A survey of Intrusion Detection System technologies

Ross Heenan* Dr Naghmeh Moradpoor
0705974@abertay.ac.uk n.moradpoor@napier.ac.uk
*School of Arts, Media and Computer Games (AMG)
Abertay University
DUNDEE, DD1 1HG, UK

ABSTRACT – *This paper provides an overview of IDS types and how they work as well as configuration considerations and issues that affect them.*

Advanced methods of increasing the performance of an IDS are explored such as specification based IDS for protecting Supervisory Control And Data Acquisition (SCADA) and Cloud networks.

Also by providing a review of varied studies ranging from issues in configuration and specific problems to custom techniques and cutting edge studies a reference can be provided to others interested in learning about and developing IDS solutions.

Intrusion Detection is an area of much required study to provide solutions to satisfy evolving services and networks and systems that support them.

This paper aims to be a reference for IDS technologies other researchers and developers interested in the field of intrusion detection.

Keywords: Intrusion Detection Systems (IDS), Host-based IDS (HIDS), Network-based IDS (NIDS), Machine Learning, Artificial Neural Network, Genetic Algorithm

1. INTRODUCTION

Intrusion Detection Systems (IDS) are a powerful and versatile tool available for Network Security Management that are able to provide an effective and efficient solution for preventing intrusions such as SQL injection, Denial of service (DoS) attacks or misuse of the Computer network or services it is deployed to protect. They are particularly useful in modern network architectures such as Cloud, Wireless and Supervisory Control And Data Acquisition (SCADA) networks due to their adaptability and functionality and the complexity of these networks requiring efficient security solutions.

IDS come in various types ranging from Host-based, Network-based, Hybrid, Distributed and Wireless and also in different form such as a physical network device or as software to be installed on a host or network device.

Host-based IDS are used to detect intrusions on a single host device using methods such as whitelisting and file integrity checking, Network based are used to detect intrusions or misuse over a network by monitoring the traffic they capture. Hybrid IDS are customised methods that use custom approach methods of detection or use a combination of IDS types as well as a combination of detection methods. Distributed systems are those that are deployed and controlled by a third party provider and Wireless IDS are clearly for the detection of intrusions or misuse over a wireless networking connection.

There are two main methods of detection used with IDS systems, Signature based and Anomaly based. Signature based detection uses known signatures or rules to generate intrusion alerts if transactions or events have occurred using the defined IP addresses, ports and protocols in the rules created.

Anomaly based detection generally uses machine learning methods such as Genetic Algorithms, Support Vector Machine (SVM) and Artificial Neural Networks in combination with knowledge bases or data sets of acceptable and attack type traffic. These methods are able to identify novel attacks that are similar to ones it is trained to by the dataset used.

Studies on Intrusion detection systems are now becoming generally more scoped and use more complex methods such as Artificial Neural network, Genetic Algorithms and Support Vector Machine and specification based detection methods to optimise the effectiveness of the Intrusion detection system solution deployed for situations that require a custom solution provided such as SCADA or Cloud networks

The aim of this survey is to provide an analysis, comparison and evaluation of types of IDS, their configurations, methods used for detection, and scenarios they are suitable to be deployed to provide an effective security solution.

The rest of the paper will be structured as follows, the next section will provide a further background of IDS, their configurations and detection methods. Related work will then be overviewed in terms of specific problems concerning IDS such as methods of evasion along with custom techniques required for custom deployments such as a Cloud or Wireless networks. State-of the art recent studies will also be reviewed in order to identify the most efficient and accurate methods currently available for detection and preventing intrusions in computer networks and their host systems.

Finally conclusion with overview of the surveys significant points will be provided to allow those interested to be able to be access a concise overview of IDS technologies and their methods of detection and prevention as well as specific areas such as issues affecting them and their solutions. This will hopefully allow for contribution to further studies also carried out in future by others.

2. BACKGROUND

More advanced IDS detection methods use machine learning techniques along with knowledge bases to train them to detect abnormal behaviour or unknown attacks.

As stated in the previous section there are many types of IDS exist such as Network-based (NIDS), Host-based (HIDS), Distributed (DIDS) and others like Hybrid, Wireless and specification based IDS's. Different IDS are able to use different methods of detection and some can use various methods.

HIDS such as Tripwire [22] and Open Source Host Based Intrusion Detection System (OSSEC) [19] use whitelists of the filesystem it is protecting to perform file integrity scans which identify any abnormalities which can classify possible intrusions.

NIDS such as Snort [20] and Bro [16] use rule sets that define a type of intrusion or unacceptable behaviour such as a port scan or a DoS attack attempt.

Wireless IDS such as Kismet [17] and OpenWIPS-ng [18] capture wireless traffic and detects and identifies standard and hidden networks in order to attempt to detect intrusions. Specification based IDS have also been designed in recent studies that concentrate on the configuring of an IDS solution that will allow the method of detection to be based on an essential part of the network such as protecting use of SCADA specific protocols as a basis for the method of detection.

More recently there has also been a customised OS created called Security Onion that contains various IDS such as Snort, Bro, Suricata and many other tools preinstalled.

Many factors affect the efficiency of an IDS solution such as the positioning of the IDS or its sensors, the mode they are operated, the way that they are configured and the resources they require to operate.

Gaedke et al [1] provides an assessment of various IDS to assess the most suitable for use as a NIDS. Storage space, bandwidth required within the network it is deployed and computing resources it requires such as memory and CPU requirements and also impact on the users of the network also need to be considered. This study by Gaedke et al proposes that a standardised and unbiased method of assessing an IDS's effectiveness in managing security of a network by using a combination of virtual and physical machines in a test environment. They also propose their method will help manage assessment of developing and optimising IDS solutions.

Snort is shown to be the least demanding in terms of CPU processing power as shown from the results of tests carried out in [1].

The placement and configuration of an IDS sensors is dependent on the requirements of the network and its topology and services in order to provide an effective intrusion detection solution.

Ayobami, Babatunde and Olumide [2] provide a research carrying out the identification of suitable placements and configurations of a Network Intrusion Detection System (NIDS). May factors such as access to the server setups, internet access points, remote access and intranet configuration and as identification of any bottlenecks affect the suitable placement of IDS sensors.

Also in order to identify an effective IDS solution it is needed to identify the critical systems and services in the network in order to identify attackers obvious target such as DNS, Web or mail servers, Host and Database systems as well as other network devices such as routers that could be flooded by a DoS attack.

Another consideration raised by Ayobami, Babatunde and Olumide is what mode to configure the sensors to capture traffic in, the two modes are inline and passive. In inline mode all traffic must transfer through it, this is used to be able to blocks specific traffic providing the advantage of not requiring other resources other than the NIDS. There are risks to running an IDS inline like affecting the network if failure occurs and also the need for effective configuration and training of the system.

Sensors can also be run in passive mode in which the traffic doesn't actually pass through the sensor rather a copied version of the traffic is received to be monitored and analysed for intrusion identification.

Ayobami, Babatunde and Olumide [2] highlight that many different strategies for placement of IDS sensors in a network depending on the type and structure of it and its resources and that no one method is perfect.

Other issues with modes of operation are ensuring that functions such as large receive offload (lro) and generic receive offload (gro) are turned off on the network in order to capture traffic at full capacity to be able to capture and monitor the traffic being received and identify intrusions.

Network Interface Cards (NIC) also require to be set to promiscuous mode to be able to capture all packets that are transmitted over the interface.

IDS solutions can be implemented in a distributed application structure by using multiple sensors deployed at important points of the network such as bottlenecks or in front of crucial network systems. These sensors are monitored and controlled by a centralised controller IDS which give better visibility of the network, allowing for intrusions or misuse to be identified with more ease and in turn providing more ease in the management of then network.

Cloud networks holding data such as personal files or medical records for example require complex IDS solutions that can be up scaled and adapted or an organisation can suffer major losses in terms of revenue, trust from its users and also third party suppliers if an attack is successful. These are usually implemented in a Distributed or Hybrid structure using combinations of detection methods and often combinations of HIDS, NIDS and use of other IDS methods such as specification based or machine learning.

Many tools assist the IDS in the management of the network it is protecting such as the use of a centralised storage database for storing data such as IDS alert logs and storing of captured events. GUI interfaces can also be used for managing the IDS with more ease and viewing intrusions or events with more clarity, this in turn will assist the administrator of the network in being able to manage and also adapt and evolve the security and acceptable use policies of the network and also better manage the users and resources available.

Flaws that exist with IDS include bypassing with encrypted or obfuscated data, or attacks that are fragmented and sent with a time delay. Also attacks that are sent over multiple paths have also proven to fool and IDS's detection.

Many advanced and custom methods of detection exist such as the use of Genetic Algorithms (GA), Artificial Neural Network (ANN), Support Vector Machine (SVM), Fuzzing and other variations, these require a knowledge base or dataset such as the DARPA '99 dataset in order to train the designed system to be able to efficiently and accurately detect intrusions.

IDS that can be used with anomaly methods such as Snort NIDS with Statistical Packet Anomaly Detection Engine (SPADE) [21], Network Anomaly Detection Engine (NETAD) [6], Bro with scripts and many others. Load balancing of servers are also possible to assist IDS in performance if heavy traffic loads are received.

3. LITERATURE STUDY

Increasing IDS performance, development of novel techniques and problems affecting IDS are key areas of IDS research.

A. *Increasing IDS performance*

Kumar and Yadav [3] use ANN neural networks method using the KDD cup '99 dataset and shows high anomaly detection by using a gradient descent with momentum back propagation algorithm to train the system in detection. Their method was as efficient in detection and classification of attacks as current methods implemented but only used random patterns of data were used for training.

Jain and Swarup [4] highlight that methods currently used are not evolving proportionately with the evolvement in complexity and networks their technologies. MLP (Multi-layer perceptron) neural network is proposed as a solution for it fast learning time and real time detection ability.

The findings by Jain and Swarup suggest that the use of an MLP Neural Network allows creation of a uniformed dataset from a dynamically created one to allow for the neural network be trained effectively for detection and be able to provide real time detection

Mahoney and Matthew [5] is an early anomaly IDS detection study proposing the use of the NETAD IDS to filter traffic captured by using modelling at the packet byte level to find the commonly used protocols such as HTTP, IP etc.…

The study highlights the need for a data set to be constantly updated and also the training of the IDS system to be sufficient in order to provide a system that is accurate and efficient in detection.

B. *Custom detection techniques*

Aggarwal and Shah [6] investigate the comparison of testing multiple IDS (Snort, Suricata and NETAD) using a method to provide Heterogeneous Fusion of alerts with the DARPA '99 dataset which is a dataset that can be used as a knowledge base for testing Intrusion Detection events to identify which is the most effective in accuracy in detecting novel attacks. Their study showed NETAD was the most improved in false alarm detection by 40% and also in efficiency of 20% in accuracy using their proposed alert fusion method

In [7] Jaidhar and Kumara suggest deploying a hybrid security solution within each VM with in order to classify attacks with more efficiency. This was done uing a mixture of an IDPS based within each VM and also an IDPS situated in the Hypervisor.

Simulated attacks such as DoS, user2root and rootkits were carried out and the results evaluated. The structure proposed can be seen in the figure proposed by (Jaidhar and Kumara 2015) with the custom IDPS deployment shown below.

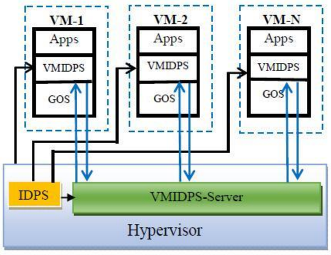

Fig .1. Architecture of VMIDPS for Virtualized Environment

Their study was successful in detection of the attacks attempted but further testing was required in a real world environment.

Aljurayban and Emam [8] propose the use of a Layered Intrusion Detection Framework (LIDF) for efficient protection of a Cloud networking environment using an Artificial Neural Network (ANN) to create a data mined knowledge base for detection. The aim to be efficient in detection and require less analysis of traffic with increased throughput in turn giving a more efficient performing system. The LIDF framework is also supposed to be able to handle mass traffic and being up scaled to a more demanding environment while retaining effective management of the Cloud network and its services.

Tests were carried out using MATLAB using GUI network simulator NS2 to provide simulation environments using ANN or other machine learning methods. The system implemented showed efficient results in training and experiments and was able to reduce false positive occurrence but also affected the true positive rate in detection. The accuracy of detection in all experiments ranged between 80 to 100%.

The method by Aljurayban and Emam [8] provides an optimised solution for dealing with intrusions at different layers of the Cloud network infrastructure. The study will require further optimisation in areas such as training data used and also implementing in different Cloud structures to assess it universal effectiveness.

C. Specific problems

In [9] Garuba, Grima and Goel provide an analysis and of DDoS occurrence in the Cloud networking environment and the proposition of IDS solutions.

IDS are proposed as a key tool for detection of these types of attacks using machine learning techniques to identify patterns of abnormality. The design is a hybrid IDS solution to address these attacks by deploying various methods and detection algorithms for comparison to identify the most suitable.

Research by Jiefei, Lobo and Russo [10] investigates the occurrence of Multi-path routed attacks where an attack is fragmented and sent over multiple routes to attempt to fool an IDS system. This is made possible due to multi path TCP (MPTCP) which allows transmissions to route over multiple paths between a source and target.

A Distributed IDS (DIDS) using an algorithm to attempt to match signatures over multiple routes by string matching automation through analysis of each packet received locally. Then broadcasting its state to other sensors in the network allowing them carry out synchronised scans for traffic targeted to the same location over the multiple routes.

The proposed algorithm performs well but loses efficiency in detecting signature but also minimises resources required for communication with the more volume of traffic being received.

Optimisation of the algorithm is required to reduce delay in detection, delay encountered when out of sequence packets are received and also when obfuscation is used.

Di Martino, et al [11] proposes in order to protect SCADA systems effectively a solution needs to be developed to protect them using a IDS solution that will concentrate on being able to manage protocols used to control SCADA devices in order to provide a better solution.

Bro IDS is used with an incorporated DNP3 parser along with a protocol validation policy to be able to manage and understand the SCADA device's transmitted data was used.

The aim being able to better craft policies for the IDS from being able to better understand the network connections or transmissions between devices and also to be able to identify abnormal or malicious activity especially that specific to SCADA systems.

Inter packet validation also ensures that required patterns of communication are occurring between transmission of packets, if they are not then this can constitute the occurrence of a replay attack or a DoS. The method was successful although 30% slower at processing packets than without the use of the policy introduced. This however ensured thorough analysis of all data within a DNP3 packet and suggests that the model would be able to handle the resource demand of the amount of devices contained within a standard SCADA network environment.

D. State-of-the-art studies

Kim, Kim and Gyu [12] propose a customised solution of the use of a framework using an Aho-Corasick pattern matching algorithm that generates malware signatures and rules corresponding to them for detection malware and others similar. The method uses The Major Block Comparison algorithm to create signatures from variants of common malware, these signatures contain information such as the signatures of the malware payloads and offset values.

An issue raised with the method in the study is that if a payload received is split across multiple packets then the system will not recognise it, to resolve this Delay method is used to allow part matching of signatures in packets.

In [13] Hofstede et al propose the use of an in development IDS named NetFlow/IPFIX for identification of dictionary attacks aimed at the HTTP sites and provide accuracies in detection reaching close to 100% but with an issue with false positives. The solution Hofstede et al proposes shows through analysis shown encouraging results in detecting brute force attacks.

Barolli et al [14] investigates the use of IDS using neural network for providing IDS solution in a Tor (The Onion Router) network. Tests carried out used a Tor server and client with back propagation NN to simulate transactions over the Tor network while capturing for analysis.

The system proposed is a trained ANN with data captured from Wireshark, then the server and client data are compared, differences will identify an intrusion or exploitation.

The results from testing were successful in providing effective accuracy when evaluated in the test environment.

Bellekens et al [15] propose the use of pattern matching at high speeds by employing the use of a General Purpose Graphics Processing Unit (GPGPU) and the use of an Aho-Corasick algorithm which is an algorithm that allow the searching for of strings. GPU's are proven to be able to process volumes of data and provide services such as pattern matching with great efficiency making them suitable for IDS techniques as they require efficient computational processing abilities.

The method uses a powerful compression to reduce memory required for data storage for training of pattern matching and uses the string matching algorithm in conjunction with the GPU to maximise processing throughput of data for the IDS. An advantage of this approach is that it allows the GPU to provide increased throughput of up to 8Gbps in processing of data between it and the IDS and is proposed to allow efficiency to be sustained in different deployment situations. This would provide a suitable solution in handling attacks such as DoS or DDoS as the processing of higher volumes of traffic would be capable in the use of a GPU or multiple GPU's.

4. DISCUSSION & CONCLUSION

This paper provides a background of IDS types and their use along with considerations in configuration and placement of them and issues affecting them such as encryption and fragmentation.

Many methods of detection have been reviewed in related work showing there are many different methods in detection depending on the environment the solution is deployed to and its requirements.

Many of the studies reviewed propose the use of hybrid methods incorporating machine learning or specification based techniques for efficient detection. This shows there is a requirement for much further study to be carried out in the area of intrusion detection as networks and their services are constantly evolving requiring new solutions provided for them. This study also aims to act as a reference to others for a background in IDS technologies and those developing or studying IDS solutions.

5. REFERENCES

[1] Gaedke, Matt; Hu, Lihui; Kordas, Alex; Smith, Derrick; Wang, Xinli.(2013). Administrative Evaluation of Intrusion Detection System. *Proceedings of the 2nd annual conference on Research in information technology*, p47-52.

[2] Ayobami, Ibitola; Babatunde, Lawal; Olumide Babatope, Longe. (19-21 April 2012). Strategic Sensor Placement for Intrusion Detection in Network-Based IDS. *MECS*, p61-68.

[3] Kumar, S.; Yadav, A.. (8-10 May 2014). Increasing Performance Of Intrusion Detection System Using Neural Network. *Advanced Communication Control and Computing Technologies (ICACCCT), 2014 International Conference on*, p546-550.

[4] Jain, Pritesh; Swarup Sen, Anand. (8-9 March 2014). Technique of Intrusion Detection Based on Neural Network- A Review. *IT in Business, Industry and Government (CSIBIG), 2014 Conference on*, (2), p1-3.

[5] V. Mahoney, Matthew. (2003). Network Traffic Anomaly Detection Based on Packet Bytes. *Proceedings of the 2003 ACM symposium on Applied computing*, p346-350.

[6] Aggarwal, A.K., Shah, V. (26-27 Feb. 2015). Heterogeneous Fusion of IDS alerts for Detecting DOS Attacks. *Computing Communication Control and Automation (ICCUBEA), 2015 International Conference on*, p153-158.

[7] Jaidhar, C.D; Kumara M A, A. (26-28 May 2015). Hypervisor and virtual machine dependent Intrusion Detection and Prevention System for virtualized cloud environment. *Telematics and Future Generation Networks (TAFGEN), 2015 1st International Conference on*, p28-33.

[8] Aljurayban, N.S.; Emam, A. (21-23 March 2015). Framework for Cloud Intrusion Detection System Service. *Web Applications and Networking (WSWAN), 2015 2nd World Symposium on*, p1-5.

[9] Garuba, M.; Goel, R; Grima, A. (7-9 April 2014). Cloud Computing Vulnerability: DDoS as its main Security Threat, and Analysis of IDS as a Solution Model. *Information Technology: New Generations (ITNG), 2014 11th International Conference on*, p307-312.

[10] Jiefei Ma; Le, F.; Lobo, J.; Russo, A. (April 26 2015-May 1 2015). Detecting Distributed Signature-based Intrusion: The Case of Multi-Path Routing Attacks. *Computer Communications (INFOCOM), 2015 IEEE Conference on*, p558-566.

[11] Di Martino, Catello; Lin, Hui; Kalbarczyk, Zbigniew; K. Iyer, Ravishankar; Slagell, Adam;. (2012). l*CSIIRW '12*, p1-4.

[12] Kim, SunWoo; Kim, TaeGuen; Gyu Im, Eul. . (October 1–4, 2013). Real-time Malware Detection Framework in Intrusion Detection Systems. *Proceedings of the 2013 Research in Adaptive and Convergent Systems*, p351-351.

[13] Hofstede, R.; Jonker, M.; van der Toorn, O.; Sperotto, A. (11-15 May 2015). A First Look at HTTP(S) Intrusion Detection using NetFlow/IPFIX. *Integrated Network Management (IM), 2015 IFIP/IEEE International Symposium on,* p862-865.

[14] Barolli, Leonard; Elmazi, Donald; Ishitaki, Oda, Tetsuya; Taro; Yi Liu, Uchida, Kazunori. (24-27 March 2015). Application of Neural Networks for Intrusion Detection in Tor Networks. *Advanced Information Networking and Applications Workshops (WAINA), 2015 IEEE 29th International Conference on*, p67-72.

[15] Xavier J. A. Bellekens, Christos Tachtatzis, Robert C. Atkinson, Craig Renfrew, and Tony Kirkham. 2014. A Highly-Efficient Memory-Compression Scheme for GPU-Accelerated Intrusion Detection Systems. In Proceedings of the 7th International Conference on Security of Information and Networks (SIN '14). ACM, New York, NY, USA, , Pages 302 , 8 pages. DOI=http://dx.doi.org/10.1145/2659651.2659723

[16] *Bro.* Available: http://bro-ids.org/. Last accessed 10th Sep 2015

[17] *Kismet.* Available: https://www.kismetwireless.net/. Last accessed 8th Sep 2015.

[18] *OpenWIPS-ng.* Available: http://www.openwips-ng.org/. Last accessed 10th Sep 2015.

[19] *OSSEC.* Available: http://www.ossec.net. Last accessed 10th Sep 2015.

[20] *Snort.* Available: http://www.snort.org. Last accessed 10th Sep 2015.

[21] *SPADE.* Available: https://sourceforge.net/projects/snortspade/. Last accessed 7th Sep 2015.

[22] *Tripwire.* Available: www.tripwire.org/. Last accessed 10th Sep 2015.