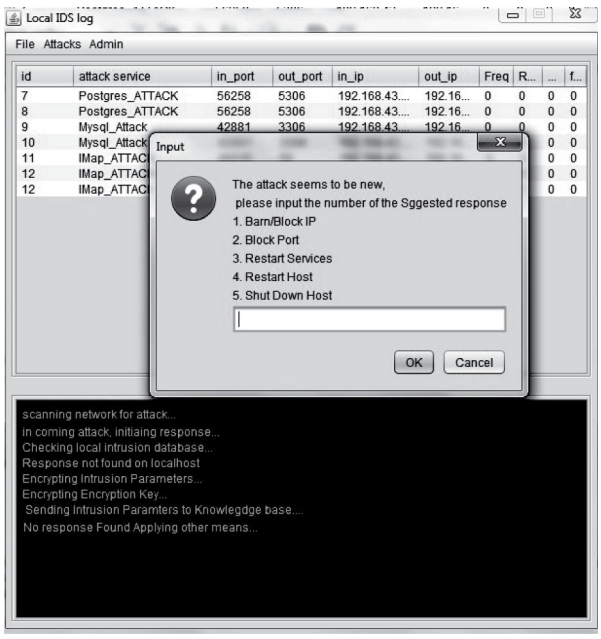

Figure 4. Response to unknown attack from KRS using the Knowledge Recommender Module.

On the other hand, where an unfamiliar attack does not have any replica in KRS, the KRS will suggest possible intrusion response that can neutralize that type of attack by using the knowledge recommender module, as shown in Figure 4.

4.1.4. Knowledge Repository Server Intrusion Response Phase

In the Knowledge Repository Server, whenever an intrusion request comes in, the Knowledge Base (KB) performs the following steps:

1. Decrypt the cypher text to get the intrusion parameter
2. Check if there is a similar attack parameter in the database
3. If there are similar parameters, it will pass them all to the knowledge recommender module to recommend the best suggestion among the responses.
4. Else if the parameters combinations are new, it also passes them to the knowledge recommender module to recommend the possible effective responses.

4.1.5. Node Intrusion Response Retriever and Application Phase

At this phase, the node retrieves the signed encrypted response and verifies signature on the response before decrypting the response and applying the response to the intrusion.

4.2. Performance Evaluation

The performance of CK-RAID was measured based on known and benchmarked metrics of Collaborative IDS which are timeliness, accuracy and robustness. Timeliness in collaborative intrusion detection system is the time taken for a node to get the analysis results of the intrusion parameters after initiating the request. Accuracy in intrusion detection is an essential property of the security system which is the question of information privacy and information filtering. These two features are combined to give accurate intrusion response to request by any node, while robustness is the question of how resilient is the collaborative system to attack, especially the insider attack.

The experiment consists of three nodes with a minimum of three hosts each. IDSs were placed on each node to detect and respond to intrusion alert. The experiment was performed using putty to telnet connections to systems with twenty different types of simulated attacks.

Some of the alerts were known by some IDS in the environment and recorded as known attack by the different IDS in the simulated environment. Ten (10) out of the simulated attacks were picked at random and launched to the three nodes at random. The time taken for the knowledge server to generate a response in seconds was recorded for the ten experiment runs, which is shown in Table 1.

Table 1. Intrusion response time.

Number of experiment	Type of attack	Response time in seconds
1.	Locally known attack	0.313
2.	Globally known attack	1.011
3.	Locally known attack	0.314
4.	Locally known attack	0.312
5.	Globally known attack	1.133
6.	Globally known attack	1.311
7.	Globally known attack	1.331
8.	Unknown attack	1.578
9.	Locally known attack	0.314
10.	Locally known attack	0.314

Suppose the intrusion category sets: Global Intrusion G, Local intrusion L and unknown intrusion U are defined as:

$$G = \{g_1, g_2, ..., g_n : g_k \neq g_n \ \forall k, n \in \mathbb{Z}^+\},$$
$$L = \{l_1, l_2, ..., l_n : l_k \neq l_n \ \forall k, n \in \mathbb{Z}^+\},$$
$$U = \{u_1, u_2, ..., u_n : u_k \neq u_n \ \forall k, n \in \mathbb{Z}^+\}.$$

From Table 1, the Average Response Time (ART) for each intrusion category set was estimated for globally known intrusion as:

$$\mathrm{ART}(G) = \frac{1}{|G|} \cdot \sum_{i=1}^{n} G_i$$
$$= \frac{1.011 + 1.133 + 1.311 + 1.331}{4}$$
$$= \frac{4.786}{4} = 1.197 \text{ secs.}$$

4.2.1. Comparison of CK-RAID with Related Work

CK-RAID was compared with the existing related framework of intrusion detection system, in order to measure its effectiveness. Specifically, 5% of KDD Cup 99 dataset which contains about 2,454,431 intrusion set of about 20 different intrusion types were used as an input to conduct further experiment. 600,000 data points out of the intrusion set were considered as the input, 70% for training and 30% for testing. The results yielded 97.2% accuracy and false positive rate of 3.12%. The result of the experiment is represented in Table 2.

Table 2. KDD Cup 99 intrusion set metrics.

Metrics	Values	Performance Metric	%
TP	474300	FPR	3.12
TN	108900	FNR	2.73
FP	3507	TPR	97.27
FN	13293	TNR	96.88
		Accuracy	97.20

CK-RAID result shows 0.8% increase in detection accuracy when compared with Ranjan and Sahoo (2014) who presented clustering approach for anomaly intrusion detection with the accuracy of 96.38% and false positive rate of 3.2%. Figure 5 is a graphical description of the accuracy (%) and false alarm comparison between CK-RAID, medoid clustering, and three other algorithms.

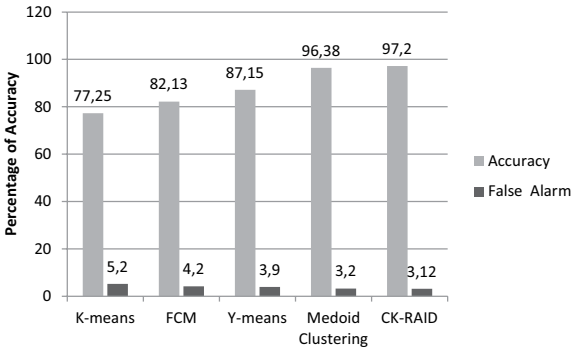

Figure 5. Comparison with medoid clustering.

5. Conclusion

This research work presents a collaborative knowledge filtering driven by decision tree algorithm to classify intrusions so as to improve the accuracy of intrusion detection and intrusion response. RSA encryption and digital signature were used to secure the information exchange among the nodes in the distributed network environment. The pfSense firewall was used to simulate the distributed network environment. The result obtained shows an increase in the timeliness of the intrusion detection and response, as well as the accuracy in intrusion detection and response.

References

[1] S. A. Onashoga *et al.*, "A Strategic Review of Existing Mobile Agent-Based Intrusion Detection Systems", *Issues in Informing Science & Information Technology*, pp. 669–683, 2009. https://doi.org/10.28945/1088

[2] A. H. Farooqi and F. A. Khan, "A Survey of Intrusion Detection Systems for Wireless Sensor Networks", *International Journal of Ad*

Hoc and Ubiquitous Computing, vol. 9, no. 2 pp. 69–83, 2012. https://doi.org/10.1504/ijahuc.2012.045549

[3] M. Uddin and A. A. Rahman, "Dynamic Multi Layer Signature-Based Intrusion Detection System Using Mobile Agents", arXiv preprint arXiv:1010.5036., 2010. https://doi.org/10.5121/ijnsa.2010.2411

[4] D. K. Sharma and N. K. Singh, "An Approach for Collaborative Decision in Distributed Intrusion Detection System", *International Journal of Computer Application*, vol. 133, no. 13, 2016. https://doi.org/10.5120/ijca2016908026

[5] R. Tiwari, and R. Gour, "Mobile Agent-Based Distributed Intrusion Detection System: A Survey", *International Journal of Computer Applications in Engineering Sciences*, 2012.

[6] U. A. Sandhu *et al.*, "A Survey of Intrusion Detection and Prevention Techniques", *2011 International Conference on Information Communication and Management*, 2011, pp. 66–71.

[7] K. Tiwari *et al.*, "Intrusion Detection Using Data Mining Techniques", *International Journal of Advanced Computer Technology*, vol. 2, no. 4, pp. 21–25, 2013.

[8] J. Yang *et al.*, "CARDS: A Distributed System for Detecting Coordinated Attacks", in *Proc. of IFIP International Information Security Conference*, Springer, Boston, MA, 2000, pp. 171–180. https://doi.org/10.1007/978-0-387-35515-3_18

[9] C. J. Fung *et al.*, "Bayesian Decision Aggregation in Collaborative Intrusion Detection Networks", in *Proc. of 2010 IEEE Network Operations and Management Symposium-NOMS*, 2010, pp. 349–356. https://doi.org/10.1109/noms.2010.5488489

[10] C. J. Fung *et al.*, "Effective Acquaintance Management Based on Bayesian Learning for Distributed Intrusion Detection Networks", *IEEE Transactions on Network and Service Management*, vol. 9, no. 3, pp. 320–332, 2012. https://doi.org/10.1109/tnsm.2012.051712.110124

[11] M. G. Pérez *et al.*, "RepCIDN: A Reputation-Based Collaborative Intrusion Detection Network to Lessen the Impact of Malicious Alarms", *Journal of Network and Systems Management*, vol. 21, no. 1, pp. 128–167, 2013. https://doi.org/10.1007/s10922-012-9230-8

[12] R. Ranjan and G. Sahoo, "A New Clustering Approach for Anomaly Intrusion Detection", arXiv preprint arXiv:1404.2772., 2014.

[13] S. Y. Rashida, "Hybrid Architecture for Distributed Intrusion Detection System in Wireless Network", *International Journal of Network Security & Its Applications*, vol. 5, no. 3, p. 45, 2013. https://doi.org/10.5121/ijnsa.2013.5305

[14] S. A. Onashoga *et al.*, "An Adaptive and Collaborative Server-Side SMS Spam Filtering Scheme Using Artificial Immune System", *Information Security Journal: A Global Perspective*, vol 24, pp. 133–145, 2015. https://doi.org/10.1080/19393555.2015.1078017

Received: March 2019
Revised: May 2019
Accepted: June 2019

Contact addresses:
Saidat Adebukola Onashoga
Federal University of Agriculture
Abeokuta
Nigeria
e-mail: onashogasa@funaab.edu.ng

Adio Taofiki Akinwale
Federal University of Agriculture
Abeokuta
Nigeria
e-mail: aatakinwale@yahoo.com

Opeyemi L. Amusa
Federal University of Agriculture
Abeokuta
Nigeria
e-mail: amusaol@funaab.edu.ng

Gboyega Adebayo
Federal University of Agriculture
Abeokuta
Nigeria
e-mail: adebayoga@funaab.edu.ng

Saidat Adebukola Onashoga, a Chartered IT Professional (CITP), received the BSc degree in mathematical sciences (Computer Science option) from the University of Agriculture, Abeokuta, Nigeria, as the best graduating student, subsequently getting there an appointment as a Graduate Assistant. At the same institution she enrolled in 2005 in the PhD study in Computer Science with a focus on computer security, eventually receiving the PhD degreee in April 2010. Dr. Onashoga's professional interests include data mining, computer security and information management. She has published both in national and international journals, and has attended national and international workshops, seminars and conferences. Her current research focuses on applying computer security to different domains, like privacy-preserving data mining algorithms in health care management and development of secure mobile applications. She is member of the Informing Science Institute, USA, Computer Professionals of Nigeria (CPN), Nigeria Computer Society (NCS).

Adio Taofiki Akinwale received the MSc and PhD degrees from Oskar Langer University, Wroclaw, Poland in 1990 and 1994, respectively. He is currently a Professor of Computer Science in the Federal University of Agriculture, Abeokuta, Nigeria. His main areas of research interest are database management systems and optimization of query algorithms. Prof. Akinwale is member of the Nigerian Computer Society, Computer Professionals of Nigeria, as well as the Polish Academy of Sciences.

Opeyemi L. Amusa is a Chartered Computer Professional (MCPN). She holds a post-secondary degree in Computer Technology with Higher National Diploma Certificate. She also received the postgraduate diploma in Computer Science from the University of Agriculture, Abeokuta, Nigeria, as well as the MSc in Computer Science. In 2009, she also completed the Cisco Certified Instructor Associate course. Ms Amusa's academic and research interests include data mining, computer information technology and network security, in which areas she has published in local and international journals, as well as attended a number of national workshops, seminars and conferences. She is member of Computer Professionals of Nigeria (CPN), Nigeria Computer Society (NCS), and Association of Applied Information Management Professionals.

Gboyega Adebayo is a Professor of Condensed Matter Physics. He received the BSc degree (with honors) in physics from the University of Agriculture, Abeokuta, Nigeria, in 1992; the MSc degree in solid state physics from the University of Ibadan, Ibadan, Nigeria, in 1997; and the PhD degree in condensed matter physics from the University of Ibadan, Ibadan, Nigeria, in 2005. He has held lecturing positions at the University of Agriculture, Abeokuta, Nigeria. His research interests cover condensed matter physics, materials physics and computational physics, with over 50 technical publications. Professor Adebayo is a member of the Nigerian Institute of Physics (NIP), American Physical Society (APS), and German Physical Society, (DPG).