Received: 1 July 2020 | Revised: 25 October 2020 | Accepted: 19 November 2020 | Published on: 24 January 2021

DOI: 10.1002/spy2.147

RESEARCH ARTICLE

WILEY

Introduced a new method for enhancement of intrusion detection with random forest and PSO algorithm

Mahdi Ajdani | **Hamidreza Ghaffary**

Department of Computer Science, Ferdows Branch, Islamic Azad University, Ferdows, Iran

Correspondence
Hamidreza Ghaffary, Department of Computer Science, Ferdows Branch, Islamic Azad University, Ferdows, Iran.
Email: hamidghaffary53@yahoo.com

Abstract

As computer networks expand, attacks and intrusions into these networks have increased. In addition to firewalls and other intrusion prevention equipment, other systems, such as IDS (Metrics), are designed to provide enhanced security in computer systems, including the purpose of monitoring intrusive and intrusive activities. Intrusive allocation system can be considered effective if the high intrusion rate is slightly misleading, and in this article a new way to classify it is abnormal (infiltration) in the host or network. UNSW-NB15 and KDD-Cup'99 Datasets Introducing Random Forest and PSO algorithm. In this paper, training data and label data used with the random forest algorithm. After creating a random forest algorithm, provide test data. We use the data stored in train step, which is actually a copy of the data, so that when performing the test step, the same training data can be compared with categorize using a PSO algorithm. In in order to show the accuracy of the proposed method, an example of the confusion matrix formed in the code that showed performance of all methods and modes studied is compared based on accuracy and time that the PSO algorithm has always been able to take less time, which is quite acceptable and predictable. Improves correct diagnosis of correct detection rate in that report was 75.94% and in the proposed method in this article it reached 97%. With the proposed method, learning speed has been greatly increased and accuracy is acceptable.

KEYWORDS

behavioral pattern allotment, data mining, PSO algorithm, random Forest algorithm

1 | INTRODUCTION

Today, with the growth of use and increasing the importance of computer networks, attacks and intrusions into these networks are increasingly expanding and taking many forms. Influence, all illegal actions that are true, confidential or access to a source Includes endangerment.[1] Infiltrators are divided into two categories, external and internal. External influencers are those who try to access them without the use of system resources, and internal influencers are those who, with limited authority in the system, try to access resources outside their reach. In the past, influences were mostly done by people who were interested in testing their skills and abilities, but now, the desire to influence has increased with financial, political and military motives.[2] The Internet and online processes are one of the essential tools of everyday life today and are used as an important part of business operations. In today's modern world, influence takes place in a fraction of a second. Intruders are quite intelligent using the modified version of the commands to infiltrate and then erase

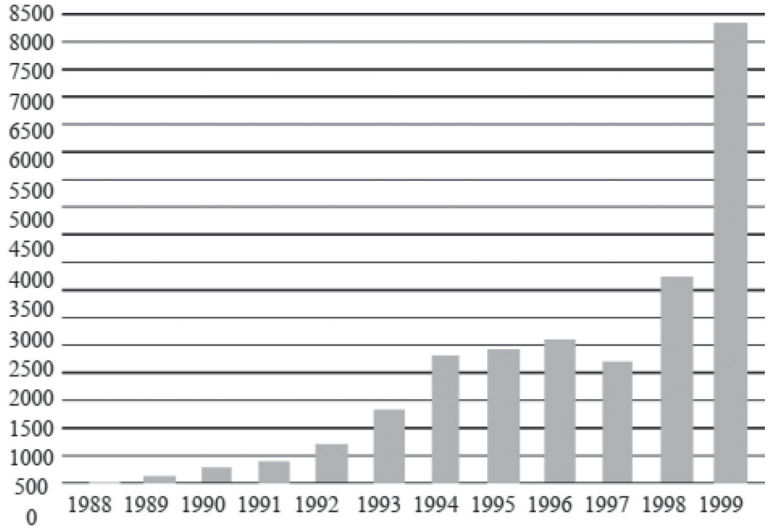

FIGURE 1 The growth rate of CERT/CC recorded attacks

their footprints from the reports, therefore, network security must be carefully considered to provide secure information channels.[3] Detection is the process of monitoring events that occur in a computer system or network and analyzing them to find signs of accidents. These incidents can be imminent violations or threats to security work policies, accepted policies in the use of the system, or security standards.[4] Influence detection systems focus on identifying possible incidents, incidental information about them, trying to stop them, and reporting them to security managers. In addition, organizations use intrusion detection systems for other purposes. These include identifying problems with security policies, documenting existing threats, and preventing individuals from violating security policies.[5]

This argument is combined with another obvious fact. The rise of such attacks on the Internet itself reflects the growing popularity of the Internet in recent years. Figure 1 shows the number of incidents reported to CERT/CC over a 12-year period it has been shown. E-commerce alone has been able to intensify this process.[6]

In recent years, however, foreign attacks have often been used to conduct personal patrols and test the skill of hackers. These days, military, economic, and political goals have played a major role in provoking hackers. The goal of intrusion detection systems is to detect unusual access or attacks to an internal network. Network-based intrusion detection systems are a valuable tool for deep defense of computer networks. The intrusion detection system searches for known or potentially malicious activities in network traffic and rings an alarm whenever it detects a suspicious activity.[5]

In general, intrusion detection systems are divided into two methods: pattern-based detection and anomaly detection. The template needs to learn the pattern of the average user to distinguish it from the behavioral pattern of attackers and intruders. This requires the behavioral data of each of these two groups.[7] In[1] a classification model based on Xgboost and PSO used to adaptively search for the optimal structure of Xgboost. The benchmark NSL-KDD dataset is used to evaluate the proposed model and results demonstrate that PSO-Xgboost model outperforms other comparative models. Also[2] examines the attack detection mechanism by using three data mining algorithms based on particle swarm optimization (PSO), namely PSO-K Nearest Neighbor, PSO-Random Forest, and PSO-Decision Tree in the Canadian Institute for Cybersecurity Dataset (CICIDS2017). Results showed that the approach using the PSO-RF method was able to produce the highest accuracy of attack detection. Paper[4] presents a method to solve the tradeoff among these conflicting objectives using multi-objective particle swarm optimization approach with multiple machine learning classifiers. Results reveal that this technique with J48 classifier gives the highest gbest value.

By timely detection of measures taken to infiltrate the system, it can be hoped that security will be established to some extent in the network. One of the tools that can be used to achieve this goal is data mining. So far, various data mining methods have been used to learn the behavioral patterns of ordinary users and disruptive users. This paper aims to improve the speed and accuracy of intrusion detection using data mining methods that have been performed on sample UNSW-NB15 and KDD-Cup99 data.

1.1 | Random forest

Random Forest is an ensemble method, which predicts based on the results of a collection of Decision Trees. Resampling using the bootstrap approach is used for the creation of each tree in the "forest." Also, on each node split a subset of

features is selected randomly and the selection of the split variable occurs over this subset. The predicted value is the majority vote, for classification, and the average, for regressions. The approach is due to Breiman (2001), which was based on prior ideas of Amit and Geman (1997) and Ho (1995, 1998). Essentially, there are two parameters for tuning on Random Forest models: mtry - the number of randomly selected features to consider in each split; and ntree - the number of trees in the model. There is a tradeoff in mtry: large values increases the correlation among trees, but improves the strength (accuracy) of each tree (Breiman 2001). A bootstrapped subset of the training dataset is created to train each tree in the "forest." Due to this fact, on average, each tree makes use of around two-thirds of the training dataset. The unused elements are called by the Out Of Bag (OOB) samples. The OOB samples can be used for validation. In this case, each tree predicts over its respective OOB samples and the final result is an average over the trees' outcomes.[8] The OOB samples can be used to estimate the importance of each variable and create a rank for them. For each tree and its respective OOB samples, it computes the accuracy over this set, permutes randomly a variable between samples, and recomputes the accuracy on the new set. Performing this for all trees and averaging for each variable, it is possible to have a relevance comparison metric, which is usually referred to as Variable Importance Measure (VIM) or Permutation Importance Index (PIM). Another way to create an importance rank is, on each tree and each node split, to calculate the split improvement by a measure (eg, Gini Index) and use these values to compare the variables' importance.[9]

1.2 | PSO algorithm

Particle swarm optimization (PSO) algorithm is a stochastic optimization technique based on swarm, which was proposed by Eberhart and Kennedy (1995) and Kennedy and Eberhart (1995). PSO algorithm simulates animal's social behavior, including insects, herds, birds and fishes. These swarms conform a cooperative way to find food, and each member in the swarms keeps changing the search pattern according to the learning experiences of its own and other members.

Main design idea of the PSO algorithm is closely related to two researches: One is evolutionary algorithm, just like evolutionary algorithm; PSO also uses a swarm mode which makes it to simultaneously search large region in the solution space of the optimized objective function. The other is artificial life, namely it studies the artificial systems with life characteristics.

In studying the behavior of social animals with the artificial life theory, for how to construct the swarm artificial life systems with cooperative behavior by computer, Millonas proposed five basic principles.[9]

1. Proximity: the swarm should be able to carry out simple space and time computations.
2. Quality: the swarm should be able to sense the quality change in the environment and response it.
3. Diverse response: the swarm should not limit its way to get the resources in a narrow scope.
4. Stability: the swarm should not change its behavior mode with every environmental change.
5. Adaptability: the swarm should change its behavior mode when this change is worthy.

Note that the fourth principle and the fifth one are the opposite sides of the same coin. These five principles include the main characteristics of the artificial life systems, and they have become guiding principles to establish the swarm artificial life system. In PSO, particles can update their positions and velocities according to the environment change, namely it meets the requirements of proximity and quality. In addition, the swarm in PSO does not limit its movement but continuously search the optimal solution in the possible solution space. Particles in PSO can keep their stable movement in the search space, while change their movement mode to adapt the change in the environment. So particle swarm systems meet the above five principles.

1.3 | UNSW-NB15 dataset

In this proposed work, UNSW-NB15, the recently developed benchmark dataset for evaluating IDSs,[10] is used to avoid the limitation of the above-mentioned dataset. UNSW-NB15 includes nine modern attacks and features of realistic normal traffic with a balanced distribution set. The UNSW-NB15 dataset has been recently released.[11] This dataset contains nine different modern attack types and a wide variety of real normal activities.[12] The dataset contains real modern normal behaviors and contemporary synthesized attack activities and consists of 49 features with their class labels. This dataset comprises 2 540 044 observations. In this study, the UNSW-NB15 was divided into a training set and testing set.

TABLE 1 Statistics of the UNSW-NB15 dataset

	Effective	%
Normal	102 000	40
Attack	155 673	60
Total	257 673	100

Furthermore, this new dataset ensures an accurate evaluation of IDSs.[10] Table 1 shows the distribution of the records in the UNSW-NB15 dataset.

It is a complete representative of existing real networks. However, it is believed that despite all these shortcomings, it can be used as an effective reference criterion data to help researchers compare different methods of intrusion detection.

1.4 | Entropy function

The authors of this article seek to maximize the content of information from all parameters. One way to achieve the above goal is to optimize entropy. In this method, minimizing entropy leads to maximizing the support level of each parameter of evidence, while minimizing entropy Uncertainty in statistics and cross-entropy between algorithms and data.[13]

Entropy can be defined as a measure of the content of expected information or uncertainty about the probability distribution. It can also include the amount of disorder within a system or the uncertainty of a partition.

The philosophy of entropy reduction in pattern recognition can be used to classify, analyze, and analyze data, one of the tasks of which is to discover the pattern or order in a large data set. Data structure rules are determined by small entropy values and random values by large entropies. In the field of data mining, the most well-known application of entropy is the collection of decision tree data.[14,15]

In information theory and machine learning, a concept called interest Information is mentioned. The mathematical expectation of information interest is a function of the interaction of information or the same amount of entropy reduction if information is obtained. In machine learning and artificial intelligence, this concept is used to determine features.[16,17] The utility of an attribute is the amount of entropy reduction obtained by separating the instances through this attribute.

1.5 | Evaluation

The performance of the methods can be compared based on accuracy, feature sensitivity, accuracy and time. It is possible to calculate the above criteria by successfully diagnosing, ignoring an acceptable misdiagnosis behavior, and failing to detect an attack. None of the cases of ignoring an acceptable behavior and successful diagnosis are dangerous as long as the penetration detection system works as expected. In fact, failure to detect an attack plays the most serious and dangerous role possible when the intrusion detection system considers an acceptable behavior. This is actually the failure of the system to detect an attack, FP FN outputs. TF, TN is the result of the implementation of the proposed algorithms of the current research to calculate the observation accuracy.[18]

2 | PROPOSED METHOD

The proposed system, by analyzing the behavior of its users, identifies destructive behaviors according to the given training. After receiving the tagged data in the preprocessing stage, the system reduces the number of features to the minimum possible by combining the methods of selecting features and applying conversions to features. In the second phase, by applying the random forest algorithm, it learns these patterns (Figure 2).

In this system, after selecting the data set containing user behavior, first select the most effective features from the available features, here the selected features include the first to sixth features of the UNSW-NB15 dataset. In tree-based algorithms, if features that are non-numeric have a large number of unique values, then applying tree-based methods with the challenge of slowing down (in both the training and testing phases) also reduces Care will be taken. And since in the random forest algorithm package in R language the maximum number of values of a non-numerical attribute

FIGURE 2 One phase

is limited and, in most cases, it can have a value equal to 53, in the next step, identify the attribute or attributes with the number of non-numerical values.[19] The number is more than 53. Studies have shown that the third feature belongs to this category and requires changes in order to reduce the number of values to an acceptable limit by the random forest algorithm.

In the PSO algorithm, each birds (answer) is a subset of the values for a property and the fit function is also an entropy function, which for each bird (answer) determines how much a class is likely to occur for a group of values and when it is minimized that classes corresponding to these values belong to only one class for the selected attribute. In other words, the amount of entropy function for a group of values is minimal when the probability of a class occurring for that group of values is close to 1, and at the same time the probability of the occurrence of other classes for that group is close to zero. The next step was to group the attribute values. Here, the goal was to automatically group the attribute values that were non-numeric, so that the number of unique values was reduced and at the same time did not lead to a decrease in accuracy (one group the values were considered as a value, and after preparing the data with the new attribute or values, the resulting values are given to the next phase, which is the same as the random forest method.

The last phase of this method consists of two stages of learning and testing, in which the system is taught by a part of the data at its disposal in order to prepare for the next stage, which is the same stage of the test. We provided the data prepared in the previous step to the forest, and by performing a random forest with a different number of trees, we examined the effectiveness of the proposed method in predicting the possible behavior of users based on previous learning. Using this system and based on the predictions made in the test phase, it is possible to be aware of the dangerous behaviors in the network in a short time and to make a suitable decision to deal with it. In the following chapter, we will explain how to implement the proposed method in the R environment (Figure 3).

3 | RESULTS AND DISCUSSION

As previously explained, among the available standard test data, in this study, UNSW-NB15 and KDD-Cup'99 data with all hypotheses of the problem were selected for review. One of the steps in implementing the proposed intrusion detection algorithm is feature selection. After loading the data, we extract the dataset structure and see that of the 42 Dataset features, 38 are numeric and 4 are non-numeric, and all features are numerical except for the B, C, D, and AP rows.

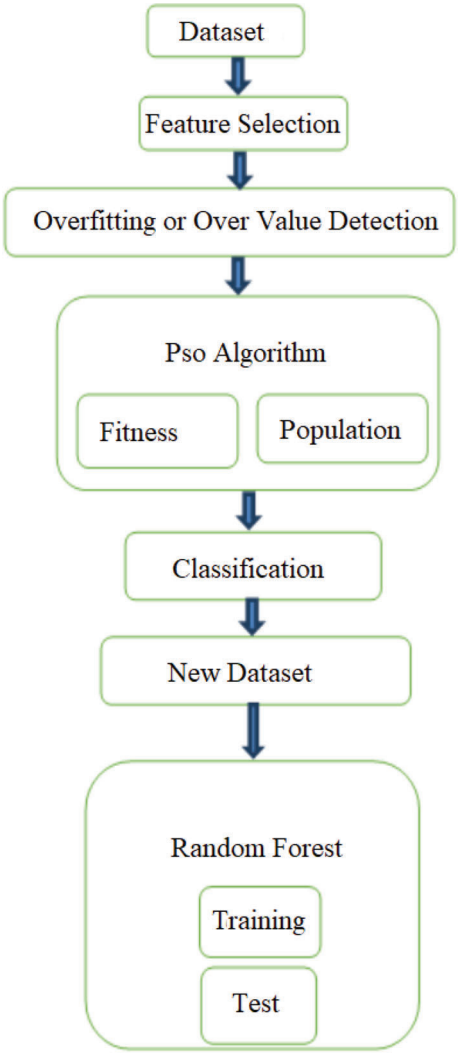

FIGURE 3 Proposed method

TABLE 2 Fitness function

Algorithm: Fitness Function
Input:x, train
Values <−get_non_zero_indices(x)
Group <− concat(values)
Group_target <−get_target_for_group(train)
Return (entropy(Group_target))

The first non-numeric attribute has 3 values, the second non-numerical attribute has 65 values, and the third non-numerical attribute has 11 values. The last non-numerical attribute, which is actually the data label, has 38 values. Some of the data in dataset can be seen in Table 2.

As can be seen in pseudo-code 1, the function fits at the input of the birds or the candidate's solutions, which are an array of zeros and ones. The length of this array is the same as the properties themselves (65). Another input is a function of fitting training data. The fit function first takes the index and the value of non-zero houses and puts them together. Eventually, entropy results. Obviously, the lower the entropy, the better the candidate.

In order to perform the process of selecting properties in R, it is done in such a way that first the properties of columns one to six are stored in one variable and then the number of repetitions of each value of the third column is calculated and

TABLE 3 PSO algorithm

```
Algorithm: PSO
Input: fitness_function, train
Initialize(population)
While (not_stop_condition) do:
Pop_fitness <– fitness_function(population)
Cell_particle <–select_particle{pop_fitness)
particle <– (cell_particle)
Population < – survival_select(population, particle)
End
Return(max_fitness(population))
```

TABLE 4 Display of a 99-KDD dataset

Protocol type	Service	Flag	Src_byte	…	Label
tcp	http	SF	223	…	Normal
udp	private	SF	105	…	Snmpget Attack
tcp	http	SF	230	…	Normal
udp	private	SF	105	…	Normal
udp	private	SF	105	…	Snmpget Attack
tcp	smtp	SF	3170	…	Normal
tcp	http	SF	297	…	Normal

stored. Then save the values of the third column whose number of repetitions was less than 54 in another variable and the rows whose third column contained the values found above the number of repetitions less than 54) were removed from the data set under consideration in Out of 65 possible values, only 50 values remain in the third feature. However, the new production dataset replaced the previous dataset and the number of rows and columns was calculated and stored. As a result, 310 501 rows of the first data in seven columns of features remained to be tested in the next steps.

At this step, we present the training data and label data to the random forest algorithm and form forests consisting of 1 to 30 trees. After creating a random forest algorithm, we provide test data that we did not provide to the algorithm during the training phase without label values in the random forest to predict the algorithm in relation to the type of attack.

In this part of the code, we use the data stored in train2, which is actually a copy of the data stored in train, which we saved at the beginning of the process, so that when performing the test step, the same training data can be compared with Categorize using a PSO algorithm.

As can be seen in pseudo-code 2, the input of the PSO algorithm is a function of fit and training data. First, we randomly generate an initial population of birds, or candidate solutions. Then we get a better solution from a population by combining solutions. Finally, the top 100 candidates Select and send to the next step. This operation is repeated until the condition of stopping is true.

At the end of the random forest, a matrix is formed whose rows represent the algorithm's prediction and its columns represent the actual value of the tag to compare the algorithm's prediction result with reality. In fact, the clutter matrix is a table that describes the performance of classifiers. The logic behind this is because the numbers in the main diameter of this matrix represent the number of correct algorithm predictions and any non-exponent values outside the display matrix diameter. In the following, in order to show the accuracy of the proposed method, an example of the confusion matrix formed in the code in Table 3 is presented.

As can be seen in pseudo-code 3, at this point we have a bird that has a number of zeros and ones. The meaning of a value in array one is that there is a corresponding property. Amounts of a feature that if we combine and add a new value, nothing is lost (according to entropy testimony. So, we get the sum of the values. Then we find the number of their lines in the training data and the amount of the desired feature we change to a constant value. We give the new training data

TABLE 5 Proposed algorithm

Algorithm: PSO aided random forest
Input: PSO_best_solution, train, test
Values <– non_zero_indices(x)
Group <– union(values)
Group_idx = get_group_idx(group, train)
Train[group_idx] < – new_value Rfmodel <– random Forrest(train, train_label)
Test_label <– predict (rfmodel, test)

TABLE 6 Compare the results in KDD-Cup'99

		Decision tree	10 decision tree	20 decision tree	25 decision tree	30 decision tree
Precision	—	0.658	0.745	0.899	0.963	0.987
	PSO	0.958	0.980	0.986	0.991	0.995
Time (sec) of Training	—	0.23	0.35	0.38	0.39	0.39
	PSO	0.22	0.27	0.27	0.28	0.29
Time (sec) of testing	—	2.15	0.63	15.65	18.10	22.5
	PSO	2.01	6.94	12.84	14.50	18.01

TABLE 7 Compare the results in UNSW-NB15

		Decision tree	10 decision tree	20 decision tree	25 decision tree	30 decision tree
Precision	—	0.723	0.756	0.899	0.965	0.988
	PSO	0.959	0.983	0.987	0.993	0.997
Time (sec) of Training	—	0.23	0.35	0.37	0.38	0.40
	PSO	0.22	0.27	0.27	0.28	0.31
Time (sec) of testing	—	2.12	0.65	14.72	17.12	20.5
	PSO	2.11	5.62	11.99	13.85	17.01

to a random forest algorithm and create a predictive algorithm based on it. We also test the built-in algorithm and test it (Tables 4 and 5).

The performance of all methods and modes studied is compared based on accuracy and time in Tables 6 and 7. As can be seen, the application of the PSO algorithm has significantly improved all comparison factors, namely accuracy and time, and in the case of using a decision tree, while grouping by PSO algorithm, there is a significant difference between the diagnoses. There are two ways to use and not to use a PSO algorithm. In fact, at all times of implementation, the PSO algorithm has always been able to take less time, which is quite acceptable and predictable.

Compared to the previously reported algorithm,[13] which did not use the concept of information interest and was based on a conventional decision tree, the method used in this study was about 25.2%.

Improves correct diagnosis of correct detection rate in that report was 75.94% and in the proposed method in this article it reached 97%. Compared to another paper, which reported an error of about 25.2%, the results of this study show a reduction in quality of about 0.74% (Tables 6 and 7). Instead, the proposed method in this study allows for faster implementation due to the use of feature extraction (rather than feature selection). Comparison based Area under curve (AUC) based on TP and FP values and accuracy with other study showed in Table 8, that showed proposed method was better results.

TABLE 8 Comparison of the proposed model with other models

Model	Accuracy	AUC
PSO-Xgboost,[1]	0.89	0.785
PSO-DT[2]	0.78	0.699
CS-PSO[3,6,19]	0.85	0.768
Proposed model	0.93	0.881

4 | CONCLUSION

This paper presents a new model, namely, an PSO-RF that focuses on the applicability of the new cosmology inspired algorithm PSO to train random forest. The AUC the proposed model have been obtained using the KDD-cup99 and UNSW-NB15 datasets. The performance of the model was compared with those of popular intrusion detection techniques, based on such as PSO-Xgboost, PSO-DT and classification techniques. UNSW-NB15 dataset trained with the KDD-cup99 dataset obtained of 96.3% and 96.5%, respectively. These values are higher than those obtained by other methods tested using the UNSW-NB15 and KDD-cup99 datasets. The main idea of the article is to automate grouping of values a property (nominal type and 65 unique states) and to generate new values with a much smaller number of unique values. This idea is most useful in increasing the speed of tree-based methods. So the first point is that the speed of the proposed method for each method based on the decision tree increases. In comparing the accuracy of the two methods, the same train and test data should be used for both methods, and even the number of experiments should be repeated 10, 20 times in order to compare the accuracy of the two methods; Therefore, the accuracy mentioned in the article cannot be considered. The point to be made is that for decision-making methods, when a nominal attribute has a large number of unique values, the learning time will be too long and it will be slow to learn. With the proposed method, learning speed has been greatly increased and accuracy is acceptable.

ORCID

Mahdi Ajdani https://orcid.org/0000-0002-6969-2901
Hamidreza Ghaffary https://orcid.org/0000-0003-0627-4080

REFERENCES

1. Jiang H, He Z, Ye G, Zhang H. Network intrusion detection based on PSO-Xgboost model. *IEEE Access*. 2020;8:58392-58401.
2. Budilaksono S, Riyadi AA, Lukman Azhari DD, et al. Comparison of data mining algorithm: PSO-KNN, PSO-RF, and PSO-DT to measure attack detection accuracy levels on intrusion detection system. *JPhCS*. 2020;1471(1):012019.
3. Dickson A, Thomas C. Improved PSO for optimizing the performance of intrusion detection systems. *J Intell Fuzzy Syst*. 2020;38(5):1-11.
4. Masdari M, Khezri H. A survey and taxonomy of the fuzzy signature-based intrusion detection systems. *Appl Soft Comput*. 2020;106:301.
5. Rajadurai H, Gandhi UD. A stacked ensemble learning model for intrusion detection in wireless network. *Neural Comput Appl*. 2020. https://doi.org/10.1007/s00521-020-04986-5.
6. Ghosh P, Karmakar A, Sharma J, Phadikar S. CS-PSO based intrusion detection system in cloud environment. *Emerging Technologies in Data Mining and Information Security*. Singapore: Springer; 2019:261-269.
7. Tan X, Su S, Huang Z, et al. Wireless sensor networks intrusion detection based on SMOTE and the random forest algorithm. *Sensors*. 2019;19(1):203.
8. Bhavani TT, Rao MK, Reddy AM. Network intrusion detection system using random forest and decision tree machine learning techniques. *First International Conference on Sustainable Technologies for Computational Intelligence*. Singapore: Springer; 2020:637-643.
9. Li X, Chen W, Zhang Q, Wu L. Building auto-encoder intrusion detection system based on random Forest feature selection. *Comput Secur*. 2020;101:851.
10. Moustafa N, Slay J, 2015. Unsw-nb15: a comprehensive data set for network intrusion detection systems (unsw-nb15 network data set). In *2015 Military Communications and Information Systems Conference (MilCIS)* pp. 1-6.
11. Moustafa N, Slay J. The significant features of the unswnb15 and the kdd99 data sets for network intrusion detection systems. *2015 Fourth International Workshop on Building Analysis Datasets and Gathering Experience Returns for Security (BADGERS)*. IEEE; 2015:25-31. https://doi.org/10.1109/BADGERS.2015.014.
12. Moustafa N, Slay J. The evaluation of network anomaly detection systems: statistical analysis of the unsw-nb15 data set and the comparison with the kdd99 data set. *Inf Secur J A Global Perspect*. 2016;25(1-3):18-31.
13. Lu T, Huang Y, Zhao W, Zhang J. The metering automation system based intrusion detection using random Forest classifier with SMOTE+ENN. *2019 IEEE Seventh International Conference on Computer Science and Network Technology (ICCSNT)*. IEEE; 2019:370-374. https://doi.org/10.1109/ICCSNT47585.2019.8962430.

14. Singh RK, Dalal S, Chauhan VK, Kumar D. Optimization of FAR in intrusion detection system by using random Forest algorithm. *Proceedings of Second International Conference on Advanced Computing and Software Engineering (ICACSE)*; 2019. https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3350276.
15. Pervez M, Farid D. Feature selection and intrusion classification in NSL-UNSW-NB15 cup 99 dataset employing SVMS. *8th Int. Conf. Software, Knowledge Information Management and Applications* 2014, pp. 1-6.
16. Najafi M, Rafeh R. A new light weight intrusion detection algorithm for computer networks. *Adv Defence Sci Technol*. 2017;10:191-200.
17. Stein G, Chen B, Wu A, Hua K. Decision tree classifier for network intrusion detection with GA-based feature selection. *Proceedings of the 43rd Annual Southeast Regional Conference* 2005, 2, pp. 136-141.
18. Yang L, Cai M, Duan Y, Yang X, 2019. Intrusion detection based on approximate information entropy for random forest classification. In *Proceedings of the 2019 4th International Conference on Big Data and Computing*, pp. 125-129. https://doi.org/10.1145/3335484.3335488.
19. Tama BA, Rhee KH. An integration of PSO-based feature selection and random forest for anomaly detection in iot network. *MATEC Web of Conferences*. Vol 159. EDP Sciences; 2018:1053. https://doi.org/10.1051/matecconf/201815901053.

How to cite this article: Ajdani M, Ghaffary H. Introduced a new method for enhancement of intrusion detection with random forest and PSO algorithm. *Security and Privacy*. 2021;4:e147. https://doi.org/10.1002/spy2.147