Available at: www.researchpublications.org

Auto Detection of Attacks on Network

Anup G. Kadu[#] , Dr. A.S.Alvi[*]

[#] *M.E., Information Technology,P.R.M.I.T.& R,Badnera* - [*] *Information Technology,P.R.M.I.T.& R,Badnera*

[1]anupkadu@gmail.com

[2]abraralvi@gmail.com

Abstract -- The two knowledge-based approaches are not sufficient to tackle the anomaly detection problem, and that a holistic solution should also include knowledge-independent analysis techniques. There are some algorithms, and it becomes critical in the case of unsupervised detection, because there is no additional information to select the most relevant set some approaches can be easily extended to detect other types of attacks, considering different sets of traffic features. In fact, more features can be added to any standard list to improve detection and characterization results. The of Knowledge Independent Detection of Network Attack is simply to detect the attacks which are completely unknown to us. There is no previous knowledge about that data. There are some algorithms in existence which are used for network security but they are inefficient as they are knowledge based (Signature Based and Anomaly Based) whenever there is a vast amount of continuous incoming data then it is a big risk regarding the network attacks which are knowledge based. Our particular goal is to identify those attacks with the help of Robust Clustering Algorithm and make whole data secure.

***Keywords*---** **Signature Based, Anomaly Based, Robust Clustering**

I. INTRODUCTION

The unsupervised detection of network attacks represents an extremely challenging goal**.** The detection of network attacks is a paramount task for network operators in today's Internet. Denial of Service attacks (DoS), Distributed DoS (DDoS), network/host scans, and spreading worms or viruses are examples of the different attacks that daily threaten the integrity and normal operation of the network. The principal challenge in automatically detecting and analysing network attacks is that these are a moving and ever-growing target.

Two different approaches are by far dominant in the literature and commercial security devices: signature-based detection and anomaly detection. Signature-based detection systems are highly effective to detect those attacks which they are programmed to alert on. However, they cannot defend the network against unknown attacks. Even more, building new signatures is expensive and time-consuming, as it involves manual inspection by human experts. Anomaly detection uses labelled data to build normal-operation-traffic profiles, detecting anomalies as activities that deviate from this baseline. Such methods can detect new kinds of network attacks not seen before. Signature engines also have their disadvantages. Because they only detect known attacks, a signature must be created for every attack, and novel attacks cannot be detected. Signature engines are also prone to false positives since they are commonly based on regular expressions and string matching. Both of these mechanisms merely look for strings within packets transmitting over the wire. A disadvantage of anomaly-detection engines is the difficultly of defining rules. Each protocol being analyzed must be defined, implemented and tested for accuracy. The rule development process is also compounded by differences in vendor implementations of the various protocols. Custom protocols traversing the network cannot be analyzed without great effort. Moreover detailed knowledge of normal network behaviour must be constructed and transferred into the engine memory for detection to occur correctly. On the other hand, once a protocol has been built and a behaviour defined, the engine can scale more quickly and easily than the signature-based model because a new signature does not have to be created for every attack and potential variant.[1]

Our approach relies on robust clustering algorithms to detect both well-known as well as completely unknown attacks, and to automatically produce easy-to-interpret signatures to characterize them, both in an on-line basis. The analysis is performed on packet-level traffic, captured in consecutive time slots of fixed length ΔT and aggregated in IP flows (standard *5-tuples*). IP flows are additionally aggregated at 9 different *flow* levels l_i. These include: *source IPs*, *destination IPs*, *source Network Prefixes*, *destination Network Prefixes*, and *traffic per Time Slot*. The complete detection and characterization algorithm runs in three successive stages. The first step consists in detecting an anomalous time slot where an attack might be hidden. The unsupervised detection and characterization algorithm begins in the second stage, using as input the set of IP flows captured in the flagged time slot. The method uses robust clustering techniques based on Sub-Space Clustering (SSC), Density-based Clustering, and Evidence Accumulation (EA) to blindly extract the suspicious flows that compose the attack. In the third stage, the evidence of traffic structure provided by the clustering algorithms is used to produce filtering rules that characterize the detected attack and simplify its analysis. The characterization of an attack can be a hard and Time-consuming task, particularly when dealing with unknown attacks. Even expert operators can be quickly overwhelmed if simple and easy-to-interpret information is not provided to prioritize the time spent in the analysis. To alleviate this issue, the most relevant filtering rules are combined into a new traffic signature that characterizes the attack in simple terms. This signature can ultimately be integrated to any standard security device to

detect the attack in the future, which constitutes a major step towards autonomous security: in a nutshell, our algorithm automatically produces new signatures without any previous data about traffic or knowledge about the attack. [2]

II. RELATED WORK AND BACKGROUND

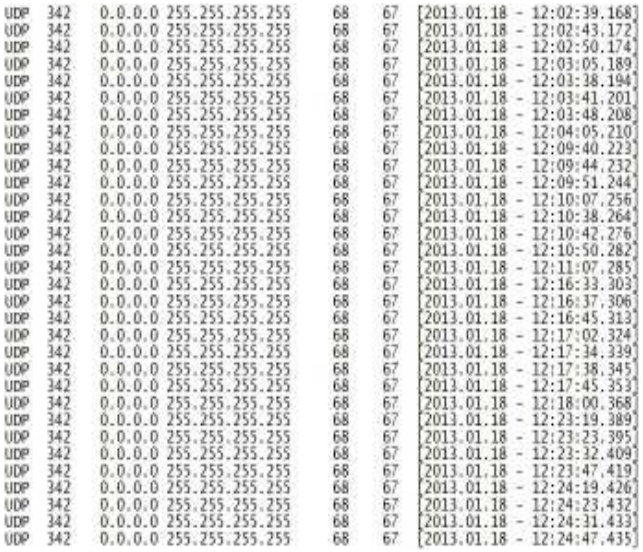
Two different approaches are by far dominant in current research literature and commercial detection systems: signature-based detection and anomaly detection. Signature-based detection is the de-facto approach used in standard security devices such as IDSs, IPSs, and firewalls. When an attack is discovered, generally after its occurrence during a diagnosis phase, the associated anomalous traffic pattern is coded as a signature by human experts, which is then used to detect a new occurrence of the same attack. Signature-based detection methods are highly effective to detect those attacks which they are programmed to alert on. However, they cannot defend the network against new attacks, simply because they cannot recognize what they do not know.[3][4]

In addition, building new signatures is a resources-consuming task, as it involves manual traffic inspection by human experts. On the other hand, anomaly detection uses labelled data to build normal- operation-traffic profiles, detecting anomalies as activities that deviate from this baseline. Such methods can detect new kinds of network attacks not seen before. Nevertheless, anomaly detection requires training for profiling, which is time- consuming and depends on the availability of purely anomaly-free traffic data sets. Labelling traffic as anomaly-free is not only time consuming and expensive, but also prone to errors in the practice, since it is difficult to guarantee that no anomalies are buried inside the collected data. In addition, it is not easy to keep an accurate and up-to-date normal-operation profile. Our thesis is that these two knowledge-based approaches are not sufficient to tackle the anomaly detection problem, and that a holistic solution should also include knowledge-independent analysis techniques. To this aim we propose UNADA, an Unsupervised Network Anomaly Detection Algorithm that detects network traffic anomalies without relying on signatures, training, or labelled traffic of any kind. Based on the observation that network traffic anomalies are, by definition, sparse events that deviate markedly from the majority of the traffic, UNADA relies on robust clustering algorithms to detect outlying traffic flows.[5]

III. ALGORITHM

Algorithm define the specific steps as follows:

Log File-



UDP	342	0.0.0.0	255.255.255.255	68	67	[2013.01.18 - 12:02:39.168]
UDP	342	0.0.0.0	255.255.255.255	68	67	[2013.01.18 - 12:02:43.172]
UDP	342	0.0.0.0	255.255.255.255	68	67	[2013.01.18 - 12:02:50.174]
UDP	342	0.0.0.0	255.255.255.255	68	67	[2013.01.18 - 12:03:05.189]
UDP	342	0.0.0.0	255.255.255.255	68	67	[2013.01.18 - 12:03:38.194]
UDP	342	0.0.0.0	255.255.255.255	68	67	[2013.01.18 - 12:03:41.201]
UDP	342	0.0.0.0	255.255.255.255	68	67	[2013.01.18 - 12:03:48.208]
UDP	342	0.0.0.0	255.255.255.255	68	67	[2013.01.18 - 12:04:05.210]
UDP	342	0.0.0.0	255.255.255.255	68	67	[2013.01.18 - 12:09:40.223]
UDP	342	0.0.0.0	255.255.255.255	68	67	[2013.01.18 - 12:09:44.232]
UDP	342	0.0.0.0	255.255.255.255	68	67	[2013.01.18 - 12:09:51.244]
UDP	342	0.0.0.0	255.255.255.255	68	67	[2013.01.18 - 12:10:07.256]
UDP	342	0.0.0.0	255.255.255.255	68	67	[2013.01.18 - 12:10:38.264]
UDP	342	0.0.0.0	255.255.255.255	68	67	[2013.01.18 - 12:10:42.276]
UDP	342	0.0.0.0	255.255.255.255	68	67	[2013.01.18 - 12:10:50.282]
UDP	342	0.0.0.0	255.255.255.255	68	67	[2013.01.18 - 12:11:07.285]
UDP	342	0.0.0.0	255.255.255.255	68	67	[2013.01.18 - 12:16:33.303]
UDP	342	0.0.0.0	255.255.255.255	68	67	[2013.01.18 - 12:16:37.306]
UDP	342	0.0.0.0	255.255.255.255	68	67	[2013.01.18 - 12:16:45.313]
UDP	342	0.0.0.0	255.255.255.255	68	67	[2013.01.18 - 12:17:02.324]
UDP	342	0.0.0.0	255.255.255.255	68	67	[2013.01.18 - 12:17:34.339]
UDP	342	0.0.0.0	255.255.255.255	68	67	[2013.01.18 - 12:17:38.345]
UDP	342	0.0.0.0	255.255.255.255	68	67	[2013.01.18 - 12:17:45.353]
UDP	342	0.0.0.0	255.255.255.255	68	67	[2013.01.18 - 12:18:00.368]
UDP	342	0.0.0.0	255.255.255.255	68	67	[2013.01.18 - 12:23:19.389]
UDP	342	0.0.0.0	255.255.255.255	68	67	[2013.01.18 - 12:23:23.395]
UDP	342	0.0.0.0	255.255.255.255	68	67	[2013.01.18 - 12:23:32.409]
UDP	342	0.0.0.0	255.255.255.255	68	67	[2013.01.18 - 12:23:47.419]
UDP	342	0.0.0.0	255.255.255.255	68	67	[2013.01.18 - 12:24:19.426]
UDP	342	0.0.0.0	255.255.255.255	68	67	[2013.01.18 - 12:24:23.432]
UDP	342	0.0.0.0	255.255.255.255	68	67	[2013.01.18 - 12:24:31.433]
UDP	342	0.0.0.0	255.255.255.255	68	67	[2013.01.18 - 12:24:47.435]

We have to create log file which contain the specific feature of the data.

We can create log file both by using software and hardware is on our choice and we create log by using software.

Then we have find out the data flow in the log file weather it may maximum or not.

Apply sliding time windowing scheme for specific amount of time .

Aggregation process for traffic flow

we have created feature space matrix

X(1)=[sip dip sp dp nsip/ndip y(1)/ndip]

II^{ly} we have to create feature space matrices for all time windows data set.

i.e., X= € (x1, x2…….xn)

And then apply clustering algorithm and declare smallest group of cluster as outlier.

Detect outlier using outlier detection algorithm.

Trace back outlier in feature space matrix, aggregation and log file.

Use trace data to Create signature for anomalous flow.

Signature will be logged and updated the signature table.

Signature table can be use for online detection anomalous flow

IV SYSTEM DESIGN

In the system design input data at first that contain the data packets. A data set is an ordered sequence of object, this may contain anomaly and we have to detect anomalies in the data set to detect that anomalies in the huge dataset we have to apply robust clustering approach which will create automatic signature. In my proposed work I am going to implement completely blind approach so for that no any previous knowledge about the anomaly and to detect such types of blind attack I am going to apply robust clustering approach for the detection of network anomaly in an completely unsupervised fashion .

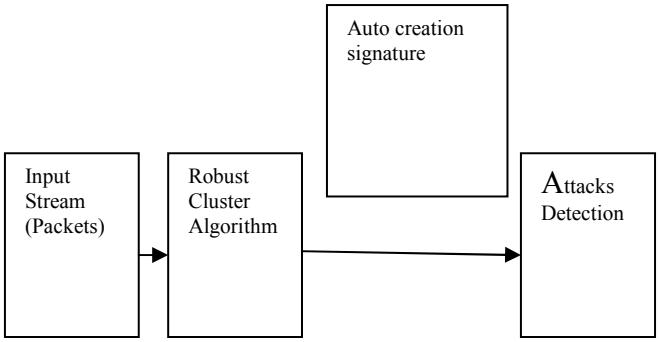


Fig.1 Organization of the system

A. ***Advantages the proposed system on the existing Approaches:***

Our unsupervised algorithm has several terms w. r. t. the state of the art:

I) First and most important, it works in a completely unsupervised fashion, which means that it can be directly plugged-in to any monitoring system and start to work from scratch, without any kind of calibration or previous knowledge.

II) It combines robust clustering techniques to avoid general clustering problems such as sensitivity to initialization, specification of number of clusters, or structure-masking by irrelevant features.

III) It automatically builds compact and easy-to-interpret signatures to characterize attacks, which can be directly integrated into any traditional security device.

IV) It is designed to work on-line, using the parallel structure of the proposed clustering approach.

V CONCLUSION

An Auto Detection of Attacks on Network presents many interesting feature with respect to previous algorithms in the field unsupervised anomaly detection. It uses completely unlabeled data to detect traffic anomalies, without considering any specific model or any standard traffic flow. It detect anomaly without using previous signatures of anomalies or any kind of training by using labelled traffic. Rather using ordinary clustering techniques to identify anomalies. This approach avoids the general clustering problem of sensitivity. Many Unsupervised Network Anomaly Detection Algorithm have the lack of robustness of general clustering approaches. Density-based Clustering and multiple Evidence Accumulation. It also work in an on-line basis. It is very effective way detecting network changes.

REFERENCES:

[1] S. Hansman, R. Hunt "A Taxonomy of Network and Computer Attacks", in *Computers and Security*, vol. 24 (1), pp. 31-43, 2005.

[2] Paul Barford, Jeffery Kline, David Plonka and Amos Ron "A Signal Analysis of Network Traffic Anomalies" In procedeengs of ACM Sigcomminternet Measurment Workshop 2002

[3] A. Soule et al., "Combining Filtering and Statistical Methods for Anomaly Detec-tion", in Proc. ACM IMC, 2005

[4] Balachander Krishnamurthy, Subhabrata Sen, Yin Zhang Yan Chen_ AT&T Labs–Research; "Sketchbased Change Detection: Methods, Evaluation, and Applications"
180 Park Avenue University of California *IMC'03,* October 27–29, 2003.

[5] Ana L.N. Fred Telecommunications Institute Instituto Superior T´ecnico, Portugal and Anil K. Jain Dept. of Computer Science and Engineering Michigan State University, USA "Data Clustering Using Evidence Accumulation"2009.