On improving the efficiency of intrusion detection systems using game theoretic approaches

Thesis submitted in partial fulfilment of the requirements
for the award of the degree of

Doctor of Philosophy

in

Computer Science and Engineering

by

Basant Subba

Under the supervision of

Dr. Sushanta Karmakar
Dr. Santosh Biswas

Department of Computer Science and Engineering

Indian Institute of Technology Guwahati

Guwahati - 781039, India

JANUARY, 2018

Dedicated to

***Parents**, family and friends !*

For their unconditional love, patience, sacrifices and continued support during my successful journey

Two roads diverged in a yellow wood,

And sorry I could not travel both

And be one traveler, long I stood

And looked down one as far as I could

To where it bent in the undergrowth;

Then took the other, as just as fair,

And having perhaps the better claim,

Because it was grassy and wanted wear;

Though as for that the passing there

Had worn them really about the same,

And both that morning equally lay

In leaves no step had trodden black.

Oh, I kept the first for another day!

Yet knowing how way leads on to way,

I doubted if I should ever come back.

I shall be telling this with a sigh

Somewhere ages and ages hence:

Two roads diverged in a wood, and I-

I took the one less traveled by,

And that has made all the difference.

Robert Frost

Declaration

I certify that

- The work contained in this thesis is original and has been done by myself and under the general supervision of my supervisor(s).
- The work reported herein has not been submitted to any other Institute for any degree or diploma.
- Whenever I have used materials (concepts, ideas, text, expressions, data, graphs, diagrams, theoretical analysis, results, etc.) from other sources, I have given due credit by citing them in the text of the thesis and giving their details in the references. Elaborate sentences used verbatim from published work have been clearly identified and quoted.
- I also affirm that no part of this thesis contains plagiarised contents to the best of my knowledge and I understand and take complete responsibility if any complaint arises.
- I am fully aware that my thesis supervisor(s) are not in a position to check for any possible instance of plagiarism within this submitted work.

July 3, 2018 Basant Subba

Department of Computer Science and Engineering
Indian Institute of Technology Guwahati
Guwahati - 781039, India

Dr. Santosh Biswas
Associate Professor
Email : santosh_biswas@iitg.ernet.in
Phone : +91-361-258-2364

Dr. Sushanta Karmakar
Associate Professor
Email : sushantak@iitg.ernet.in
Phone : +91-361-2582368

Certificate

This is to certify that this thesis entitled "**On improving the efficiency of intrusion detection systems using game theoretic approaches**" submitted by **Basant Subba**, in partial fulfilment of the requirements for the award of the degree of Doctor of Philosophy, to the Indian Institute of Technology Guwahati, Assam, India, is a record of the bonafide research work carried out by him under my guidance and supervision at the Department of Computer Science and Engineering, Indian Institute of Technology Guwahati, Assam, India. To the best of my knowledge, no part of the work reported in this thesis has been presented for the award of any degree at any other institution.

Date: July 3, 2018
Place: IIT Guwahati

Dr. Sushanta Karmakar

Dr. Santosh Biswas

Publications Related to Thesis

Journals

1. **Basant Subba**, Santosh Biswas and Sushanta Karmakar. "A game theory based multi layered intrusion detection framework for VANET." *Future Generation Computer Systems*, Elsevier, 2018. (Available online: https://doi.org/10.1016/j.future.2017.12.008)

2. **Basant Subba**, Santosh Biswas and Sushanta Karmakar. "False Alarm Reduction in Signature based IDS: Game Theory Approach." *Security and Communication Networks*, Wiley, 9(18), 4863–4881, 2016.

3. **Basant Subba**, Santosh Biswas and Sushanta Karmakar. "Intrusion Detection in Mobile Ad-hoc Networks: Bayesian Game Formulation." *Engineering Science and Technology, an International Journal (JESTECH)*, Elsevier, 19(2), 782–799, 2016.

4. **Basant Subba**, Santosh Biswas and Sushanta Karmakar. "A game theory based multi layered intrusion detection framework for wireless sensor networks." *International Journal of Wireless Information Networks*, Springer, 2018. (Available online: https://doi.org/10.1007/s10776-018-0403-6)

Conferences

1. **Basant Subba**, Santosh Biswas and Sushanta Karmakar. "*Intrusion Detection Systems using Linear Discriminant Analysis and Logistic Regression.*" Annual IEEE India Conference (INDICON) held at Jamia Millia Islamia University, India, 17 Dec – 20 Dec, 2015.

2. **Basant Subba**, Santosh Biswas and Sushanta Karmakar. "*A Neural Network based system for Intrusion Detection and attack classification.*" National Conference on Communication (NCC) held at Indian Institute of Technology Guwahati, India, 4th – 6th March, 2016.

3. **Basant Subba**, Santosh Biswas and Sushanta Karmakar. "*Enhancing performance of anomaly based intrusion detection systems through dimensionality reduction using principal component analysis.*" International Conference on Advanced Networks and

Telecommunications Systems (ANTS), held at the Indian Institute of Science, Bangalore, India 6th – 9th Nov, 2016.

4. **Basant Subba**, Santosh Biswas and Sushanta Karmakar. "*Enhancing effectiveness of intrusion detection systems: A hybrid approach.*" International Conference on Advanced Networks and Telecommunications Systems (ANTS) , held at the Indian Institute of Science, Bangalore, India 6th – 9th Nov, 2016.

5. **Basant Subba**, Santosh Biswas and Sushanta Karmakar. "*Host Based Intrusion Detection System Using Frequency Analysis of N-Gram Terms.*" IEEE Region 10 Conference (TENCON), held at Penang, Malaysia, 5th–8th Nov., 2017

Acknowledgements

This thesis is a result of culmination of efforts of many people who have helped me directly or indirectly during my exciting journey of PhD.

First and foremost I would like to profusely thank my supervisors Dr. Santosh Biswas and Dr. Sushanta Karmakar. For Dr. Santosh Biswas I would like to say that I am very fortunate to have him as my thesis supervisor. He provided me with enough autonomy and valuable guidance to formulate my research ideas, which were crucial towards the successful completion of my PhD work. He has always been a call away whenever I felt like discussing my research related problems with him. There were times when I felt like I had reached a dead end and saw no way forward but he has always been there to motivate and guide me during such difficult times. I would also like to thank him immensely for his guidance on helping me write my research papers and for his invaluable feedbacks while improving and addressing my journal papers' reviews.

I will always remain indebted to Dr. Sushanta Karmakar for taking me in as his PhD student and for helping me at various stages of my PhD journey, specially during my initail PhD days, when I was under stress and had troubles figuring out my research directions. He has always been supportive of my work and a constant source of motivation. He has provided me with new research directions at different stages of my PhD life and helped me look at research problems from various perspectives. I will always be thankful to him for his invaluable advices and feedbacks while correcting my journal papers' reviews.

It was a previlage to have Prof. Diganta Goswami, Prof. Sajith Gopalan and Dr. Partha Sarathi Mandal as the honourable members of my thesis doctoral committee. I would like to express my earnest gratitude to all of them for their invaluable time in evaluating my work progress and for their fruitful advices towards improving the quality of my reseach work. It is a matter of great pride and honour to be a student of IIT Guwahati. I express my sincere gratitude to all the academic and non-academic personnel of IIT Guwahati whose persistent efforts have ensured that this place remains conducive for research and academic development. I owe my sincere thanks to Prof. Shivashankar B. Nair (former Head, Deptt. of CSE), Prof. S.V. Rao (Current Head, Deptt. of CSE), Prof. Gautam Barua (former Director), Prof. Gautam Biswas (present Director), all the Deans and administrative staffs of the institute for all their support. I also express my sincere regards to all the respected faculties and staffs of the CSE department for their extended help and support.

The office staffs at the department of CSE are always cordial and supportive. I would like to thank Mr. Souvik, Mr. Prashanta, Mr. Monojit Bhattacharjee, Ms. Gauri and Mr. Prabin Bharali for handling all our office related matters with utmost sincerity. I would like to ac-

knowledge the support and assistance provided by the scientific officers of CSE department Mr. Nanu Alan Kachari, Mr. Bhriguraj Borah and Mr. Raktajit Pathak to all the PhD research scholars in the department. They have always strived hard to ensure that we do not face any issues related to Internet connectivity, computer peripherals, printer cartridges etc. At a personal level, I would like to thank them all for their troubleshooting assistance while addressing the networking and system related issues in the research scholars lab.

A special thanks must go to all security guards, janitors, housekeeping staffs, mess staffs, canteen staffs, food court staffs, hostel caretakers, wardens, doctors, medical staffs, and drivers of the institute for making our life and stay at IIT Guwahati comfortable and easy. These people form the lifelines of IIT Guwahati and are really the unsung heroes. When one stays at a residential campus for an extended period of time, our friends and colleagues become our family. We share our happiness, sorrows, griefs, festivals, special moments, birthdays with them. This acknowledgment would be incomplete without mentioning their names.

Among friends, I would first like to thank four of my seniors Shirshendu Das, Nilkanta Sahu, Shashi Shekhar Jha and Mayank Agarwal. Right from my initial days at IIT Guwahati they have always been there for me. Shirshendu Das and Nilkanta Sahu are like a father figure to me. Both of them were always there to help me out, whenever I found myself in troubles (academics and personal). Their guidance and counsellings helped me cope with stress and anxieties at various stages of my PhD career. Shashi Shekhar Jha and Mayank Agarwal were like my extended supervisors. They were always there whenever I needed any help regarding academic and technical issues. Additionally, I would like to thank Mayank Agarwal for helping me out with my journal reviews and for all his technical troubleshooting assistance, which unburdened and made my PhD journey a lot simpler. I will always cherish the Shillong and Kaziranga trips that I took with you all. I will also forever remember our innumerable weekend outings for foods and movies. Life at IIT Guwahati would not have been so memorable without the four of you.

Among my other peers at the CSE department, I would like to thank Shilpa Budhkar for all her life lessons and pep talk, which have broaden my perspective about life and made me a better person. I would also like to thank Hema K Yarnagula, Sukarn Agarwal, Dipika Deb and Nayantara Kotoky for all their love and support. I will always cherish the discussions we had during our evening tea sessions, which were always fun filled and refreshing after a long day's work at our cubicles. Amidst all the joys, sorrows, turmoil, hardships and frustrations during my PhD journey you all have always been there for me. Life at IIT Guwahati would not have been the same without you all. I would also like to thank my batch mates Durgesh Kumar and Khushboo Rani for their support and help during my PhD

course works. Satish Kumar and Sibaji Gaj also deserve acknowledgment for their support and advices regarding career prospectives at different stages of my PhD journey.

When one stays at a residential campus like IIT for an extended period of time, one ends up creating a big circle of friends comprising fellow research scholars and campus residents. This is true in my case too. My acknowledgment would be incomplete without mentioning their contributions toward successful completion of my PhD journey. I would like to thank my entire badminton fraternity, Shyam Trivedi, Anand Agrawal, Subrat Kumar Mallick, Pankaj Singh, Mukul Parmanand, Vinod Pandey, Siddesh Desai, Shatrughan Jaiswal, Rajendra Soni, Kishor Gajrani and Uttam Kumar Tarai from various departments and centers of IIT Guwahati. Playing badminton with you all has been one of the most enriching experiences of my life. We went on to win back to back Spardha tournaments together. I will always cherish those winning moments and countless match analysis that ensued those victories. I will also forever remember our cycling trips that we took on weekends, which were always refreshing and rejuvenating after tiring and exhausting weekdays academic activities. I would like to acknowledge Dr. Anamika Barua mam from the HSS department of IIT Guwahati for all her love and support. You are the linchpin that holds us all together. I will forever cherish the dinner parties that we had at your quarter. I must admit that I always looked forward to our weekend get-togethers at your place, which helped us get through some of our most difficult times at IIT Guwahati. I would also like to thank Swapnali Bora mam for her love and support. I will cherish the delicious food and occasional music sessions that we had at her place.

Among other notable mentions include Samten Doma, Debanjan, Mamata Di, Mrs. Amrita Bose Paul, Shounak Chakraborty, Lalatendu Behera, Awnish Kumar, Pradeep Kumar Biswal, Achyut Mani Tripathi, Deepak, Saptarshi, Piyoosh, Mohit, Rakesh Pandey, Swarup, Shuvendu Rana, Tushar Semwal, Sonia Sharma, Mahesh Patel and Akash Anil whom I need to acknowledge for being part of this wonderful journey of doctoral research. I would also like to thank my M.Tech supervisor Prof. D.Ghosh at National Institute of Technology (NIT) Durgapur, whose encouragement and guidance were instrumental in my decision to seek admission for PhD degree programme. I would also like to thank everyone who have helped me directly or indirectly in my PhD journey and whose names I did not mention here. I sincerely acknowledge all of your efforts too. I would like to acknowledge those anonymous users, bloggers, forum moderators whose answers to various queries have helped me when I got stuck. A big thanks also goes out to Google, Wikipedia and online cloud storage providers like Dropbox and MEGA who ensured that our data remained safe and available 24×7. Many thanks to my hostel warden, mess workers, cleaners, janitors, canteen and juice centers personnel of Brahmaputra hostel, which has been my second home for the last 6 years.

Without the adequate support of my family members, my doctoral journey could only have remained a dream. I would like to thank my sister Sheela Subba for her constant support and motivation right from schooling days to completion of my PhD journey. Knowing that you were there to take care of our parents, while I was away from home pursuing my PhD degree provided me with a sense of security and enabled me to concentrate on my research work. I would also like to thank my younger brother Sarad Subba who has always done my bidding and has been a perfect younger sibling one could ever have asked for. I would like to thank my uncle, Mr. Laydong Simik Lepcha, who always motivated me to strive for best things in life. Watching him succeed in his academic life inspired me to set similar goals for myself while growing up. I would also like to extend my sincere thanks to all my extended family members, who have been involved in my PhD journey directly or indirectly. Their constant love and support kept me going even during the most difficult times in my life.

The highest amount of acknowledgment goes out to my parents. Their belief in my abilities helped me succeed throughout my academic life. Their sacrifices ensured that no obstacles could hinder the pace of my journey. Thank you Mother (Mrs. Sangkit Subba) and Father (Mr. Suk Bahadur Subba) for being so kind, generous, supportive, caring and for all your love. This thesis is a culmination of your blessings!

Last, but not the least I would like to thank Almighty for his grace and blessings which enabled me to successfully complete my PhD journey.

July 3, 2018 Basant Subba

Abstract

The cost of cyber crime has seen exponential growth in recent years and is already at a phenomenal level. According to Forbes, the global cost of cyber crime is expected to reach $2 trillion by 2019. These figures indicate the huge threat posed by cyber criminals and underscores the severity of the risk associated with the malicious cyber activities. Intrusion Detection Systems (IDSs) have been proposed in the literature to address these security threats. IDS is basically a hardware device or a software application that monitors a host system or a computer network for sign of anomalous system processes and malicious network activities.

The thesis consists of three distinct contributions. As the first contribution, a novel game theory-based false alarm minimization scheme for signature based IDS is proposed. The proposed framework models the intrusion detection process as a two player non-cooperative game between the IDS and the attacker. It uses various network context information like, IDS's detection rate, criticality levels of the host machines, severity levels of network vulnerabilities, attacking and monitoring costs etc., to devise efficient IDS monitoring strategies based on the Nash Equilibrium (NE) of the game. The proposed framework is shown to filter out most of the false positive alarms generated by the signature based IDS and thereby significantly improve the IDS's accuracy, without adversely affecting its detection capabilities.

The second contribution of the thesis proposes a Bayesian game theory-based hybrid intrusion detection framework for resource constrained Mobile Ad-hoc Networks (MANETs). It uses a combination of simple threshold based rules and complex data mining based association rules to detect various type of attacks in MANETs. In addition, the proposed intrusion detection framework models the interaction between the IDS and the node being monitored as a two player non-cooperative Bayesian game. Such non-cooperative game theoretic modeling enables the MANET nodes operating the IDS to minimize their overall energy consumption by adopting probabilistic monitoring strategies based on the Bayesian Nash Equilibrium (BNE) of the game, without adversely affecting their detection rate. The framework is also shown to significantly reduce the volume of IDS traffic introduced into the network.

As a final contribution of the thesis, a novel clustering algorithm and a game theory-based multi-layered intrusion detection framework for Vehicular Ad-hoc Networks (VANETs) are

proposed. VANETs are characterized by high vehicular mobility and operate in a narrow bandwidth wireless radio spectrum. High vehicular mobility results in unstable vehicular clusters with intermittent network connectivity among vehicles. Therefore, introduction of high volume of intrusion detection related traffic can cause congestion in VANETs. The proposed clustering algorithm uses various vehicular information like vehicles' velocities, their direction of movements, real-time coordinates etc., to produce stable vehicular clusters, which enhances the overall stability of the vehicular network. On the other hand, the proposed IDS framework uses a combination of specification rules and a neural network based classifier module to detect various type of attacks in VANETs. Additionally, the proposed IDS framework models the intrusion detection process in VANET as a two player non-cooperative game between the IDS and the vehicle being monitored. This enables the IDS to devise efficient monitoring strategy based on the Nash Equilibrium of the game and thereby, significantly reduce the volume of IDS traffic in the vehicular network.

Contents

List of Figures

List of Tables

List of Symbols

Symbols	Description
μ	Mean
σ	Standard deviation
C_a	Cost of attacking
C_m	Cost of monitoring
p	attacking probability distribution
q	monitoring probability distribution
$U_A(p, q)$	Payoff utility of attacker
a	Detection rate of the vulnerability scanner
b	False alarm rate of the vulnerability scanner
$U_D(p, q)$	Payoff utility of defender
S_s	Sensible vulnerability set
N_A	Number of vulnerability sets in S_s
S_q	Quasi-sensible vulnerability set
S_D	Network's vulnerability set
S_A	Alarm set
C_L	Cluster leader node
R_i	Reputation value of node n_i
p_i	Maliciousness level of node n_i
Θ_i	Type of node n_i (normal or malicious)
α	Detection rate of IDS module
γ	False alarm rate of IDS module
w_k	Asset value of node n_k

p_o	Prior probability of node n_k being malicious
SB_{n_i}	Sampling budget of node n_i
E_{ids}	Energy required for operating IDS
Cst_i	Cost incurred by node n_i for performing monitoring operation
λ	Sampling budget weighing factor
E_{n_i}	Energy level of node n_i
P_i	payment received by node n_i for acting as cluster head
ϕ_j^i	Packet reception rate of node n_j from node n_i
ψ_i	Packet forwarding rate of node n_i
$D_{v_i}^{PCH}$	Distance between primary cluster head (PCH) and vehicle v_i
SCF_{v_i}	Social Choice Function (SCF) table of vehicle v_i
A_{v_t}	Actual velocity of vehicle v_t
F_{v_t}	Forecast velocity of vehicle v_t
Ck_i	Checker node n_i
R_{Ck_i}	Reputation of checker node n_i
$U_{v_j}^{v_i}$	Utility function value computed by vehicle v_i for vehicle v_j
Rv_{avg}^i	Average aggregated utility function value of vehicle v_i
Rwd_{v_i}	Reward obtained by vehicle v_i for acting as the cluster head
P_{v_i}	Payment received by vehicle v_i when it act as cluster head
PDR_{v_i}	Packet drop rate of vehicle v_i
Ver_{msg}	Verification message
δ	Average number of vehicles accepting and forwarding information from malicious vehicles
$Agg_{v_m}^{RSU^i}$	Aggregated reputation of vehicle v_m computed by the i^{th} RSU
$Glb_{v_m}^{RSU}$	Global aggregated reputation of vehicle v_m computed by all RSUs

List of Abbreviations

Terms	Abbreviations
IDS	Intrusion Detection System
LAN	Local Area Network
WAN	Wide Area Network
MAC	Media Access Control
GT	Game Theory
NE	Nash Equilibrium
SVM	Support Vector Machine
AODV	Ad-hoc On-Demand Distance Vector
MANET	Mobile Ad-hoc Network
WSN	Wireless Sensor Network
VANET	Vehicular Ad-hoc Network
VCG	Vickrey-Clarke-Groves
BNE	Bayesian Nash Equilibrium
RSSI	Radio Signal Strength Indicator
DoS	Denial of Service
PDR	Packet Delivery Ratio/Packet Drop Rate
PFR	Packet Forwarding Rate
RO	Routing Overhead
SB	Sampling Budget
PF	Power Factor
SCF	Social Choice Function
DL	Detection Level

FP	False Positive
FN	False Negative
GVT	Global Vector Table
BCV	Binary Correlation Vector
SVS	Sensible Vulnerability Set
DARPA	Defense Advanced Research Projects Agency
IDEVAL	Intrusion Detection Evaluation
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
FTP	File Transfer Protocol
SQL	Structured Query Language
RF	Risk Factor
IP	Internet Protocol
OS	Operating System
CVE	Common Vulnerabilities and Exposures
DR	Detection Rate
SUMO	Simulation of Urban Mobility
NS3	Network Simulator 3
TraCI	Traffic Control Interface
OSM	OpenStreetMap
FAR	False Alarm Rate
IVT	IDS Traffic Volume
ACMD	Average Cluster Membership Duration
DSRC	Dedicated Short Range Communication
WAVE	Wireless Access in Vehicular Environment
SCHs	Service Channels
CCH	Control Channel
WSM	WAVE Short Messages
HIDS	Heavyweight Intrusion Detection System

LIDS	Lightweight/Local Intrusion Detection System
CIDS	Cluster Intrusion Detection System
GDS	Global Detection System
V2V	Vehicle to Vehicle
V2I	Vehicle to Infrastructure
RSU	Road Side Unit
SCF	Social Choice Function
CH	Cluster Head
CM	Cluster Member
CG	Cluster Gateway
CJR	Cluster Join Request
MSE	Mean Squared Error
PCH	Primary Cluster Head
SCH	Secondary Cluster Head
LTE	Long Term Evolution
FAR	False Alarm Rate

"This riparian stuff is not rocket science . . . it's much more complex than that"

Steve Nelle

Retired NRCS Wildlife Biologist San Angelo, Texas

1

Introduction

1.1 Network Security

Interconnection of various heterogeneous networks through Internet, Local Area Networks (LANs), Wide Area Networks (WANs) etc., has enabled a seamless flow of information and communication between them. Such an integration between networks has led to increased productivity and connectivity among various organizations and corporations. However, on the flip side, it has also introduced many security issues and vulnerabilities into these interconnected networks. The distributed and heterogeneous nature of the contemporary networks, coupled with the complexity of their underlying communication environment have made the network control and security task much more challenging than ever before. Most of the corporates and organizations are under constant threats from cyber criminals and hackers, who work either individually or as part of a larger collective groups in pursuit of financial rewards. Their activities take many forms, and include theft of financial credentials, credit card fraud, corporate espionage, identity theft, data theft, data ransoming activities etc. Cyber crimes have seen a huge surge in recent times. In 2015, a group called the "The Impact Team" stole and leaked more than 25 gigabytes of user data from a commercial website called the Ashley Madison that promoted extramarital affairs in what came to be known as the "Ashley Madison data breach". The "WannaCry" ransomware attack on May 2017 targeted and infected computers running the Microsoft Windows operating system by encrypting data and demanding ransom payments in the Bitcoin cryptocurrency. This attack is believed to have infected more than 230,000 computers

in over 150 countries. The 2016 US presidential election saw the Democrat candidate and the White House front runner Hillary Clinton lose out to her Republican candidate Donald Trump, after being embroiled in an email controversy believed to have been orchestrated by the Russian hackers. In July 2017, the hackers compromised the HBO's network systems and threatened to leak the final two episodes of its popular and widely viewed TV series 'Game of Thrones', unless the HBO agrees to pay a multi million dollar ransom. All these incidents show that cyber crimes and data thefts pose real threats across wide range of spectrum like, social network breach, personal data theft, corporate espionage, entertainment production house ransoming etc.

The cost of cyber crime has seen an exponential growth in recent years and is already at a phenomenal level. According to Forbes, the global cost of cyber crime is expected to reach $2 trillion by 2019. This figure indicates the huge threat posed by cyber criminals, and underscores the severity of the risk associated with malicious cyber activities. The network security problems are much more conspicuous in wireless networks like Mobile Ad-hoc Networks (MANETs), Wireless Sensor Networks (WSNs) etc., which are prone to various type of attacks like eavesdropping, jamming, sniffing, data packet alteration, packet dropping etc. In addition, the inherent computational and resource constraints of these wireless networks further exacerbate their security issues. Nodes in these wireless networks are usually battery powered with limited processing capabilities and memory storages. Moreover, they mostly operate in a narrow and crowded 2.4 GHz ISM band. All these limitations make the wireless networks extremely vulnerable to wide range of attacks. The attackers can easily exploit the vulnerabilities in these networks, if adequate security mechanisms are not put in place to secure them. Protecting these vulnerable networks require understanding of vulnerabilities that exist in every layer of these networks.

A network attack can be defined as any set of malicious actions that attempt to compromise the availability, integrity and confidentiality of the network. Network attacks can originate either from external or internal sources. External attacks are launched by entities/attackers that do not have legitimate permission to access the network. On the other hand, internal attacks are carried out by the legitimate but malicious users of the network, who have full access to all of the network's resources. Internal attackers are more difficult to detect than the external attackers, as the former have full access and knowledge about the security components put in place to detect the network intrusions. This enables them to easily circumvent preventive security measures like firewall, proxy servers, user authentication etc., as these security measures only monitor the incoming data traffic from the

external networks. The growing internal threat, mobile workforce, deployment of critical servers on the network, and more attacks coming in on common ports have exploited flaws in the preventive centric security solutions. Therefore, an additional complementary security mechanism in the form of Intrusion Detection System (IDS) is required for providing a comprehensive network security and detecting insider attacks. IDS technologies have seen wide scale adoption across many corporates and government organizations owing to their ability to detect internal attacks and external intrusions that have permeated through the first layer of network's defense. Based on their targeted applications and severity levels, network attacks can be categorized into various classes, some of which are enumerated below:

- *Footprint and enumeration*: This is the first step adopted by an attacker before launching a full scale network attack. During this attack, the attacker performs a reconnaissance task to gather various information about the network like, un-patched applications, operating systems, active open ports, protocols etc.
- *Unauthorized network access*: In this type of attack, the attacker gains an unauthorized access to the network and then modifies, deletes or introduces false data into the network.
- *Sniffing and eavesdropping*: This attack allows the attacker to monitor or listen to the network traffic in a promiscuous mode and intercept all the transmission data to suit its need.
- *Identity spoofing*: Each host in the network is identified by a unique IP and MAC address pair. In this attack, the attacker impersonates to be someone else by creating spoofed data packets containing the IP and MAC addresses of other genuine hosts in the network.
- *Denial of Service*: By executing this attack, the attacker prevents the legitimate users from accessing the network resources. The attacker executes this attack by using multiple malicious nodes to flood the network with request messages for network's resources and thereby making them unavailable for the genuine users of the network.
- *Black hole attack*: In this attack, the malicious node (attacker) claims to have the shortest path to the destination node. However, upon receiving the data packets, it drops all the data packets instead of forwarding them to next hop/destination node, thereby leading to decreased network throughput.

Intrusion and Intrusion Detection System (IDS): Intrusion is defined as any set of actions that attempt to compromise the availability, integrity and confidentiality of the system or network resources. An IDS is a hardware device or a software application that monitors the system processes or network traffic for sign of malicious activities and policy violations. When any intrusive activities are detected, the IDS informs the administrator by raising alerts and generating log reports of the intrusions. The administrator can then take appropriate counter measures to contain the intrusions before any significant damage is inflicted to the system/network. IDS can also be configured to take elementary defensive measures like dropping malicious packets, blocking malicious hosts, resetting connection to the port on which the attack was detected etc. Although IDSs were initially deployed in "detection-only" mode, security practitioners grew increasingly confident in the their ability to accurately detect attacks, and "blocking mode" was turned on, which led to emergence of the Intrusion Prevention System (IPS). The IDS/IPS market accounted for $1.43 billion in revenue at the end of 2015, and this number is expected to grow to more than $1.80 billion by the end of 2020.

Owing to the huge security benefits and unparalleled protections provided by them, IDSs have widely been adopted by many organizations and cooperations as an integral part of their overall network security. IBM's Internet Security Systems (ISS), Cisco's Sourcefire, McAfee's Stonesoft and Extreme Network's Enterasys are some of the well known commercially available IDS/IPS solutions. However, there is a caveat associated with the usage of IDS. With the increased usage of encryption over the years, IDSs have lost their significant potential, as the contents of each packet are now often obfuscated. As a result, modern IDS relies on data such as length of connection, connection characteristics, protocol type etc., to provide the majority of the decision features. Nevertheless, in resources constrained wireless networks like MANETs and WSNs, encryptions are often cost prohibitive and usually avoided. As such, IDSs are able to effectively analyze the non encrypted data packets in these networks and provide a comprehensive network security.

Based on their mode of operations, IDS can be classified as either Host based IDS (HIDS) or Network based IDS (NIDS). HIDS monitors a standalone computer system by analyzing its system call traces and audit log files to detect malicious system processes. On the other hand, NIDS monitors the network traffic flow in a promiscuous mode for signs of malicious network activities. This thesis mainly addresses the issues in NIDS and therefore, unless otherwise specified explicitly, the term IDS in the thesis refers to NIDS.

Host based intrusion detection system (HIDS): HIDSs are used to determine if the system has been compromised and warn the system administrator when such an event takes place. Based on their input sources, HIDs can be divided into following three types:

- *File system monitors* : HIDS implementations that use filesystem monitoring compare files on the host machine with previously gathered information about these files, such as permissions, sizes, ownerships, inode numbers, checksums, number of links, last modification date etc. Therefore, when an attacker gains access to the system and changes the system files, they will be detected by the HIDS. Advanced Intrusion Detection Environment (AIDE) [1] and Mtree [2] are two well known file system monitoring based HIDS. Although HIDSs based on file system monitoring can detect break-in on systems, there are certain drawbacks associated with them. The contents of directories like **/tmp** tend to change a lot. Therefore, these directories may not be incorporated in a filesystem check. This gives an attacker a good place to store their files. Additionally, file system monitors generally do not work in realtime. This enables the attacker to restore things and cover up its track. For example, the attacker can modify the last access time (*atime*) and last modify time (*mtime*) of a file after making changes to it. Moreover, the technique of logging every system call made by every running process imposes a significant overhead on the host's resources.

- *Logfile analyzers*: HIDSs based on this approach use system log files to determine the potential intrusion attempts made by the attacker to compromise the system and warn the administrator when such intrusion attempts are detected. Swatch [3] and SEC [4] are two log file analyzer based HIDS. They use regular expressions for pattern matching analysis and can be configured to mail an alert to a predefined e-mail address and execute predefined preventive commands on a match. They also have the ability to compress multiple events of the same type into one and thereby, effectively throttle the volume of alerts generated. Most modern operating systems use some form of logging software. As this software is already running, the overhead in data collection is low and much of the data cleansing is inherently performed by the logging process. However, the drawback of the logfile analyzer based HIDS is that it only takes an action when a predefined number of matches have taken place during a given time frame. For example, it could be set to warn an administrator when five failed login attempts take place in one minute. However, if an attacker carrying out a brute force password attack limits his login attempts to only four per minute, the

attack will go undetected. Moreover, logfile analyzer based IDS is less accurate than a comparable system call based IDS due to several issues associated with the logfile's entry data format.

- *Connection analyzers*: Connection analyzer based HIDSs monitor incoming network connections to a system from the external sources. This enables them to detect unauthorized connections to TCP ports and report this in the system log files. They are also capable of detecting SYN, FIN and XMAS type port scan attacks. PortSentry [5] and Scanlogd [6] are two well known connection analyzer based HIDS. Although, connection analyzer based HIDSs are good at detecting unauthorized TCP port scans, they require too much management overhead compared to the actual advantages they bring to the administrator. Another issue with this approach is that a friendly host might accidentally end up getting blocked because they tried connecting to a wrong port, or to services that no longer exist on the system. Additionally, this approach is prone to Denial of Service (DoS) attack, wherein the attacker floods the connection analyzer with deluge of connection requests and renders it non-responsive. The attacker can execute the DoS attack by either filling up the log files with a large number of probes or by spoofing other hosts' addresses and have them blocked.

Network based intrusion detection system (NIDS): Unlike HIDS, which monitors individual hosts, NIDS monitors the data traffic of the entire network (or segment of network with multiple hosts). It operates in a promiscuous mode by wiretapping the network traffic stream, as the data packets traverse along the network segment. An overview of NIDS deployment layout is shown in Fig. 1.1. NIDS detects attacks and anomalous network behaviors by inspecting the header information and payload contents of the data packets. Most of the NIDS modules come equipped with attack signature rules that are used for detecting network attacks. This class of NIDS are referred to as signature or misuse based NIDS. Signature based NIDS correlates the header and payload information of the data packets being monitored against the signature rules to identify the malicious data traffic. On the other hand, anomaly based NIDS uses the normal behavior of the network traffic to build the baseline profile of the network. The real time network data traffic is then correlated against the learned baseline profile to identify the malicious data traffic. The third class of NIDS called the event based NIDS detects network intrusions by keeping track of the sequence of data packet events. Event based IDSs basically act as state estimators and observe the sequence of data packet events in the network to decide whether the observed

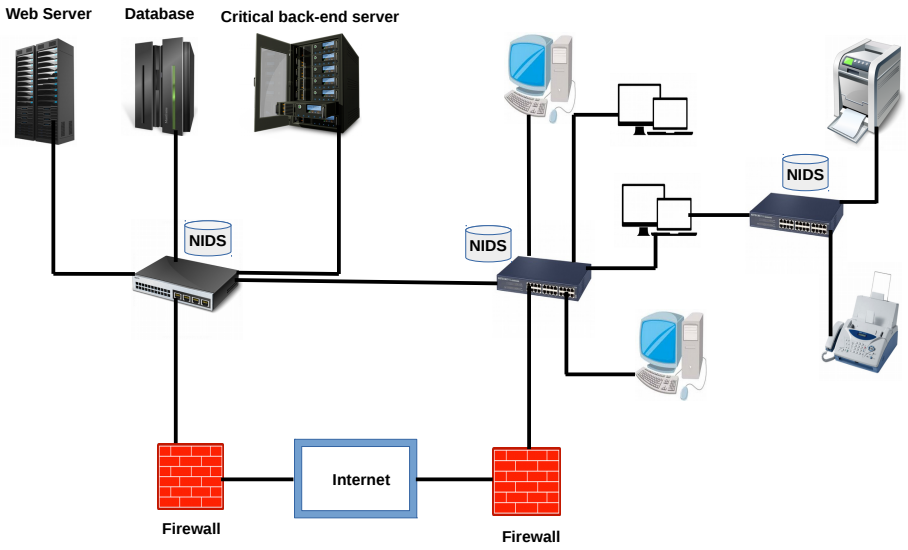

Figure 1.1: NIDS deployment layout

progression of the states corresponds to a normal or attack scenario. They work under the principle that the sequence of data packet events under attack is different from that under normal network conditions.

The work on this thesis primarily focuses on improving the performance of the NIDS. Therefore, unless otherwise specified explicitly, the term IDS in the thesis refers to NIDS. In the subsequent paragraphs, we provide a brief overview about the signature, anomaly and event based IDSs.

Signature based IDSs: These detection systems use a set of well known attack signatures stored in their rule database to detect malicious network activities. The attack signatures are sourced from various publicly available vulnerability databases like BugTraq [7], CVE [8], Nmap [9], Nessus [10] etc. When the data traffic being monitored matches with on or more of the attack signatures, an alert is raised by the IDS to inform the network administrator about the attack. Snort [11], EMERALD [12] and BRO [13] are some of the well known signature based IDSs. Snort's attack signature for detecting the "*land*" attack, which is one form of DoS attack is given below:

alert ip any any → any any (msg : "BAD TRAFFIC same SRC/DST" ; sameip; reference : cve, CVE - 1999 - 0016; reference : url, www.cert.org/advisories/CA - 1997 - 28.html; classtype : bad - unknown; sid : 527; rev : 3;)

This signature raises an alert, whenever an Internet Protocol (IP) packet having the same source and destination address is observed in the network traffic.

Although, signature based detection systems have high detection rate and accuracy against known attacks, they have their share of drawbacks. Since, they use the attack signatures stored in their rule database to detect malicious data traffic, they are incapable of detecting new zero day attacks, for which the signatures have not yet been developed. Therefore, they require frequent updates to their rule database to remain effective. Moreover, since the attack signatures have to be developed manually by human experts, they are time consuming and prone to error. The time between the vulnerability announcement and the development of the corresponding attack signature creates a window of opportunity for malicious attackers to breach and exploit the network. In addition, since most of the attack signatures use pattern matching to identify malicious data packets, the attacker can circumvent and avoid detection by making minor modifications to the syntax of the attack data. Signature based detection systems also produce a large number of false positive alarms, when deployed with default settings, without considering the context of the underlying network. For example, Windows based attacks are ineffective against a Linux based system. However, a signature based IDS running on a Linux system still generates alarms for such attacks, if there are corresponding attack signatures for these attacks in its rule database. High volume of false alarms put a strain on the network by potentially blocking the legitimate traffic, which is completely unacceptable to an organization that derives significant revenues from on-line activities. High false alarm rate also arouses a level of distrust regarding the effectiveness of the signature based detection systems to the end users and makes the network administrator wary of allowing an ineffective technology to make decisions regarding which traffic to be allowed to progress through the network. However, notwithstanding all these drawbacks, most of the commercially available IDSs are predominantly signature based. Therefore, from the above discussions, it can be concluded that effective signature update mechanism and false alarm minimization are two major aspects of signature based detection systems that need to be improved for enhancing their effectiveness and enabling their wide-scale adoption.

Anomaly based IDSs: These detection systems create a normal baseline profile of the

network during their training phase and then applies it to the network data traffic at real time to detect anomalous network activities. The network's normal baseline profile is developed using various system and network parameters like CPU usage, memory utilization, disk activity, processor usage, network bandwidth usage, number of users logged in, count of failed logins, amount of mails sent and received etc. Unlike signature based detection systems, anomaly based detection systems do not require any predefined attack signatures or syntax knowledge to detect attacks. As such they are capable of detecting previously unseen and zero day attacks. Anomaly based detection systems use various methods based on data-mining[14][15][16], statistics[17][18][19] and machine learning[20][21][22] to develop the network's baseline profile. They have been shown to achieve high accuracy and detection rate across wide range of both known and unknown attacks, which make them highly desirable for deployment in providing a comprehensive network security. However, there are certain drawbacks associated with them. They require a pure attack free training data to develop the normal baseline profile of the network. However, obtaining such pure training data is difficult. They also incur a significant computational overhead in training and generating the baseline profile of the network. Moreover, anomaly based detection systems require periodic retraining to develop new baseline profile of the network to prevent them from correlating the real time data traffic on stale baseline profile models, which can significantly increase their false alarm rate.

Event based IDSs: There are certain type of attacks for which the anomaly detection model or attack signatures cannot be developed, since they do not cause any change in network's syntax under the normal and attack scenarios. Address Resolution Protocol (ARP) spoofing attack in a Local Area Network (LAN) is an example of such an attack [23]. ARP spoofing attack does not cause any change in the pattern of the network's communication. However, upon successful execution of the ARP spoofing attack, all the data packets addressed to the host with the given IP address will instead be delivered to another host with a spoofed MAC address. An event based IDS can be used to detect such attack by keeping track of sequence of data packet events in the network. Event based IDS basically acts as a state estimator and observes the sequence of data packet events in the network to decide whether the observed progression of the states corresponds to a normal or attack scenario.

Drawbacks of signature, anomaly and event based IDSs: Although all the three classes of IDSs perform well against wide range of network attacks, there are many drawbacks associated with them, which are summarized as below:

- Signature based IDSs are incapable of detecting zero day attacks, whose signatures are not present in their rule databases. In order to detect these new attacks, their signature databases need to be updated periodically.
- Since most of the attack signatures are developed manually, they are prone to errors and mis-configurations. They also produce a large number of False Positive (FP) alarms when deployed with default settings, without considering the context of the underlying network. FPs are generated when benign network traffic is classified as attacks by the IDS.
- Anomaly based IDSs require pure training data that is free from malicious data traffic to develop the normal baseline profile of the network. However, obtaining such pure attack free data is difficult. Additionally, training anomaly detection systems incurs a significant amount of computational overhead.
- Anomaly based IDSs need to be retrained periodically to incorporate the changes in the underlying network parameters for staying effective. Without retraining, the IDS assessment will be based on stale training data, which results in their poor performance.
- The major drawback of the event based IDSs is their scalability. The number of states to be kept track by the event based IDS can grow exponentially with the increase in network's size being monitored. This introduces a tremendous amount of computational overhead and limits their overall effectiveness in resource constrained networks.
- Event based IDSs also require active techniques like sending out probe packets to identify the differences in sequencing of data packets under normal and attack conditions, which violate the standard operations of the protocol under consideration.

In addition to the drawbacks listed above, both the anomaly and the signature based IDSs require a significant amount of energy and computational resources to perform intrusion detection operations. However, many wireless networks like MANETs, WSNs etc., are characterized by energy and resource constrained nodes. Therefore, IDS frameworks that require persistent monitoring operation can drain out the energy level of nodes in such networks. This results in premature death of the nodes operating the IDS, which effectively shortens the overall lifetime of these wireless networks. Moreover, wireless networks operate in a narrow bandwidth radio spectrum. Therefore, high volume of IDS traffic can easily

cause congestion and prevent the flow of normal data traffic in such networks. Hence, IDS frameworks designed for these wireless networks need to be tailored to meet their computational, energy and bandwidth constraints.

Game theory [24] can be used as a mathematical tool for obtaining the best trade-off between energy consumption and IDS performance in the resource constrained wireless networks. Game theory deals with the study of deriving mathematical models of cooperation and conflict between intelligent decision-makers in cooperative and non-cooperative situations, respectively. Cooperative games model the competitive interaction between a given set of players (coalition) with similar set of objectives. They aim to maximize the overall well being of the coalition, while ensuring fairness for each individual member of the coalition. Players in cooperative games form coalition due to the possibility of enforcement of cooperative behavior by an external entity, which may not be in the best interest of the players. On the other hand, non-cooperative games model the interaction between two or more players with conflicting and contradictory set of objectives. They can be used to model the intrusion detection process as a competitive game between two players (attacker and IDS). Non-cooperative games are characterized by the solution concept called the Nash Equilibrium (NE). The NE of a non-cooperative game corresponds to the strategy combinations of the players, such that no player has any profitable incentive to unilaterally deviate from the chosen NE strategy, while the other players keep their strategies fixed. Modeling the intrusion detection process as a non-cooperative game enables the IDS to adopt a probabilistic monitoring strategy based on the NE of the game. This significantly reduces the volume of IDS traffic introduced into the network and also minimizes the energy consumption required for operating the IDS, without degrading its overall detection capabilities.

Signature based IDSs are prone to high false alarm rate. We have shown for the first time the application of game theory in minimizing the volume of false positive alarms generated by the signature based IDSs. In the proposed game theory-based false alarm minimization scheme, multiple vulnerability scanners are used to scan the network for identifying all possible vulnerabilities and create a *Threat profile* of the network. The network's *Threat profile* comprises multiple vulnerability sets, with each set consisting of one or more network vulnerabilities found during the scan. Each vulnerability set is assigned a unique criticality weight based on the severity of the vulnerabilities contained in it. A game theoretic procedure is then used to create a *Sensible Vulnerability Set* (SVS) of the network. The SVS comprises a subset of high criticality weight vulnerability sets from the network's *Threat*

profile. Alarms generated by the signature based IDS are initially correlated with vulnerabilities in the *Threat profile* to identify the potential true positive alarms. These alarms are then correlated with vulnerabilities in the SVS to determine the final TP alarms. This two phase correlation procedure filters out the false positive alarms generated due to unsuccessful attack attempts. It also filters out the low priority alarms, which do not require immediate intervention of the network administrator. As such, the proposed game theory-based false alarm minimization scheme significantly reduces the volume of false positive alarms generated by the signature based IDS, without degrading its overall detection rate.

In the sub-sequent section, we provide a brief overview of various game theory-based IDS frameworks proposed in the literature followed by discussions on the drawbacks and issues in these frameworks, which provides the motivation for the work carried out in this thesis.

1.2 Game Theory preliminaries

Game theory has extensively been used in wide areas of research like, economics, political science, computer science, psychology, biology etc., to study the events of conflict and cooperation between two or more rational decision makers (players) with common or contradicting set of objectives. It aims to formulate logical actions that the players should adopt to obtain the best possible outcomes for themselves when faced with the choice of series of strategies. A strategy for a player is a complete plan of actions in all possible situations throughout the game. If the player's strategy specifies to take an unique action in a given situation then it is called a pure strategy. On the other hand, if the strategy specifies a probability distribution for all possible actions in a given situation then the strategy is referred to as a mixed strategy. In general, games are characterized by three parameters namely, set of players, collection of strategies for each individual player and the players' payoff utilities corresponding to their chosen strategies. Interestingly, the primary objective of game theory is not to device an absolute desired outcome but rather to decide an optimal strategy that achieves better outcome compared to other strategies.

Game theory can broadly be classified into two main categories namely, cooperative and non-cooperative games. Cooperative games are used to model the competitive interaction between group of players (coalition) with a similar set of objective functions. Cooperative games consider utilities of all the players with the goal of maximizing the entire coalition's pay-off, while ensuring fairness for each individual member of the coalition. The central notion in the cooperative game theory is the concept of the *core*. The *core* is a set of payoff

allocations, which guarantees that no group of players have any incentive to leave their current coalition to form another coalition. *Shapley value* is another important solution concept in cooperative game theory, which specifies the marginal contribution of each individual player in the overall well-being of the coalition.

On the other hand, non-cooperative games are used to model the interaction between two or more competing players with conflicting set of objectives. Non-cooperative games are characterized by self-enforcing alliances among players due to absence of external mechanism to enforce cooperative behaviors among players. Non-cooperative games are competitive in nature with each player trying to maximize its payoff utility by choosing its best strategy, without considering the effect of its choices on the overall well being of other players of the game. They can further be classified as complete and incomplete information games. In a complete information non-cooperative game, each player has a complete knowledge about the utility functions, payoffs and strategies of all the other players in the game but the players may not know all the moves made by the other players. On the other hand, incomplete information non-cooperative games are those in which at least one of the player is unaware about the utility functions, possible payoffs and strategies of at least one other player of the game. In such incomplete information games, some of the players possess private information (for example their types), which are unknown to other players of the game. Such games are modeled as Bayesian games, wherein the players have prior associated belief values about the nature of other players. The belief values of the players are then updated over a period of time using the Bayes rule. Non-cooperative games are characterized by the solution concept called the Nash Equilibrium (NE), which corresponds to the strategy combination of the players, such that no player would prefer to change its strategy, given that the other players adhere to their prescribed chosen strategies, as doing so would result in a lower payoff for the deviating players.

1.3 Game Theory based IDSs and their issues

Game theory has extensively been used in the literature for addressing various IDS related issues like minimizing the energy consumption required for operating the IDS, reducing the volume of IDS traffic, minimizing the IDS's computational overhead etc. Game theory-based IDS frameworks that model the cooperation and selfishness of nodes in an Ad-hoc network are proposed in [25] [26]. In these frameworks, each node decides whether to forward or drop data packets based on the trade-offs involved in the cost (energy consumption) and

the benefit (network throughput) in collaborating with other nodes in the network. They enforce cooperation mechanism, which ensures that selfish nodes that do not abide by the network rules receive low throughputs. A Bayesian game theory-based IDS framework for analyzing the interaction between the malicious node (attacker) and the IDS (defender) in wireless Ad-hoc networks is proposed in [27]. The framework uses a Bayesian hybrid detection approach, wherein a less powerful lightweight module is used to ascertain the type (normal or malicious) of the nodes being monitored, and a more powerful heavyweight module based on Support Vector Machine (SVM) acts as the last line of defense against various type of attacks. A game theory-based scheme for economic deployment of IDS agents in a wireless Ad-hoc network is proposed in [28]. The scheme models the interaction between the attacker and the intrusion detection agent within a non-cooperative game theoretic settings and derives the security risk value of the network using a mixed strategy NE solution concept. A game theoretic IDS framework that models the interaction between the service provider and the attacker as a zero-sum intrusion detection game is proposed in [29]. In this framework, the service provider tries to maximize its payoff by increasing its probability of successful detection of the attacker, while the attacker tries to minimize its probability of being detected by the IDS. The optimal solution for both the player in such a situation is to play the min-max strategy of the game. Game theoretic intrusion detection frameworks proposed for Wireless Sensor Networks (WSNs) in [30] [31] [32] model the intrusion detection process as complete information non-cooperative games between the malicious senor nodes and the IDS. These frameworks enable the sensor nodes operating the IDS to adopt probabilistic monitoring strategies based on the mixed strategy NE of the non-cooperative games. A game theory-based efficient lightweight intrusion detection and prevention schema for Vehicular Ad-hoc Networks (VANETs) is proposed in [33]. The scheme accurately predicts the future malicious behavior of an attacker and categorizes it into an appropriate list based on its past observed behaviors. It models the attack-defense problem in the vehicular network as a Bayesian game formulation between the attacker and the Road Side Unit (RSU) and makes prediction about the future states of suspected malicious vehicles using the Bayesian Nash Equilibrium (BNE). A cooperative game theory-based strategy to address the problem of greediness in IEEE 802.11 CSMA/CA protocol in VANETs is proposed in [34]. This strategy is immune to frequent interpretation errors and enforces the selfish nodes to cooperate under the threat of retaliation. It is able to maximize the payoff of the cooperative nodes by 20% compared to the classical strategy [35] and by 9% compared to the reputation based strategy [36]. It also minimizes the throughput of

selfish nodes by 76% compared with the generous strategy [37].

Although, game theory-based IDS schemes proposed in the literature address many of the IDS related issues like, minimization of IDS traffic volume, reduction of energy consumption required for operating IDS etc., there are several drawbacks associated with them. Most of the non-cooperative game theory-based IDS frameworks model the intrusion detection process as a complete information game, wherein every node in the network is assumed to have a complete information about all the network parameters. However, such assumptions are not valid in real networks, wherein nodes only have partial information about different network parameters. Moreover, many of these IDS frameworks model the intrusion detection problem as a static game, wherein the strategies of the players (malicious nodes and IDS) are assumed to be static and fixed. However, such static game theoretic modeling results in poor IDS performance, since the players' strategies might vary dynamically depending on the context of the underlying network's parameters. Additionally, some of the game theory-based IDS frameworks are geared towards detection of specific class of attacks and cannot be generalized for detection of other type of attacks. On the other hand, most of the cooperative game theory-based IDS frameworks proposed in the literature use *Core* and *Shapley values* as their main solution concepts. However, evaluating these solution concepts are computation intensive in large scale networks. In this thesis, we aim to address these issues in the existing game theory-based IDS frameworks. In the subsequent sub-sections of the chapter, we provide brief overviews about various contributions of the thesis.

1.4 Game theory-based false alarm minimization scheme for signature based IDS

Signature based IDSs are prone to high false alarm rate with sometimes more than 90% of the alerts being generated by these IDSs turning out to be false positives. Therefore, false alarm minimization of signature based IDSs is an important issue that needs to be addressed for enabling their wider acceptance among the networking community. Towards this end, a novel game theory-based false alarm minimization scheme for signature based IDS is proposed as the first contribution of the thesis. The proposed scheme uses multiple vulnerability scanners namely, Nessus [10], Nmap [9] and Common Vulnerability Database (CVE) [8] to scan the network for all possible vulnerabilities and create a vulnerability *Threat profile* of the network. The network's *Threat profile* comprises multiple vulnerability sets, with each set containing one or more network vulnerabilities found during the network scan. Each vulnerability set is assigned a unique criticality weight based on the

severity of the vulnerabilities contained in that set. The interaction between the attacker and the defender (IDS) is then formulated as a non-cooperative game between two competing players. Various attacking and monitoring strategies of the attacker and the defender are examined using different network parameters like attacking and monitoring costs, false alarm rate of IDS, detection rate of IDS etc. These strategies are then used to evaluate the Nash Equilibrium of the game and build the *Sensible Vulnerability Set (SVS)* of the network. The network's *SVS* is a subset of its vulnerability *Threat profile* comprising high criticality weight vulnerability sets. IDS alarms that pass the *Threat profile's* correlation test are eventually correlated with the vulnerabilities in the *SVS* to determine the final TP alarms. This two phase correlation procedure filters out most of the FP alarms and low priority alarms, which do not require immediate intervention of the network administrator. As a result, the proposed framework significantly reduces the false alarm rate of the signature based IDS, without degrading its overall detection rate.

1.5 A game theory-based hybrid intrusion detection framework for MANET

MANETs are characterized by resource and energy constrained battery powered nodes. Moreover, they operate in a narrow bandwidth wireless radio spectrum. As such, a high volume of IDS traffic can cause congestion and prevent the flow of normal data traffic in MANETs. Therefore, in addition to possessing high accuracy and detection rate, any IDS framework proposed for MANETs must also be energy efficient and must not introduce a significant volume of IDS traffic into the network. To address these issues, a novel Bayesian game theory-based intrusion detection framework for MANET is proposed as a second contribution of the thesis. The proposed framework uses a combination of threshold based rules and a data mining based association rules to detect wide range of attacks in MANET. It models the interaction between the attacker (malicious node) and the defender (node operating IDS) as a two player multi-stage, non-cooperative and incomplete information Bayesian game. The Bayesian representation model allows the node operating the IDS to adopt the most efficient monitoring strategy in an incomplete information game settings by examining the maliciousness history profile of the node being monitored and by evaluating the Bayesian Nash Equilibrium of the game. It allows the IDS to adopt a probabilistic monitoring strategy, which significantly reduces the energy consumption required for operating the IDS and also minimizes the volume of IDS traffic introduced into the network, without degrading the overall performance of the IDS.

1.6 A game theory-based multi layered IDS framework for VANET

Vehicular Ad-hoc Network (VANET) uses clustering as a predominant transmission strategy, wherein multiple vehicles group together to form a cluster. A Cluster Head (CH) is elected for each cluster, which receives data packets from its cluster members and relays them outside (and vice versa). The performance of any cluster based VANET intrusion detection framework largely depends upon the stability of its clusters. Unfortunately, in a dynamic environment like VANET, cluster reconfigurations and CH changes are unavoidable, which makes the clustering process in VANET a difficult task. Nevertheless, there are certain unique characteristics of VANET, which can be exploited to develop an efficient clustering algorithm. The vehicular movement in VANET is topologically constrained by road conditions, users' driving pattern and roadside equipment such as signs and traffic lights, which leads to a predictable traffic patterns, with vehicles often moving in naturally formed groups. Moreover, vehicles in VANET are equipped with electronic license plates and GPS device, which enable them to identify other vehicles and determine their coordinates in real time. Using all these vehicular data, along with the position and velocity information of the vehicles, a stable clustering algorithm for VANET can be developed.

The third contribution of this thesis proposes a novel clustering algorithm and a multi layered game theory-based intrusion detection framework for VANET. The proposed clustering algorithm generates stable vehicular clusters and reduces the overhead involved in cluster formation process. On the other hand, the proposed intrusion detection framework uses a set of specification rules and a lightweight neural network based classifier module to detect various type of attacks in vehicular networks. The framework also employs a novel Cluster Head (CH) election algorithm that uses an incentive structure based on the VCG mechanism [38] to motivate vehicles to actively participate in the CH election process by offering them payment in the form of enhanced reputation gain for taking up the role of CH. The framework models the interaction between the IDS and the vehicle being monitored as a two player non-cooperative game. Such game theoretic modeling enables the IDS to adopt a probabilistic monitoring strategy based on the NE of the game, which minimizes the overall volume of intrusion detection related traffic in a bandwidth constrained vehicular network, without compromising the performance of the intrusion detection framework.

1.7 Organization of Thesis

The thesis work has been documented in the following six chapters:

Chapter 2 provides an overview of the existing IDS frameworks proposed in the literature for different networks. This chapter also discusses various limitations of the existing IDS frameworks, which provides us with the motivation for the work carried out in this thesis.

Chapter 3 provides a game theory-based false alarm minimization scheme for signature based IDS.

Chapter 4 presents a Bayesian game theory-based multi-layered intrusion detection framework for MANETs.

Chapter 5 provides a novel clustering algorithm and a game theory-based intrusion detection framework for VANETs.

Chapter 6 summarizes the work described in this thesis and provides direction for future work.

"This riparian stuff is not rocket science . . . it's much more complex than that"

Steve Nelle

Retired NRCS Wildlife Biologist San Angelo, Texas

2

Background and literature Survey

2.1 Introduction

Intrusion Detection Systems (IDSs) have emerged as invaluable and indispensable asset for network security in recent years. IDSs monitor and possibly prevent network intrusions by monitoring the network's data traffic for sign of anomaly. Based on their mode of operations and detection capabilities, IDSs can be categorized into three main classes, namely signature based, event based and anomaly based IDSs. Signature based IDSs [11] [13] [12] (also referred to as misuse based IDSs) correlate the header and payload information of the data traffic being monitored with predefined set of attack signatures to detect network intrusions. However, they can only detect those known attacks for which the corresponding attack signatures are present in the IDSs' rule databases. On the other hand, event based IDSs [39] [40] [41] can detect those known attacks for which the attack signatures cannot be generated. These known attacks do not cause any change in the syntax of the network traffic under normal and compromised situations. However, they change the intended behavior of network communication. Event based IDSs use the difference in sequence of events occurring under normal and compromised conditions to detect these known attacks. The third class of IDS are the anomaly based IDSs. They are capable of detecting both known and unknown attacks [14] [15] [17] [42] [20]. Anomaly based IDSs initially develop the normal baseline profiles of networks during their training phase and then apply them to the real time network traffic to identify anomalous data traffic. The main advantage of anomaly based IDSs over their signature and event based counterparts

is their ability to detect previously unseen zero day attacks, since they do not require any pre-existing attack signatures and attack syntax knowledge to detect network intrusions.

Although, all the three classes of IDSs (signature, event and anomaly based) are capable of detecting wide range of network attacks, there are several drawbacks associated with them. Signature based IDSs can only detect those attacks, which have been encountered previously and whose syntaxes are well known. However, they are ineffective against unknown zero day attacks. Event based IDSs have scalability issues. The number of system states to be kept track by the event based IDS can grow exponentially with the increase in network's size being monitored. This results in a tremendous computational overhead and limits their effectiveness in resource constrained networks. Anomaly based IDSs require extensive training periods and pure training data to develop the normal baseline profile of the network, which puts limitations on their deployment in real time networks. They also produce a significant volume of IDS traffic, which can cause congestion in bandwidth constrained networks. The work in this thesis aims to address these and various other IDS related issues like, false alarm minimization, reducing energy consumption required for operating IDS, minimizing the volume of intrusion detection related traffic etc., using a game theoretic approach.

The rest of the chapter has been organized in following ways. Section 2.2 provides a taxonomy about various type of IDSs, along with their strengths and drawbacks. Section 2.3 d0iscusses major issues in the existing IDS frameworks. Section 2.4 provides an introduction to the game theory. Section 2.5 discusses various game theory-based IDS frameworks proposed in the literature, along with their drawbacks. Section 2.6 lists the scope and contribution of the thesis and Section 2.7 provides the conclusion of the chapter followed by a brief overview of the next chapter.

2.2 IDS Taxonomy

Based on their detection capabilities, IDS frameworks proposed in the literature can be categorized into three different classes namely, signature based, event based and anomaly based. Signature and event based IDSs can only detect known attacks, which have been encountered previously. On the other hand, anomaly based IDSs can detect both known as well as previously unseen zero day attacks. In the subsequent sub-sections, we provide detailed descriptions about each class of IDS.

2.2.1 Signature based IDS

Signature based IDSs (also known as misuse based IDSs) correlate the network's data traffic with a set of well known attack signatures stored in their rule databases to detect network intrusions. These attack signatures are developed by network experts after executing the known exploits several times and observing the occurrences of unique patterns during their executions. When the data traffic being monitored matches with one or more attack signatures in the rule database, an alarm is raised by the signature based IDS to inform the network administrator about the security breach. The alarms generated by the signature based IDSs contain various information describing the type of attacks detected, targeted applications, port numbers and IP addresses of the victim machines etc. Typically, a signature based IDS acts like a firewall and looks out for known attacks that can be detected using signature rules. SMTP/SSH exploits, port scans, abuse of user's system command etc., are some examples of such attacks.

An ideal attack signature should be as simple as possible and should be capable of detecting different variations of the corresponding attack for which it was written. A simple attack signature makes it easier to search for a match in the network's data stream without overburdening the IDS's monitoring component. Many signature based detection systems use regular expressions for pattern matching and for identifying different variations of the same attack. Regular expressions allow wild card and complex pattern matching which results in more accurate detection of attacks. For example, in order to detect mails with executable attachments, signature detection system looks for the pattern "*name* = < *file-name* >.*exe*", where "file-name" is any valid filename. If the signature detection system is not equipped with regular expression capabilities, the attacker can avoid detection by inserting any number of spaces and tabs before or after the "=" sign. Therefore, by surrounding "=" with spaces and tabs, an attacker can send a virus infected executable, while hiding the attack from a signature detection system (not equipped with regular expression matching capabilities). Snort [11], BRO [13] and EMERALD [12] are some of the prominent signature based detection systems, which come equipped with regular expression based pattern matching capabilities. Snort's attack signature to detect a '*land*' attack, which is one form of Denial of Service (DoS) attack is given below :

alert ip any any -> any any (msg : "BAD TRAFFIC same SRC/DST"; sameip; reference : cve, CVE-1999-0016; reference : url, www.cert.org/advisories/CA-1997-28.html; classtype : bad - unknown; sid : 527; rev : 3;)

This Snort signature raises an alert whenever it detects an IP data packet with the same source and destination IP address. As can be observed from the signature's description, the details of this exploit is documented in the CVE database with reference number *CVE-1999-0016*.

2.2.2 Event based IDS

There are certain type of attacks for which the attack signatures cannot be written, as these attacks do not cause any change in the network's syntax under the normal and attack scenarios. Address Resolution Protocol (ARP) spoofing attack is one example of such an attack. In the ARP spoofing attack, the attacker broadcasts a fake ARP message containing a false Media Access Control (MAC) - Internet Protocol (IP) address pair. Since, ARP is a stateless protocol, the host machines receiving these fake ARP messages update their ARP cache with a false MAC-IP pair. Altering IP-MAC pairs with fake MAC addresses do not cause any change in the syntax of ARP messages. However, it causes a change in the intended behavior of the network communication. Upon successful execution of the ARP spoofing attack, all the data packets addressed to the host with the given IP address will instead be delivered to another host with a spoofed MAC address.

Several custom solutions for detecting ARP spoofing attack have been proposed in the literature. The solution proposed in [43] uses a static APR entries to prevent the ARP spoofing attack by maintaining a static IP-MAC address pairing of all the hosts in the network. When an ARP cache table is marked as static, the operating system's kernel ignores all the ARP replies with modified IP-MAC pairing and instead uses the static entries in the ARP cache table for mapping IP addresses to MAC addresses. However, this solution is not scalable and cannot be applied to large size networks, as maintaining static IP-MAC pairing entries for all the hosts becomes infeasible in such large scale networks. Another solution to prevent the ARP spoofing attack is by enabling security features offered by the switch, which allows the physical ports on the switch to be tied to a predefined MAC addresses using a Content Addressable Memory (CAM) tables. Under this scheme, the port-MAC address pairings are immutable and any change in their pairings are ignored. However, the drawback of this method is that it does not allow genuine changes in port-MAC address pairing. In addition to these custom solutions, several cryptography based techniques have also been proposed to prevent ARP spoofing attack [44] [45]. However, cryptography based techniques are computation intensive and require modification of standard ARP protocol, which is not

desirable.

An event based IDS can address the drawbacks associated with the custom and cryptography based solutions for detecting ARP spoofing attack. Event based IDS detects ARP spoofing attack by keeping track of the sequence of data packet events. The sequence of data packet events under the attack condition is different from that under the normal scenario. Event based IDS basically acts as a state estimator and observes the sequence of data packet events in the network to decide whether the observed progression of the states corresponds to normal or attack scenario. A generic system theory framework known as Failure Detection and Diagnosis (FDD) of Discrete Event Systems (DES) has been used to develop a fault identification models [39] [40] [46] [41]. Such frameworks use a state estimator called the diagnoser that monitors the sequence of events generated by the system/network to determine whether the states through which the system/network traverses corresponds to normal or failed (attack) conditions.

2.2.3 Anomaly based IDS

The major drawback of the signature and the event based IDSs is that they can only detect those attacks whose behaviors and syntaxes are well known. Anomaly based IDSs can be used to address this issue, as they do not require any predefined attack signatures or syntax knowledge to detect network attacks. They are capable of detecting previously unknown zero day attacks, which cannot be detected by signature and event based IDSs. Anomaly based IDS models the normal behavior of the network during the training phase to create the baseline profile of the network, which quantifies the full range of normal network activities. The baseline profile is then applied to real time data traffic and any deviation of the network's data traffic from the baseline profile is construed as anomalous. Attacks like extreme bandwidth usages, excessive system calls from a process, unusually high volume of incoming network traffic etc., cannot be detected using signature and event based IDSs, since it is not possible to write signature rules and define syntaxes for detecting such attacks. However, they can be detected using an anomaly based IDS since it uses the learned baseline profile to monitor the network's traffic.

The performance of anomaly based IDS strongly depends on the accuracy of the network's baseline profile; if there are any major changes in the underlying network environment from the time the baseline network profile was generated, false alarm rate will increase significantly resulting in the low accuracy of the IDS. Anomaly-based IDSs typically have lower

detection rate and higher false alarm rate than signature-based IDSs, but are capable of detecting zero day attacks. Anomaly based IDSs proposed in the literature can be categorized into data-mining based methods, statistical based methods and machine learning based methods. A brief description about each of these methods are provided below:

1. *Data-mining methods*: Data mining techniques take as input a set of data and output a set of patterns observed in the input dataset. Therefore, they can be applied to build a normal profile of the network traffic and detect network anomalies. FIRE [14], an anomaly based IDS uses data mining techniques to process the network traffic and generates a set of fuzzy rules corresponding to every feature in the dataset, which are then employed to detect attacks. Similarly, data mining based anomaly detection systems employing genetic algorithms use specific features in the dataset to detect attacks by observing any deviation in their values from the learned profile [16] [47]. Clustering is another major data mining technique that has been successfully employed to detect anomalies in the network traffic [15] [48].

2. *Statistical methods*: Anomaly based IDSs that use statistical methods work on the principle that anomaly results in deviation of network traffic characteristics such as change in number of packets transmitted, high frequency usage of certain IP addresses and ports etc. [17]. Statistical method that uses entropy measures to analyze various network traffic feature values can be used to build a fairly accurate anomaly detection model [49]. Statistical method based IDS schemes that examine the packet header contents, instead of the packet payload, using wavelet analysis to detect anomalous data packets are proposed in [18] [19]. In addition, statistical anomaly detection engines can be used in conjunction with signature based systems to detect unknown attacks and generate signatures. SPADE [42] is one such system that can be added to Snort which is a signature based IDS [11]. However, the major drawback of the statistical method based IDS schemes is that they fail to detect network attacks when the attacker keeps the disruptions caused by the attacks below the threshold levels.

3. *Machine learning methods*: Machine learning based anomaly detection methods automatically learn from the input network data and provide feedbacks to improve their performance over time. A machine learning based anomaly detection system that uses a Bayesian network model to detect Distributed Denial of Service (DDoS) attack is proposed in [20]. Bayesian network model assigns a probabilistic relationship between various features of network traffic under consideration, which enables it to determine

the interdependencies between different features. Additionally, they can also predict the future interdependencies between the network features. Anomaly detection systems employing various other machine learning algorithms like neural networks [50] [21], support vector machines [22] [51], Logistic regressions [52] etc., have also been proposed in the literature. These machine learning based classifier models have been shown to achieve high accuracy and detection rate across wide range of network attacks. However, the main drawback of the machine learning based anomaly detection systems is the overhead involved in training them.

In addition to signature, event and anomaly based IDSs, there is another class of IDS called the hybrid IDS, which uses a combination of signature based and anomaly based detection components to detect network intrusions. Hybrid IDSs have been shown to achieve high detection rate and accuracy across wide range of network attacks. A hybrid IDS framework that uses a combination of signature based and anomaly based detection components is proposed in [53]. It uses Snort as the signature based component and Packet Header Anomaly Detection (PHAD) and Network Traffic Anomaly Detection (NETAD) as the anomaly based components. Performance evaluation of this hybrid IDS framework on the MIT Lincoln Laboratories network traffic data (IDEVAL) [54] showed that it achieves higher detection rate compared to the misuse-based IDS and anomaly based IDS operating alone on their own. Similar result on the IDEVAL dataset was obtained by the hybrid IDS framework proposed in [55], which validates the effectiveness of hybrid IDSs in detecting network intrusions. However, hybrid IDSs are energy intensive and incur significant computational overhead as the data traffic needs to be analyzed and correlated by two different classes of IDSs to determine the network intrusions.

A summarized list of various IDS frameworks proposed in the literature based on different methodologies is provided in Table 2.1.

2.3 Issues with the existing IDS frameworks and motivation for thesis

IDS frameworks are characterized by many parameters namely, their detection rate, accuracy, energy efficiency, volume of IDS traffic generation, transparency, ease of use, security, interoperability, scalability etc. [56] [57]. Among these parameters, detection rate and accuracy are two important features that define the effectiveness of the IDS framework. Detection rate and accuracy are defined by the following three features: (i) True Positive

Table 2.1: List of various IDS frameworks

IDS Framework	Methodology
Snort [11]	Signature based
BRO [13]	Signature based
Emerald [12]	Signature based
Kozierok [43]	Event based
Gouda [44]	Event based
TARP [45]	Event based
Cassandras [39]	Event based
Sekar [40]	Event based
Whittaker [41]	Event based
FIRE [14]	Data mining based
MINDS [15]	Data mining based
Ramaswamy [48]	Data mining based
Genetics1 [16]	Data mining based
Genetics2 [47]	Data mining based
Valdes [20]	Machine learning based
Bequiri [21]	Machine learning based
KhanSVM [22]	Machine learning based
HongmeiSVM [51]	Machine learning based
Basant [50]	Machine learning based
Barford [17]	Statistical based
Lakhina [49]	Statistical based
Kim [18]	Statistical based
Kohler [19]	Statistical based
Spade [42]	Statistical based
BasantHybrid [55]	Hybrid
Aydin [53]	Hybrid

(TP), (ii) False Positive (FP) and (iii) False Negative (FN). TPs are the genuine alarms corresponding to real attacks; FPs are false alarms raised by the IDS for benign cases and FNs are the cases in which the IDS fails to identify the actual attacks. Detection rate of an IDS is a measure of to what degree the IDS can effectively detect the network intrusions. It is represented as the ratio of the number of attacks correctly identified by the IDS to the total number of attacks in the network during a given time period and is given as *TP/(TP + FN)*. On the other hand, accuracy of an IDS is the measure of how many genuine attacks are detected by the IDS. It represents the ratio of the number of genuine attacks versus all the attacks detected by the IDS and is given as *TP/(TP + FP)*. A perfect IDS would have a 100% detection rate and a 0% false alarm rate i.e., it would detect all the attacks without ever misclassifying any normal network behavior as anomalous. However, developing such a perfect system is rarely possible.

The IDS's false alarm rate is a function of its internal threshold. As the detection rate increases, the false alarm rate increases as well. Given this interdependency between the detection rate and the false alarm rate, IDS evaluation is usually conducted by plotting the Receiver Operating Characteristics (ROC) curve for a given implementation. As shown in Fig. 2.1, the ROC curve is created by plotting the TP rate against the FP rate at various threshold settings, with greater area under the curve indicating a better performance of the IDS model. An area between 0.9 to 1 under ROC curve represents a perfect anomaly detection model, an area between 0.8 to 0.9 represents good model, while an area under 0.5 represents a poor model. In the subsequent subsections, major issues in signature, event and anomaly IDS frameworks with respect to various parameters like detection rate, accuracy, energy efficiency and IDS traffic volume are discussed.

2.3.1 Issues with the signature based IDSs

Signature based IDSs maintain a set of attack signatures for known attacks in their rule database. The attack signatures characterize the profile of known security threats and are used to monitor the data streams of various flows traversing through the network. When the flow being monitored matches with one or more attack signatures, appropriate actions are taken (e.g. block the flow or limit the flow rate). Signature based IDSs have high detection rate and accuracy against known attacks for which the corresponding attack signatures are present in their rule database. However, there are many challenges and drawbacks associated with signature based IDSs, some of which are listed below:

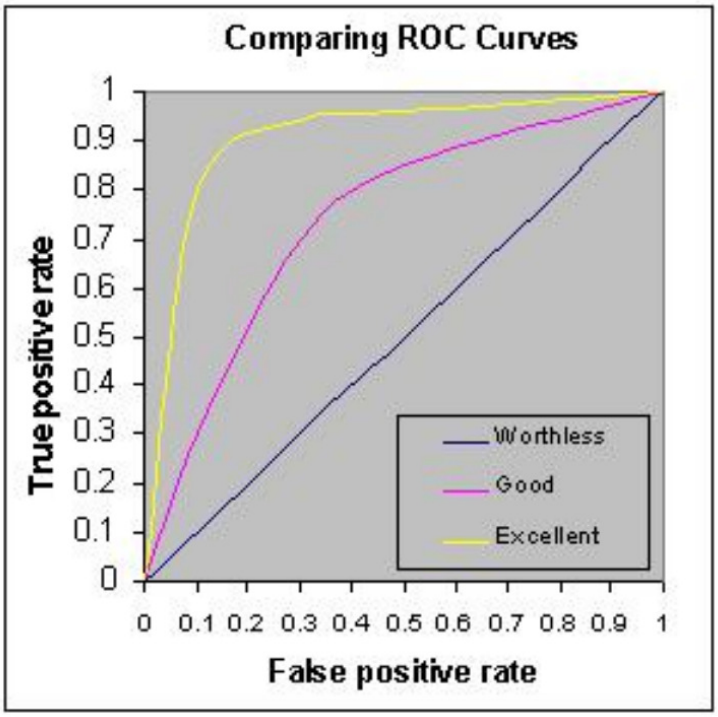

Figure 2.1: ROC with different threshold settings

- *Incapable of detecting zero day attacks*: Since signature based IDSs use attack signatures to detect intrusions, they are incapable of detecting new attacks for which the signatures have not yet been developed. Therefore, they have very low detection rate against new zero day attacks. Moreover, since most of the attack signatures use regular expressions for pattern matching, the attacker can circumvent and avoid detection by minutely altering and modifying the attack patterns.

- *Manual signature development*: Developing good quality attack signatures is a difficult task and requires a thorough expertise and knowledge about the network domains. A signature should be capable of detecting all possible variations of the attack for which it has been written, while at the same time avoid misclassification of benign network traffic as attacks. Since, the attack signatures have to be developed manually by human experts, they are prone to errors and mis-configurations.

- *High false alarm rate*: When signature based IDSs are deployed with default configurations, without considering the context of the underlying networks, large number of false positive alarms is generated. Therefore, prior to their deployment, attack signatures need to be customized for detection of network specific attacks to avoid

misclassification of normal data traffic as anomaly. For example, if it is known that all the machines on the network runs on the Windows operating systems then the attack signatures written for detection of the Linux operating system specific attacks can be removed or disabled from the signature database.

- *Frequent signature updates*: Signature based IDSs require frequent update to their rule database for detecting newly discovered network vulnerabilities. However, developing good quality attack signatures for the newly discovered exploits takes time, since it requires deeper understanding of the attacks' semantics. Lengthy signature development phase provides an attacker with enough time to exploit the newly discovered network vulnerabilities without being detected.

Amongst all the drawbacks listed above, generation of large number of FP alarms is one of the most pertinent issue facing the signature based IDSs. High false alarm rate (FAR) is absolute nightmares of IDS researchers and practitioners. Voluminous amount of FP alarms can overwhelm the network administrator, who has to go through each and every alarm to verify its genuineness. In the absence of an alert verification and filtering mechanism, the attacker can launch a DoS attack against the IDS infrastructure and render it non-responsive, thereby defeating the very purpose of having an IDS in the first place itself. The severity of this issue can be gauged from the fact that sometimes up to 98% of the alarms generated by the signature based IDSs are FPs [58] [59]. FP alarms are generated under following two circumstances:

1. When a network host is attacked and the alarm is generated by the IDS but the attack fails to exploit any vulnerabilities of the targeted host. For example, when a Windows based attack is launched on a Linux machine, an alarm is generated by the IDS due to matching signature. However, the attack fails to exploit any vulnerabilities of the Linux system.

2. When benign network activities are misclassified as attacks by the IDS. For example, the administrator may occasionally ping the host machines in the network to verify whether they are up and running. However, an attack signature written to detect execution of ping command raises an alert, which results in a FP alarm in such cases.

Various techniques have been proposed in the literature for minimization of FP alarms of signature based IDS, which can be classified into following categories:

- *Signature enhancement*: In this method, attack signatures are enhanced with additional network context information for better correlation of network data traffic with the attack signatures [60] [61]. For example, consider the following Snort signature used for detecting buffer overflow that triggers a DoS attack on "Microsoft distributed transaction" service.

  alert tcp EXTERNAL_NET any → HOME_NET 3372 (msg:DOS MSDTC attempt; flow: to server, established; dsize: > 1023; reference: bugtraq,4006; reference: cve,2002-0224; reference: nessus, 10939; classtype: attempted-dos; sid: 1408; vrev: 10;)

  This Snort rule monitors the TCP connections from external network to home network on port 3372. If any data packet in an open TCP session whose size is greater than 1023 bytes is detected, then Snort generates a *DOS MSDTC attempt* alarm message. However, this attack signature raises an alert, irrespective of the type of the operating system (OS) that the host machine on the home network is running on. This leads to generation of FP alarms when the underlying machine is running on Linux OS, as this attack is ineffective on Linux machines. The generation of FP alerts in such cases can be avoided by enhancing the attack signatures to raise an alert, only when the attack is detected against hosts running on Windows OS.

- *Alarm correlation from different IDSs*: This technique correlates the alarms generated by the signature based IDS with the alarms produced by other IDS (anomaly or event based) to verify the TP alarms. A framework that correlates syslog information with alarms generated by HIDS and NIDS is proposed in [62]. Correlation process begins by removing alarms through a filtering process which are known to be non effective. Following this, both HIDS and NIDS events are matched to determine success of the attack attempt. An architecture for automatic alert verification called ATLANTIDES is proposed in [63]. It consists of a network IDS that monitors the incoming data traffic and an Output Anomaly Detector (OAD), which compares the output traffic with the model it has created during the training phase. It verifies whether the incoming data traffic that raises an alert in the input network IDS actually produces an anomaly in the outgoing traffic too. If it does, then the alert is forwarded to the administrator as a TP alarm, otherwise it is discarded as a false alarm.

- *Alarm correlation with reference number*: This method maintains a list of vulnerabilities present in the host system and correlates the alarms generated by the signature based IDS with the vulnerabilities of the host system based on the vulnerability ref-

erence numbers (as recorded in some public vulnerability databases like CVE [8], BugTraq [7], Nessus [10] etc.). A correlation scheme that correlates IDS alarms with network vulnerabilities in the Threat profile based on the reference number is proposed in [57]. Alarms that cannot be correlated with any network vulnerabilities in the Threat profile are filtered out as FPs. However, instead of directly discarding them as FPs, a check is made to determine if these alarms correspond to attacks on critical applications. If the filtered FP alarms are against the critical applications then they are restored back as TP alarms.

Although false alarm minimization schemes proposed in the literature improve the accuracy of signature based IDSs, they have several drawbacks associated with them. In the signature enhancement based scheme, attack signatures have to be modified and the intrusion detection engine has to be restarted every time the network context information changes. Additionally, modifying attack signatures is a tedious task and prone to error, which can adversely affect the performance of the signature based IDSs. On the other hand, improving the accuracy by correlating alarms from different IDSs is a challenging and non-trivial task, since events from different domains need to be matched. For example, correlation scheme proposed in [62] matches the alarms generated by the signature based NIDS with the alerts produced by the anomaly based HIDS. However, such correlation process is difficult due to differences in the response time and syntaxes of the NIDS and HIDS. Therefore, the improvement in the accuracy achieved by this scheme is limited. Although, the technique based on correlating the signature based IDS alarms with the host vulnerabilities on the basis of reference numbers do not require any signature modifications or comparison with other type of IDSs (anomaly or event based), they have their own set of drawbacks. Some of the alarms generated by the signature based IDS may not have valid reference numbers [64]. Therefore, this technique fails when correlation is not possible between the IDS alarms and the host vulnerabilities due to missing reference numbers.

Any false alarm minimization scheme proposed for signature based IDS to improve its accuracy also leads to drop in its detection rate. Decrease in detection rate against non-critical applications and services can be tolerated. However, decrease in detection rate against critical applications and services may lead to severe network security breach as some of the attacks against these critical applications might pass undetected by the IDS. The works proposed in [65] [66] [67] provide a prioritized list of alarms to the administrator, instead of filtering out some alarms as FPs. The administrator then uses some threshold

values to determine the TP alarms. However, the effectiveness of these schemes largely depend on the threshold values selected to differentiate between the TP and FP alarms. Setting a low threshold value increases the false alarm rate, wherein the benign network traffic are classified as attacks. On the other hand, setting a high threshold value decreases the detection rate, as most of the alarms corresponding to genuine attacks are dropped as FPs.

2.3.2 Issues with event based IDSs

Although event based IDSs are capable of detecting various attacks for which signatures cannot be developed (e.g., ARP spoofing attack in LAN, Internet Control Message Protocol (ICMP) attack using error messages etc.), they have their set of limitations. Event based IDSs act as a state estimator and monitor the sequence of events generated by the network to decide whether the states through which the system traverses correspond to normal or compromised conditions. However, the main drawback of the event based IDSs is their scalability. The number of states to be kept track by the event based IDS can grow exponentially with the increase in network's size being monitored. This introduces a tremendous computational overhead and limit their overall effectiveness in a resource constrained networks. Moreover, they require active techniques like sending out probe packets to identify the differences in sequencing of data packets under normal and attack conditions, which violates the standard operations of the protocols[68] [69]. Such active techniques can cause congestion in a bandwidth constrained wireless networks. All these issues put a severe limitation in their deployment on large scale real time networks.

2.3.3 Issues with anomaly based IDSs

Anomaly based IDSs create baseline profile of networks during their training phase and apply them to the real time network traffic. Any deviation of the monitored data traffic from the learned baseline profile is flagged as anomalous. Anomalous events are caused by network activities that fall outside the predefined or accepted model of learned behavioral patterns. Although anomaly based IDSs are more effective in detecting wider range of network attacks in comparison to the signature or event based IDSs, they have their share of drawbacks. Training an anomaly detector model to develop the network's baseline profile is computation intensive and requires a pure training data that is free from contam-

ination and attack data. However, obtaining such attack free and non-contaminated data is difficult. Moreover, the anomaly detection model needs to be retrained periodically at regular intervals to remain effective. If they are not re-trained periodically to incorporate the changes in the underlying network parameters, the network's profile may eventually outgrow the established baseline profile, leading to subsequent IDS assessment based on stale training. Anomaly based IDSs are also known to produce a significant volume of IDS traffic. This can adversely affect the flow of normal data traffic and cause congestion in bandwidth constrained networks.

In addition to accuracy and detection rate, various other factors need to be taken into consideration, while designing an IDS framework. IDS's architecture and working principles largely depend on the nature of the underlying networks for which they are being designed. Wired networks have comparatively higher computational capabilities and network resources at their disposal compared to their wireless counterparts. Therefore, IDS frameworks designed for wired networks are not constrained in terms of energy consumption and computational resources. However, they have to be designed to minimize the number of FP alarms. High false alarm rate can overwhelm the network administrator, who has to check every alert generated by the IDS to verify their genuineness and authenticity.

On the other hand, wireless networks like Mobile Ad-hoc Networks (MANETs) and Wireless Senor Networks (WSNs) are usually energy and resource constrained. Nodes in such networks have limited computational capabilities and are usually powered by batteries. Therefore, persistent monitoring operations can result in premature death of nodes operating the IDS in such networks. In addition, wireless networks like Vehicular Ad-hoc Networks (VANETs) operate in a narrow bandwidth wireless radio spectrum. Therefore, high volume of intrusion detection related traffic can cause congestion and hinder the flow of normal data traffic in VANETs. Hence, IDS frameworks proposed for wireless networks must possess the following key properties:

1. *High detection capabilities*: The open and dynamic nature of wireless networks make them extremely vulnerable to different type of attacks like black hole attack, wormhole attack, sybil attack, DoS attack etc. Due to high cost and overhead involved in encryption process, data packets in these networks are usually transmitted in plain text form, without encrypting them. The attacker can easily intercept these plain text data packets and forge them before reintroducing them back into the network. This can have an adverse ramification on the integrity and confidentiality of the wireless

networks. Therefore, due to high stakes involved in their security, any intrusion detection framework (signature or anomaly based) proposed for wireless networks must be capable of detecting such attacks with high detection rate.

2. *Reduced false alarm rate*: Signature based IDSs produce a large number of false positive alarms when they are deployed with out of the box configurations, without considering the context of the underlying networks. Similarly, anomaly based IDSs also produce a large number of false positive alarms when they are not trained properly. Due to the limited processing capabilities of nodes in wireless networks, high false alarm rate can result in the breakdown of the nodes operating the IDS. Moreover, the attacker can launch a DoS attack against the IDS infrastructure itself and render it ineffective. Therefore, IDS frameworks must incorporate appropriate false alarm minimization scheme to prevent the IDS from generating a deluge of FP alarms.

3. *Reduced energy consumption*: Wireless networks like MANETs and WSNs are characterized by energy and resource constrained nodes. Therefore, any IDS framework that requires substantial amount of computing power can drain out the energy level of the nodes in these networks. Computation intensive monitoring operation results in premature death of the nodes operating the IDS, which effectively shortens the life span of these networks. Therefore, IDS frameworks proposed for energy constrained wireless networks must adopt appropriate measures to reduce the energy consumption required for operating the IDS.

4. *Minimized IDS traffic volume*: Wireless networks usually operate in a narrow bandwidth wireless radio spectrum. Therefore, large volume of IDS traffic can cause congestion and prevent the flow of normal data traffic in these networks. Hence, IDS frameworks proposed for wireless networks must not generate voluminous amount of intrusion detection related traffic.

Game theory-based IDS frameworks : Game theory can be used to address various IDS related issues discussed in the earlier sections of this chapter like, false alarm minimization, minimizing energy consumption required for operating IDS, reducing the volume of IDS traffic [25] [27] [29] [31] etc. Game theory has extensively been used in the literature for developing IDS frameworks with great results [70] [71] [72]. Game theory can be employed as a mathematical model for developing a false alarm minimization scheme for signature based IDS by identifying the potential network vulnerabilities and assigning

priorities to them. It can also be used for designing efficient IDS monitoring strategies to minimize the energy consumption required for operating IDS in energy and resource constrained wireless networks like MANETs and WSNs, without degrading the IDS's overall performance. In addition, it can also be used to minimize the overall volume of IDS traffic and prevent network congestion in bandwidth constrained wireless networks like VANETs. Reduced IDS traffic volume improves the throughput and performance of these wireless networks by enabling a seamless flow of normal data traffic.

In the next section, we provide a detailed description about game theory and show how it can be used to address various IDS related issues in wired and wireless networks.

2.4 Game Theory

Game theory is the study of mathematical models of conflict and cooperation between intelligent decision-makers [24]. The roots of the game theory can be traced back to 1944, when mathematicaian John Von Neumann and economist Oskar Morgenstern published their seminal work entitled "Theory of Games and Economic Behaviour", which established game theory as a full-fledged sub-discipline in the field of mathematics. Their seminal work outlined mathematical theories involving broad range of economical problems and propagated the usage of game theory in other disciplines apart from mathematics. Another milestone in the history of game theory came in 1949 when mathematicaian John Nash theorized the concept of Nash Equilibrium (NE) in non-cooperative games. Non-cooperative games are widely used to model the situations of conflict between two of more competitive players, wherein the players of the game are assumed to be highly rational. The cooperative solution concept in such non-cooperative settings is provided by the NE of the game.

Game theory has been used in wide areas of research namely, economics, political science, computer science, psychology, biology etc., to study the events of conflict and cooperation between two or more rational decision makers (players) with common or contradicting set of objectives. It has experienced a tremendous success in both theoretical results and variety of real world applications, which is vindicated by the fact that a total of eight Nobel prizes have been awarded to economic sciences for work primarily on game theory. The first such recognition was awarded in 1994 to John Harsanyi, John Nash, and Reinhard Selten for their pioneering analysis of equilibria in the theory of non-cooperative games. Robert Aumann and Thomas Schelling were awarded the Nobel Prizes for enhancing the understanding of conflict and cooperation through game-theory analysis in 2005. In 2007,

Leonid Hurwicz, Eric Maskin, and Roger Myerson were awarded with Nobel Prizes for having laid the foundations of mechanism design theory. Recently, in 2012, Llyod S. Shapley and Alvin E. Roth were awarded the Nobel Prize in Economics for their seminal work on introduction to concept of payoff distribution in a cooperative coalition games.

Game theory attempts to formulate the logical and mathematical actions that the players should adopt to obtain the best possible outcomes for themselves when faced with the choice of series of strategies. Game theory can be categorized into cooperative and non-cooperative games. The work in this thesis mainly deals with non-cooperative games, wherein players have conflicting and contradictory objectives. Non-cooperative games involving two or more players are characterized by the solution concept called the Nash Equilibrium (NE). NE of a non-cooperative game corresponds to the players' strategy set in which no player can benefit by changing its chosen strategy while the other players keep their strategies unchanged. Therefore, in a non-cooperative game the players do not have any profitable incentives to deviate from the established NE strategy, as it leads to reduced payoff for the deviating player. However, this solution concept only specifies the steady state but does not specify how that steady state is reached in the game. Every finite game with finite set of players and strategies has a NE in mixed strategies. The complexity of finding a NE in a normal form game is a PPPAD-complete problem [73]. PPAD is an acronym for "polynomial party argument (directed case)". The formal definition of PPAD and other examples of PPAD problems can be found in [74]. It is believed that PPAD-complete problems are not solvable in polynomial time. However, they are simpler than NP-complete problems, although this remains an open problem to be verified.

More precisely, a non-cooperative game is specified by three important parameters: (i) the number of players, (ii) the possible actions available to each player along with a set of constraints imposed on them, and (iii) the objective function of each player which it attempts to optimize (maximize or minimize). Accordingly, a non-cooperative game can be represented by the triplet $\langle$N, S, U$\rangle$, where

- $N = \{P_1, P_2, ..., P_n\}$ are the n players of the game.
- $S = S_1 \times S_2 \times .... \times S_n$ is the strategy space of the game with S_i being the the possible action set of player $P_i \in N$.
- $U = U_1 \times U_2 \times .... \times U_n$ is the payoff utility corresponding to the strategy space S. U_i is the payoff utility of the player P_i corresponding to its chosen strategy $s_i \in S_i$.

For each player $P_i \in N$, its utility function U_i is defined as a function of its chosen strategy $s_i \in S_i$ and the set of strategies chosen by the other players denoted by s_{-i}. The strategy combination (s_i, s_{-i}) corresponds to the NE of the game if it satisfies the following properties:

$$U_i(s_i, s_{-i}) \geq U_i(s_i^*, s_{-i}) \quad \forall \quad s_i^* \in S_i$$

Therefore, NE is identified as a strategy combination of the players, wherein no player will rationally choose to deviate from its chosen strategy, while the other players keep their chosen strategies fixed, as doing so will diminish the payoff utility of the deviating player. In the subsequent subsection, we present two well known examples of non-cooperative games to elaborate the concept of NE.

2.4.1 Prisoner's dilemma

The prisoner's dilemma is a classic example of a game analyzed in game theory that shows why two completely "rational" individuals might not cooperate, even if it appears that it is in their best interests to do so. Consider two prisoners being interrogated simultaneously in two separate cells. Each prisoner has two options: (i) cooperate with the other prisoner (ii) defect by betraying the other prisoner. If both the prisoners defect then they would serve a longer jail sentence compared to when both of them say nothing. The payoff of each prisoner corresponding to his chosen strategy is given by Table 2.2.

Table 2.2: Prisoner's dilemma payoff matrix

Prisoner 1 \ Prisoner 2	**Cooperates**	**Defects**
Cooperates	1,1	3,0
Defects	0,3	2,2

From the Table 2.2, it can be observed that the maximum reward (0 years jail term) is achieved by the prisoner when it defects and the other prisoner cooperates i.e., when their decisions are different. The Prisoners' dilemma has one single NE, which is for both players to "defect". It can be observed that although the best outcome would be achieved when both the players "cooperate". However, it is not a stable solution, as each player has an incentive to change his strategy to "defect" from "cooperation".

2.4.2 Matching pennies

Matching pennies is a simple game comprising two players (Player 1 and Player 2) with each player tossing a penny to get heads or tails. If the pennies match i.e., if both heads or both tails show up then Player 1 wins one from player 2. However, if the pennies do not match i.e., one head and one tail show up then Player 2 wins and receives one from Player 1. The payoff matrix corresponding of this game is shown in Table 2.3

Table 2.3: Matching pennies payoff matrix

Player 1 \ Player 2	**Head**	**Tail**
Head	+1,-1	-1,1
Tail	-1,+1	+1,-1

Matching Pennies is a classic example of a zero-sum game, wherein one player's gain is exactly equal to the other player's loss. It can be observed from Table 2.3 that this game does not have any pure strategy NE since there is no pure strategy (head or tail) of the player that is the best response to the other player's chosen strategy (head or tail). The unique NE of this game is a mixed strategies, wherein each player chooses head or tail with equal probability. Such a mixed strategy NE makes one player indifferent between choosing its strategy of head or tail in response to the other players chosen strategy.

In the subsequent subsections, we provide a brief discussions on taxonomy of games and various methods used to solve them.

2.4.3 Non cooperative games

Non-cooperative games are used to model the interaction between two or more competing players with conflicting set of objectives. In such games, individual players might act selfishly by unilaterally deviating from a proposed solution if it is in their own selfish interest, without coordinating their actions with other players. There are no external mechanisms to enforce cooperative behavior among the players in non-cooperative games, which leads to a self-enforcing alliances among players (e.g. through credible threats or through competition between group of players). Non-cooperative games are competitive in nature, wherein each player tries to choose its best available action (the one which gives a player the highest payoff, called best response), irrespective of the effects that their choices may have on other players. The best action for any given player in a non-cooperative game de-

pends in general, on the other players' actions. Therefore, a player must keep in mind the actions the other players will choose, while choosing its own action.

Nash equilibrium (NE) is a central solution concept in non-cooperative game theory. It captures the notion of a stable solution, from which no single player can individually improve his welfare by unilaterally deviating, while the other players keep their strategies unchanged [75]. Nash equilibrium represents a certain stable operating point that is robust to unilateral deviations. It is not necessarily the best solution concept, but it is at least the one which all players agree upon. Nash theorem states that every finite game in strategic form has a Nash equilibrium in either mixed or pure strategies [76]. A game has a Nash equilibrium in a pure strategy, when each player deterministically plays its chosen strategy. When players are allowed to randomize and each player picks a certain probability distribution over his set of strategies, such a choice is called mixed strategy.

Non-cooperative games can further be classified as complete or incomplete information games. In a complete information game, each player has a complete knowledge about the utility functions, payoffs and strategies of all the other players in the game but the players may not see all of the moves made by other players. On the other hand, incomplete information games are those in which at least one of the player is unaware about the utility functions, possible payoffs and strategies of at least one other player of the game. In such games, some of the players possess some private information (for e.g., their type), which is unknown to other players of the game.

In addition, there is a class of an incomplete information game called the Bayesian game, in which each player has a belief value about the type of the other players with a certain priori probability distribution. Bayesian games are characterized by the presence of a special player called Nature that assigns a type to each player according to the probability distribution across each player's type space [77]. Such modeling enables Bayesian game to convert the incomplete information game to an imperfect information game (in which the history of the game is not available to all players). In the Bayesian game, each player has initial beliefs about the type of every other players, which are later updated according to the Bayes' rule as the game progresses, i.e., the belief a player holds about another player's type changes based on the observed action of that another player. The resulting NE of this class of games is called the Bayesian Nash Equilibrium (BNE).

2.4.4 Cooperative games

Cooperative games (also known as a coalition games) are used to model the competitive interaction between group of players with a same set of objective functions. Cooperative games consider the payoff utility of all the players with the goal of maximizing the entire coalition's pay-off, while maintaining the fairness for each individual player of the coalition. Coalition of players in cooperative games arise due to the possibility of external enforcement of cooperative behavior (e.g., through contract law). Such enforced cooperative behavior may not be in the best interest of the players of the coalition. A central notion in the cooperative game theory is the concept of the *core*. The *core* is a set of payoff allocations that guarantees that no group of players have any incentive to leave its coalition to form another coalition. However, the *core* solution can suffer from some drawbacks, like having an unfair allocation and being difficult to achieve. Another solution concept in cooperative game theory is the *Shapley value*, which is used to calculate the marginal contribution of the individual player in the coalition. However, the complexity of computing the *Shapley value* increases with the increase in the number of players in a cooperative game. Therefore, it is recommended only for cases where the number of players is low.

Another widely applicable concept of cooperative games is *bargaining games*. The bargaining problem studies a situation where two or more players need to select one of the many possible outcomes of a joint collaboration. For example, players trying to come to an agreement on a fair resource sharing inside a cluster. If the individuals reach an agreement, they can gain a higher benefit than playing the game without cooperation. The solution of this type of games is called the Nash Bargaining Solution (NBS), in which no action taken by one of the individual is imposed without the consent of the others.

2.4.5 Cooperation enforcement games

This class of game considers players that would normally behave selfishly but they are enforced to cooperate, while still striving to maximize their outcomes from the game. Cooperation enforcement mechanisms are also designed to encourage greater cooperation among individuals. For example, in multi-hop wireless networks, each node serves as a source/destination for traffic as well as a router to forward data packets. Applying game theory in such environments raises the following question: What are the incentives for nodes to cooperate, particularly when there may be natural disincentives such as higher

energy consumption? Incentive mechanisms are used to address this question. Incentive mechanisms are generally divided in two schemes: reputation-based systems and credit-based systems [75] [78]. In credit-based systems, cooperation is induced by means of payments received every time a node relays or forwards a packet, and such credit can later be used by these nodes to encourage others to cooperate. In reputation-based systems, each node assigns a reputation value to all other nodes in its neighborhood. As the node's reputation decreases, its neighbors may refuse to perform services for it, leading to its gradual exclusion from the network. Nodes decide independently the extent of their cooperation with the network, trying to balance their reputation (too little cooperation might lead a node to become an untrustworthy node) and resource considerations (too much cooperation may lead to a fast battery depletion).

Mechanism design [75] [79] is another branch of cooperative games which aims to enforce cooperation between nodes, and design games that have dominant strategy solutions leading to a desirable outcome (either socially desirable, or desirable for the mechanism designer). The idea is to run an algorithm in an environment with multiple owners of resources. This algorithm takes into account the preferences of the different owners. The larger goal of the mechanism design is often to design structures that lead to socially optimal outcome of the game even under selfish behaviors of the players. Mechanism design could be with money (auctions), like Vickery-Clarke-Groves mechanisms [38], or without money, like House Allocation problem. It is analogous to Bayesian games in terms of privacy of owners information, but mechanism design makes the solution of a game much simpler. The overall taxonomy of the game theory and different methods used to solve them is shown in Fig. 2.2.

2.5 Game theory-based IDS frameworks and their issues

In this section, we discuss a number of game theory-based IDS frameworks proposed in the literature. Game theory has successfully been used to address various IDS related issues like false alarm minimization, energy consumption reduction and minimization of IDS traffic volume [25] [27] [28] [29] [31]. Game theory-based IDS frameworks model the intrusion detection process as a two player non-cooperative game between the IDS and the attacker. Such game theoretic modeling takes into account various factors like monitoring and attacking cost, detection rate and false alarm rate of IDS, network vulnerabilities, network's operating systems and applications etc., for developing the IDS's monitoring

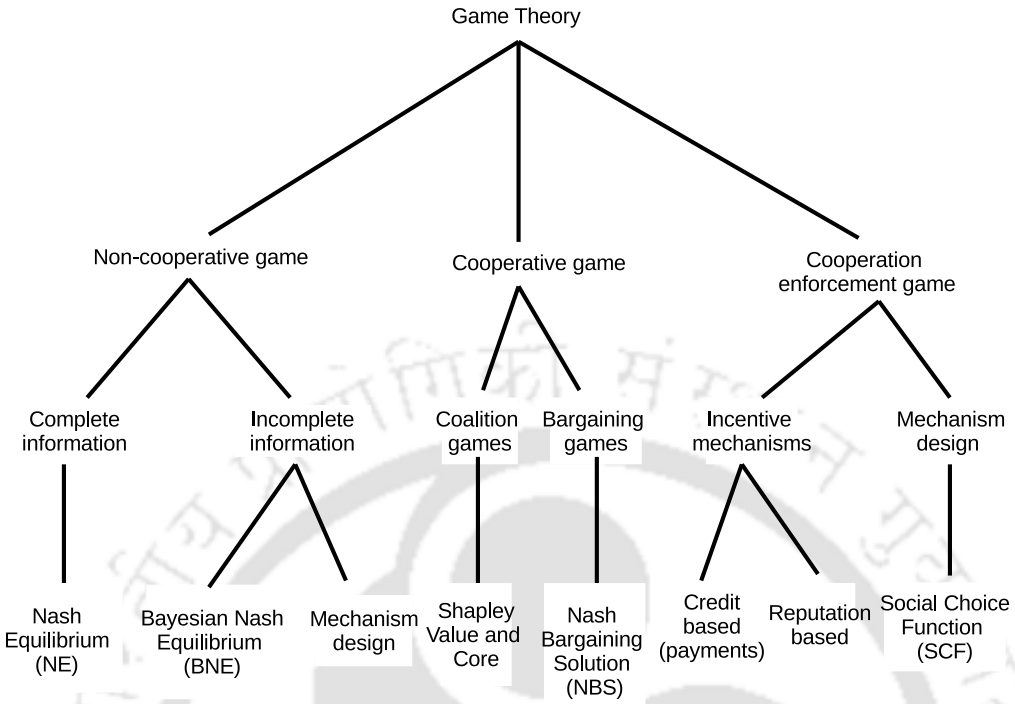

Figure 2.2: Taxonomy of games and different methods to solve them

strategies. Game theory allows the IDS to assess the type of the nodes being monitored and adopt a dynamic monitoring strategy based on their maliciousness level. It enables the IDS to adopt a probabilistic monitoring strategy based on the Nash Equilibrium of the game, instead of adopting an alway on monitoring strategy. Such probabilistic monitoring strategy significantly minimizes the volume of IDS traffic and also reduces the energy consumption required for operating the IDS, without significantly degrading the detection rate and accuracy of the IDS. Similarly, cooperative game theory-based solution concepts like VCG mechanism and Shapley value have been used for enforcing cooperative behaviors among network entities to achieve a coalition based IDS frameworks.

Game-theoretic solutions for Ad-hoc networks that model the cooperation and selfishness of the networks are discussed in [25] [26]. In these schemes, each node decides whether to forward or drop packets based on the trade-offs involved in cost (energy consumption) and benefits (network throughput) involved in collaborating with other nodes in the network. Therefore, enforcing a cooperation mechanism ensures that a selfish node which does not abide by the network rules receives a low throughput. However, the drawback of this scheme is that it assumes a complete information game, wherein nodes are assumed to

have a full knowledge about all the network parameters. Such assumptions are usually invalid in real networks, wherein nodes only have partial information about various network parameters.

A game theory-based IDS framework to analyze the interactions between pair of attacking/defending nodes using a Bayesian formulation in wireless Ad-hoc Networks is proposed in [27]. The framework suggests a Bayesian hybrid detection approach for the defender, wherein a less powerful lightweight module is used to estimate the opponent's type, and a more powerful heavyweight module acts as a last line of defense. It analyzes the obtainable Nash Equilibrium (NE) for the attacker/defender Bayesian game in both static and dynamic settings. It concludes that the dynamic approach is a more realistic model, since it allows the defender to continuously update its belief about the maliciousness level of the opponent player as the game evolves. However, the drawback of this framework is the difficulty involved in determining a reasonable prior probability about the maliciousness level of the attacker player.

A general incentive-based scheme to model Attacker's Intent, Objectives and Strategies (AIOS) based on game theoretic formalization is proposed in [80]. The scheme develops an incentive-based conceptual framework for AIOS modeling, which can capture the inherent inter-dependency between AIOS and defender objectives/strategies in such a way that AIOS can be automatically inferred. The AIOS modeling enables the defender to predict which kind of strategies are more likely to be taken by the attacker than the others, even before such an attack happens. The AIOS inferences lead to more precise risk assessment and harm prediction. However, the major issue of this framework is the complete information game assumption, wherein the players are assumed to have a complete information about all the other players.

A framework that applies two game theory-based schemes for economic deployment of intrusion detection agent is proposed in [28]. In the first scheme, the interaction between an attacker and the intrusion detection agent is modeled and analyzed within a non-cooperative game theory settings. The mixed strategy Nash Equilibrium solution is then used to derive the security risk value of the network. The second scheme uses the security risk value derived by the first scheme to compute the Shapley values of the intrusion detection agents, while considering the various threat levels. This allows the network administrator to quantitatively evaluate the security risk of each IDS agent and select the most critical and effective IDS agent deployment to meet the various threat levels to the

network. The drawback of this scheme is its high computational overhead, as it involves analyzing the intrusion detection process as a non-cooperative game and then evaluating the Shapley values of the intrusion detection agents.

A game theory-based framework that models the interaction between the service provider and the attacker as an intrusion detection game is proposed in [29]. In this scheme, the game is represented as a two person zero-sum game, wherein the service provider tries to maximize its payoff by increasing its probability of successful detection, while the attacker tries to minimize its probability of being detected by the IDS. The optimal solution for both the player is to play the minmax strategy of the game. However, the drawback of this model is its assumption that both the players (attacker and defender) have complete information about the entire network topology and links. Such assumptions allow the players to choose optimal paths for playing the minmax strategy. However, these assumptions are usually not valid in real networks, where the players do not have a complete information about all the network parameters.

IDS frameworks that model the attack-defense problem in Wireless Sensor Network (WSN) as two players non-cooperative, non-zero-sum game between the attacker and the IDS is proposed in [31]. In this IDS framework, players (attacker and the IDS) are assumed to have a complete information about each other along with the payoff utilities corresponding to different strategies of the game. However, the drawback of these frameworks is the complete information assumption about the game. A game theory-based intrusion detection framework proposed for WSN in [30] models the intrusion detection process as a two-player non-cooperative and nonzero-sum game between an attacker and the sensor node. The framework achieves better performance compared to the Markov Decision Process (MDP) based IDS framework by predicting the highly vulnerable senor nodes of the network and adopting a probabilistic monitoring strategy to monitor those vulnerable nodes. However, the major drawback of this framework is the assumption that at any given time the attacker only attacks the nodes in one of the given clusters. However, such assumption is impractical in WSN, as the attacker can attack nodes in multiple clusters at a given time. A game theory-based intrusion detection framework for preventing Denial of Service (DoS) attacks in WSN is proposed in [32]. The framework models the intrusion detection process as a repeated game between an intrusion detector and sensor nodes, where some of the nodes act maliciously. Sensor nodes are classified into different categories based on their dynamically measured behavior. The framework is shown to identify malicious sensor nodes performing DoS attacks with high detection rate and accuracy. However, the

drawback of this framework is that it can only detect DoS attacks.

A game theory-based intrusion detection framework for preventing malicious sensor nodes from launching attacks is proposed in [81]. The framework uses a sub-game perfect collusion resistant punishment mechanism to enforce sensor networks to reach a cooperative Nash Equilibrium. Similarly, an active defense model for WSNs based on evolutionary game theory is proposed in [82]. In this framework, the sensor nodes adjust their monitoring strategy based on different policies of the attacker. However, the drawback of this framework is that there can be multiple intrusion attempts to the WSN but only one of them would be detected the framework, while leaving other intrusions undetected. A probabilistic game theory-based model, which makes use of cooperation between IDSs among neighborhood nodes to reduce their individual active time is proposed in [83]. The model reduces the active duration of the IDSs, without compromising on their effectiveness to detect network intrusions. It models the interactions between IDSs as a multi-player cooperative game in which the players have partially cooperative and conflicting goals.

A game theory-based efficient lightweight intrusion detection and prevention schema for Vehicular Ad-hoc Networks (VANETs) is proposed in [33]. The schema has the ability to accurately predict the future malicious behavior of an attacker and categorize it into an appropriate list according to the future attack severity. It models the attack-defense problem in VANET as a Bayesian game formulation between the attacker and RSU, and the future states prediction of a suspected behavior is determined using the Bayesian Nash Equilibrium (BNE). The schema is shown to exhibit a high detection rate (over 98%) and low false positive rate (close to 2%). Additionally, it incurs a marginal overhead, even with a large number of malicious vehicles in the network to achieve a high-level of security.

A game theoretic framework to address the problem of greediness (in presence of selfish nodes) in IEEE 802.11 CSMA/CA MAC access protocol for VANETs is proposed in [34]. The framework encourages the selfish vehicles to behave normally under the threat of retaliation and motivates them to cooperate if they aim to maximize their obtained payoff. It is shown to improve the misbehavior detection decision and thus impose the MAC-layer cooperation in VANETs.

In summary, we found the following drawbacks associated with the game theory-based IDS frameworks proposed in the literature:

1. Most of the game theory-based IDS frameworks proposed for wireless networks like MANETs and WSNs assume complete information game settings, wherein each node

in the network is assumed to have a complete information about all the network parameters. This concept enables the nodes to make informed decisions and adopt optimal strategies based on the available resources and network constraints. However, due to the dynamic nature of MANETs and WSNs, nodes in these networks only have partial information about different network parameters. Additionally, many game theory-based IDS frameworks proposed for MANETs and WSNs assume a static game, wherein strategies of the players (attacker and IDS) are assumed to be static and fixed. However, players' strategies in these networks vary dynamically depending on the context of the underlying network parameters. Therefore, IDS frameworks based on static game settings do not perform well in dynamic networks like MANETs and WSNs.

2. Some of the game theory-based IDS frameworks proposed for Ad-hoc and WSNs are geared toward detection of specific class of attacks and cannot be generalized for detection of other type of attacks. This puts a severe limitation on their overall effectiveness in detecting network intrusions.

3. Most cooperative game theory-based IDS frameworks proposed for MANETs and WSNs in literature are characterized by the solution concepts of *Core* and *Shapley values*. However, evaluating these solution concepts become computation intensive, when appropriate optimization mechanisms are not employed.

4. Game theory-based IDS frameworks proposed for VANETs produce significant volume of intrusion detection related traffic. Since, VANETs are characterized by narrow bandwidth wireless radio channels therefore, significant volume of IDS traffic can cause congestion and prevent the flow of normal data traffic in the vehicular network.

In this thesis, we aim to address these issues in the existing IDS frameworks. Towards this end, we propose various game theory-based IDS frameworks to address different IDS related issues like false alarm minimization, minimizing the energy consumption required for operating the IDS, reducing the volume of IDS traffic etc. The proposed game theory-based IDS frameworks model the intrusion detection process as a two player non-cooperative and incomplete information game between the IDS and the attacker. Such game theoretic formulation allows the IDSs to adopt dynamic monitoring strategies based on the history profiles and observed behaviors of nodes being monitored, which greatly enhance their effectiveness and efficiency. The proposed IDS frameworks are shown to achieve high ac-

curacy and detection rate across wide range of network attacks. In addition, the thesis also proposes a game theory-based false alarm minimization scheme for signature based IDS. To the best of our knowledge, this is the first game theory-based false alarm minimization scheme to be proposed.

2.6 Scope and contribution of thesis

The work in this thesis aims to address the following IDS related issues using game theoretic approaches:

- ***False alarm minimization of signature based IDS***: Signature based IDSs are known to produce a large volume of FP alarms when operated with default set of rules, without considering the context of the underlying networks. To address this issue, a game theory-based false alarm minimization is proposed which models the interaction between the attacker and the IDS as a two player non-cooperative game. The NE of the said non-cooperative game is evaluated using various parameters like, attacking and monitoring costs, false alarm rate, detection rate of IDS etc., to develop effective monitoring strategies for IDS. Such game theoretic modeling significantly reduces the number of FP alarms generated by the signature based IDSs, without degrading their overall detection rate.
- ***Minimization of energy consumption required for operating IDS***: Wireless networks like MANETs are characterized by energy and resource constrained nodes. Therefore, continuous monitoring by the nodes operating the IDS can result in their premature death in such networks. To address this issue, a novel Bayesian game theory-based intrusion detection framework for MANET is proposed. It uses a combination of threshold rule based module and a data mining based association rule module to detect wide range of network attacks in MANETs. The proposed IDS framework models the interaction between the attacker (malicious node) and the defender (node operating IDS) as a two player multi-stage, non-cooperative and incomplete information Bayesian game. Such Bayesian game representational model allows the IDS to adopt probabilistic monitoring strategies instead of an 'always on' monitoring strategy by examining the maliciousness history profile of the nodes being monitored and by evaluating the Bayesian Nash Equilibrium of the game. This significantly reduces the energy consumption required for operating the IDS in MANETs, without adversely

affecting the IDS's detection capabilities.

- ***Minimization of the IDS traffic volume***: Wireless networks like VANETs operate in a narrow bandwidth wireless radio spectrum. Therefore, large volume of IDS traffic can cause congestion in these networks. To address this issue, a game theory-based multi-layered intrusion detection framework for VANET is proposed. The proposed IDS framework uses three different monitoring entities namely, agent nodes, Cluster Heads (CHs) and Road Side Units (RSUs) to carry out the intrusion detection task at three different levels of the vehicular network. Additionally, the framework models the interaction between the malicious vehicle (attacker) and the IDS (defender) as a two player non-cooperative game. The NE of the said non-cooperative game is then used to derive a probabilistic IDS monitoring strategy, which significantly reduces the volume of IDS traffic that is introduced into the vehicular network, without degrading the overall performance of the IDS framework.

2.7 Conclusion

This chapter presented a detailed taxonomy of the IDSs. Various IDS frameworks reported in the literature for each class of IDS (signature, event and anomaly based) were then analyzed followed by discussions on the drawbacks associated with them. Subsequently, the chapter provided an introduction to game theory along with description of various type of games namely, non-cooperative, cooperative and cooperation enforcement games. Various game theory-based IDS frameworks proposed in the literature were then discussed followed by the analysis of their drawbacks. Finally, research direction taken in this thesis was pointed out vis-à-vis the drawbacks in existing IDS frameworks, followed by a brief discussion on application of game theory to address these issues.

In the next chapter, we address the issue of high false alarm rate in the signature based IDSs. Signature based IDSs are prone to high false alarm rate with sometimes more than 90% of the alerts being generated by them turning out to be false positives. Therefore, false alarm minimization of signature based IDSs is an important issue that needs to be addressed for enabling their wider acceptance. Towards this end, we propose a novel game theory-based false alarm minimization scheme for a signature based IDS as the first contribution of the thesis.

"Big whorls have little whorls;
Which feed on their velocity,
And little whorls have lesser whorls,
And so on to viscosity
(in the molecular sense)"

Richardson (1922)

3

False Alarm Reduction in Signature based IDS: Game Theory Approach

3.1 Introduction

The distributed and heterogeneous nature of the contemporary networks coupled with the complexity of their underlying communication environment has made network control and security much more challenging than ever before. Traditional preventive mechanisms like data encryption, user authentication, firewall etc., act as the first line of defense against network intrusions. However, these mechanisms have several limitations. For instance, a weak password can render the user authentication ineffective, thereby allowing unauthorized access to the network. Similarly, firewalls are vulnerable to configuration errors and ambiguous security policies. Therefore, additional security mechanism in the form of Intrusion Detection Systems (IDSs) are required to effectively counter network intrusions and complement traditional preventive techniques like data encryption, user authentication, firewall etc.

IDSs are preventive security mechanism that focus on identification of network intrusions and adoption of appropriate counter measures before any significant damage can be inflicted to the network. A class of IDS known as the signature based IDS (also known as misuse based IDS) uses a database of known attack signatures to detect network intrusions. It monitors the network traffic and raise an alarm wherever there is a malicious traffic that

matches with one or more attack signatures in the signature database. Although signature based IDSs are effective in detecting wide range of network attacks, they are prone to high false alarm rate, wherein normal data traffic is incorrectly classified as intrusion by the IDS. Sometimes up to 90% of alarms generated by the signature based IDSs are False Positives (FPs) [59]. Hence, it is necessary to reduce the volume of FP alarms generated by the signature based IDS in order to mitigate the burden on the network administrator who has to manually evaluate each and every IDS alarm and take appropriate measures. High false alarm rate also defeats the very purpose of having an IDS in the first place, as the attacker can launch a DoS attack against the IDS infrastructure itself and render it ineffective.

The most simple and straightforward solution to reduce FP alarms is to manually turn off or deactivate some attack signatures in the database. However, manually turning off the attack signatures can adversely effect the IDS performance by increasing its False Negative (FN) rate and thereby decreasing its detection capabilities. Another approach to minimize the FP alarm rate is to enrich the attack signatures with various network context information parameters like IP addresses, port numbers, protocol details, OS types, etc. However, this approach requires alteration of attack signatures and rebooting of the IDS engine. Therefore, it is not considered to be a feasible option as well. Numerous works on false alarm minimization of signature based IDS have been reported in the literature with varying degree of success [62] [57] [58] [84] [59] [85]. In general, enhancing the IDS's accuracy by minimizing its FP alarm rate also decreases its detection rate by increasing its overall FN rate. However, as the increase in accuracy due to FP alarm reduction is much higher than the related fall in the detection rate, false alarm minimization techniques are considered to be a viable option for increasing the efficiency of signature based IDSs and continues to be researched upon.

In this chapter, we propose a novel game theory-based false alarm minimization scheme for signature based IDS. Game theoretic modeling of intrusion detection process allows the network administrator to dynamically configure the network's security settings based on the underlying network environment and available resources, which significantly reduces the number of FP alarms generated by the signature based IDS.

The rest of the chapter has been structured in following ways. Section 3.2 discusses related works on false alarm minimization in signature based IDS. The drawbacks associated with these works are listed out, which provides the motivation for the work carried out in this chapter. Section 3.3 provides a detailed description of the proposed game theory-based

false alarm minimization scheme for signature based IDS. Section 3.4 presents the performance analysis of the proposed false alarm minimization scheme on the benchmark IDEVAL [86] and testbed datasets to validate its effectiveness in reducing the FP alarms generated by the signature based IDS. Finally, we conclude with conclusion and a brief introduction to Chapter 4 in section 3.5.

3.2 Related Works

Signature based IDSs are known to produce a large number of FP alarms when operated with default settings without considering the underlying network environment they are operating under. Extensive studies on improving and optimizing the performance of signature based IDS through various false alarm minimization techniques have been carried out in the literature [57] [87] [88] [89]. This section discusses various related works on false alarm minimization in signature based IDS and their related drawbacks.

IDS with identification capability, called the IDSIC was proposed by *Pei-Te Chen et al.* [90]. They proposed the concept of security auditors who take charge of discovering the potential system vulnerabilities and modifying the testing packets with fingerprints so that they can be recognized by IDSs. They also introduced an extended IDS with Identification Capability (IDSIC) so that the security tests can work with IDSs. With this capability, the IDSIC can reduce the overall false alarms being generated.

One of the earliest formal treatment for integration of vulnerability context informations with alarms was proposed by *B.Morin et al.* [91].They proposed a data model for IDS alert correlation called M2D2. Their scheme uses reference numbers to identify network vulnerabilities along with other context information parameters to verify TP alarms. However, some IDS alarms do not have a corresponding reference number. Therefore, these alarms with missing reference numbers cannot be correlated with any of the vulnerabilities.

N.Hubballi et al. [57] proposed a method to reduce the FP alarm rate of the IDS without manipulating the default attack signature set (i.e., neither altering the signatures nor turning them off). Their scheme basically correlates IDS alarms with vulnerabilities in the Threat profile. Alarms that cannot be correlated with any network vulnerabilities in the Threat profile are filtered out and marked as FPs. The filtered FP alarms are then correlated with the application being targeted based on the parameters generated with alarm to model the criticality of the applications. If the filtered FP alarm was against a critical application

then it is restored back as TP. The drawback of their approach is that it decreases the detection rate of the IDS due to misclassification of some effective alarms as non-effective by the correlation engine filter.

A scheme to enhance attack signatures with additional context information and thereby augmenting their expressiveness and ability to reduce FP alarms was proposed by *Sommet et al.* [61]. They termed these modified signatures as contextual signatures. These additional contexts are provided as full regular expressions instead of fixed strings, which allow them to correlate multiple interdependent matches. However, signatures are usually complex and modifying them is tedious and prone to error. It also requires the IDS to be put off when the signatures are being updated thus leading to downtime of signature detection systems.

Weizhi Meng et al. [92] proposed a mechanism to enhance the performance of the signature based network intrusion detection systems using an Enhanced Filter Mechanism (EFM). Their proposed mechanism consists of three major components namely, a context-aware blacklist-based packet filter, an exclusive signature matching component and a KNN-based false alarm filter. They showed that their proposed mechanism enhances the overall performance of the signature based IDS. The drawback of their scheme is the computational overhead involved in processing of the same data traffic by multiple processing units.

An architecture for automatic alert verification called ATLANTIDES was proposed by *Bolzoni et al.* [63]. Their proposed architecture consists of a network IDS that monitors the incoming data traffic and an Output Anomaly Detector (OAD), which compares the output traffic with the model it has created during the training phase. To minimize the overall false positive alarms, ATLANTIDES verifies whether the incoming data traffic that raises an alert in the input network IDS actually produces an anomaly in the outgoing traffic too. If it does, then the alert is forwarded to the administrator as a TP alarm, otherwise it is discarded as a false alarm. The correlation engine stores the alarms generated by the network IDS in a hash table for a predefined period before discarding them. This time window is a critical factor that determines the accuracy of the scheme. A small time window leads to misclassification of some attacks because their corresponding alarms were removed from the hash table before correlation can be performed. On the other hand, a large time window may result in an enormous number of false positive alarms being generated.

F. Massicotte et al. [60] proposed a Passive Network Monitoring Tool (PNMT) that is capable of passively acquiring network context information and allowing the inclusion of such context in network intrusion detection rules. Their proposed approach is model-driven

and relies on the modeling of packet and network information as UML class diagrams, and the definition of intrusion detection rules as OCL expressions constraining these diagrams. The drawback of this approach is that it requires manipulation of attack signatures through addition of various network context information parameters like configuration of a node, its operating system, running applications etc. This process is usually prone to error and requires the IDS to be put off when signatures are being updated.

In summary, we found the following drawbacks in our survey of previous works on false alarm minimization in signature based IDS:

- Some schemes propose enhancement to attack signatures with additional context information. However, such schemes are prone to error and require the IDS to be put off while signatures are being updated.
- In some false alarm minimization schemes, only the attacks against the critical applications are taken into account for re-evaluation, while even the severe attacks against non-critical applications are not take into consideration. This allows the attacker to easily compromise the non-critical applications and then launch attacks against critical applications using the compromised non-critical applications.
- Some schemes use the reference numbers to correlate the IDS alarms with the attack signatures. However, some of the alarms generated by the signature based IDS may not have associated reference numbers and therefore they cannot be correlated with any of the network vulnerabilities. This results in some of the potential TP alarms being discarded as FPs thereby, compromising the IDS's detection capabilities.
- The performance of any false alarm minimization scheme depends on the detection rate and accuracy of the vulnerability scanner being used to generate the Threat profile of the network. In our survey, we found that none of the previous works have taken this aspect into consideration while developing the false alarm minimization scheme.

To address these issues in the related works, we propose a novel game theory-based false alarm minimization scheme for signature based IDSs. The proposed scheme uses multiple vulnerability scanners to scan the network and create the network's Threat profile. The Threat profile consists of multiple vulnerability sets, with each set containing one or more network vulnerabilities found during the network scan. Each vulnerability set of the Threat

profile is assigned a unique criticality weight based on the severity of the vulnerabilities contained in it. In addition, the proposed scheme models the interaction between the IDS and the attacker as a two player non-cooperative game. The Nash Equilibrium of the said game is then used to compute the Sensible Vulnerability Set (SVS) of the network. The SVS is a subset of the network's Threat profile and comprises high criticality weight vulnerability sets. Initially, the alarms generated by the signature based IDS are correlated with the vulnerabilities in the Threat profile to identify the potential TP alarms, which are then eventually correlated with the vulnerabilities in the SVS to determine the final TP alarms. The detailed description of the proposed false alarm minimization scheme is discussed in the Section 3.3.

3.3 Proposed false alarm minimization scheme

Signature based IDSs are known to produce large number of FP alarms when operated with default set of rules, without considering the context of the underlying network's environment. In most of the cases, signature based IDSs are deployed with out of the box configuration, without customizing the attack signatures in their signature database. This results in many of the ineffective attacks being treated as potential intrusions by the signature based IDSs, which increase their false alarm rate. High false alarm rate results in massive utilization of the network's resources for performing monitoring operation against ineffective network threats. In addition, high false alarm rate also defeats the very purpose of having an IDS in the first place itself as the administrator has to manually analyze each and every IDS alarms to verify them, which incurs high overhead and is also prone to error. Therefore, it is necessary to reduce the false alarm rate of the signature based IDS to prevent the network administrator from being overwhelmed with inundation of FP alarms.

The overall architecture of the proposed signature based IDS false alarm minimization scheme is shown in Fig. 3.1. A brief description about various components of the proposed false alarm minimization scheme are provided below:

1. **Packet sniffer**: This module captures the incoming network traffic packets for analysis. It is built using the Libpcap or Wincap libraries.

2. **Signature based IDS**: This component correlates the packets captured by the packet sniffer against the attack signatures in the signature database. If the parameters of the captured packets match with that of one or more attack signatures, then alarms

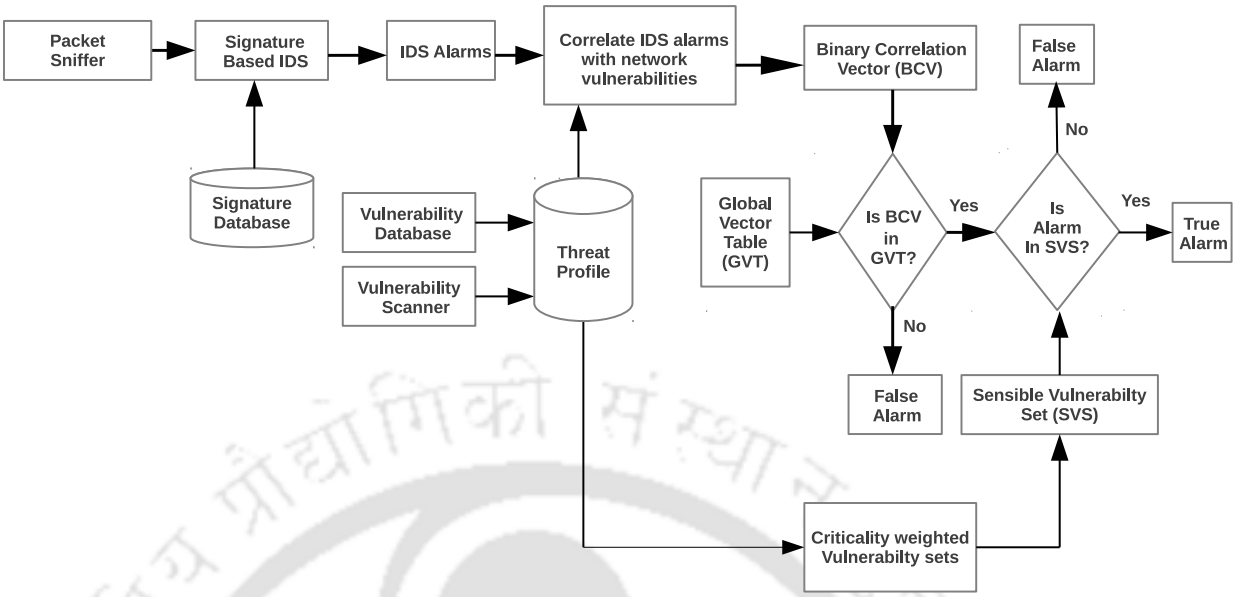

Figure 3.1: Architecture of the proposed false alarm minimization scheme

are raised by the IDS to signal the captured packets as anomalous.

3. **IDS Alarms**: It is the output of signature based IDS and contains both the TP and FP alarms.

4. **Vulnerability database**: This module contains a detailed description about various network vulnerabilities. These vulnerability details are populated from various well known sources such as CVE database [8], Nessus [10], Nmap [9] etc.

5. **Vulnerability Scanner**: Multiple vulnerability scanners are used to scan the network under consideration and obtain an exhaustive list of all possible network vulnerabilities that can be exploited by the attacker.

6. **Threat profile**: The network vulnerabilities identified by the scanners are categorized into different vulnerability sets. These vulnerability sets form the network's Threat profile. Each vulnerability set comprises multiple vulnerabilities found during the network scan and is assigned a unique critical weight based on the severity of vulnerabilities contained in it. Alarms generated by the signature based IDS are correlated with the vulnerabilities in the Threat profile to determine the final TP alarms. A detailed description about the network's Threat profile is provided in sub-subsection 3.3.1.

7. **Binary Correlation Vector**: IDS alarms are correlated with the vulnerabilities in the

Threat profile based on various features like, IP address, port number, protocol type, OS etc., to generate the Binary Correlation Vectors (BCVs). The BCVs are binary strings of predefined length. The positional index in the BCV is set to 0 if there is a match in the corresponding feature value between the IDS alarm and the vulnerability being correlated, else it is set to 1.

8. **Global Vector Table**: The Global Vector Table (GVT) comprises a list of BCVs corresponding to relevant and valid attacks on network vulnerabilities. The BCVs generated after correlating the IDS alarms with the network vulnerabilities in the Threat profile are verified whether they belong to the GVT or not. If they do not belong to the GVT, then the corresponding IDS alarms are dropped as FPs else they are treated as potential TP alarms and forwarded to the next stage for further processing.

9. **Sensible Vulnerability Set**: The Sensible Vulnerability Set (SVS) comprises a subset of high criticality weight vulnerability sets from the network's Threat profile. The SVS is generated by using a game theoretic procedure that takes into account various parameters like detection rate and false alarm rate of the scanner, monitoring and attacking costs, criticality weights of the vulnerability sets etc. IDS alarms that pass the network's Threat profile correlation test are eventually correlated with the vulnerabilities in the SVS to determine the final TP alarms. It can be shown that focusing only on the vulnerabilities in the SVS is enough to maximize the attacker's payoff. Other vulnerabilities outside the SVS are not attractive enough to draw attacker's attention due to their low security asset values. A detailed description about the SVS generation using a game theoretic procedure is provided in sub-section 3.3.3.

In the subsequent sub-sections, we describe the main components of the proposed signature based IDS false alarm minimization scheme namely, the network's Threat profile and the GVT. We also formulate the network intrusion detection problem as a two player non-cooperative game between the network administrator and the attacker. The Nash Equilibrium of the said game is then used to develop the SVS of the network.

3.3.1 Network's Threat profile

In this sub-section, we provide an elaborate discussion about the network's Threat profile generation process. The proposed false alarm minimization scheme uses multiple vulnerability scanners to scan the network for all possible vulnerabilities and create a Threat

Table 3.1: Snapshot of network Threat profile

Vul. Set	IP Address	Ref. No.	Protocol	Port No.	Risk Factor	OS	CriticalWeight
1	172.16.26.251	CVE-1999-0874	TCP	4502	High	Windows	0.9
	172.16.112.149	CVE-2000-0677	TCP	3302	High	Windows	
	172.16.112.100	NessusID :10173	UDP	1503	Low	Windows	
2	172.16.26.249	CVE-1999-0021	TCP	3302	High	SunOS	0.7
	172.16.115.234	NessusID:10360	TCP	1206	Med	Windows	
. . .	. . .	. . .	. . .	. . .	. . .	. . .	. . .
X	172.16.112.207	NessusID :10280	TCP	90	High	Windows	0.08
	172.16.112.50	CVE-1999-0619	TCP	1055	High	Linux	

profile of the network. The Threat profile consists of multiple vulnerability sets with each set containing one or more network vulnerabilities found during the network scan. Each vulnerability set is assigned a unique criticality weight based on the severity of the vulnerabilities contained in that set. The Threat profile provides various information about the hosts present in the network such as, operating system types, open ports, protocol types, applications, criticality weight of vulnerabilities and other external information populated from various vulnerability databases like, Nessus [10], Bugtraq [7], CVE [8], Nmap [9] etc. The snapshot of the network's Threat profile is shown in Table 3.1. Only those IDS alarms which can be correlated with one or more vulnerabilities in the network's Threat profile are considered as potential TP alarms, while others are discarded as FPs.

3.3.2 Global Vector Table

In this sub-section, we provide a detailed description about the Global Vector Table (GVT) of the proposed false alarm minimization scheme. We represent the IDS alarm by an alarm vector $A_i = \langle a_1, a_2, ..., a_n \rangle$ and the vulnerability in the network's Threat profile by a threat vector $T_i = \langle t_1, t_2, ..., t_n \rangle$. Here, a_i and t_i represent the i^{th} feature of the alarm vector (A_i) and threat vector (T_i), respectively. The positional match or mismatch between features of A_i and T_i is represented by the Binary Correlation Vector (BCV) $C = \langle c_1, c_2, ..., c_n \rangle$, where $c_i = 0$, if $a_i == t_i$, else it is set to 1. Alarms generated by the signature based IDS are correlated with vulnerabilities in the network's Threat profile based on their IP addresses and reference numbers. However, some of the IDS alarms may not have an associated reference number. In such cases, the correlation between the IDS alarms and the network vulnerabilities are carried out using other network parameters like, protocol type, port number and OS type.

Table 3.2: Global Vector Table (GVT)

Binary Correlation Vector ⟨ **IP, RefID, Prot, PNo, RF, OS** ⟩	**Description**
⟨ 0, 0, 0, 0, 0, 0 ⟩	*All feature matches*
⟨ 0, 1, 0, 0, 0, 0 ⟩	*Mismatch in RefID*
⟨ 0, 0, 1, 0, 0, 0 ⟩	*Mismatch in Protocol*
⟨ 0, 0, 0, 1, 0, 0 ⟩	*Mismatch in Port No.*
⟨ 0, 0, 0, 0, 1, 0 ⟩	*Mismatch in RF value*

Table 3.3: Snapshot of Alarms generated by IDS

IP Address	**Ref No**	**Prot**	**Port No**	**Risk Factor**	**OS**
172.16.26.251	CVE-1999 -0874	TCP	4502	High	Windows
172.16.26.241	CVE-2000 -0347	TCP	1523	High	Windows
172.16.26.249		TCP	3302	High	SunOS
172.16.26.250	CVE-2000-0677	TCP	1055	High	Linux
172.16.26.243		TCP	90	High	Solaris
172.16.26.241		TCP	1503	Low	Windows
172.16.26.244	CVE-2002-0012	TCP	1402	Med	Redhat
172.16.26.245	CVE-1999-0197	TCP	1206	Med	Windows
172.16.26.249	CVE-1999-0127	TCP	3302	High	SunOS

The GVT of the network is defined as a set of BCVs corresponding to relevant and valid attacks on network vulnerabilities. If the BCV obtained after correlating a given IDS alarm vector with a threat vector has an entry in the GVT, then the corresponding IDS alarm is marked as a potential TP alarm. Table 3.2 shows the GVT with a predefined list of BCVs. The parameters ⟨ *IP, RefID, Prot, PNo, RF, OS* ⟩ in the table are abbreviations for IP address, Reference ID, Protocol, Port Number, Risk factor and Operating System, respectively. In general, the IDS alarms are flagged as potential TP under following two conditions:

1. IP address and OS type of the IDS alarm and vulnerability being correlated matches.
2. There is at most one mismatch in any other parameters (RefID, Prot, PNo, and RF) being correlated.

For instance, consider the 1^{st} alarm (A_1) of Table 3.3. Its IP address matches with that of the 1^{st} vulnerability (V_1) in the 1^{st} vulnerability set of the Threat profile in Table 3.1. The BCV obtained after correlating the features of A_1 and V_1 is ⟨ 0, 0, 0, 0, 0, 0 ⟩. This BCV corresponds to one of the binary vectors in GVT of Table 3.2. Therefore, the corresponding alarm A_1 is flagged as potential TP.

Next, consider the BCV ⟨ 0, 1, 0, 0, 0, 0 ⟩. This BCV can be generated under following two conditions: (i) when the RefID parameter is missing in the IDS alarm being correlated, (ii) when there is a mismatch in the RefID parameter between the IDS alarm and the vulnerability being correlated. The IDS alarm corresponding to the 1^{st} case could be for one of the valid vulnerability in the Threat profile. However, the IDS alarm corresponding to the 2^{nd} case cannot be for any valid vulnerability in the Threat profile. Treating the alarm corresponding to the 2^{nd} case as a TP increases the false alarm rate of the IDS. Therefore, to resolve this issue, whenever there is a mismatch in RefID parameter between the IDS alarm and the vulnerability being correlated, all other preceding parameters in the BCV are set to one. For example when the 3^{rd} alarm in Table 3.3 is correlated with the 1^{st} vulnerability of 2^{nd} set in Table 3.1, the corresponding generated BCV is ⟨ 0, 1, 0, 0, 0, 0 ⟩ but when the 9^{th} alarm in Table 3.3 is correlated with the same vulnerability in Table 3.1, the generated BCV is ⟨ 0, 1, 1, 1, 1, 1 ⟩.

IDS alarms that are marked as potential TPs after correlating them with vulnerabilities in the network's Threat profile are examined whether they belong to the SVS or not. If they belong to the SVS then they are considered as final TP alarms and are forwarded to the network administrator who takes appropriate measures to address them. However, if they do not belong to the SVS then they are discarded as FP alarms. This two layered correlation process is shown to significantly reduce the number of false positive alarms generated by the signature based IDS, without adversely affecting its detection rate.

3.3.3 Intrusion detection game model to generate the Sensible Vulnerability Set

In this sub-section, we provide a detailed description about the Sensible Vulnerability Set (SVS) of the proposed false alarm minimization scheme. The SVS is a subset of network's Threat profile and consists of high critical weight vulnerability sets. Alarms generated by the signature based IDS are initially correlated with vulnerabilities in the Threat profile to identify the potential TP alarms. The IDS alarms that pass the Threat profile correlation test are eventually correlated with vulnerabilities in the SVS to determine the final TP alarms. This two layered correlation process filters out most of the alerts generated from unsuccessful attacks and low priority attacks, which do not require immediate attention of the network administrator and hence, significantly reduces the overall false positive alarms generated by the signature based IDS.

We use a game theoretic formulation to determine the SVS of the network. Towards

this end, we formulate the intrusion detection problem as a two player non-cooperative game between the network administrator (defender) and the attacker. Game theory allows modeling situations of conflict between competing players and predict the best action for individual player by identifying the stable outcome of the game through Nash Equilibrium analysis. We make an implicit assumption that both the attacker and the defender players are rational decision makers, i.e., given set of strategies, both the players will always choose a strategy that maximize their overall payoff utilities.

Consider a network $N_t = \langle S_D, S_A \rangle$, where S_D and S_A are the network's vulnerability sets and the alarm sets, respectively. $S_D = \langle V_1, V_2, V_3, ..., V_X \rangle$ is the network's Threat profile consisting of multiple vulnerability sets. Each vulnerability set $V_i \in S_D$ is assigned a unique criticality weight based on the severity of vulnerabilities contained in it and consists of a one or more network vulnerabilities discovered during the network scan. $S_A = \langle A_1, A_2,...., A_n \rangle$ is the alarm set generated by the signature based IDS in response to the attacks against various network vulnerabilities. The main objective of the attacker is to exploit the vulnerabilities in the high critical weight vulnerability sets of S_D, without being detected by the IDS. To achieve this objective, the attacker probes the network for possible loopholes and also makes some educated guess about various vulnerabilities present in the network. It then chooses a mixed attack strategy $\boldsymbol{p} = \langle p_1, p_2, ..., p_X \rangle$, which is the attacker's attack probability distribution over the X vulnerability sets in S_D, such that $\sum_{i=1}^{X}(p_i) \leq P \leq 1$.

On the other hand, to counter the attacks against various network vulnerabilities in S_D, the defender uses the Threat profile information to develop its monitoring strategies. The defender allocates network resources for monitoring the vulnerabilities in S_D with the probability distribution $\boldsymbol{q} = \langle q_1, q_2, ..., q_X \rangle$, where q_i is the probability of monitoring vulnerabilities in the vulnerability set $V_i \in S_D$, such that $\sum_{i=0}^{X}(q_i) \leq Q \leq 1$.

Table 3.4 shows the payoff matrix of the attacker and the IDS (defender) interacting over the network vulnerability $v_i \in V_i$ in the strategic form. The idea behind the formulation of this payoff matrix has been borrowed from [93]. In the payoff matrix, *'a'* and *'b'* denotes the detection rate and the false alarm rate of the vulnerability scanner, respectively ($0 \leq a, b \leq 1$). The cost of attacking and monitoring the vulnerability v_i is assumed to be proportional to the criticality weight of the vulnerability set V_i $\big(W_{cr}(i)\big)$. Accordingly, the attacking and monitoring costs are denoted by $C_a W_{cr}(i)$ and $C_m W_{cr}(i)$, respectively, where C_a and C_m are the costs associated with attacking and monitoring the vulnerability v_i. C_f denotes the cost associated with false alarm and $C_f W_{cr}(i)$ denotes the defender's cost (energy and

resources cost) due to false alarm. We make an implicit assumption that the values of both C_a and C_m are relatively less than that of $W_{cr}(i)$, otherwise the attacker and the defender will have no profitable incentive to exploit and defend the vulnerability v_i.

Table 3.4: Strategic form of the game for vulnerability v_i

	Defend	**Not-Defend**
Attack	(1 - 2*a*)$W_{cr}(i)$ - $C_aW_{cr}(i)$, - (1 - 2*a*)$W_{cr}(i)$ - $C_mW_{cr}(i)$	$W_{cr}(i)$ - $C_aW_{cr}(i)$, -$W_{cr}(i)$
Not-Attack	0, -$bC_fW_{cr}(i)$ - $C_mW_{cr}(i)$	0, 0

If the attacker plays its strategy *"attack"* with probability distribution (***p***) and the defender plays its strategy *"monitor"* with probability distribution (***q***), then the overall payoff utilities of the attacker and the defender are given by the utility functions U_A(***p, q***) and U_D(***p, q***), respectively, where

$$U_A(\boldsymbol{p, q}) = \sum_{i=1}^{X} p_i q_i \Big[(1-2a)W_{cr}(i) - C_aW_{cr}(i)\Big] + p_i(1-q_i)\Big[W_{cr}(i) - C_aW_{cr}(i)\Big] \tag{3.1}$$

$$U_D(\boldsymbol{p, q}) = \sum_{i=1}^{X} p_i q_i \Big[-(1-2a)W_{cr}(i) - C_mW_{cr}(i)\Big] - p_i(1-q_i)W_{cr}(i) - (1-p_i)q_i\Big[bC_fW_{cr}(i) + C_mW_{cr}(i)\Big] \tag{3.2}$$

The intrusion detection game with an attacker and the defender player is defined by the following parameters:

- **Players**: Attacker, Defender.
- **Vulnerability set**: X vulnerability set $\langle V_1, V_2, \ldots, V_X \rangle$, where each set V_i consists of one or more network vulnerabilities.
- **Game strategy**: Strategy set of the attacker (A_A) and the defender (D_D) interacting over X network vulnerability set given by:

$$A_A = [\ \mathbf{p} : \mathbf{p} \in [0, P]^X, \sum_{i=0}^{X} p_i \leq P\ ]$$

$$D_D = [\ \mathbf{q} : \mathbf{q} \in [0, Q]^X, \sum_{i=0}^{X} q_i \leq Q\ ]$$

- **Payoff Utility**: U_A for attacker, U_D for defender.
- **Game rule**: The attacker and the defender player adopts strategies that maximize their respective payoff utilities U_A and U_D.

In the said non-cooperative intrusion detection game between the attacker and the defender player, Nash equilibrium (NE) corresponds to the steady state of the game in which no player has any profitable incentive to deviate from its current strategy while the other player keeps its strategy fixed. The strategy profile $S = \langle p^*, q^* \rangle$ is said to be the NE of the game if neither the attacker nor the defender player can improve their payoff utilities by unilaterally deviating from the NE strategy S. In the subsequent sub-section, we develop the NE strategy for the proposed non-cooperative game between the attacker and the defender player. We then employ the NE strategy to develop the Sensible Vulnerability Set (SVS) of the network and minimize the overall false alarm rate of the signature based IDS.

3.3.4 Sensible Vulnerability Set

In the proposed false alarm minimization scheme, multiple vulnerability scanners are used to obtain an exhaustive list of all possible network vulnerabilities. The vulnerabilities detected by the scanners are categorized into N different sets with each set containing one or more vulnerabilities to form the Threat profile of the network. Each vulnerability set is assigned a unique criticality weights ($W_{cr}(i)$) based on the severity of the vulnerabilities contained in it. The network's Threat profile vulnerability sets are sorted based on their criticality weights. Therefore, $W_{cr}(1) \geq W_{cr}(2) \geq ... \geq W_{cr}(N)$, where $W_{cr}(i)$ is the criticality weight of the i^{th} vulnerability set. Given a network with N vulnerability sets, we define the sensible vulnerability set (S_s) and quasi-sensible vulnerability set (S_q) of the network as follows:

$$\begin{cases} W_{cr}(i) > \frac{|S_s|(1-C_a)-2aQ}{(1-C_a)(\sum_{j\in S_s}\frac{1}{W_{cr}(j)}))}, & \forall\, i \in S_s \\ W_{cr}(i) = \frac{|S_s|(1-C_a)-2aQ}{(1-C_a)(\sum_{j\in S_s}\frac{1}{W_{cr}(j)}))}, & \forall\, i \in S_q \\ W_{cr}(i) < \frac{|S_s|(1-C_a)-2aQ}{(1-C_a)(\sum_{j\in S_s}\frac{1}{W_{cr}(j)}))}, & \forall\, i \in N \text{ - } S_s \text{ - } S_q \end{cases} \tag{3.3}$$

where Q is the defender's monitoring probability distribution over N vulnerability sets of the network's Threat profile, | S_s | is the cardinality of the set S_s and N - S_s - S_q are the set of vulnerabilities in N but neither in S_s nor in S_q.

Lemma 1 : *Given a network with N vulnerability sets, both S_s and S_q can uniquely be determined. S_s consists of N_A vulnerability sets ($N_A \subseteq$ N) with the largest criticality weights such that if* :

1. $W_{cr}(N) > \frac{N(1-C_a)-2aQ}{(1-C_a)(\sum_{i=1}^{N}\frac{1}{W_{cr}(i)})}$, then $N_A = N$, $S_q = \emptyset$
2. $W_{cr}(N) \leq \frac{N(1-C_a)-2aQ}{(1-C_a)(\sum_{i=1}^{N}\frac{1}{W_{cr}(i)})}$, then N_A is determined by the following equations:

$$\begin{cases} W_{cr}(N_A) > \frac{N_A(1-C_a)-2aQ}{(1-C_a)(\sum_{k=1}^{N_A}\frac{1}{W_{cr}(k)})} \\ W_{cr}(N_{A+1}) \leq \frac{N_A(1-C_a)-2aQ}{(1-C_a)(\sum_{k=1}^{N_A}\frac{1}{W_{cr}(k)})} \end{cases} \tag{3.4}$$

and S_q consists of vulnerabilities from the vulnerability set S_i such that:

$$W_{cr}(S_i) = \frac{N_A(1-C_a)-2aQ}{(1-C_a)(\sum_{k=1}^{N_A}\frac{1}{W_{cr}(k)})}$$

The proof of ***Lemma 1*** has been adopted from [93] and provided below.

Proof: The proof consists of showing that S_s comprises of *n* vulnerability sets with the largest criticality weights and then proving that $n = N_A$ by showing that neither $n < N_A$ nor $n > N_A$ is possible.

Case 1 of ***Lemma 1*** can be proven straightforwardly. Here, we prove the 2^{nd} case of the ***Lemma 1***. The N_A vulnerability sets with largest criticality weights satisfying Equation

(3.4) consists of a sensible vulnerability set S_s and Equation (3.3) holds in such a case. We need to show that the set S_s can be determined uniquely.

We first show that if the vulnerability set $i \in S_s$, then $\forall j < i$ ($W_{cr}(j) \geq W_{cr}(i)$); it holds that $j \in S_s$. If not, $\exists\ j_o < i$ ($W_{cr}(j_o) \geq W_{cr}(i)$), such that $j_o \in N$ - S_s. It then implies that $W_{cr}(j_o)$ $\leq$ ($|S_s|$.(1-C_a)-2aQ)/((1-C_a)$\sum_{k \in S_s}(1/W_{cr}(k))$). But from Equation 3.3, we have $W_{cr}(i)$ $>$ ($|S_s|$.(1-C_a)-2aQ)/((1-C_a)$\sum_{k \in S_s}(1/W_{cr}(k))$). It implies that $W_{cr}(i) > W_{cr}(j_o)$, which contradicts with the assumption that $W_{cr}(j_o) \geq W_{cr}(i)$. Hence S_s consists of n vulnerability sets with the largest criticality weights. We then show that $n = N_A$ and it is not possible that $n < N_A$ or $n > N_A$. If $n < N_A$, then from Equation (3.4), we have :

$$
\begin{aligned}
W_{cr}(N_A) &> \frac{N_A(1-C_a) - 2aQ}{(1-C_a)(\sum_{k=1}^{N_A} \frac{1}{W_{cr}(k)})} \\
\implies W_{cr}(N_A)\left(\sum_{k=1}^{N_A} \frac{1}{W_{cr}(k)}\right) &> \frac{N_A(1-C_a) - 2aQ}{(1-C_a)} \\
\implies W_{cr}(N_A)\left(\sum_{k=1}^{N_A} \frac{1}{W_{cr}(k)}\right) - (N_A - n) &> n - \frac{2aQ}{1-C_a}
\end{aligned}
$$

Noticing that $W_{cr}(N_A) \leq W_{cr}(i), \forall\, i \leq N_A$ and since $n < N_A$ (i.e. $W_{cr}(n+1) \geq W_{cr}(N_A)$), we have:

$$
\begin{aligned}
W_{cr}(n+1)\left(\sum_{j=1}^{n} \frac{1}{W_{cr}(j)}\right) &\geq W_{cr}(N_A)\left(\sum_{j=1}^{n} \frac{1}{W_{cr}(j)}\right) \\
\geq W_{cr}(N_A)\left(\sum_{j=1}^{N_A} \frac{1}{W_{cr}(j)}\right) &- W_{cr}(N_A)\left(\sum_{j=n+1}^{N_A} \frac{1}{W_{cr}(j)}\right) \\
&\geq W_{cr}(N_A)\left(\sum_{j=1}^{N_A} \frac{1}{W_{cr}(j)}\right) - (N_A - n) \\
&> n - \frac{2aQ}{(1-C_a)}. \\
\implies W_{cr}(n+1) &> \frac{n(1-C_a) - 2aQ}{(1-C_a)\sum_{j=1}^{n} \frac{1}{W_{cr}(j)}}
\end{aligned}
$$

But from Equation (3.4), we have $W_{cr}(n+1) \leq$ *(n*(1 - C_a) - 2aQ)*/((1 - C_a)$\sum_{j=1}^{n}(1/W_{cr}(j))$). This contradiction shows that it is not possible that $n < N_A$. Similarly, we can show that it is not possible that $n > N_A$. Hence, $n = N_A$ is uniquely determined, and so is S_s. It logically follows that S_q can also be uniquely determined. This concludes the proof of ***Lemma 1***. ■

We now state the following Theorems:

- ***Theorem 1***: The attacker has no incentive to attack any vulnerabilities in the set N - S_s - S_q.
- ***Theorem 2***: The defender only needs to monitor the vulnerabilities in the set S_s + S_q to improve its overall payoff U_D.

Proof Theorem 1: Let $\mathbf{p} \in A_A$ be the attacker's strategy such that $\exists$ i $\in N$ - S_s - S_q with $p_i > 0$. Let $\mathbf{q} \in D_D$ be the defender's strategy and let $\mathbf{p^*}$ be another strategy of attacker such that $p_i^* = 0$, $\forall$ i $\in N$ - S_s - S_q. The proof of theorem lies in showing that $U_A(\mathbf{p},\mathbf{q}) < U_A(\mathbf{p^*},\mathbf{q})$

.

If $W_{cr}(N) \geq \frac{N_A(1-C_a)-2aQ}{(1-C_a)(\sum_{k=1}^{N_A}\frac{1}{W_{cr}(k)})}$, then N - S_s - $S_q = \emptyset$ and the theorem holds evidently. We now need to prove that the theorem holds for the case when:

$W_{cr}(N) < \frac{N_A(1-C_a)-2aQ}{(1-C_a)(\sum_{k=1}^{N_A}\frac{1}{W_{cr}(k)})}$, i.e. when N - S_s - $S_q \neq \emptyset$.

Consider the defender's strategy $\mathbf{q^*} = \langle\, q_1^*, q_2^*, ..., q_N^* \,\rangle$, where

$$q_i^* = \begin{cases} \frac{1}{2a}\left(1 - C_a - \frac{N_A(1-C_a)-2aQ}{\left((1-C_a)W_{cr}(i)(\sum_{k=1}^{N_A}\frac{1}{W_{cr}(k)}\right)}\right), & \text{if } i \in S_s \\ 0, & \text{if } i \in N \text{ - } S_s \end{cases}$$

It holds that $q_i^* \geq 0$ and $\sum_{i=1}^{N_A}(q_i^*) = Q$. Let $\mathbf{q} = \langle\, q_1, q_2, ..., q_N \,\rangle$ be the monitoring probability distribution of the defender over N vulnerability sets. By Pigeon Hole Principle, it holds that $\sum_{i=1}^{N_A}(q_i) \leq Q$. Thus $\exists\ m \in S_s$ such that $q_m \leq q_m^*$. Now consider any attack strategy $\mathbf{p} = \langle\, p_1, p_2, ..., p_N \,\rangle \in A_A$ satisfying $\sum(p_i) > 0$ with $i \in N$ - S_s - S_q i.e., the attacker attacks at least one target outside the sensible set S_s with non-zero probability. Let $\mathbf{p^*} = \langle\, p_1^*, p_2^*, ..., p_N^* \,\rangle$ be another attacker strategy profile based on $\mathbf{p}$ such that

$$p_i^* = \begin{cases} p_i\ , & i \in S_s \text{ and } i \neq m \\ p_m + \sum\limits_{j \in\ N - S_s - S_q} p_j\ , & i = \text{m} \\ p_i\ , & i \in S_q \\ 0, & i \in N \text{ - } S_s \text{ - } S_q \end{cases}$$

Now, noticing that $W_{cr}(i) < \frac{(N_A(1-C_a)-2aQ)}{((1-C_a)\sum_{j=1}^{N_A}\frac{1}{W_{cr}(j)})}$, $\forall\, i \in N$ - S_s - S_q. and comparing attacker's

payoff with strategy **p** and strategy **p*** we get:

$$
\begin{aligned}
U_A(\mathbf{p}) - U_A(\mathbf{p}*) &= \sum_{i\in N} p_i W_{cr}(i)(1-2aq_i - C_a) \;-\; \sum_{i\in N} p_i^* W_{cr}(i)(1-2aq_i - C_a) \\
&= \sum_{i\in N} p_i W_{cr}(i)(1-2aq_i - C_a) \;-\; \Big[\sum_{i\in S_s+S_s, i\neq m} p_i W_{cr}(i)(1-2aq_i - C_a) \\
&\quad + (p_m \;+ \sum_{i\in N-S_s-S_q} p_i) W_{cr}(i)(1-2aq_m - C_a)\Big] \\
&= \sum_{i\in N} p_i W_{cr}(i)(1-2aq_i - C_a) \;-\; \Big[\sum_{i\in S_s+S_s, i\neq m} p_i W_{cr}(i)(1-2aq_i - C_a) \\
&\quad + \Big(p_m \;+ \sum_{i\in N-S_s-S_q} p_i\Big) W_{cr}(i)(1-2aq_m - C_a)\Big] \\
&= \sum_{i\in N-S_s-S_q} p_i W_{cr}(i)(1-2aq_i - C_a) \;-\; \sum_{i\in N-S_s-S_q} p_i W_{cr}(m)(1-2aq_m - C_a) \\
&\leq \sum_{i\in N-S_s-S_q} p_i W_{cr}(i)(1-2aq_i - C_a) \;-\; \sum_{i\in N-S_s-S_q} p_i W_{cr}(m)(1-2aq_m^* - C_a) \\
&= \sum_{i\in N-S_s-S_q} p_i W_{cr}(i)(1-2aq_i - C_a) \;-\; \sum_{i\in N-S_s-S_q} p_i \frac{N_A(1-C_a) - 2aQ}{(1-C_a)\sum_{k=1}^{N_A}\frac{1}{W_{cr}(k)}} \\
&\leq \sum_{i\in N-S_s-S_q} p_i W_{cr}(i) \;-\; \sum_{i\in N-S_s-S_q} p_i \frac{N_A(1-C_a) - 2aQ}{(1-C_a)\sum_{k=1}^{N_A}\frac{1}{W_{cr}(k)}} \\
&= \sum_{i\in N-S_s-S_q} p_i \Big(W_{cr}(i) \;-\; \frac{N_A(1-C_a) - 2aQ}{(1-C_a)\sum_{k=1}^{N_A}\frac{1}{W_{cr}(k)}}\Big) \\
&< 0.
\end{aligned}
$$

Hence the attack strategy **p*** provides the attacker more payoff than the strategy **p**. Therefore, the rational attacker has no incentive to choose **p** over **p***. ■

Theorem 1 shows that focusing only on targets in the sets S_s and S_q is enough to maximize the attacker's payoff. Other targets in the set *N* - S_s - S_q are not attractive enough to draw attacker's attention due to their low security asset values. The proof of ***Theorem 2*** logically follows from proof of ***Theorem 1***.

Let **p*** and **q*** be the attack and monitor probability distribution of the attacker and defender over the vulnerability set (S_s + S_q), respectively. The Nash Equilibrium (NE) of the non-cooperative game between the attacker and defender corresponds to the strategy profile **(p*, q*)**, such that if the attacker change its attack strategy to **p** from **p***, while the

defender maintain its strategy as **q***, then the attacker gets the same payoff by attacking any other monitored vulnerabilities in the vulnerability set (S_s + S_q) and gets less payoff by attacking any non-monitored vulnerabilities in the set (N - S_s - S_q). Similarly, if the defender unilaterally deviates from the NE strategy by changing its monitoring strategy from **q*** to q, then it gets same payoff by monitoring any other vulnerabilities in the set (S_s + S_q) and gets less payoff by monitoring other vulnerabilities in the set (N - S_s - S_q).

3.4 Performance Analysis

In this section, we analyze the performance of the proposed false alarm minimization scheme. We have used the benchmark DARPA Intrusion Detection Evaluation (IDEVAL) dataset [86] and an in-house testbed dataset to evaluate the performance of the proposed false alarm minimization scheme. Snort [11] with default configuration and all its attack signatures enabled was used as the signature based IDS to detect network intrusions. Snort rule set deployed for the evaluation was VRT certified rules for Snort v2.8. Experimental results on both the IDEVAL and the testbed dataset validate that the proposed scheme significantly reduces the false alarm rate of the signature based IDS without degrading its overall detection rate.

3.4.1 Analysis on the IDEVAL dataset

To analyze and evaluate the performance of the proposed framework on the IDEVAL dataset, the fourth and the fifth week's IDEVAL test dataset consisting of 201 instances of about 56 types of attacks were used. A detailed description about the IDEVAL dataset can be found in [54]. According to the documentation of the IDEVAL network, there are a total of 34 internal hosts that were subjected to various type of attacks. Multiple vulnerability scanners like CVE [8], Bugtraq [7] and Nessus [10] were used to scan the IDEVAL network's host operating systems and create the Threat profile of the network. Table 3.5 shows the sample snapshot of the IDEVAL network's Threat profile. Vulnerabilities in the Threat profile are grouped into 10 different vulnerability sets. Each vulnerability set comprises multiple vulnerabilities found during the network scan and is assigned a unique criticality weight based on the severity of the vulnerabilities contained in it (CVSS scores and risk factor values). Vulnerabilities in the vulnerability set with high criticality weights are extremely severe, which upon successful exploitation can inflict extensive damage to the network. On

the other hand, vulnerabilities in the vulnerability set with low criticality weights are less severe and may not cause any extensive damage to the network even if they are exploited successfully by the attacker. The criticality weight ($W_{cr}(i)$) of the vulnerability sets are set according to following formula:

$$W_{cr}(i+1) = \frac{10-i}{10}, \quad where \;\; i = 0, 1, 2, 3, ..., 9$$

Alarms generated by the signature based IDS are initially correlated with vulnerabilities in the Threat profile using the procedure described in Section 3.3 to determine the potential TP alarms. The game theoretic procedure discussed in sub-section 3.3.3 is then used to determine the Sensible Vulnerability Set (SVS) of the IDEVAL network. The potential TP alarms obtained after correlating the IDS alarms with the vulnerabilities in the Threat profile are eventually verified whether they correspond to any of the network vulnerabilities in the SVS. If they correspond to the vulnerabilities in the SVS, they are forwarded to the network administrator as TP alarms for appropriate actions else they are discarded as FP alarms.

We performed a numerical analysis on two typical scenarios (1^{st} case and 2^{nd}case). We first consider a network with a high requirement on security, e.g., a military network that usually requires a high level of confidentiality (1^{st} case). Such network needs to be resistant against most vulnerabilities and therefore requires the detection rate of the vulnerability scanner to be very high. In such a scenario, the critical weights of the network vulnerabilities ($W_{cr}(i)$) are much higher than the related cost of monitoring and attacking them i.e., C_a, $C_m \ll W_{cr}(i)$. Accordingly, we set C_a= C_m = 0.001. In such high security network, the defender is usually equipped with more than one vulnerability scanners and hence can detect most of the vulnerabilities present in the network. Therefore, we set small values for FP alarm rate (b) and FP alarm cost (C_f) i.e., b = 0.06 and C_f = 0.01. We also choose a relatively large value of scanner's detection rate (a = 0.95) for this case.

In the second scenario (2^{nd}case), we consider a wireless LAN network where the attack and monitoring costs are important (C_a = C_m = 0.10). This scenario corresponds to the case where both the attacker and the defender are highly constrained in terms of their energy resources. Therefore, a high cost is associated with attacking and monitoring vulnerabilities in such a network. Accordingly, relatively high values are set for false alarm rate (b = 0.25) and cost due to false alarm (C_f = 0.20). The monitoring nodes in such networks are not very efficient, as they only have access to a limited number of vulnerability scanners. Hence, a low value is associated with the scanner's detection rate (a = 0.45).

Table 3.5: Threat profile snapshot of IDEVAL dataset

Vulnerability Set	IP Address	Ref No.	Protocol	Port No.	CVSS Score /Risk Factor	Criticality Weight	OS Type
1	172.16.113.105	CVE-2000-0677	TCP	23,25,110	10	1.00	RedHat 5.0
	172.16.112.100	CVE-1999-0874	TCP	21,25,139	10		Windows NT 4.0
	172.16.112.100	CVE-1999-0509	TCP	21,25,139	10		Windows NT 4.0
	...	...	...	...	..		..
	172.16.112.149	CVE-2000-0677	TCP	20,23,25,80	10		Redhat 5.0
2	172.16.112.100	NessusID :10173	TCP	21,22,139	Severe	0.90	Windows NT 4.0
	172.16.113.50	CVE-1999-0197	TCP	21,25,80	10		SunOS 4.1.4
	...	...	...	...	...		...
	172.16.112.207	CVE-1999-0146	TCP	23,25,110,143	7.5		SunOS 4.1.4
3	172.16.112.194	CVE-1999-0149	TCP	23,25,110,143	7.5	0.80	Solaris 2.5.1
	...	...	...	...	...		...
	172.16.112.50	NessusID:11032	TCP	20,21,23	High		Solaris 2.5.1
4	172.16.116.201	CVE-1999-0021	TCP	254,80	7.5	0.70	Windows 95
	172.16.113.50	BugTraqID:4132	TCP	23,25,110,143	High		SunOS 4.1.4
	...	...	...	...	...		...
	172.16.112.207	NessusID:11395	TCP	23	Medium		SunOS 4.1.4
5	172.16.115.234	NessusID:10360	TCP	23,25,80,110	High	0.60	Windows NT 4.0
	172.16.116.194	CVE-1999-0191	TCP	23,25,50,110	6.4		Windows 95
	...	...	...	...	...		...
	172.16.116.194	Bugtraq ID: 1448	TCP	23,25,80,110	High		Windows 95
6	172.16.112.20	CVE-1999-0146	TCP	53	7.5	0.50	Redhat 5.0
	...	...	...	...	...		...
	172.16.112.149	CVE-1999-0149	TCP	80	7.5		Redhat 5.0
7	172.16.113.50	CVE-2000-0915	TCP	80	5.0	0.40	SunOS 4.1.4
	172.16.113.84	CVE-2001-0731	TCP	80	5.0		SunOS 4.1.4
	...	...	...	...	...		...
	172.16.112.194	CVE-2000-1036	TCP	23,25,80,110	5.0		Solaris 2.5.1
8	172.16.112.100	NessusID :10360	TCP	21,25,53	High	0.30	Windows NT 4.0
	172.16.112.100	CVE-1999-0191	TCP	21,25,80	6.4		Windows NT 4.0
	172.16.116.194	CVE-2000-0347	TCP	23,25,80,110	5.0		Windows 95
	...	...	...	...	...		...
9	172.16.112.194	CVE-2000-0382	TCP	23,25,50,110	2.6	0.20	Solaris 2.5.1
	...	...	...	...	...		...
	172.16.112.207	CVE-1999-0105	TCP	23,25,110,143	2.1		SunOS 4.1.4
10	172.16.112.010, 172.16.112.207, 172.16.113.50, 172.16.113.84	NessusID :10280	TCP	23	Low	0.10	SunOS
	172.16.112.20, 172.16.112.50, 172.16.112.149, 172.16.112.194	CVE-1999-0619	TCP	23	1.0		Redhat 5.0 Solaris 2.5.1

Table 3.6: Snapshot of alarms generated on IDEVAL dataset by Snort

IP Address	Ref No.	Prot	Port No.	CVSS Score /Risk Factor	OS
172.16.112.149	CVE-2000-0677	TCP	25,80	10	Redhat 5.0
172.16.112.100	NessusID :10173	TCP	22,139	Severe	Windows
172.16.112.207		TCP	25,110,	7.5	SunOS 4.1.4
172.16.113.50	CVE-1999-0021	TCP	254,80	7.5	SunOS 4.1.4
172.16.112.207	NessusID:11395	TCP	23	Medium	SunOS 4.1.4
172.16.116.194	CVE-2000-0347	TCP	25,80	5.0	Windows 95
...	...	...	...	...	...

Table 3.7: Performance of proposed framework on IDEVAL dataset

Scheme	Critical Vulnerability		Non-Critical Vulnerability	
	Accuracy (%)	Detection Rate (%)	Accuracy (%)	Detection Rate (%)
Snort	83.24	37.47	71.31	35.67
Proposed	97.85	37.47	95.56	32.43

Using the procedure described in sub-section 3.3.3, we found that the SVS for the 1^{st} case comprises vulnerabilities from the top 6 vulnerability sets of Table 3.5 with criticality weights in the range 0.50 to 1.0. Table 3.6 shows the snapshot of alarms raised by Snort on the IDEVAL dataset. Correlating the 1^{st} alarm (A_1) of Table 3.6 with the last vulnerability of category 1 in Table 3.5, produces the Binary Correlation Vector (BCV) $\langle$ 0, 0, 0, 0, 0, 0 $\rangle$. This BCV belongs to the Global Vector Table (GVT) in Table 3.2. Therefore, the alarm A_1 is flagged as potential TP alarm. On verifying the criticality weight of the vulnerability corresponding to the alarm A_1 in the Threat profile (Table 3.5), we found that it belongs to the vulnerability set in the SVS. Therefore, the alarm A_1 is declared as TP alarm and forwarded to the network administrator for further actions.

Fig. 3.2 and Fig. 3.3 show the defender's and attacker's payoff under different strategies for the 1^{st} case. It can be observed from Fig. 3.2 that when the attacker employs its best strategy by attacking only the vulnerabilities in the SVS, the defender's best response is to monitor only the vulnerabilities in the SVS. Similarly, from Fig. 3.3 it can be observed that the attacker's best response when the defender plays its best strategy by defending only the vulnerabilities in the SVS is to attack only the subset of vulnerabilities in the SVS. Therefore, the Nash Equilibrium (NE) for the 1^{st} case corresponds to the strategy combination, where both the defender and the attacker defends and attacks only the vulnerabilities in the SVS with a certain probability distributions. Any unilateral deviation by either the attacker or the defender from this NE strategy results in the degradation of the payoff for the deviating player.

For the 2^{nd} case, the SVS consist of the top 4 vulnerability sets of Table 3.5 with criticality weights in the range 0.70 to 1.0. Fig. 3.4 and Fig. 3.5 show the defender's and attacker's payoff for the 2^{nd} case, respectively. It can be observed from these figures that in this case too, the optimal strategy for both the defender and the attacker is to monitor and attack the vulnerabilities in the SVS with a certain probability distribution.

Table 3.7 shows the performance of the proposed false alarm minimization framework on the IDEVAL dataset (using parameters from the 1^{st} case). It was found that the perfor-

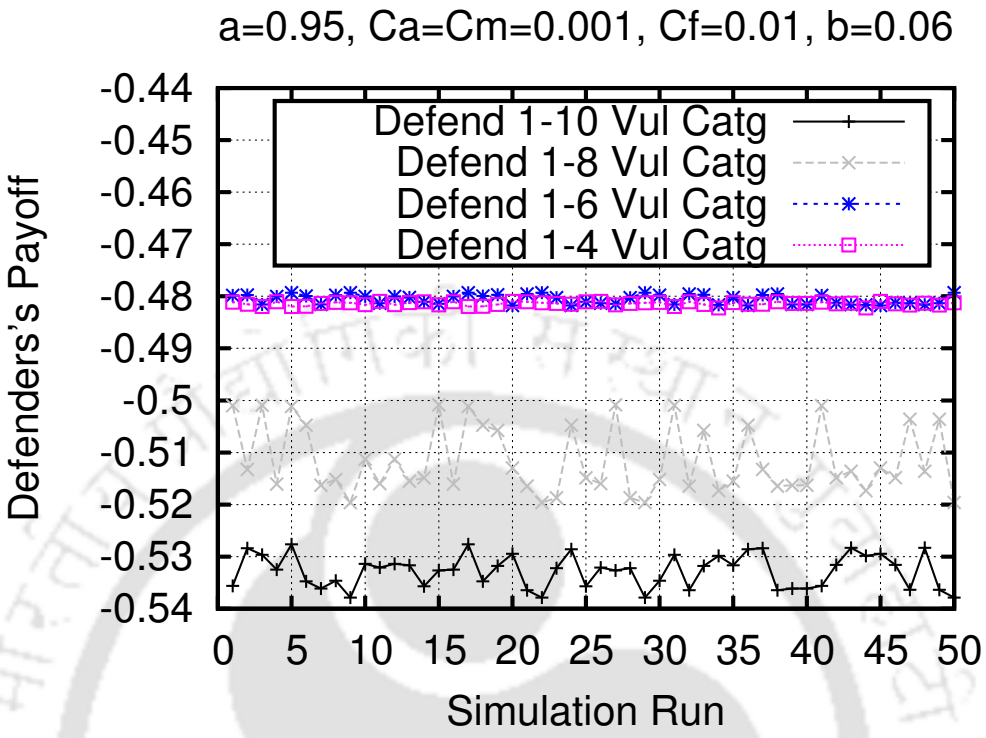

Figure 3.2: Defender's payoff for Case 1 under different strategies

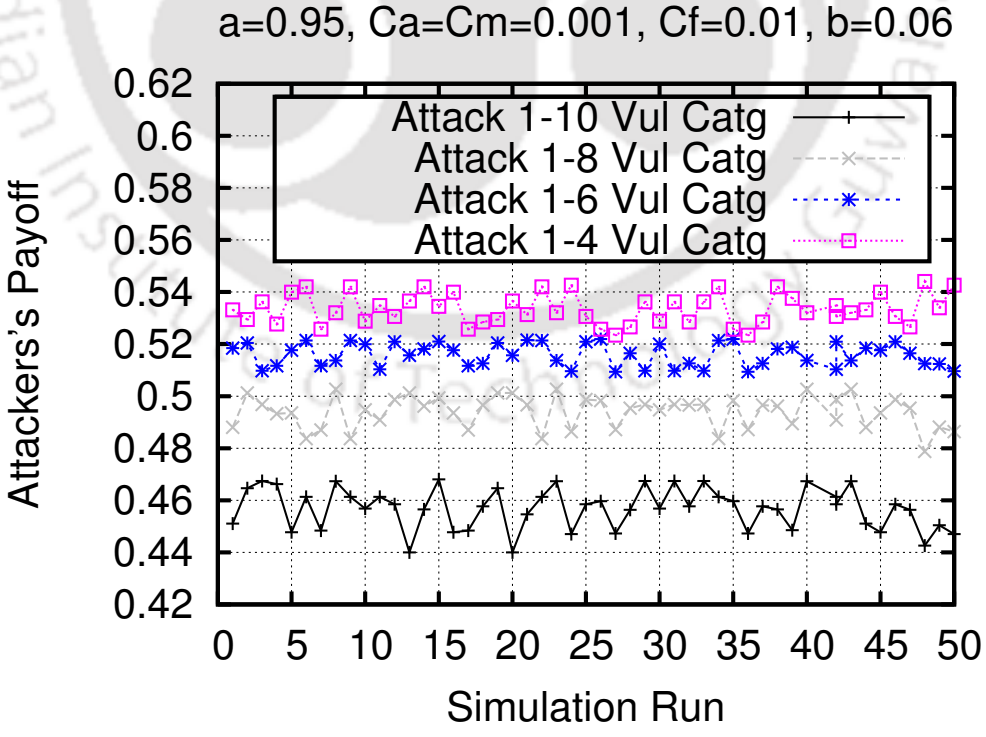

Figure 3.3: Attacker's payoff for Case 1 under different strategies

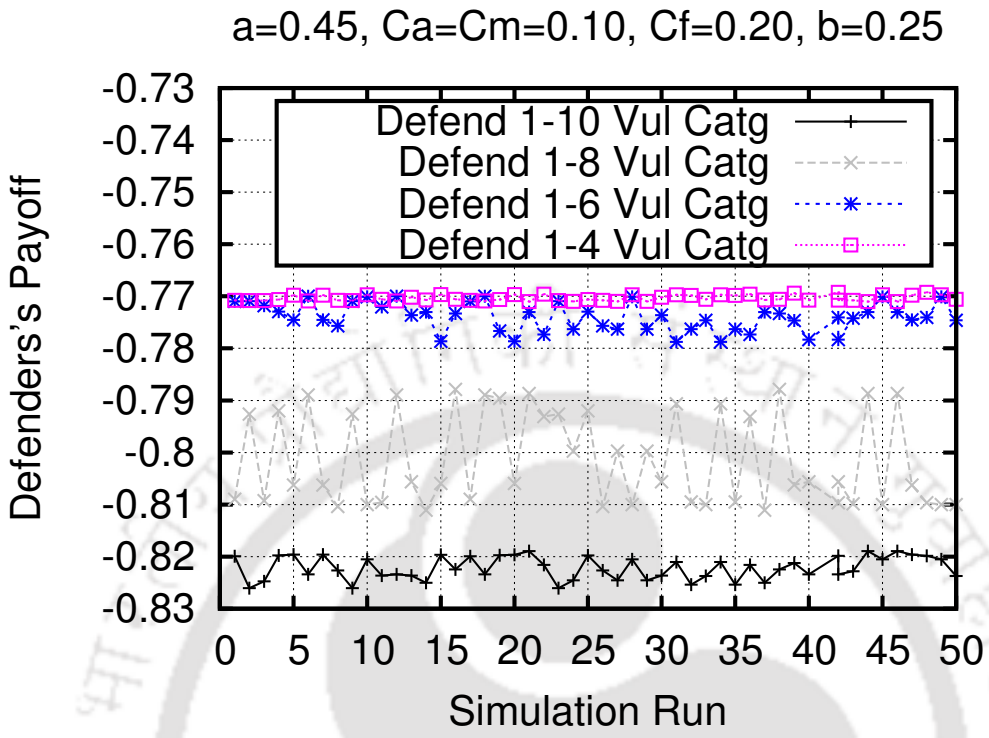

Figure 3.4: Defender's payoff for Case 2 under different strategies

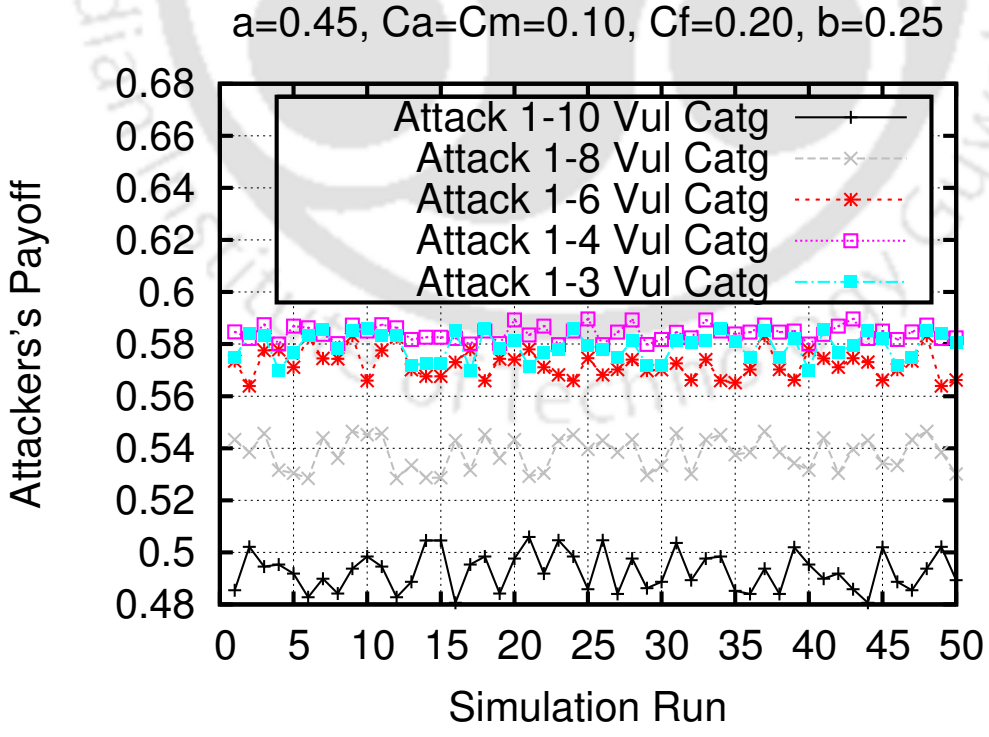

Figure 3.5: Attacker's payoff for Case 2 under different strategies

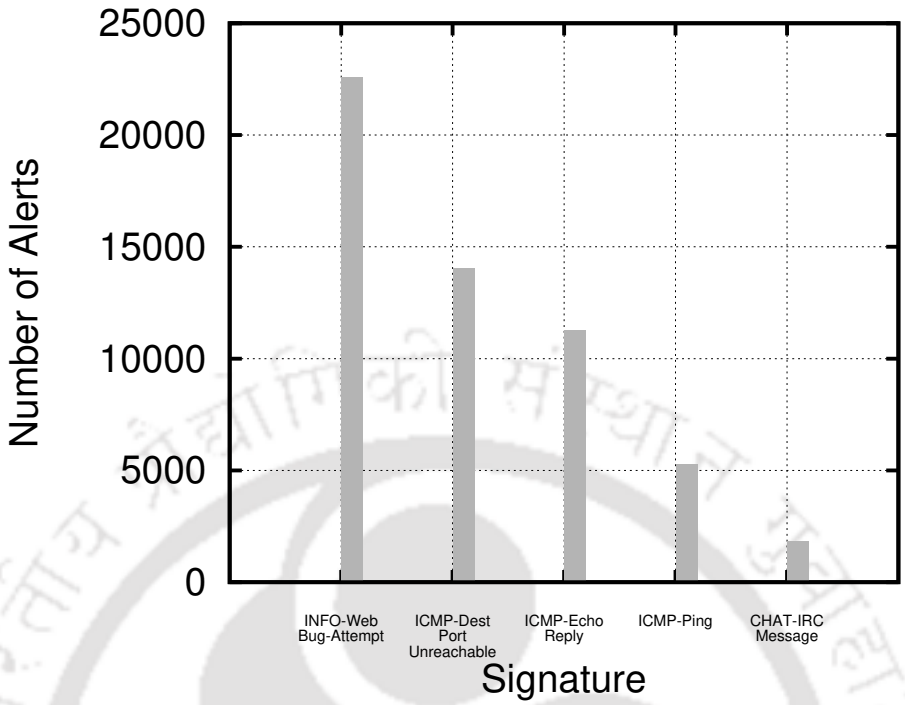

Figure 3.6: Top 5 Signatures generating largest number of FP alarms in the IDEVAL dataset

mance of Snort on the IDEVAL dataset was relatively poor. About 67% of the total alarms produced by Snort on the IDEVAL dataset were false positives. Figure 3.6 shows the top five Snort signatures generating the largest number of FP alarms on the IDEVAL dataset. The highest number of false alarms were triggered by INFO Web Bug attempt signature. This signature rule raises an alarm whenever the privacy policy violation are detected. However, none of these web bug alerts are related to any attack instances. Therefore, no true alarms were generated by this signature. Another major cause of false alarms were ICMP alerts. Logging every connection associated with probing, for example all ping activities, generates a huge number of false alarms. Large number of false alerts were also are generated from unsuccessful attempts or unrelated vulnerability, which do not require any immediate actions from the administrators.

In summary, Snort was able to detect 34 out of 56 type of attacks but at the same time produced a large number of unnecessary alarms. Many of the false alarms generated by the Snort were due to its disregard about the context of the underlying network environment. For example, consider the following Snort signature that detects the “Microsoft distributed transaction” attack which generates a buffer overflow and triggers a denial of service for the Microsoft distributed transaction services.

alert tcp EXTERNAL_NET any - >HOME_NET 3372 (msg : "DOS MSDTC attempt"; flow : to_server, established; dsize :> 1023; reference : bugtraq, 4006; reference : cve, 2002-0224; reference : nessus,10939; classtype : attempted - dos; sid : 1408; rev : 10;)

This Snort rule searches for TCP packets coming from any external network from any port to any machine inside the network on port 3372. If any packet with these characteristics is part of an open TCP session and also if the size of this packet is bigger than 1023 bytes, then Snort generates an alarm for DOS MSDTC attempt attack. However, this attack is effective only against the Windows based systems but are ineffective against the Linux based systems. Therefore, operating Snort with default settings, without considering the context of the underlying network environment produces a large number of false alarms.

It can be observed from Table 3.7 that the accuracy of the proposed false alarm minimization framework is significantly high for both critical and non-critical vulnerabilities of the IDEVAL dataset. This implies that the proposed framework successfully filters out most of the FP alarms generated by the Snort.

Although, many works [94] [95] in the literature have reportedly pointed out various flaws in the DARPA's IDEVAL dataset, it still remains one of the few large scale attempt at an objective evaluation of IDS systems. As such, it does provide a basis for making a rough comparison of existing IDS systems under a common set of circumstances and assumptions. Moreover, in absence of better and openly available benchmark datasets, vast amount of IDS research is based on the experiments performed on the DARPA's IDEVAL dataset.

The reason for choosing Snort as the default signature based IDS was because of its large community base and its rich set of attack signatures. The attack signatures in Snort are populated from various publicly available vulnerability databases like BugTraq [7], CVE [8], Nmap [9], Nessus [10] etc. This makes Snort an ideal candidate for evaluation of the signature based IDS.

3.4.2 Analysis on the in-house testbed dataset

To further analyze the proposed framework in a practical setup, we deployed it on the inhouse testbed comprising several hosts with various operating systems like Windows 2000 Server, Windows NT, SunOS, Windows XP, Windows 7, Ubuntu 12.04, Redhat 7.2 and Fedora 12. Various applications such as Telnet, FTP server, SQL server etc. were installed on the host machines of the testbed network. Fig. 3.7 shows the testbed network setup

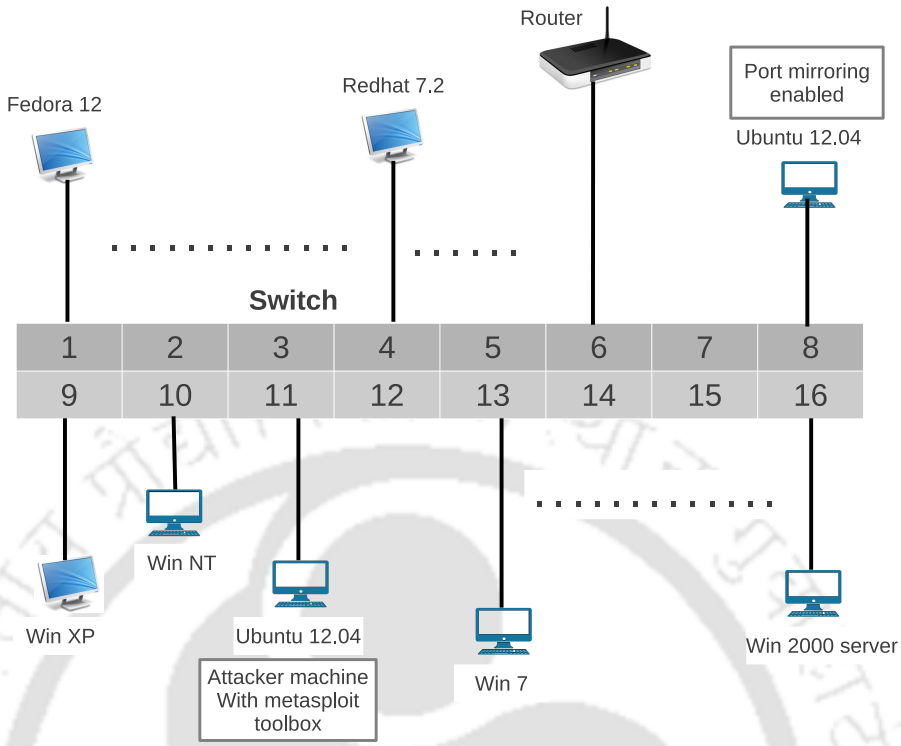

Figure 3.7: Configuration of the in-house testbed network setup

configuration. The host machines in the testbed network were connected in a LAN with a CISCO catalyst 3560 G series switch. Port mirroring facility was enabled at port 8 of the switch to capture the data packets of the testbed network as a tcpdump file. One of the machine in the testbed network running Ubuntu 12.04 and connected to port 11 of the switch was used to generate attacks using a metasploit toolbox [96], which is a freely available open source exploit toolbox. Some of the attacks were also launched from outside the testbed network by external attackers connected through router on port 6 of the switch. The Threat profile of the testbed network was generated using various vulnerability scanners like, Nmap [9], CVE [8] and Nessus [10]. Snort with default set of rules was used as signature based IDS.

Table 3.8: Performance of the proposed framework on the IITG Lab. dataset

	Critical Vulnerabilities				**Non-Critical Vulnerabilities**			
Attack Class	Alarms	After Corr.	Acc (%)	DR (%)	Alarms	After Corr.	Acc (%)	DR (%)
FTP	458	37	100	92.12	353	29	97.65	78.83
SQL	349	25	98.14	92.33	415	32	98.89	75.66
Telnet	328	31	98.85	94.66	456	17	100	83.25
DoS	526	41	100	93.12	786	23	99.24	85.34
Probe	431	37	100	99.12	567	39	99.54	81.34

The vulnerabilities in the Threat profile of the testbed network were categorized into 10 different vulnerability sets, with each set containing one or more vulnerabilities. The criticality weights of the vulnerability sets were set between 0.1 to 1. Severe vulnerabilities were assigned to higher criticality weight vulnerability sets. The cost of attacking (C_a) and monitoring (C_m) the network vulnerabilities were both set to 0.002. The false alarm cost was set to 0.003. The detection rate and the false alarm rate of the vulnerability scanners on the subset of the testbed network's host operating systems were found to be 0.98 and 0.04, respectively. The generated testbed dataset consist of 297 instances of 30 different types of attacks along with normal data traffic collected over a period of 5 hours. Following categories of attacks were considered in the testbed network setup:

- **Denial of Service (DoS)**: Teardrop, Land, Smurf, Ping of death, Win-nuke, Syndrop, Back, Mailbomb, Udpstorm, Arppoison, Crashiis, SYN Flood, tcpreset, selfping, ICMP Flood.
- **FTP & SQL**: Finger redirect, FTP server overflow, FTP format string, Freeftpd, username overflow, SQL server overflow, SQL injection.
- **Telnet & Probe**: Telnet buffer overflow, Telnet Resolve host conf, Ipsweep, Nmap, Mscan, Reset scan.

Table 3.8 shows the accuracy and detection rate of the proposed false alarm minimization scheme against different type of attacks on both the critical and non-critical vulnerabilities of the testbed network. It can be observed from the table that the proposed framework achieves high *accuracy* across all categories of attacks for both critical and non-critical vulnerabilities. This implies that most of the false alarms generated by the Snort were filtered out by the correlation engine of the proposed scheme. The detection rate of the proposed false alarm minimization scheme is relatively high for attacks against critical vulnerabilities, whereas its detection rate for attacks against non-critical vulnerabilities is comparatively low. However, the low detection rate against non-critical vulnerabilities is acceptable as the attacker is very unlikely to attack them due to their low asset values.

We compare the performance of the proposed false alarm minimization scheme with various other frameworks to validate its effectiveness. Table 3.9 and Table 3.10 show the performance comparison of the proposed false alarm minimization scheme with that of alarm verification based [97], alarm classification based [99], data summarization based [98] and hybrid [63] frameworks on the IDEVAL dataset and the testbed dataset, respectively. The

Table 3.9: Comparison of proposed framework with other false alarm minimization frameworks on the IDEVAL dataset

	Alarm verification [97]	Data summarization [98]	Alarm classification [99]	Hybrid [63]	Proposed Scheme
Accuracy (%)	95.57	94.72	95.53	97.91	98.83
Detection Rate (%)	67.51	71.29	66.87	69.23	68.28

Table 3.10: Comparison of proposed framework with other false alarm minimization frameworks on the IITG Lab. dataset

	Alarm verification [97]	Data summarization [98]	Alarm classification [99]	Hybrid [63]	Proposed Scheme
Accuracy (%)	97.39	96.17	94.47	95.29	98.55
Detection Rate (%)	89.73	90.78	90.29	90.91	91.87

reason for choosing these frameworks for comparison with the proposed false alarm minimization framework is because of the similarity of the dataset (DARPA's IDEVAL dataset) used in these frameworks for their evaluations. Moreover, to the best of our knowledge, there are no other game theory based false alarm minimization frameworks proposed in the literature for signature based IDSs.

It can be observed from Table 3.9 that the proposed framework has the highest accuracy among all the frameworks on the IDEVAL dataset. However, its detection rate is less than that of frameworks proposed in [98] and [63]. The low detection rate of the proposed framework on the IDEVAL dataset is primarily due to the inability of its signature based IDS (Snort) to detect the attacks in the IDEVAL dataset in the first place and does not necessarily imply poor performance of the proposed framework. The low detection rate of Snort on the IDEVAL dataset can be attributed to the fact that most of the attacks in the IDEVAL dataset are obsolete and Snort no longer contains signatures to detect these attacks.

Similarly, it can be observed from the Table 3.10 that the proposed framework has the least false alarm rate (highest accuracy) amongst all the frameworks on the testbed dataset. The proposed framework is able to achieve this high accuracy since it uses various network context information parameters like alarm reference numbers, IP addresses, protocol types, port numbers, severity levels of vulnerabilities corresponding to the IDS alarms, OS types etc., along with a game theory-based monitoring strategy to filter out most of the false positive alarms generated by the signature based IDS (Snort). Since all the schemes (except [98]) use a common signature based IDS (Snort), their detection rates are comparable to each other. All the schemes have relatively high detection rate on the testbed dataset

since their signature based IDS contains most of the attack signatures to detect the attacks on testbed dataset. However, the proposed framework has a better accuracy compared to other frameworks since it correctly identifies most of the TP alarms, while other frameworks incorrectly classifies some of the normal data traffic as attacks.

3.5 Conclusion

Signature based IDSs produce a large number of FP alarms that outnumbers the TP alarms by a ratio of almost 2:1. To address this issue, we proposed a novel game theory-based false alarm minimization scheme for signature based IDS. The proposed scheme uses multiple vulnerability scanners to scan the network and create a Threat profile of the network. The Threat profile comprises multiple vulnerability sets and each vulnerability set is assigned a unique criticality weight based on the severity of vulnerabilities contained in it. The IDS alarms are correlated with vulnerabilities in the Threat profile to determine the potential TP alarms. The proposed scheme also models the interaction between the attacker and the IDS (defender) as a two player non-cooperative game. Various attacking and monitoring strategies are examined to evaluate the Nash equilibrium of the game and build the Sensible Vulnerability Set (SVS) of the network. The SVS consists of a subset of high criticality weight vulnerability sets from the network's Threat profile. The IDS alarms that pass the network's Threat profile correlation test are eventually correlated with vulnerabilities in the SVS to determine the final TP alarms. Experimental results on the benchmark IDEVAL dataset and the testbed dataset show that the proposed framework significantly reduces the false alarm rate of the signature based IDS.

In the next chapter, we propose a novel Bayesian game theory-based hybrid intrusion detection framework for Mobile Ad-hoc Networks (MANETs). Persistent monitoring can result in premature death of nodes operating the IDS in MANETs. Additionally, a high volume of IDS traffic can cause congestion and prevent the flow of normal data traffic in MANETs. To address this issue a novel game theory based IDS framework for MANET is proposed in the next chapter. The proposed framework minimizes the energy consumption required for operating the IDS and also reduces the volume of IDS traffic introduced into the network by adopting probabilistic monitoring strategies based on the Bayesian Nash Equilibrium of the game.

"The river dragged both of them out and,
current took them down the stream..."

Robert Whalen

4

Intrusion Detection in Mobile Ad-hoc Networks: Bayesian Game Formulation

4.1 Introduction

Mobile Ad-hoc Network (MANET) is a collection of IEEE 802.11 / Wi-Fi enabled self-configuring and infrastructure-less network of battery powered and energy constrained wireless mobile devices. MANET is ad hoc in nature because it does not rely on any pre-existing infrastructure, such as routers in wired networks or access points in managed (infrastructure) wireless networks. Each node in a MANET is equipped with a wireless transmitter and receiver, which enables it to communicate with other nodes within its wireless transmission range without using any centralized structure. Due to their limited communication range, mobility and constrained computational capabilities, nodes in MANET must cooperate with each other to provide networking services among themselves. Therefore, each node in a MANET acts both as a transmitter and a receiver simultaneously. Fig. 4.1 shows the overall architecture of the MANET. As shown in the figure, mobile devices in MANET are connected to the access network (which provides various Quality of Service (QoS), multimedia and access to security applications) through an access router. The access routers are in turn connected to the core router, which provides Internet access to the wireless devices in MANET. MANETs typically communicate at radio frequencies range of 30 MHz - 5 GHz. Minimal configuration and quick deployment coupled with dynamic and adaptive routing protocols of MANETs make them suitable for deployment in extreme and

volatile environmental conditions, where it is difficult to have an infrastructure oriented wired connectivity.

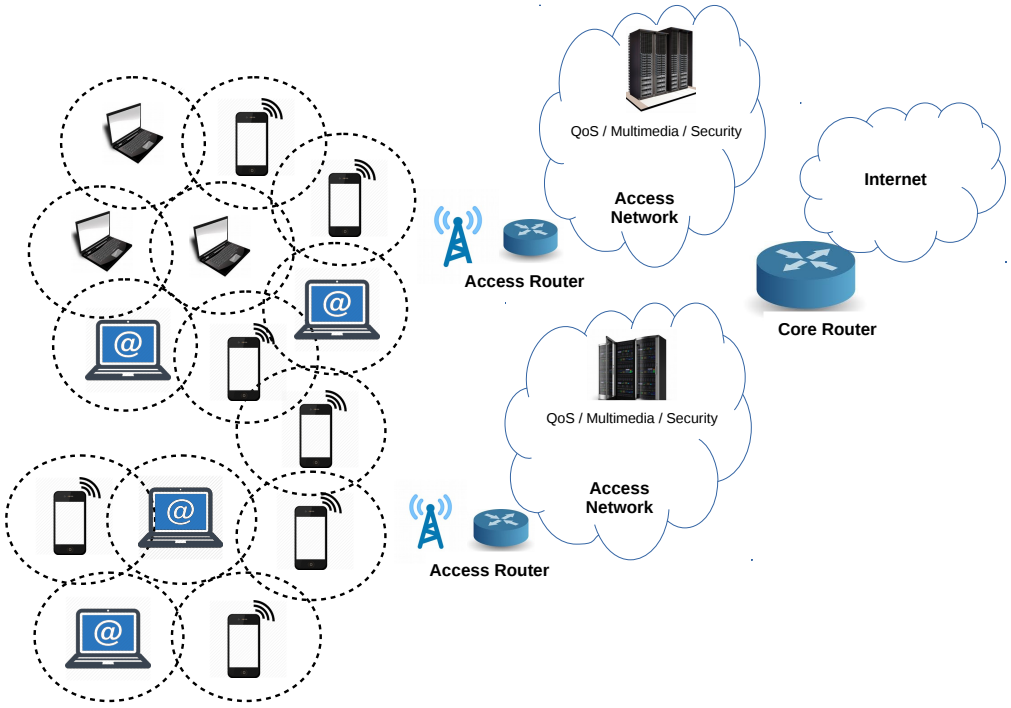

Figure 4.1: Mobile Ad-hoc Network (MANET) Architecture

MANETs are extremely useful for setting up an ad-hoc and infrastructure-less network connectivity on a go and have found applications in diverse domains such as military operations, environmental monitoring, vehicular ad hoc communications, disaster rescue operations, peer-to-peer messaging etc. However, on the flip side, the dynamic and distributed nature of MANETs make them vulnerable to various types of attacks like black hole attack, traffic distortion, IP spoofing, DoS attack etc., [100] [101] [102]. Intrusion Detection Systems (IDSs) have been proposed in the literature to address these security threats in MANETs. However, unlike in wired networks with well established infrastructure, there are no fixed checkpoints like routers and switches in MANETs, where the IDSs can be deployed [103] [104]. Therefore, nodes in MANET must cooperate among themselves and adopt a distributed intrusion detection mechanism to address various security threats for their overall well being [105] [106] [107]. Due to the absence of a centralized monitoring entity in MANET, each node runs its own IDS. However, owing to their limited battery life, it is

not feasible to keep the IDS running continuously on MANET nodes. Operating the IDS in a promiscuous mode drains out the energy level of MANET nodes, which results in their premature death and disconnected network problem. Additionally, MANETs operate in a bandwidth constrained wireless radio spectrum. Therefore, introduction of large volume of intrusion detection related traffic can cause congestion and limit the flow of normal data traffic in MANETs.

In this chapter, we propose a novel game theory-based hybrid IDS framework for MANETs. The proposed framework uses a combination of lightweight and heavyweight IDS modules to achieve high accuracy and detection rate against wide range of attacks. The lightweight module uses simple threshold based rules to detect malicious nodes, while the heavyweight module uses a powerful data mining based association rules to identify the malicious nodes. Additionally, the proposed framework models the intrusion detection process in MANET as a two player non-cooperative Bayesian game, which enables the node operating the IDS to adopt a probabilistic monitoring strategy based on the history profile of node being monitored. This helps the node operating the IDS to conserve its energy while at the same time minimize the overall volume of IDS traffic introduced into the network, without adversely affecting the performance of the IDS.

The rest of the chapter has been structured in the following way. Section 4.2 discusses various related works on MANET intrusion detection frameworks proposed in the literature. The drawbacks of these frameworks are then enumerated, which provides the motivation for the work carried out in this chapter. Section 4.3 presents the brief overview of our proposed MANET IDS framework. Bayesian game model used for developing the energy efficient IDS monitoring strategies is discussed in Sub-section 4.3.1. Distributed and energy efficient MANET leader election mechanism is discussed in Sub-section 4.3.2. The details of the proposed hybrid IDS framework along with its main components are discussed in Sub-section 4.3.3. Experimental results and performance evaluation of the proposed IDS framework is provided in Section 4.4. Finally, we conclude with the conclusion and a brief introduction about Chapter 5 in Section 4.5.

4.2 Related Works

A MANET IDS framework called the Enhanced Adaptive Acknowledgment (EAACK) is proposed in [108]. The framework requires all acknowledgment packets to be digitally signed by the sender and verified by the receiver. It uses DSA and RSA as digital signatures

and is shown to achieve high detection rate against wide range of attacks. However, the main drawback of this framework is its requirement to digitally sign all the acknowledgments which increases its computational overhead. A light weight, energy efficient and non-cryptographic intrusion detection solution against the gray hole attack in MANET is proposed in [109]. However, their scheme requires the IDS to operate in a promiscuous mode to detect intrusions, which results in high power consumption for operating the IDS. Additionally, their scheme can only detect gray hole attack and cannot be generalized for detecting other class of attacks.

A hybrid MANET IDS framework comprising two different modules, namely, the Watchdog and Pathrater is proposed in [110]. In this framework, the Watchdog module acts as an IDS and detects malicious node behaviors in the network by promiscuously listening to its next hop's transmission. If the Watchdog notices that its immediate next node fails to forward the packet within a specified period of time then it increments the node's failure counter. If the failure counter of the node being monitored exceeds a threshold value, then the Watchdog reports the node as malicious. On the other hand, the Pathrater module informs the routing protocol to avoid transmission of data through the malicious nodes. The main issue of this framework is that it requires continuous monitoring by the Watchdog module to detect malicious nodes, which can drain out the energy level of the monitoring node.

A TWOACK MANET IDS framework that requires every data packets transmitted over three consecutive nodes along the source to destination path to be acknowledged is proposed in [111]. In this framework, every node along the route has to send back an acknowledgment packet to the node that is two hop count away from it in the route. The arrival of TWOACK packet at first node X (in the three consecutive nodes along the route) indicates a successful transmission of packet from node X to node Z via the intermediate node Y. However, if this TWOACK packet is not received within a specified time interval, both node Y and Z are reported as malicious by the framework. The main drawback of this scheme is the increased routing overhead due to frequent TWOACK packet generation.

Game theory-based IDS frameworks for Ad-hoc networks that model the cooperation and selfishness of the networks are discussed in [25] [26]. In these frameworks, each node decides whether to forward or withhold the packet based on the trade-offs involved in cost (energy consumption) and benefits (network throughput) for collaborating with other nodes in the network. Therefore, enforcing a cooperation mechanism ensures that a selfish

node that does not obey the network rules receives a low throughput. However, the main drawback of these frameworks is the assumption that each node has a full information about all the network parameters.

A game theoretic IDS framework for analyzing the interactions between pairs of attacking/defending nodes using a Bayesian formulation in wireless Ad-hoc Networks is proposed in [27]. The framework uses a Bayesian hybrid detection approach, wherein a less powerful lightweight module is used to estimate the node being monitored, and a more powerful heavyweight module acts as a last line of defense. It analyzes the obtainable Nash Equilibrium (NE) for the attacker/defender Bayesian game in a dynamic settings, which allows the defender (IDS) to consistently update its belief about the maliciousness of the opponent player as the game evolves. However, the main issue associated with this framework is the difficulty involved in determining a reasonable prior probability about the maliciousness of the attacker (malicious node).

A general incentive-based method to model attacker's intent, objectives and strategies (AIOS) based on game theoretic formalization is proposed in [80]. The framework uses an incentive based conceptual structure for AIOS modeling which can capture the inherent inter-dependency between AIOS and defender objectives and strategies in such a way that AIOS can be automatically inferred. The AIOS modeling enables the defender to predict which kind of strategies are more likely to be taken by the attacker than the others, even before such an attack happens. The AIOS inferences lead to more precise risk assessment and harm prediction. However, the drawback of this framework is its complete information game assumption.

Chen et. al [28] proposed a framework that applies two game theoretic schemes for economic deployment of intrusion detection agent. In the first scheme the interaction between an attacker and the intrusion detection agent is modeled and analyzed within a non-cooperative game theory settings. The mixed strategy Nash Equilibrium solution is then used to derive the security risk value. The second scheme uses the security risk value derived by the first scheme to compute the Shapley value of the intrusion detection agent while considering the various threat levels. This allows the network administrator to quantitatively evaluate the security risk of each IDS agent and easily select the most critical and effective IDS agent deployment to meet the various threat levels to the network. A game theoretical framework to model the interaction between the service provider and the attacker as an intrusion detection game was proposed by *Kodialam et al.* [29]. In this

framework, the game is represented as a two person zero-sum game, wherein the service provider tries to maximize its payoff by increasing its probability of successful detection while the attacker tries to minimize its probability of being detected by the IDS. The optimal solution for both the player is to play the min-max strategy of the game. The drawback of these frameworks is the assumption that both players (attacker and defender) have complete information about the network topology and all links in the network. However, such an assumption is usually invalid in real networks where the players do not have a complete information about all the network parameters.

Summarizing our studies on the related works, we found that most of the non-game theory-based MANET IDS frameworks are computation intensive. They also require the nodes operating the IDS to perform the monitoring operation continuously, which leads to high energy consumption as well as introduction of large volume of IDS traffic into the network. These issues are addressed to certain extent by the game theory-based MANET IDS frameworks. However, most of the game theory IDS frameworks proposed in the literature assume a complete information game, wherein the players (attacker and defender) are presumed to have a complete information about the game, i.e., they make an implicit assumption that all the network parameters are known *a priori*. However, such an assumption is impractical in MANETs, as nodes only have partial information about various network parameters. Moreover, most of the game theory-based MANET IDS frameworks are static in nature, wherein the strategies and utilities of players are fixed and repeated over a period of time. Such static representational model fails in dynamic environments, wherein nodes adopt different strategies at various stages of the game. In addition, most of the MANET IDS frameworks proposed in literature are geared towards detection of specific class of attacks like blackhole attack, wormhole attack, selective forwarding etc. [110] [112] and cannot be generalized for detecting other class of attacks. All these drawbacks in the related works provide us with the motivation to propose a new MANET IDS framework based on incomplete information game to address them.

In this chapter, we propose a novel MANET IDS framework comprising two different modules namely, the *cluster leader election* module and the *game theory-based hybrid IDS* module. The *cluster leader election* module uses the Vickery-Clarke-Groves (VCG) mechanism [38] and elects the node with the highest reputation and energy level value as the cluster leader node. The cluster leader node is designated with the responsibility of providing intrusion detection services to all the other cluster nodes for a specified period of time, after which a new leader node is elected in its place. This re-election process minimizes the overall energy

consumption required for operating the IDS in MANET, while at the same time ensures a uniform energy dissipation across multiple nodes for performing the monitoring operation. The second component i.e., the *game theory-based hybrid IDS module* performs the actual intrusion detection task to identify various malicious nodes in the network.

The cluster leader node operates the *game theory-based hybrid IDS module*, which consists of a lightweight and a heavyweight components. The lightweight component is less powerful and uses simple rules based on threshold values to detect intrusions. On the other hand, the heavyweight component is more powerful and uses complex association-mining rule techniques to detect anomalies. Initially only the lightweight component of the hybrid IDS module is activated for performing the monitoring operation. When the actions of the node being monitored by the lightweight component is found to be malicious, the heavyweight component is activated for further analysis and to verify whether the node being monitored is indeed malicious. However, if the action of the node being monitored by the lightweight component is deemed to be normal then the heavyweight component is activated with certain probability. In this case, the probability of activating the heavyweight component is determined by modeling the interaction between the IDS and the node being monitored as a two player non-cooperative Bayesian game. The decision to activate the heavyweight component is determined by evaluating the Bayesian Nash Equilibrium (BNE) of the non-cooperative game.

4.3 Proposed MANET IDS Framework

In this section, we provide a detailed description of the proposed MANET IDS framework. First, the flowchart of the proposed IDS framework is provided followed by the description of its two main modules namely, the *cluster leader election module* and the *game theory-based hybrid IDS module*. We make the following assumptions with respect to our proposed MANET IDS framework:

- MANET topolgy is divided into multiple clusters using a standard clustering algorithm [113]. All the nodes in a given cluster are within the transmission range of each other.
- Each node n_i in any given cluster has the following associated parameter values: maliciousness value (p_i), reputation value (R_i) and energy level value (E_i).
- The elected cluster leader node (C_L) provides the intrusion detection services to all the other cluster nodes for a predefined period of time.

Fig. 4.2 shows the flowchart of the proposed MANET IDS framework. Let Cls = {C_1, C_2, .., C_k} denote the set of k clusters in MANET. Initially the framework elects the cluster leader node C_L and a set of checker nodes for each $C_i \in Cls$ using the Vickrey-Clarke-Groves (VCG) mechanism [38]. C_L is entrusted with the responsibility of providing the intrusion detection services to all the other nodes in C_i. A new cluster leader node is elected after every predefined time interval to ensure a uniform energy consumption across multiple nodes for performing the monitoring operation. The intrusion detection services provided by C_L to a node n_j depends on n_j's reputation value (R_j). Nodes with higher reputation values are entitled to more intrusion detection services from C_L compared to nodes with lower reputation values. The services provided by C_L to n_j includes monitoring the n_j's incoming traffic which are received from its neighboring nodes, as well as monitoring the n_j's outgoing traffic.

To address the issue of a presence of a malicious C_L, a set of checker nodes are elected to monitor the operations of C_L. C_L may misbehave after being elected as the leader node by declining to provide intrusion detection services to cluster nodes or by reporting the normal node as malicious. If majority of the checker nodes find the C_L to be misbehaving then it is replaced with a new cluster leader node. In addition, the malicious leader node is also punished by lowering its reputation value. Here, we make an assumption that majority of the checker nodes in the cluster are non-malicious and no collusion takes between the malicious checker nodes. The detailed description about the cluster leader election and punishment mechanism is provided in sub-section 4.3.2.

After being elected as the cluster leader, C_L assigns an initial maliciousness belief value (p_i) to node n_i being monitored and activates its lightweight IDS component, which uses the Packet Forwarding Rate (PFR) of n_i to determine whether it is normal or malicious. The PFR of n_i is defined as the ratio of total number of packets received by n_i to the total number of packets forwarded by n_i over a given interval of time. If the PFR of n_i is less than the threshold value T_{PFR}, then n_i is assumed to be malicious. The p_i value of n_i is then updated using the Bayes rule and C_L's heavyweight IDS component is activated for further monitoring of n_i. However, if the PFR of n_i is greater than or equal to the threshold value T_{PFR}, then n_i is assumed to be normal. In this case, the p_i value of n_i is updated using the Bayes rule but the decision to activate the C_L's heavyweight IDS component is determined by representing the interaction between C_L and n_i as a two player non-cooperative Bayesian game followed by evaluating the Bayesian Nash Equilibrium (BNE) of the game. The BNE of this non-cooperative game corresponds to the strategy combination (q^*, p^*), where q^* is

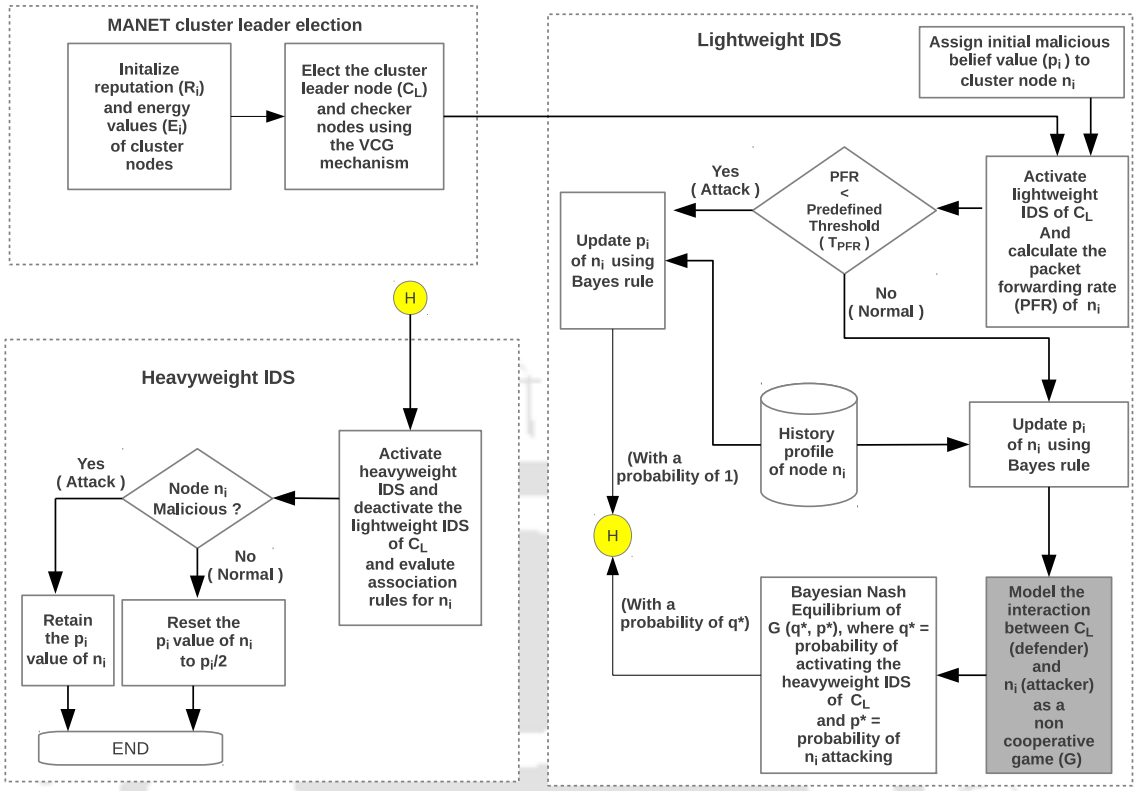

Figure 4.2: Flowchart of the proposed MANET IDS scheme

the probability of C_L to activate its heavyweight IDS component and p^* is the probability of n_i to play its strategy *Attack*, if it is malicious. Therefore, in this case the decision to activate the C_L's heavyweight IDS component is probabilistic and depends on the BNE of the game. The C_L's heavyweight IDS component is an anomaly based IDS that uses association-rule mining techniques to determine whether n_i is malicious or normal. If n_i is found to be normal by the heavyweight IDS component then the p_i value of n_i is reset to $p_i/2$, else the p_i value of n_i is retained. Moreover, when the malicious belief value of any node n_j being monitored by C_L falls below a predefined threshold value ($Mal_{th}^{C_L}$) then n_j is removed from the cluster by C_L. In addition, C_L also informs other nodes in the cluster to avoid communication with the malicious node n_j. The value of $Mal_{th}^{C_L}$ is set equal to one third the average maliciousness value of all the nodes being monitored by C_L.

Data packets in MANET can be dropped due to various reasons like network congestion, depletion of node's resources, presence of malicious nodes etc. Nevertheless, excessive packet dropping is a strong indicator about the presence of malicious nodes in the network. Therefore, determining node n_i's PFR value by using the C_L's lightweight IDS component provides a strong insight into n_i's nature (normal or malicious). Although, the C_L's heavy-

weight IDS component is more powerful compared to its lightweight IDS component, the energy required for operating the former is comparatively higher than that required for operating the latter. Therefore, using the lightweight IDS component as a precursor before activating the heavyweight IDS component reduces the overall energy consumption required for operating the IDS in MANET. More elaborate details about the proposed hybrid MANET IDS framework is provided in sub-section 4.3.3. In the subsequent sub-sections, we introduce the preliminaries of the game theory which is a prerequisite for developing the monitoring strategies of the proposed hybrid MANET IDS framework.

4.3.1 Bayesian game model for proposed MANET IDS framework

Game theory allows modeling events of conflict between two or more rational decision makers (players) with different set of objectives and competing for the same set of resources. Therefore, game theory is concerned with finding the best strategies for individual decision makers and recognizing the stable outcomes in such situations. The interaction between the IDS and the node being monitored can be represented as a two player non-cooperative Bayesian game. The said game comprises two players namely, P_i and P_j with their set of strategies. The player P_i is a potential malicious node (attacker), while the other player P_j is the cluster leader node (defender). The private information of player P_i is its type θ_i (*normal or malicious*). The type $\theta_i = 1$, if P_i is normal and $\theta_i = 0$, if it is malicious. This private information regarding the type of P_i is unknown to the P_j. The type of P_j is always normal and denoted by $\theta_j = 1$, which is a common knowledge known to both the players. The attacker player of type $\theta_i = 0$ has two pure strategies: {*Attack, Not attack*} while the normal player of type $\theta_i = 1$ has only one pure strategy: {*Not attack*}. Similarly the defender player P_j has two pure strategies: {*Monitor, Not monitor*}.

In the beginning, both the players simultaneously choose their strategies, with prior knowledge about the costs involved in monitoring and attacking nodes in the network. The defender player (P_j) assigns a maliciousness belief value to the node being monitored (P_i), which is assumed to be a common knowledge known to both the players. This non-cooperative incomplete information game between the players P_i and P_j can be represented as a triplet $G = \langle N, S, U \rangle$, where

- $N = \{P_i, P_j\}$ are the two players of the game.
- $S = S_i \times S_j$ is the strategy space of the game with S_i and S_j being the strategy space

of players P_i and P_j, respectively.

- $U = U_i \times U_j$ is the payoff utility corresponding to the strategy space S. U_i and U_j are the payoffs of players P_i and P_j corresponding to their strategies spaces S_i and S_j, respectively.

In the subsequent sections, the term player and node refers to the same entity and we use them interchangeably. Let $C = \{n_1, n_2, ..., n_t\}$ be the set of t nodes in a given MANET cluster. Consider any given node $n_k \in C$ $(1 \leq k \leq t)$ and let its associated asset value be w_k. The loss of asset when the attacker player P_i successfully exploits n_k represents the loss, whose value is equivalent to degree of damage such as loss of reputation, compromise of data integrity, cost of controlling damages etc. The defender player P_j is equipped with an IDS module to monitor the actions of the attacker player P_i. Let the detection rate and the false alarm rate of P_j's IDS module be denoted by α and γ, respectively where α, $\gamma \in$ [0, 1]. Additionally, let the cost involved in attacking and monitoring n_k be denoted by C_{a_k} and C_{m_k}, respectively.

Table 4.1: Payoff Matrix when player P_i is malicious

	Monitor	Not Monitor
Attack	$(1 - 2\alpha)w_k - C_{a_k}$, $(2\alpha - 1)w_k - C_{m_k}$	$w_k - C_{a_k}$, $-w_k$
Not Attack	0, $-\gamma w_k - C_{m_k}$	0, 0

Table 4.2: Payoff Matrix when player P_i is normal

	Monitor	Not Monitor
Not Attack	0, $-\gamma w_k - C_{m_k}$	0, 0

Table 4.1 and Table 4.2 shows the payoff matrices corresponding to the interaction between players P_i and P_j over the node n_k whose asset value is worth w_k, when the type of P_i is malicious and normal, respectively. These tables define various payoffs obtained by P_i and P_j when interacting over node n_k under different strategies. Following conclusions can be drawn from Table 4.1, when the type of player P_i is malicious.

- When the malicious player P_i attacks n_k and the defender player P_j does not monitor n_k, i.e., for strategy combination S_1 = (*Attack, Not Monitor*), P_j's payoff is

$$U_j(S_1) = -w_k$$

which represents the loss of asset worth w_k. On the other hand, for the strategy combination S_1, P_i receives a payoff which is its gain from the successful exploitation of n_k minus the cost involved in attacking n_k (C_{a_k}). Therefore, the payoff utility of P_i with strategy combination S_1 is

$$U_i(S_1) = w_k - C_{a_k}$$

- For the strategy combination S_2 = (*Attack, Monitor*), P_j's payoff is the gain obtained from successful attack detection against n_k minus the monitoring cost C_{m_k}. However, successful attack detection against n_k depends on the detection rate (α) of the P_j's IDS module. Therefore, the payoff utility of P_j playing strategy combination S_2 is

$$\begin{aligned} U_j(S_2) &= \alpha w_k - (1-\alpha)w_k - C_{m_k} \\ &= (2\alpha - 1)w_k - C_{m_k} \end{aligned}$$

where (1 - α) represents the false negative rate of the P_j's IDS module. On the other hand, P_i's loss after being detected by P_j's IDS module is equal to P_j's gain minus the attacking cost C_{a_k}. Therefore, P_i's payoff utility with strategy combination S_2 is

$$U_i(S_2) = (1 - 2\alpha)w_k - C_{a_k}$$

- For the strategy combination S_3 = (*Not Attack, Monitor*), P_j's expected loss is -γw_k due to false alarm of IDS plus the monitoring cost C_{m_k}, while the payoff of P_i is 0. Therefore, the payoff utilities of P_j and P_i with strategy combination S_3 are

$$\begin{aligned} U_j(S_3) &= -\gamma w_k - C_{m_k} \\ U_i(S_3) &= 0, \quad respectively \end{aligned}$$

- For the strategy combination S_4 = (*Not Attack, Not Monitor*) the payoffs of both P_i and P_j are 0, i.e., $U_j(S_4) = U_i(S_4) = 0$.

Similarly from Table 4.2, we can observe that when the type of player P_i is normal the payoff of P_i is always 0. The payoff of defender player P_j is 0 if it plays its pure strategy (*Not Monitor*). On the other hand, if it plays its pure strategy (*Monitor*) its payoff utility is -γw_k - C_{m_k}, which is the cost incurred due to false alarms and the monitoring cost.

4.3.1.1 Bayesian Nash Equilibrium (BNE) Analysis

Fig. 4.3 shows the extensive form of the Bayesian game described in the preceding subsection. In the Figure, **N** represents the nature node that determines the type of player P_i (malicious or normal). Let p_o be the prior probability of player P_i being malicious. We make an implicit assumption that both P_i and P_j are rational players and their main objective is to maximize their respective payoff utilities. P_i would want to play a strategy that minimizes its probability of being detected by the P_j's IDS module, while P_j would like to play a strategy that maximizes its probability of successfully detecting the malicious player P_i.

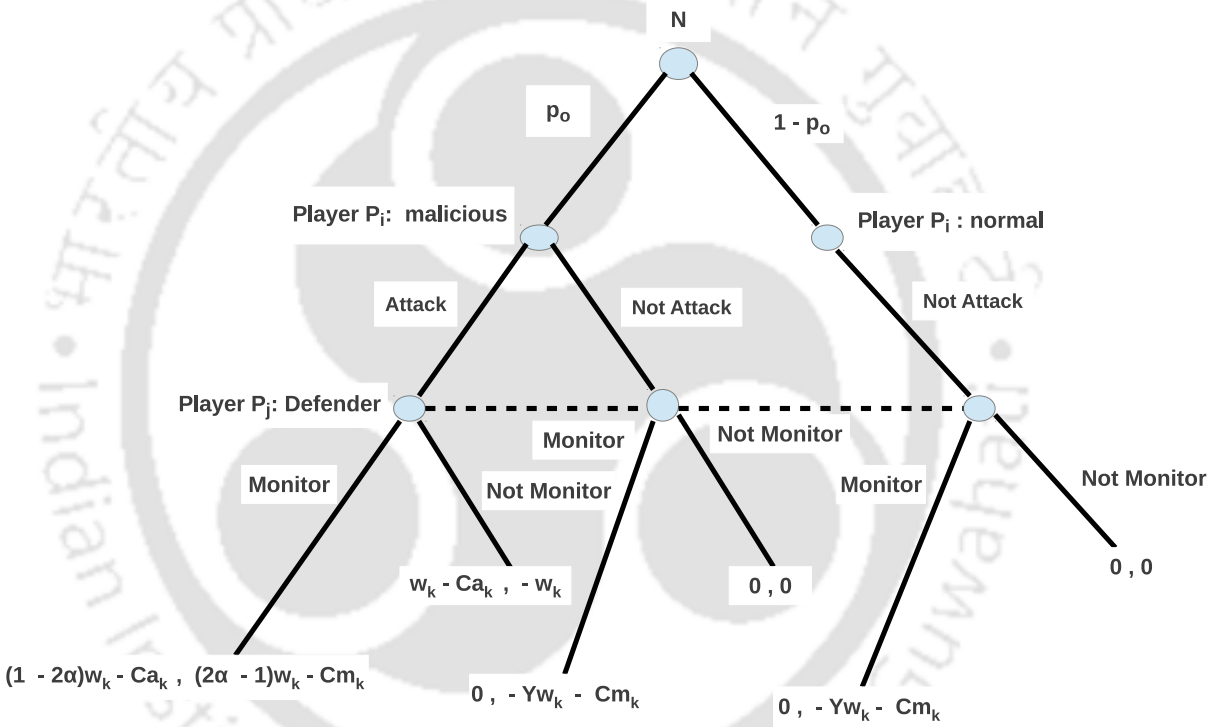

Figure 4.3: Extensive form of the Bayesian game

In the subsequent paragraphs, the BNE of the said Bayesian game is analyzed. The BNE of the game correspond to the strategy combination of the players P_i and P_j, such that no player has any profitable incentive to unilaterally deviate from its chosen strategy. We make an implicit assumption that P_j's prior belief (p_o) about P_i being malicious is a common prior, i.e., P_i (attacker) knows about the P_j's (defender) belief value about P_i being malicious. We make the following observations about the Bayesian game described by Table 4.1, Table 4.2 and Fig 4.3.

- If the type of P_i is malicious and if it plays its pure strategy *Attack* then the expected

payoff of P_j playing its pure strategy *Monitor* is:

$$U_j(\textit{Monitor}) = p_o\Big((2\alpha - 1)w_k - C_{m_k}\Big) - (1 - p_o)(\gamma w_k + C_{m_k})$$

and when P_j plays its pure strategy *Not Monitor*, its expected payoff is:

$$U_j(\textit{Not Monitor}) = -p_o w_k$$

- When P_j plays its pure strategy *Monitor*, the expected payoff of P_i playing its pure strategies *Attack* and *Not Attack* are:

$$U_i(\textit{Attack}) = p_o\Big((1 - 2\alpha)w_k - C_{a_k}\Big) \quad and$$
$$U_i(\textit{Not Attack}) = 0 \ , \quad respectively.$$

- Therefore, if $U_j(\textit{Monitor}) > U_j(\textit{Not Monitor})$, i.e., if $p_o > \frac{\gamma w_k + C_{m_k}}{(2\alpha+\gamma)w_k}$, the best response of the P_j is to play its pure strategy *Monitor*. However, when P_j plays its pure strategy *Monitor*, the best response of P_i would be to play its pure strategy *Not Attack*. Hence the strategy combination ((*Attack* if malicious, *Not Attack* if normal), *Monitor*, p_o) is not a BNE, when $p_o > \frac{\gamma w_k + C_{m_k}}{(2\alpha+\gamma)w_k}$. Similarly if $U_j(\textit{Monitor}) < U_j(\textit{Not Monitor})$ i.e., if $p_o < \frac{\gamma w_k + C_{m_k}}{(2\alpha+\gamma)w_k}$, the best response of P_j is to play its strategy *Not Monitor*, since in this case the payoff obtained by playing strategy *Monitor* is less than the payoff obtained by playing strategy *Not Monitor*. Therefore, the strategy combination ((*Attack* if malicious, *Not Attack* if normal), *Not Monitor*, p_o) is a pure strategy BNE, when $p_o < \frac{\gamma w_k + C_{m_k}}{(2\alpha+\gamma)w_k}$.

- If P_i plays its pure strategy *Not Attack*, then P_j's dominant strategy would be to play *Not Monitor* regardless of the value of p_o. However, if P_j plays its pure strategy *Not Monitor*, the best response of P_i if its type is *malicious* would be to play its strategy *Attack*. Therefore, the strategy combination ((*Not Attack* if malicious, *Not Attack* if normal), *Not Monitor*) is not a BNE.

In our previous discussion, we showed that if $p_o > \frac{\gamma w_k + C_{m_k}}{(2\alpha+\gamma)w_k}$ then there does not exist any pure-strategy BNE. But any game with a finite set of players and finite set of strategies has a Nash equilibrium of mixed strategies. In the subsequent section, we derive the mixed strategy BNE for the game, when no pure strategy BNE exists.

Let P_i play its strategy *Attack* with probability p if its type is *malicious*. In this case the expected payoff of P_j playing its pure strategy *Monitor* is :

$$U_j(\textit{Monitor}) = pp_o\Big((2\alpha-1)w_k - C_{m_k}\Big) - (1-p)p_o(\gamma w_k + C_{m_k}) - (1-p_o)(\gamma w_k + C_{m_k})$$

and the expected payoff of P_j playing its pure strategy *Not Monitor* is :

$$U_j(\textit{Not Monitor}) = -pp_o w_k$$

Similarly, the expected payoff of P_i playing its pure strategies *Attack* and *Not Attack* when P_j plays its strategy *Monitor* with probability q are :

$$U_i(\textit{Attack}) = p_o\Big(q((1-2\alpha)w_k - C_{a_k}) + (1-q)(w_k - C_{a_k})\Big) \quad \textit{and}$$

$$U_i(\textit{ Not Attack}) = 0 \ , \quad \textit{respectively.}$$

By equating $U_j(\textit{Monitor}) = U_j(\textit{Not Monitor})$, we get $p = \frac{\gamma w_k + C_{m_k}}{(2\alpha+\gamma)w_k p_o}$, which is the probability value of P_i to play its strategy *Attack* under the BNE strategy. Similarly, by equating $U_i(\textit{Attack}) = U_i(\textit{Not Attack})$, we obtain the probability value of P_j to play its strategy *Monitor* under the BNE to be $q = \frac{w_k - C_{a_k}}{2\alpha w_k}$. Therefore, when the prior probability of P_i being malicious $p_o > \frac{\gamma w_k + C_{m_k}}{(2\alpha+\gamma)w_k}$, no pure strategy BNE exists. But there exists a mixed-strategy BNE which corresponds to the strategy combination ((*Attack* with probability p if malicious, *Not Attack* if normal), *Monitor* with probability q, p_o), where $p = \frac{\gamma w_k + C_{m_k}}{(2\alpha+\gamma)w_k p_o}$ and $q = \frac{w_k - C_{a_k}}{2\alpha w_k}$.

From the mixed strategy BNE obtained above, we observe that the monitoring probability (q) of P_j does not depend on its current maliciousness belief about the player P_i, but rather influences P_i's behavior, as the probability of attack (p) is inversely proportional to the P_j's maliciousness belief about P_i. A high maliciousness belief value of the P_j about the opponent player P_i drastically reduces P_i's attacking probability p. The static Bayesian game approach described above can be used to model most types of attacks in MANETs like Denial of Service (DoS) attacks, blackhole attack [114], wormhole attack [115] etc. It enables the node operating the IDS to implement a probabilistic monitoring strategy based on the BNE of the game while at the same time maximize its expected payoff utility. However, the drawback of the static Bayesian game approach is that it is not always possible

to determine the prior maliciousness belief value (p_o) about the node being monitored in a dynamic networks like MANET. Therefore, depending on the nature of the environment, the monitoring node may assign an appropriate value for p_o. If the environment under consideration is hostile, a high value of p_o should be assigned.

4.3.2 Energy efficient MANET cluster leader node election mechanism

MANETs are characterized by energy and resource constrained wireless nodes. As a result, nodes in MANET do not have any profitable stimulus to act as the cluster leader node and perform the monitoring operation as it requires a substantial amount of energy. Therefore, a mechanism based on an incentive structure needs to be developed to encourage nodes in MANET to participate in the cluster leader node election process. Towards this end, we propose a secure MANET cluster leader election mechanism, wherein the nodes are provided with incentives in the form of enhanced reputation gain for taking up the role of the cluster leader node. In the proposed framework, the MANET topology consists of multiple clusters, with each cluster comprising a leader node that carries out the intrusion detection services for all the other cluster nodes. Re-elections are conducted after every predefined time interval to elect a new cluster leader node in order to ensure a uniform energy consumption across multiple nodes for operating the IDS.

Cluster leader election mechanism that elects a random node as the leader node [116], without considering the energy level of nodes results in premature death of nodes with low energy levels. Therefore, the election mechanism must take into consideration the energy level of nodes while electing the cluster leader node. Moreover, nodes in MANET are inherently selfish in nature in order to preserve their resources (CPU time, energy etc). Hence, in order to motivate the MANET nodes to actively participate in the cluster leader node election process by revealing their energy level values, we propose a reputation based leader node election mechanism. The elected leader node is provided with a payment in the form of enhanced reputation gain by the mechanism. Nodes with higher reputation values are considered as more trustworthy and given higher priorities in the cluster's services. The cluster leader node allocates the packet sampling budget to different nodes based on their reputation values. The sampling budget of the i^{th} node n_i (SB_{n_i}) denotes the amount of

services it is entitled to receive from the leader node and is given by:

$$SB_{n_i} = (R_i) \Big/ \sum_{j=1}^{N} R_j$$

where *N* is the total number of nodes in the cluster and R_i is the reputation value of n_i.

When a node is elected as the leader node, its reputation value increases. This motivate the nodes in the cluster to truthfully reveal their private information (energy levels) during the leader node election process. A default reputation value of R_o is assigned to all the nodes during the cluster formation period, which is later updated when a node is elected as the cluster leader. Let the energy required by the cluster leader node to operate the IDS for the elected period of time be denoted by E_{ids} and let its cost function value for carrying out the intrusion detection analysis during this period be denoted by Cst_i. We divide the *N* nodes in the cluster into *k* energy classes {$Class_1$, $Class_2$,...,$Class_k$} based on their power factor denoted by $PF_i = E_{n_i} \Big/ NT_i$, where $1 \leq i \leq N$, E_{n_i} is the energy level of node n_i and T_i is the user defined scaling factor.

$$Class\ \ of\ \ n_i = \begin{cases} Class_1, & \text{if } PF_i < \rho_1 \\ Class_d, & \text{if } \rho_{d-1} \leq PF_i < \rho_d \\ Class_k, & \text{if } PF_i \geq \rho_{k-1} \end{cases}$$

where ρ = {ρ_1, ρ_2..., ρ_{k-1}} is a set of (*k - 1*) threshold values. The cost function value of the node $n_i \in Class_a$ ($1 \leq a \leq k$) for analyzing the other cluster nodes' data packets for specified period of time is given by:

$$Cst_i = \begin{cases} \left(\frac{\lambda * SB_{n_i}}{PF_i}\right) = \lambda * \left(\frac{R_i}{\sum_{j=1}^{N}(R_j)}\right) * \frac{NT_i}{E_{n_i}}, & \text{if } E_{n_i} \geq E_{ids} \\ \infty, & \text{if } E_{n_i} < E_{ids} \end{cases}$$

where $\lambda \in$ [0,1] is the sampling budget weighing factor. If the energy level of a node n_i is less than the threshold energy required for carrying out intrusion detection analysis i.e., if $E_{n_i} < E_{ids}$, then node n_i cannot be elected as the cluster leader since its cost function value would be infinite.

To motivate the nodes in the cluster, including the selfish ones for cooperation, we model the cluster leader node election process as a game with nodes as its players. Each node n_i in the cluster holds a confidential information about its type (θ_i). The type of θ_i can be either

Normal or *Selfish*. The payoff utility function of node n_i with type θ_i when it is elected as the leader node is given by:

$$U_i(\theta_i, \theta_{-i}) = P_i - W_i(\theta_i\ ,\ O(\theta_i, \theta_{-i})) \tag{4.1}$$

where

- θ_{-i} represents the types of all other cluster nodes except node n_i
- $O(\theta_i,\theta_{-i}) = O(\theta_1,..\theta_i..,\theta_N)$ is the output of the game corresponding to the types chosen by the nodes.
- $W_i = Cst_i$ is the cost incurred by n_i for providing intrusion detection services when it is elected as the leader node.
- $P_i \in \mathbb{R}$ is the payment provided by the mechanism to n_i in the form of enhanced reputation gain when it is elected as the leader node.

Each node n_i in the given cluster seeks to maximize its payoff utility function $U_i(\ )$. It signifies the gain of n_i, if it follows the type θ_i. The node n_i might not truthfully reveal its true cost function value (Cst_i) by either under-valuing or exaggerating Cst_i, if doing so leads to a better payoff utility for n_i. Therefore, we need to devise a mechanism with truth-telling as a dominant strategy to encourage nodes to truthfully reveal their true cost function values.

The cluster leader node election process begins with each node n_i in the cluster selecting its type θ_i and evaluating its cost function value W_i. The primary objective of the leader node election mechanism is to elect the node n_i with the least cost function value (Cst_i) as the cluster leader node. Since $Cst_i \propto 1\Big/E_i$, electing the node with the least cost function value as the leader node is equivalent to electing the node with the highest energy level value. We refer to this objective as the Social Choice Function (SCF) and define it as:

$$SCF = Min\Big\{W_i(\theta_i\ ,\ O(\theta_i, \theta_{-i}));\qquad i = 1, 2, ..., N\Big\} \tag{4.2}$$

If two or more nodes in the cluster have the same cost function value, then the node with the highest reputation value amongst them will be elected as the cluster leader node by the mechanism. Payment in the form of enhanced reputation gain is made to the elected leader node by employing the VCG mechanism [38]. The payment P_i received by the leader node

n_i is equal to the second least cost function value Cst_j excluding the cost function value of the elected leader node n_i and is given by the Equation 4.3.

$$P_i = Min\Big\{W_j(\theta_j\ ,\ O(\theta_j, \theta_{-j}); \quad n_j \neq n_i\Big\} \tag{4.3}$$

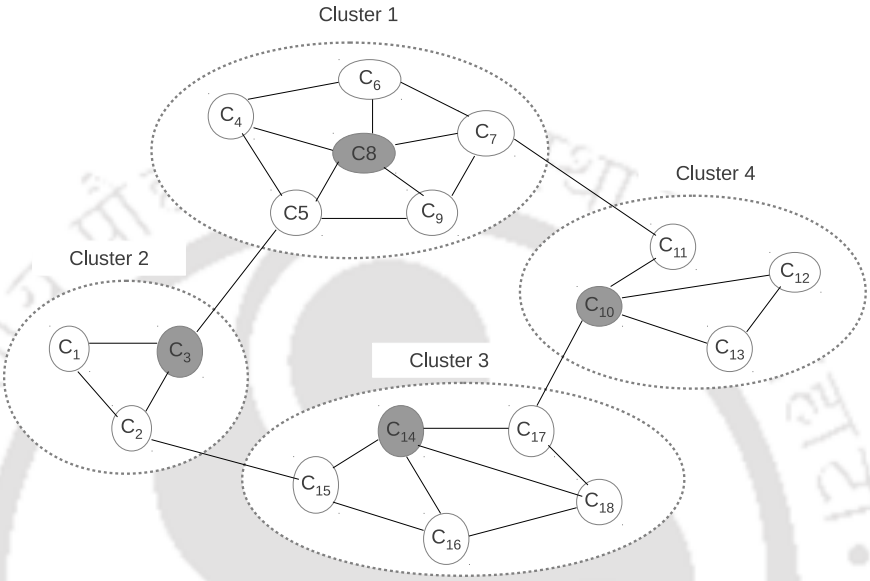

Figure 4.4: MANET topology with leader IDS

As shown in the Fig. 4.4, the proposed IDS framework models the MANET topology as a set of multiple clusters. The leader node election mechanism computes the SCF value for each cluster based on the cost function values of nodes in that cluster. This ensures that all the nodes in a given cluster elect the same leader node. Algorithm 1 illustrates the proposed distributed leader node election mechanism. Initially a random node n_i initiates the leader node election process by broadcasting the *Begin_Election( )* message to all other nodes in the cluster. The *Begin_Election( )* message contains the hash value *H( )* corresponding to the *Election( )* message to be sent by the node n_i later on. Other nodes in the cluster use this hash value to authenticate and verify the *Election( )* messages received from n_i. T_1 in the *Begin_Election( )* message specifies the duration of the election process. All the participating nodes must broadcast their *Begin_Election( )* and *Election( )* messages within the time period T_1 after the node n_i has started the election process. Those nodes that do not participate in the exchange of *Begin_Election( )* and *Election( )* messages are excluded from the cluster's services.

After successfully broadcasting the *Begin_Election( )* message, node n_i broadcasts the

Algorithm 1 Distributed cluster leader node election algorithm

Input : *Cluster I with 'N' number of nodes.*

Output : *Elected cluster leader node of I.*

$n_i \rightarrow cluster^I_{-n_i}$: *Begin_Election* $\Big(\text{H}(ID_{n_i}, Cst_i, TS_i), T_1\Big)$

$n_i \rightarrow cluster^I_{-n_i}$: *Election* $\Big(ID_{n_i}, Cst_i, TS_i\Big)$ */* Nodes in I exchange the election message containing their cost function values */*

Let node n_j ($Leader_{IDS}$) be the node with the least cost function value (Cst_j) in I.

$\forall\, n_i \in I$

if $n_i \neq Leader_{IDS}$; **then**

$n_i \xrightarrow{Elected} Leader_{IDS}$ */* n_i informs $Leader_{IDS}$ that it is the leader node. */*

$Leader_{IDS} \xrightarrow{ACK} n_i$ */* $Leader_{IDS}$ acknowledges that it is leader node */*

$n_i \xrightarrow{Pay} Leader_{IDS}$ */* n_i makes payment to $Leader_{IDS}$ */*

else

$n_i \xrightarrow{ACK} cluster^I_{-n_i}$

$cluster^I_{-n_i} \xrightarrow{Pay} n_i$

end

Election( ) message containing its identity (ID_{n_i}), its cost function value (Cst_i), and the time stamp TS_i to every other nodes in the cluster. The nodes receiving the *Election( )* message then verify that it indeed came from n_i by generating a hash value **H*( )** of the received *Election( )* message. This generated hash value is then compared with the hash value **H( )** received in *Begin_Election( )* message earlier. Upon successful verification, each node in the cluster computes the SCF value, which is the least cost function value as defined in Equation 4.2. Finally, the node with the least cost function value is elected as the leader node by the algorithm. When n_i is elected as the leader node, it is provided with a payment in the form of enhanced reputation gain by the mechanism. The payment value is equal to the second least cost function value of the node excluding that of the leader node, as defined in Equation 4.3. The leader node election process is conducted after every time interval (T_{elect}) to elect a new leader node. Re-election is also conducted when the elected leader node quits the cluster before the completion of T_{elect} time interval.

We illustrate the proposed leader election mechanism with an example in Table 4.3. The reputation values, energy levels and sampling budget rates of different nodes at the i^{th} round are shown in the 1^{st}, 2^{nd} and 3^{rd} rows of the table, respectively. The election of the new leader node for the $(i+1)^{th}$ round requires every node to compute its corresponding

cost function value Cst_i as shown in 4^{th} row of the table using the following equation:

$$Cst_i = \left(\frac{\lambda * SB_{n_i}}{PF_i}\right) = \lambda * \left(\frac{R_i}{\sum_{i=1}^{N}(R_i)}\right) \times \frac{NT_i}{E_{n_i}}$$

Table 4.3: Leader IDS election example

Nodes	N_1	N_2	N_3	N_4	N_5	N_6
i^{th} Round Reputation	7	9	2	4	5	3
i^{th} Round Energy	5	6	4	5	10	7
i^{th} Round Sampling(%)	23.33	30	6.66	13.33	16.66	10
i^{th} Round Cost Valuation (Cst_i)	0.28	0.30	0.1	0.16	0.1	0.09
$(i+1)^{th}$ Round Reputation	7	9	2	4	5	3.1
$(i+1)^{th}$ Round Energy	5	6	4	5	10	6.8

In Table 4.3, the total number of node (N) is 6. The values of λ and T_i are assumed to be 0.1, and 10, respectively. Similarly, the energy required for operating the IDS is assumed to be 0.2 units. Since node N_6 has the least cost analysis value (0.09), it is elected as the new leader node. The payment for the elected leader node N_6 is then calculated, which is equal to the 2^{nd} least cost function value i.e., P_i = 0.1 unit. All the nodes increase the reputation value of the N_6 by 0.1 unit in their reputation table. The new reputation values of various nodes at the $(i+1)^{th}$ round is shown in the 5^{th} row of the table. The payoff utility of node N_6 calculated using Equation 4.1 is 0.1 - 0.09 = 0.01, which represents the gain obtained by N_6 for taking up the role of the cluster leader node.

4.3.2.1 Cluster leader node election mechanism analysis

The primary objective of the proposed leader node election mechanism is to encourage nodes in the cluster to truthfully reveal their cost function value by providing them incentives in the form of enhanced reputation gain. In this section, we validate our mechanism design to ensure that it meets the cost-efficiency and truthfulness properties even in the presence of selfish nodes in the cluster. This is validated by demonstrating that truth-telling is the dominant strategy of the proposed mechanism design.

We consider two untruthful revelation of the selfish node n_i namely, *under-declaration* and *over-declaration* of its cost function value Cst_i. We show that in both these cases it is never better off compared to when it truthfully reveals its cost function value. Node n_i

may under-declare its cost function value by revealing a false value Cst_i^* ($Cst_i^* < Cst_i$) and win the cluster leader node election. However, under-declaring its cost function value will not benefit n_i for the following two reasons. In the 1^{st} case if Cst_i is already least among all the nodes, then under-valuing the cost function value to Cst_i^* does not increase n_i's payment, since payment is made on the basis of the second least cost function value. Therefore, n_i's payoff utility U_i remains unchanged since it is calculated with respect to its real cost analysis value Cst_i. On the other hand, if n_i does not have the least cost function value but wins the election by declaring a fake under-valued cost function value Cst_i^* then it leads to a negative payoff utility for n_i. This is because the payment P_i received by n_i is less than the real cost function value Cst_i.

Similarly, if n_i over-declares its cost function value by declaring a fake Cst_i^*, such that $Cst_i^* > Cst_i$, then such a strategy would never increase the payoff utility of n_i for the following two reasons. First, if n_i indeed has the least cost function value Cst_i, then pursuing this strategy leads to n_i not being elected as the leader node and hence it loses the payment. Second if the real cost function value Cst_i of n_i is not the least among all the nodes, then this strategy would not increase its payoff utility, as n_i would not be elected as the leader node.

4.3.2.2 Cooperative catch and punish model

A cooperative detection mechanism is required to monitor and detect misbehaving leader node. The leader node is said to be misbehaving if it does not provide intrusion detection services to the cluster nodes proportional to their reputation values. Checker nodes are used to monitor the leader node for sign of misbehavior. A distributed election algorithm to elect k checker nodes for monitoring the leader node is given by Algorithm 2 . Let Chk_{cost} be the cost incurred by the checker node to monitor the leader node. Incentives in the form of checker payments (P_{chk}) are provided to the checker nodes for monitoring the leader node, such that P_{chk} - $Chk_{cost} > 0$.

If the leader node ($Leader_{IDS}$) is found to be misbehaving by the checker nodes, the mechanism punishes $Leader_{IDS}$ by asking all the cluster nodes to decrement the reputation value of the leader node in their reputation table by value P_i as calculated in Equation 4.3. Leader node election process is then conducted to elect a new cluster leader node. To detect a misbehaving leader node, a set of detection level given by $DL = \{dl_1, dl_2, .., dl_j\}$ is proposed with each detection level representing the severity of the misbehaving leader

Algorithm 2 Distributed checker nodes election algorithm

Input : *Cluster I with 'N' number of nodes*
Output : *'k' checker nodes of I*

$n_i \xleftrightarrow{Cst_i} Cluster^I_{-n_i}$ */* Nodes in I exchange their cost function values */*

Let $\{k\text{-}CHK\}$ *be the set of 'k' nodes in I with the least cost function values (excluding* $Leader_{IDS}$*)*

$\forall\ n_i \in I$

if $n_i \notin \{k\text{-}CHK\} \parallel n_i \neq Leader_{IDS}$; **then**

- $n_i \xrightarrow{Chk_{ele}} \{k\text{-}CHK\}$ */* n_i informs nodes in $\{k\text{-}CHK\}$ that they are checkers*/*
- $\{k\text{-}CHK\} \xrightarrow{ACK} n_i$ */* nodes in $\{k\text{-}CHK\}$ acknowledges */*
- $n_i \xrightarrow{P_{chk}} \{k\text{-}CHK\}$ */* nodes in $\{k\text{-}CHK\}$ are provided with payment by n_i */*

else

- After time T_2
- $n_i \xrightarrow{ACK}$ *(N - k) non checker nodes* */* if n_i is checker */*
- *(N - k) non checker nodes* $\xrightarrow{P_{chk}} n_i$

end

node. A threshold set $T = \{t_1, t_2, ..., t_{j-1}\}$ is defined for categorizing the misbehaving detection levels. Setting the threshold value above which the leader node is considered to be misbehaving is crucial. Setting a high threshold value increases the false positive (FP) rate, wherein even the sincere leader nodes are penalized. On the other hand, setting a low threshold value increases the false negative (FN) rate, wherein the mechanism fails to detect the misbehaving leader node. Therefore, this value must be set appropriately so as to maintain a good trade-off between the FP and FN rates.

Let $Chk_{set} = (Ck_1, Ck_2, ..., Ck_k)$ be the set of 'k' checker nodes and let $S_{set} = (n_a, n_b, ..., n_k)$ be the set of nodes monitored by the checker nodes such that $|Chk_{set}| = |S_{set}|$. Each $Ck_i \in Chk_{set}$ monitors one particular node $n_j \in S_{set}$. We define an aggregate function of checker nodes in the Chk_{set} as:

$$T(n) = \sum_{Ck_i \in\ Chk_{set}\ \&\ n_j \in S_{set}} (R_{Ck_i}) * f(j) \tag{4.4}$$

where R_{Ck_i} is the reputation value of the checker node Ck_i and *f(j)* is the catch function defined as the ratio of number of node n_j's data packets analyzed by the leader node to the actual sampling budget allocation of node n_j. We then classify the action of leader node

into one of the following detection levels:

$$DL = \begin{cases} dl_1, & \text{if } T(n) < t_1 \\ dl_f, & \text{if } t_{f-1} \leq T(n) < t_f; f \in [2, j\text{-}1] \\ dl_j, & \text{if } T(n) \geq t_{j-1} \end{cases}$$

Categorizing the severity of leader node's misbehavior into *j* different levels minimizes the FP rate while determining the misbehaving leader node. If the detection level (*DL*) of the leader node falls below the threshold level (dl_{th}) then it is assumed to be misbehaving and penalized by computing its payment in negative. It is also temporarily debarred from the cluster. This catch and punish mechanism acts as a deterrence and discourages the leader node from misbehaving.

4.3.3 Proposed Hybrid MANET IDS

In sub-section 4.3.1, we modeled the interaction between the IDS and the node being monitored as a static Bayesian game, wherein the IDS has a fixed prior maliciousness belief value (p_o) about the node being monitored. However, determining this prior maliciousness belief value is usually not that straightforward. Nodes in MANET are usually energy constrained and may become less responsive as their energy levels drain out. Additionally, some of the trustworthy nodes may be compromised and made to act maliciously. Taking these factors into consideration, the IDS needs to be re-evaluate and update the maliciousness belief value of the node being monitored at regular interval. In this sub-section, we extend the static Bayesian game to a multi-stage dynamic Bayesian game, wherein the IDS periodically updates its maliciousness belief value about the node being monitored as the game evolves.

In the multi-stage Bayesian game, the game is played repeatedly after every time interval t_k. The IDS is represented as the defender player (P_j) and the node bing monitored is represented as the potential attacker player (P_i). The payoffs of the game and the identities of the players remain the same throughout each iteration of the game. However, the strategies of the players in the dynamic game depends on the history profile of the game. At any stage t_k of the game, the optimal strategy of P_i depends on the maliciousness belief value of P_j about P_i. P_j's initial belief about P_i being malicious at the first stage (t_0) of the game is given by the prior probability p_o. P_j later updates its malicious belief value about P_i at the

k^{th} stage of the game by evaluating its posterior belief $p_j(\theta_i|a_i(t_k), a_i(t_{k-1}))$, where $a_i(t_k)$ and $a_i(t_{k-1})$ represent the actions taken by P_i at the k^{th} and $(k-1)^{th}$ stage of the game. P_j evaluates its posterior belief about P_i using the following Bayes' rule.

$$p_j(\theta_i \mid a_i(t_k), a_i(t_{k-1})) = \frac{p_j(\theta_i \mid a_i(t_{k-1}))P(a_i(t_k) \mid \theta_i, a_i(t_{k-1}))}{\sum_{\tilde{\theta_i}} p_j(\tilde{\theta_i} \mid a_i(t_{k-1}))P(a_i(t_k) \mid \tilde{\theta_i}, a_i(t_{k-1}))} \tag{4.5}$$

where P($a_i(t_k)|\theta_i$, $a_i(t_{k-1})$) is the probability that P_i plays the action $a_i(t_k)$ at the k^{th} stage, given that the type of P_i is θ_i and its action at the $(k-1)^{th}$ stage was $a_i(t_{k-1})$.

From Equation 4.5, it can be observed that the IDS needs to continuously monitor the node at every stage of the game to update its maliciousness belief value. However, operating IDS in an always-on promiscuous mode is not an energy-efficient strategy and may lead to a premature death of the node operating the IDS. Therefore, to minimize the energy spent on operating the IDS, a two layered hybrid IDS detection model is proposed. The proposed hybrid model consists of one lightweight module and one heavyweight module. The former module is less powerful but requires less energy for its operation, while the latter module is more powerful but requires more energy to operate. By default only the lightweight module is activated initially.

As shown in the Fig. 4.2, the lightweight IDS module updates the malicious belief value of node n_i using the observed behaviors of n_i at the current and the previous stage of the game using the Bayes rule. The lightweight IDS module computes two parameters of n_i namely, its Packet Reception Rate (PRR) and the Packet Forwarding Rate (PFR). (In Fig. 4.2 only the PFR calculation is shown). If the PRR (or PFR) value of n_i exceeds (or falls below) the threshold value, then the action of n_i is assumed to be malicious and the heavyweight module is activated in the next stage of the game for more rigorous analysis. The maliciousness value of n_i can be unilaterally reset to lower value by the heavyweight IDS module if n_i acts normally for a predefined period of time after being reported by the lightweight IDS module. After the maliciousness value of n_i is reset to lower value, the heavyweight IDS module is turned off and the lightweight IDS module is turned on again. This process is repeated over the period of time and only one of the IDS module is activated at any given time.

4.3.3.1 Heavyweight Intrusion Detection System (HIDS)

The HIDS uses an unsupervised association-rule mining technique [117] [118] on a set of packet-level transmission events to find the association patterns. The extracted association rules are then used to build the normal profile of the network. There is a trade-off between effectiveness and efficiency while selecting the feature set for analysis. A higher number of input features help the IDS to detect various types of attacks with high accuracy and detection rate. However, it also results in a higher energy consumption and increased computational overhead. Therefore, considering the energy constrained nodes in MANET, optimal number of input features are selected for developing the normal profile of the network. The transmission events used for finding the association patterns consist of features listed in Table 4.4, which are extracted from the MAC and network layer at a predefined sampling rate. A brief description about each of these features are provided below:

- *Packet event type* : This feature represents the type of the transmission event taking place.
- *Sender Address*: This feature represents the MAC address of the sender node.
- *Destination Address*: This feature represents the MAC address of the destination node.
- *MACFrameType*: This feature represents the type of MAC frame observed in the transmission event.
- *RoutPktType*: This feature represents routing control packets (routingCtrlPkt) like Route Request, Route Reply, Route Error etc., and data packets (routingDataPkt) from network layer.
- *Route change percentage*: It is defined as ($|S_2$ - S_1 $|+$ $|S_1$ - S_2 $|)/|S_1$ $|$), where, S_2 and S_1 are the number of routing table entries before and after the observation. (S_2 - S_1) indicates the newly increased routing entries and (S_1 - S_2) indicates the deleted routing entries during the time interval (t_2 - t_1).

The HIDS uses multiple segments of training data set to extract the association rules. These rules are then aggregated to build the normal profile. The association rule describes the association of attributes within transaction records of an audit data set. Let T = {T_1, T_2,...,T_n} be the set of *n* transaction records and F = {F_1, F_2,...,F_k} be a *k* feature set

Table 4.4: HIDS feature set

Features	Values
Packet event type(Event)	SEND, RECV, DROP, FWD
Sender Address(SA)	$SrcMAC_i$
Destination Address(DA)	$DestMAC_i$
MACFrameType	RTS, CTS, DATA, ACK
RoutPktType	routingCtrlPkt, routingDataPkt
Route change percentage	PCR

defined over T. A transaction record T_i is a collection of *k*-tuple features i.e., T_i = {f_1, f_2,..., f_k}, where f_k represents a value from the k^{th} feature F_k.

Let A and B denote two disjointed item subset in T_i. The support of item subset A denoted by *sup(A)* represents the percentage of transactions containing A in T. Similarly, the support of A and B denoted by *sup(A* $\cup$ *B)* represents the percentage of transactions containing both A and B. The association rule between A and B is given as $A \Rightarrow B, (s, c)$, where s = *sup(A* $\cup$ *B)* and c = $\frac{sup(A \cup B)}{sup(A)}$ are defined as the support value and confidence value of the association rule, respectively. The rule holds good if $s \geq$ *minsup* and $c \geq$ *minconf*, where *minsup* and *minconf* denote the predefined minimum support threshold and minimum confidence threshold values, respectively.

Apriori algorithm [117] is used to build the association rules for the normal profile. The algorithm mines the frequent itemsets from the transactional dataset and uses an iterative approach to find itemsets of larger size at each iteration. The algorithm works on the principal that any subset of a frequent itemset must also be a frequent itemset. Therefore, the algorithm reduces the number of item candidates being considered by only exploring the itemsets whose support count is greater than the minimum support count. For our analysis we have used *minsup* and *minconf* values as 15% and 70%, respectively.

A transaction record is a packet level event with the following format ⟨*Event, SA, DA, MACFrameType, RoutPktType*⟩. An example association rule is *(*$SrcMAC_6$*, routingCtrlPkt* $\rightarrow$ $DestMAC_{15}$*, RECV),(0.35,1)*, which describes an event pattern related to the RECV flows of the monitoring node i.e., 35% of transaction records match the event of “node 6 sends data packets to node 15", and when node 15 receives data packets, they are 100% of the time from node 6. Another association rule example is *(*$SrcMAC_3$*, routingCtrlPkt* $\rightarrow$ $DestMAC_7$*, PCR),(0.20,0.80)*, which indicates that route change between node 3 and node 7 constitute 20% of total route change in the network and 80% of changes in node 7’s route is related with change in node 3’s route.

The association rules extracted from the test data (real time data) are correlated with the association rules in the normal profile and any deviation of the test association rules from the normal profile is considered as an anomalous by the HIDS module.

4.3.3.2 Lightweight Intrusion Detection System (LIDS)

Operating the association-rule based HIDS module in an always-on promiscuous mode consumes a significant amount of energy, since it has to analyze massive packet-level transmissions of network and MAC layers to detect intrusions. Therefore, an alternative lightweight monitoring system (LIDS) is proposed to update the maliciousness value of the node being monitored at every stage of the game. LIDS uses simple rules to detect intrusions and identify the malicious nodes. It uses two different approaches for detecting the inbound and outbound attacks. Following inbound attacks are consider in our study: *Sleep deprivation, Flooding, DoS* and *Forging attack*. The outbound attacks considered in our study are *Black hole attack* and *packet dropping attack*. Let N_j be the set of nodes monitored by the defender node P_j and let $P_i \in N_j$ be the potential attacker node. Let $R_j^i(t_k)$ denote the number of data packets received by P_j from P_i during the game stage t_k.

We define the following terminologies to determine the outbound and inbound attacks: Packet Reception Rate *(PRR)* and Packet Forwarding Rate *(PFR)*. The *PRR* of P_j from P_i for game stage ($\phi_j^i(t_k)$) is defined as the rate of inbound data traffic from P_i to P_j with respect to the total data traffic rate in the vicinity of P_j. It is given by the following equation:

$$\phi_j^i(t_k) = \frac{R_j^i(t_k)}{\sum_{a \neq b} R_{a \in N_j}^{b \in N_j}(t_k) + R_j^{b \in N_j}(t_k)} \tag{4.6}$$

If the value of *PRR* is greater than the threshold value τ, then P_i is assumed to be carrying out an inbound attack.

The *PFR* of P_i for game stage t_k is defined as the ratio of number of packets received by P_i from its neighboring nodes to the number of packets forwarded by P_i and is given by:

$$\psi_i(t_k) = \frac{R_i^{j \in N_j}(t_k)}{R_{k \in N_j}^i(t_k)} \tag{4.7}$$

P_i is assumed to be carrying out an outbound attack if the value of $\psi_i(t_k)$ is less than the threshold value Θ.

The *PRR* and *PFR* threshold values τ and Θ used for determining the inbound and the outbound attacks influence the performance of the LIDS module. These threshold values can be determined experimentally from the normal data traffic patterns. Using these simple LIDS rules as a precursor before applying the heavyweight association-rules of the HIDS can significantly lower the False Positive (FP) rate and energy consumption of the overall IDS framework.

Let the detection rate and FP rate of LIDS be α_L and γ_L, respectively. Let P($a_i(t_k)|\theta_i$, $a_i(t_{k-1})$) be the conditional probability of the player P_i to play its action $a_i(t_k)$ at the k^{th} stage of the game, given its type θ_i and its action at the $(k-1)^{th}$ stage was $a_i(t_{k-1})$. This conditional probability can be updated using the following Equations:

$$P(a_i(t_k) = Attack \mid \theta_i = 1, a_i(t_{k-1})) = p\alpha_L + (1-p)\gamma_L \tag{4.8}$$

$$P(a_i(t_k) = Not\,Attack \mid \theta_i = 1, a_i(t_{k-1})) = p(1-\alpha_L) + (1-p)(1-\gamma_L) \tag{4.9}$$

$$P(a_i(t_k) = Attack \mid \theta_i = 0, a_i(t_{k-1})) = \gamma_L \tag{4.10}$$

$$P(a_i(t_k) = Not\,Attack \mid \theta_i = 0, a_i(t_{k-1})) = 1 - \gamma_L \tag{4.11}$$

In the above equations, *p* represents the probability of the malicious player P_i to play its strategy *Attack* under Nash Equilibrium (NE). Similarly, (1 - α_L) and (1 - γ_L) represent the false negative (FN) rate and the true negative (TN) rate of the LIDS, respectively. The LIDS can determine the action of the node P_i using Equation 4.6 and Equation 4.7. It then updates the maliciousness value of the player P_i using Equation 4.5.

Numerical Example

Continuing with our standard notation, let α and γ be the detection rate and FP rate of the heavyweight IDS, respectively. Similarly, let α_L and γ_L be the detection rate and FP rate of the lightweight IDS, respectively. Consider a defender attacker game interacting

over a node n_k. Let C_{m_k} and C_{a_k} be the cost associated with monitoring and attacking node n_k. Let the asset value of n_k be w_k. In previous sections, we have developed the BNE of the game, which corresponds to the strategy combination (p^*, q^*, $p(\theta_i)$), where p^* = $\frac{\gamma w_k + C_{m_k}}{(2\alpha+\gamma)w_k p(\theta_i)}$ is the attacking probability of the attacker player (P_i), q^* = $\frac{w_k - C_{a_k}}{2\alpha w_k}$ is the monitoring probability of the defender player P_j and $p(\theta_i)$ is the maliciousness belief of P_j about P_i, which is given by Eqn. 4.5. Consider a heavyweight and a lightweight module with following values, α = 0.9178, γ = 0.0025, α_L = 0.833 and γ_L = 0.0029. Let w_k = 9.45 and C_{a_k} = C_{m_k} = w_k/1000. Assume that the initial belief of P_j about P_i being malicious is 0.5, i.e. initial value of $p(\theta_i)$ = 0.5. Therefore, the probability of player P_i playing its strategy attack for the 1^{st} stage of the game is p^* = $\frac{0.0019}{p(\theta_i)}$ = $\frac{0.0019}{0.5}$ = 0.0038. Similarly, the monitoring probability q^* = 0.5442. Next, we update the malicious belief of player P_i under following conditions:

Case 1: The observed action of P_i by the lightweight module of P_j is *Attack*:

$$p(\theta_i = 1)(t_1) = \frac{p(\theta_i = 1)(t_0)\ P(a_i(t_1) = Attack \mid \theta_i = 1, a_i(t_0))}{\sum_{\tilde{\theta}} p(\tilde{\theta_i})(t_0)\ P(a_i(t_k) = Attack \mid \tilde{\theta_i}, a_i(t_0))} = 0.6756$$

Case 2: The observed action of P_i by the lightweight module of P_j is *Not Attack*:

$$p(\theta_i = 1)(t_1) = \frac{p(\theta_i = 1)(t_0)\ P(a_i(t_1) = Not\ \ Attack \mid \theta_i = 1, a_i(t_0))}{\sum_{\tilde{\theta}} p(\tilde{\theta_i})(t_0)\ P(a_i(t_k) = Not\ \ Attack \mid \tilde{\theta_i}, a_i(t_0))} = 0.49920$$

From the above results, it can be observed that when the action of P_i is detected as an *Attack* by P_j (defender) then the maliciousness belief value of P_j about P_i increases, which in turn decreases the probability of P_i to play its strategy *Attack* in the next stage of the game. On the other hand, when the action of P_i is detected as *Not Attack* by P_j, then P_j's malicious belief about P_i decreases, which increase the probability of P_i to play its strategy *Attack* in the next stage of the game. It can also be observed that the proposed hybrid MANET IDS reduces the power consumption by activating the heavyweight IDS module 54.42% of the time instead of turning it on 100% of the time.

Summarizing the above results and discussion, we conclude that the monitoring probability of the P_j does not depend on its current maliciousness belief value about P_i, but rather influences the P_i's behavior. A high maliciousness belief value results in P_i drastically reducing its attack. This is due to fact that both P_i and P_j are rational players, and the cost and maliciousness beliefs are common information known to both the players.

4.4 Experimental Results

Since our work comprises of two different components, we classify our result and analysis into following two subsections:

- Analyze the effectiveness of the proposed MANET leader election mechanism.
- Analyze the hybrid MANET IDS on following parameters.
    1. Evaluate the detection rate and the FP rates of the lightweight module and the heavyweight module of the proposed hybrid MANET IDS.
    2. Evaluate the payoff utilities of the attacker and defender nodes under different BNE strategies.
    3. Analyze the reduction in IDS traffic volume achieved by the proposed MANET IDS framework.
    4. Analyze the performance of the proposed IDS framework with other MANET IDS frameworks.

The proposed IDS framework is implemented using the network simulator NS2 [119] on Ubuntu 12.04 running gcc version 4.6.3. The movements of the mobile nodes are restricted to a predefined flat-grid area of 15 $\times$ 15 m^2. Table 4.5 shows various parameters used in the simulation.

4.4.1 MANET leader election mechanism analysis

This subsection shows the impact of the proposed leader node election mechanism on the average life span of MANET nodes. Initially nodes in the cluster are assigned energy levels between 5-50 Joules. The energy consumed by the leader node for elected period of time (15 sec) is assumed to be 4 Joules. The energy required by nodes for their normal operations and transmissions have been ignored to simplify the analysis.

The proposed cluster leader node election model is analyzed in a cluster consisting of 12 nodes, with 3 malicious nodes. Fig. 4.5 and Fig. 4.6 show the energy levels of different nodes using the random leader election model and the proposed (VCG) leader election model, respectively. It can be observed that in the random model some of the nodes die out over a period of time, while the energy levels of other nodes remain constant or decrease

Table 4.5: Parameters used for simulation

Parameters	Value
Simulation Time	900-3000 sec
Number of Nodes	12 - 30
Simulation Area	600 $\times$ 600 m^2
Transmission Range	150 m
Mobility	Random way point
Routing Protocol	DSR
MAC layer	DCF of IEEE 802.11
Max. node movement speed	20 m/sec
Pause Time	500 sec.
Traffic type	CBR/UDP
Election Period	60 sec
Data rate	20kbps
Packet size	512 Bytes
MAC protocol	IEEE 802.11b
Sampling interval	3 sec

marginally. On the other hand, the proposed VCG mechanism based leader node election model elects the node with the least cost function value (high-energy level node) as the leader node at every stage and hence uniformly distributes the energy consumption required for performing the monitoring operation across multiple nodes. In general, it was found that the proposed model increases the average lifetime of the cluster node by 15-20% compared to the random model.

Fig. 4.7 shows the percentage of normal alive nodes versus percentage of malicious nodes in a cluster consisting of 20 nodes after 2400 sec. Malicious nodes avoid being elected as the leader node by exaggerating their cost function value. It can be observed that as the number of malicious node increases, the number of normal alive nodes decreases. This shows that the normal nodes are frequently elected as the leader node and die out faster, as the number of selfish nodes increases in the cluster.

4.4.2 Hybrid MANET IDS analysis

For analyzing the proposed hybrid MANET IDS, the *Packet Reception Rate (PRR)* threshold (τ) and *Packet Forwarding Rate (PFR)* threshold (Θ) values of the lightweight module are taken as 0.5 and 0.3, respectively. The observed detection rate (α_L) and false positive rate (γ_L) of the lightweight module against different types of attacks like *DoS, Packet drop-*

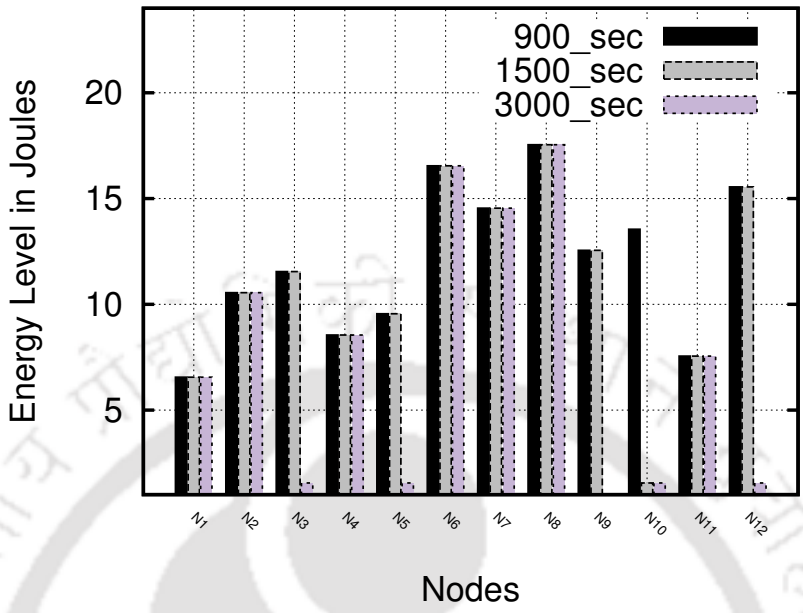

Figure 4.5: Energy consumption using random leader node election model

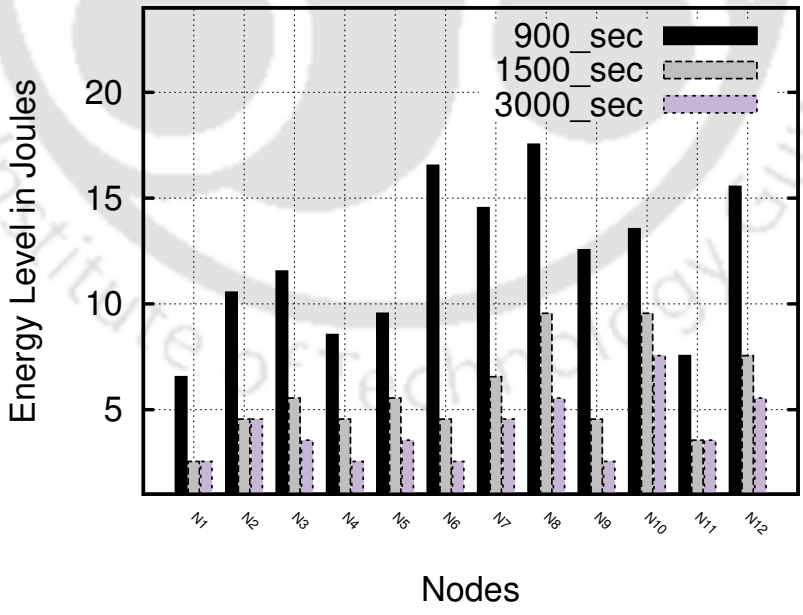

Figure 4.6: Energy consumption using proposed VCG mechanism based leader node election model

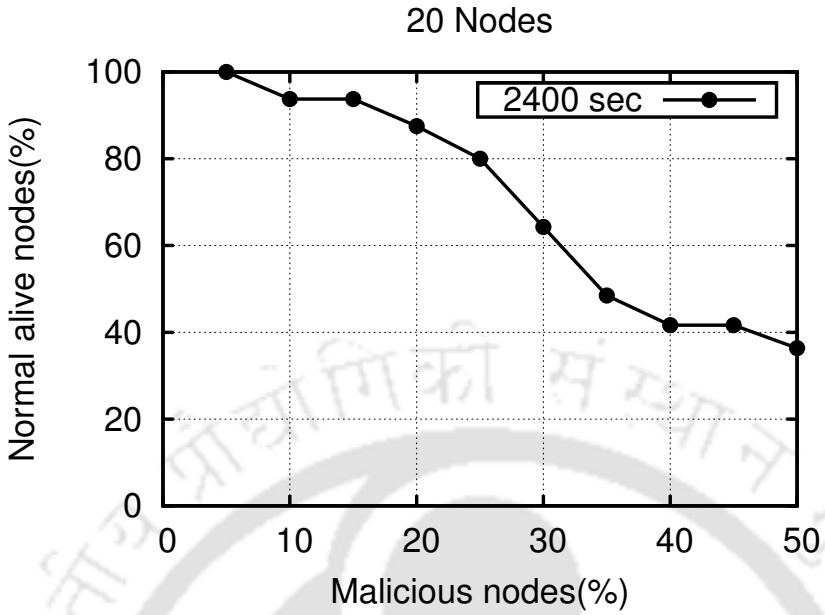

Figure 4.7: Percentage of normal alive nodes versus percentage of malicious nodes

ping, Packet distortion, Route compromise, Black-hole etc., using the above *(PRR)* and *(PFR)* threshold values are found to be 81.33% and 0.61%, respectively.

The features listed in Table 4.4 are used to build the association rules for the heavyweight IDS module. We consider different sampling intervals for creating a training dataset, with each training instance containing a summary statistics of network activities for the specified time interval. The values of minimum support threshold (*minsup*) and minimum confidence threshold (*minconf*) are taken as 15% and 65%, respectively.

The performance analysis of association-rule based *HIDS* module is carried out under different traffic conditions and against different types of attacks. Two different test scripts are used to generate training traces. 8k Trace and 5k Trace are normal training traces without any intrusions and with running time of 8000 sec and 5000 seconds, respectively. The sampling rate of five sec is used to record the feature values. The association rules extracted from these traces are then used to build the normal profile of the network.

Larger test traces with execution time from 10000 (10k) seconds to 15000 (15k) seconds are then generated. The association rules extracted from these test data (*real-time monitoring data*) are then compared with the association rules in the normal network profile. Any deviation of test association rules from the normal network profile are considered as anomaly, which triggers an intrusion alert. The test traces contain various types of attacks

like *Route compromise, Traffic distortion and Black-hole attacks*. A brief description about these attack types are provided below:

- *Route compromise*: This type of attack either involves forwarding a packet to an incorrect node or propagating false route updates.
- *Traffic distortion*: These attacks change the normal traffic behavior by randomly dropping packets, generating packets with faked source address, reporting false misbehavior against normal node, corrupting the packet contents and Denial of Service (DoS).
- *Black-hole attack*: In this attack, a malicious node advertises spurious routing information, thus receiving packets from its neighboring nodes. However, instead of forwarding the received packets, it drops them all.

Table 4.6: Performance of association-rule based heavyweight IDS module on different class of attacks

Attack Type	**Detection rate**	**False alarm rate**
Route compromise	91.4 %	0.45 %
Traffic distortion	95.3 %	0.87 %
Black-hole	99.5 %	0.35 %

Table 4.7: Performance of association-rule based heavyweight IDS module on different test traces

Test trace	**Detection rate**	**False alarm rate**
10k	92.39 %	0.45 %
12k	91.68 %	0.52 %
15k	91.28 %	0.53 %

Table 4.6 shows the performance of the proposed unsupervised association-rule based *HIDS* module against different types of attacks. It can be seen that the *HIDS* module effectively detects various type of attacks with relatively low False Positive (FP) rate. Table 4.7 shows the detection rate and FP rate of the *HIDS* module on the test traces of different sizes. The average detection rate and false alarm rate of the *HIDS* module on these test traces are 91.78% and 0.5%, respectively.

Fig. 4.8 shows the attacker's payoff corresponding to two different pure strategies of the defender. Similarly Fig. 4.9 shows the defender's payoff corresponding to two different pure strategies of the attacker. It can be observed from these figures that the payoff of the opponent player increases when the player deviates from its chosen NE strategy.

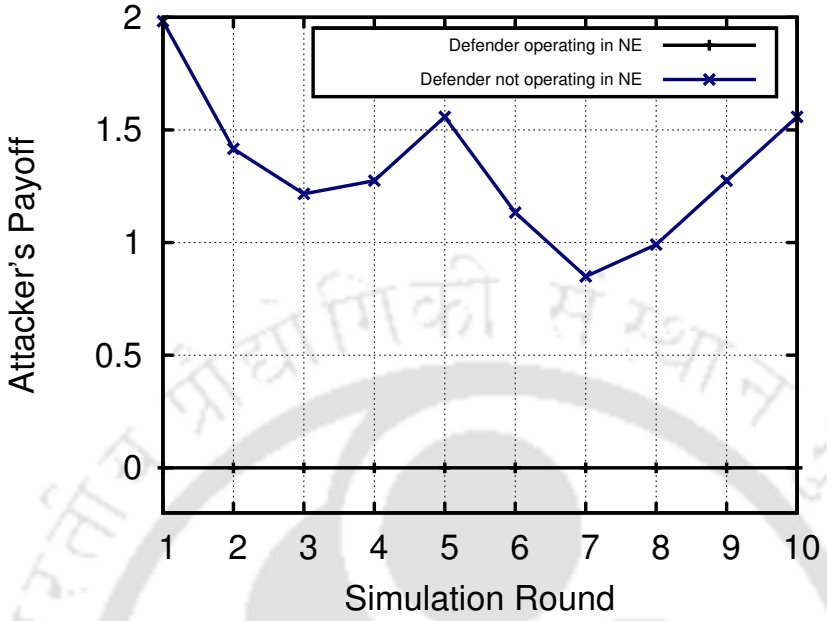

Figure 4.8: Attacker's Payoff corresponding to different strategies of Defender

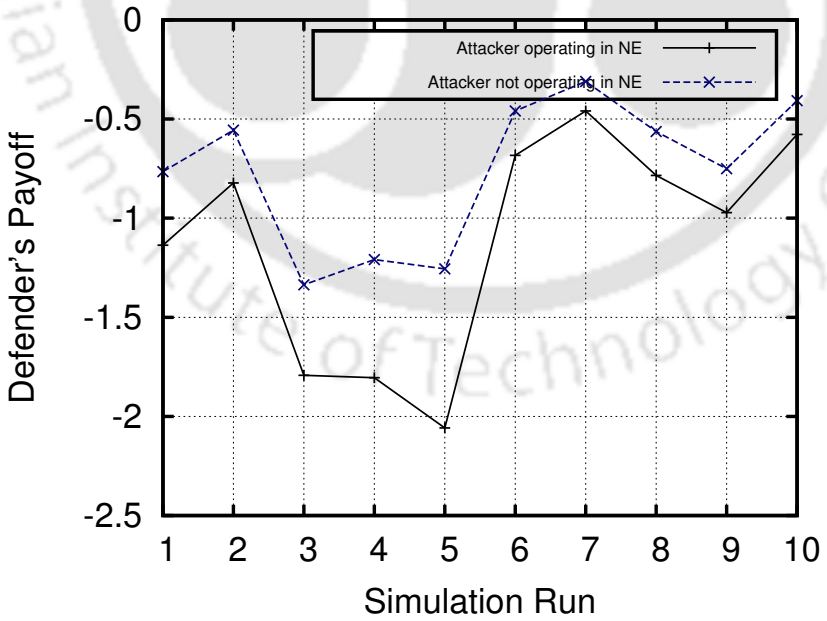

Figure 4.9: Defender's Payoff corresponding to different strategies of Attacker

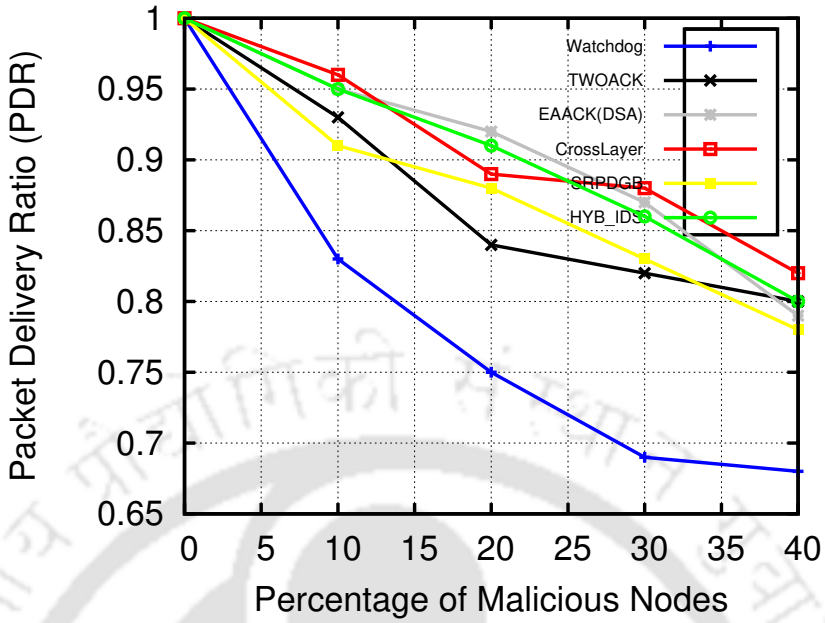

Figure 4.10: Packet Delivery Ratio

4.4.2.1 Comparison of proposed MANET IDS scheme with other methods

We have evaluated the performance of our proposed hybrid MANET IDS scheme with various other models namely, SRPDBG [120], CrossLayer [121], SPF[122], Watchdog [110], TWOACK [111] and EAACK [108]. These models were chosen for comparison since they represent a broad spectrum of MANET IDS schemes based on different methodologies like, game theory (SRPDBG), data mining (CrossLayer), specification (SPF) and rules (Watchdog, TWOACK and EAACK). Following metrics were used for evaluation and comparison of the proposed game theory-based hybrid IDS scheme with other MANET IDS schemes:

- ***Packet delivery ratio (PDR)***: It refers to the ratio of the number of packets delivered at the destination node to the total number of packets generated by the source node.
- ***Routing overhead (RO)***: It refers to the overhead involved in transmission due to introduction of additional routing control packets like Route Request (RREQ), Route Reply (RREP), Route Error (RERR), ACK etc.

Figure 4.10 and Figure 4.11 show the *PDR* and *RO* of the various IDS schemes under varying percentage of malicious nodes. It can be observed from these figures that all the four schemes (TWOACK, EAACK, SRPDBG and proposed IDS) have higher *PDR* than the simple WatchDog scheme. The *PDR* of the proposed hybrid IDS scheme is comparable to

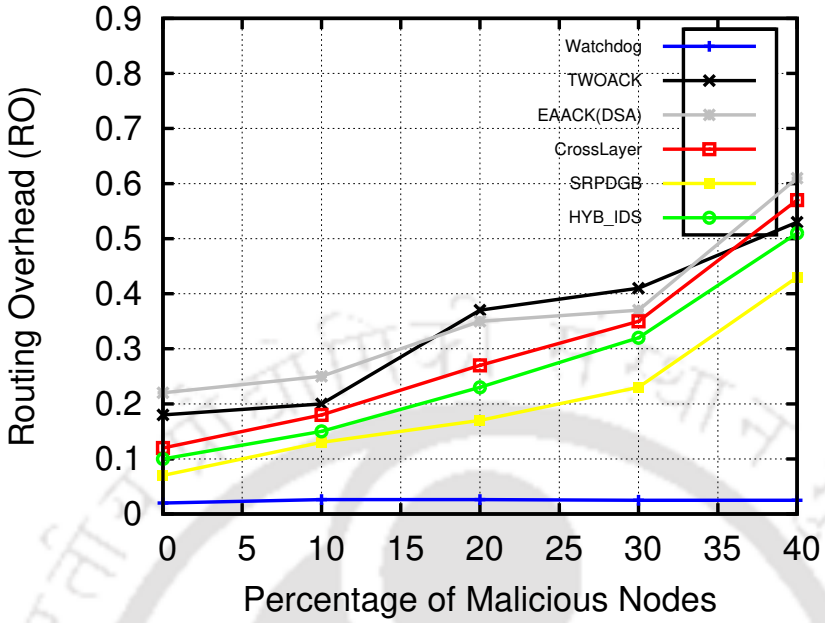

Figure 4.11: Routing Overhead

Table 4.8: Performance comparison of various IDS models

IDS Models	Attack Type	Detection Rate	False Alarm rate
SPF [122] (*Specification based*)	*Route Compromise*	47.56%	0.57%
	Traffic Distortion	43.24%	0.49%
	Black Hole	81.23%	0.51%
CrossLayer [121] (*Data mining based*)	*Route Compromise*	92.36%	0.38%
	Traffic Distortion	97.33%	0.93%
	Black Hole	99.7%	0.53%
SRPDGB [120] (*Game Theory based*)	*Route Compromise*	65.43%	0.36%
	Traffic Distortion	51.56%	0.55%
	Black Hole	99.42%	0.37%
Proposed HYB_IDS	*Route Compromise*	91.4%	0.45%
	Traffic Distortion	95.3%	0.87%
	Black Hole	99.5%	0.35%

Table 4.9: Comparison of various MANET IDS models

IDS Models	**Proposed HYB_IDS**	**CrossLayer** [121]	**SRPDGB** [120]	**SPF** [122]
Detection Rate	High	High	Low	Low
False Alarm	Low	Low	High	High
Detection Method	Game Theory based hybrid approach	Data Mining anomaly based	Game Theory and Trust based	Specification based
Attack types addressed	Routing attacks, DoS attacks, Packet dropping, Packet spoofing	Routing attacks, Packet dropping, Packet spoofing	Routing attacks, Packet dropping	Routing attacks, Packet dropping, Packet spoofing
Advantage	High detection rate, Low false alarm rate, Low power consumption	High detection rate, Low false alarm rate	Low power consumption	Detect routing attacks with high accuracy
Disadvantage	Marginal overhead incurred in cluster leader node election	High power consumption, Overhead in training the IDS model	Low detection rate, High false alarm rate	Low detection rate, High false alarm rate, High power consumption

that of EAACK and CrossLayer schemes, while it outperforms the TWOACK and SRPDBG schemes. On the other hand, the Watchdog scheme has the least *RO*, as it does not uses any acknowledgment technique to detect misbehaving nodes. The *RO* of the proposed IDS scheme (HYB_IDS) is less than that of the TWOACK, EAACK and CrossLayer schemes but higher than the SRPDBG scheme. The *RO* of the proposed IDS scheme is primarily due to exchange of election messages during the leader node and checker nodes election process.

Table 4.8 shows the detection rate and false alarm rate of various IDS models on different class of attacks (route compromise, traffic distortion and black hole attack). It can be observed from the table that the proposed game theory-based hybrid IDS framework (HYB_IDS) achieves high detection rate against all class of attacks, while at the same time produces a minimal amount of false alarms. It outperforms the SPF and SRPDGB schemes, while its performance is comparable to that of CrossLayer. However, the overhead of the proposed scheme is comparatively less than that of CrossLayer. A summarized comparison analysis of various IDS frameworks based on different features like, false alarm rate, detection method, detected attack types etc., is provided in Table 4.9.

From Table 4.8 and Table 4.9, it can be concluded that the proposed game theory-based hybrid IDS framework achieves high detection rate against wide range of attacks, while at the same time minimizes the overall volume of false alarms. It is also shown to reduce the computational overhead and energy consumption required for operating the IDS. However, the drawback of the proposed IDS framework is the marginal overhead incurred due to the cluster leader node and checker nodes election process. The high power consumption of the CrossLayer scheme [121] in Table 4.9 is because it uses data from physical layer, link layer and network layer to generate the association rules and create the baseline profile of the network. However, evaluating these association rules are computation intensive as data from multiple layers have to be taken into consideration. Additionally, the real time network data traffic has to be compared against large set of association rules in this scheme, which significantly increases its computational overhead.

4.5 Conclusion

In this chapter, we presented a novel game theory-based intrusion detection framework for MANETs. The framework models the intrusion detection process in MANET as a two player non-cooperative Bayesian game between the IDS and the node being monitored. Such game theoretic modeling allows the IDS to adopt a probabilistic monitoring strategy

based on the Bayesian Nash Equilibrium (BNE) of the game, which significantly reduces the energy consumption required for operating the IDS, without compromising the detection capabilities of the IDS. In addition, the proposed framework uses the VCG mechanism based cluster leader node election algorithm to ensure a uniform distribution of energy consumption across multiple MANET nodes for operating the IDS. This prevents the premature death of the nodes operating the IDS and hence avoids network fragmentation. The proposed IDS framework uses a combination of lightweight and heavyweight IDS modules to achieve high detection rate and accuracy across wide range of attacks. The lightweight module uses simple threshold based rules to detect malicious nodes, while the heavyweight module uses a powerful data mining based association rules to identify the malicious nodes.

In the next chapter, we propose a game theory-based multi-layered intrusion detection framework for Vehicular Ad-hoc Networks (VANETs). VANETs are formed on the fly by a network of vehicles equipped with multiple sensors and On Board Units (OBUs). VANETs operate in a bandwidth constrained wireless radio spectrum. Therefore, IDS frameworks that introduce significant volume of intrusion detection related traffic are not suitable for VANETs. Additionally, various characteristics of vehicular networks like, dynamic network topology, intermittent connectivity, communication overhead, scalability etc., also needs to be taken into consideration while developing an IDS framework for VANETs. Towards this end, a multi-layered game theory-based intrusion detection framework and a novel clustering algorithm for VANET is proposed as the third and the final contribution of the thesis in the next chapter.

"...The smaller grains remain longer in suspension, they are lifted by the inner motion of the water, while the larger grains soon settle out..."

Gotthilf Hagen
(German, 1797–1884)

5

A game theory based multi layered intrusion detection framework for VANET

5.1 Introduction

The concept of enabling vehicles with the capability to make transportation infrastructure more secure and efficient has received immense attention in recent years. This has lead to the emergence of Vehicular Ad-hoc Networks (VANETs), which are formed on the fly by the network of vehicles equipped with multiple sensors and On Board Units (OBUs). The OBUs enable vehicles to connect with Road Side Units (RSUs) through a wireless short-range direct communication link using the IEEE 802.11p based radio frequency channels. VANET uses various type of notification messages like Post Crash Notification (PCN), Road Hazard Condition Notification (RHCN), Stopped/Slow Vehicle Advisor (SVA) etc., to enable vehicular communication. Additionally, VANET allows vehicles to access various critical information like road congestions, alternate routes, road accidents, weather conditions etc., at real time to improve road safety and avoid accidents, which results in improved logistics and efficient utilization of public transportation system.

VANET uses 75 MHz of Dedicated Short Range Communications (DSRC) wireless radio spectrum operating at 5.9 GHz to support IEEE 802.11p standard for vehicular communication. DSRC provides a communication range of 300 to 1000 m, with a data rate of more than 27 Mbps and supports a vehicular mobility as high as 200 Kmph [123]. The IEEE

P1609 working group has proposed DSRC as IEEE 802.11p standard for Wireless Access in Vehicular Environment (WAVE) platform [124]. The DSRC based WAVE architecture supports two different protocol stacks namely, the WAVE Share Message Protocol (WSMP) and the traditional IPv6 protocol. Time sensitive and high priority communications are achieved using the WSMP, while the less demanding communications involving the UDP/TCP/IP data frames are achieved using the IPv6 protocol. As shown in Fig. 5.1, the DSRC spectrum band is divided into seven channels of 10 MHz each [125]. Channel 178 is the Control Channel (CCH), which is used for transmission of emergency messages. The other six channels numbered 172, 174, 176, 180, 182 and 184 are Service Channels (SCHs), which are used for transmission of both safety and non-safety related applications. If the CCH channel is active, all vehicles are bound to stop their communication during CCH time frame to receive and transmit emergency messages on CCH channel. For data exchanges, data frames containing WAVE short messages (WSMs) can be exchanged among vehicles on both the CCH and the SCH; however, IP data frames are permitted only on the SCHs.

Figure 5.1: Dedicated Short Range Communications (DSRC) spectrum with 7 channels of 10 MHz

Based on their mode of operations, the architecture of VANET can be categorized into following three types:

- ***Vehicle-to-Vehicle (V2V) network*** : This architecture (Fig. 5.2) allows direct communication among vehicles without relying on any fixed dedicated base station infrastructure and RSUs. It allows dissemination of security and safety related information

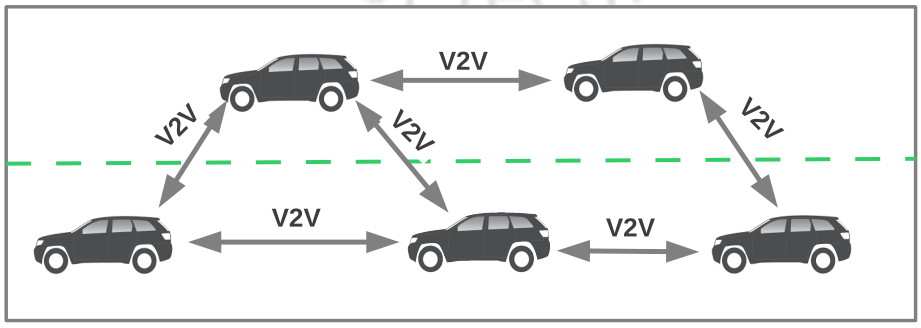

Figure 5.2: Vehicle to vehicle network

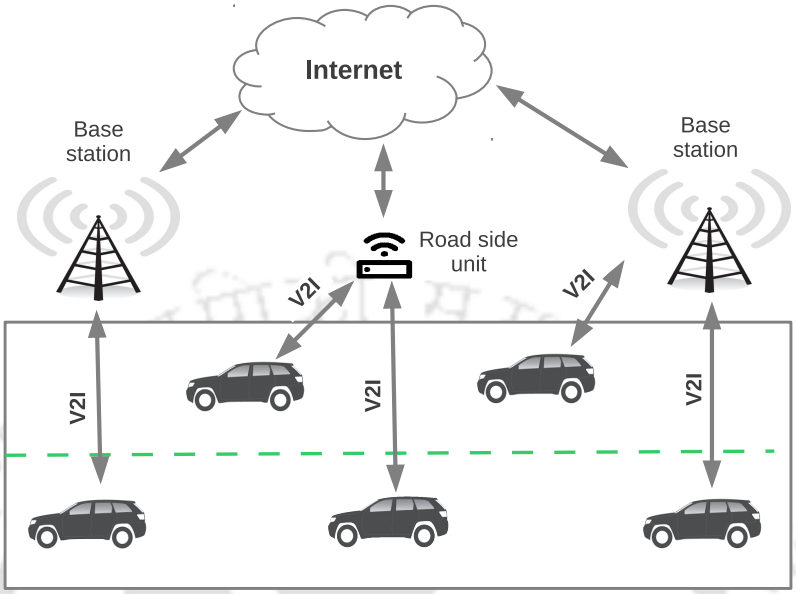

Figure 5.3: Vehicle to infrastructure network

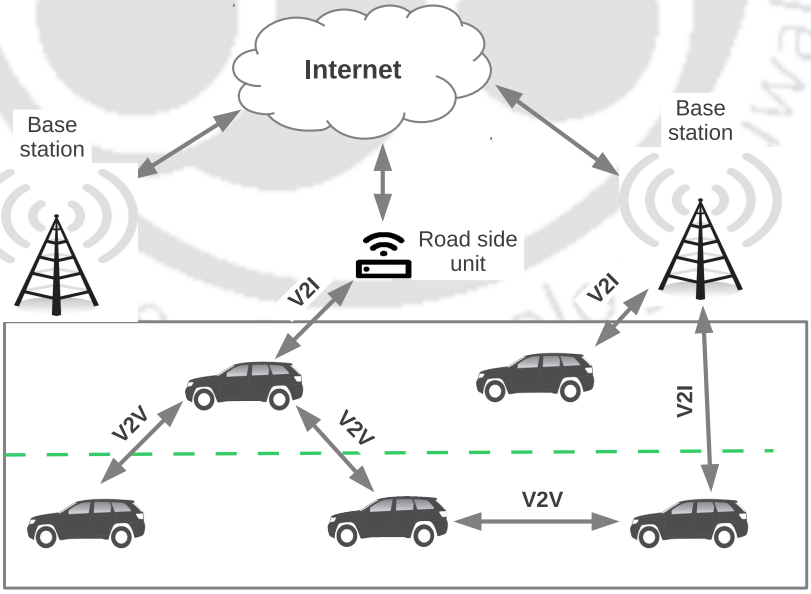

Figure 5.4: Hybrid network

among vehicles.

- ***Vehicle-to-Infrastructure (V2I) network*** : In this architecture (Fig. 5.3) the vehicles communicate with a dedicated roadside infrastructure like RSUs and base stations for obtaining important information about road safety and reporting gathered information. V2I links require more bandwidth and are less vulnerable compared to V2V links.

- ***Hybrid architecture network***: This architecture (Fig. 5.4) is the combination of V2V and V2I architectures. It allows vehicle to communicate directly with the RSU in a single hop or through multiple hops, depending on the distance of RSU from the vehicle. It also enables long distance connection to the Internet or to vehicles that are far away from RSU.

5.1.1 Security challenges in VANET

The distributed and wireless nature of VANETs coupled with their unique characteristics such as highly dynamic topologies, heterogeneous vehicular traffic, frequently disconnected networks, narrow bandwidths, short transmission range, omni directional broadcast etc., make them vulnerable to various type of security threats. The attacker can exploit the broadcast nature of VANET to carry out various types of attacks like eavesdropping, interference, jamming, masquerading, packets replay, Denial of Service (DoS), impersonation, identity disclosure etc. [126] [127] [128]. Intrusion Detection Systems (IDSs) have been proposed in the literature [129] [130] [131] [132] [133] to address these security threats in VANETs. However, intermittent network connectivity, narrow bandwidth wireless radio spectrum and absence of centralized coordinating entities for managing bandwidth usages and regulating channel access contentions make the task of formulating an effective intrusion detection framework for VANETs difficult and challenging. Therefore, any intrusion detection framework proposed for VANET must take the following key issues into consideration.

- ***Bandwidth constraints and IDS traffic volume***: VANETs operate in a narrow bandwidth wireless radio spectrum. The entire bandwidth spectrum of the DSRC band (5.850 - 5.925 GHz) used for vehicular communication in VANET is only 75 MHz with a maximum theoretical throughput of 27 Mbps and a maximum transmission distance of 1000 m. Therefore, intrusion detection frameworks that introduce signifi-

cant volume of IDS traffic and require pre stored information about the participating vehicles are not suitable for VANETs.

- ***Dynamic network topology***: Network topologies in VANETs vary depending on the traffic density and vehicular mobility. This can cause high delays in dissemination of messages due to broadcast storm at high vehicular densities and disconnected network problems at low vehicular densities. Therefore, any intrusion detection framework proposed for VANET must adopt a suitable clustering algorithm for generating stable vehicular clusters to maintain the network's stability.

- ***Communication overhead and scalability***: Due to high vehicular mobility, the association of a vehicle with other vehicles and RSUs in VANET is usually short lived and intermittent. Therefore IDS frameworks that require high communication overhead are not suitable for VANETs. In addition, VANETs consist of a network of hundreds of vehicles and are designed for supporting real time safety related applications, which require them to be up and running all the time. Therefore, IDS frameworks designed for VANETs must be scalable to vehicular networks with high vehicular densities.

Any IDS framework proposed for VANET must maintain a good trade-off between gathering enough information for effectively detecting network intrusions and preventing the overburdening of IDS's logging component with high volume of IDS traffic. To achieve this trade-off, a distributed game theory-based multi layered intrusion detection framework for VANET is proposed in this chapter with the following features:

1. ***Stable clustering algorithm***: A novel clustering algorithm is proposed as part of an overall IDS framework that takes into account various vehicular parameters like velocities, direction of movements, coordinates and reputation values to produce stable vehicular clusters. Stable clusters enhance the robustness of the IDS framework by reducing the overhead involved in the cluster formation process and by allowing vehicles enough time frame to exchange their information for making informed decisions.

2. ***Hierarchical IDS framework***: In the proposed framework, intrusion detection is carried out at three different levels in a decentralized manner. At the lowest level, the agent nodes use a set of specification rules to detect malicious vehicles. At the intermediate level, the Cluster Head (CH) uses a combination of specification rules and a neural network based classifier module to identify malicious vehicles. Finally, at the

highest level, the base stations/RSUs use the information received from their respective CHs to determine the malicious vehicles in the network.

3. ***Game theory-based IDS traffic minimization scheme***: The proposed IDS framework uses a game theory-based monitoring scheme to minimize the overall volume of intrusion detection related traffic in the vehicular network. The framework models the interaction between the IDS and the vehicle being monitored as a two player non-cooperative game and adopts a probabilistic monitoring strategy based on the Nash Equilibrium (NE) of the game, which significantly reduces the volume of IDS traffic.

The rest of the chapter has been organized in following ways. Section 5.2 discusses related works on VANET intrusion detection frameworks and their drawbacks. Section 5.3 provides an overview of the proposed game theory-based multi layered intrusion detection framework. Sections 5.3.4, 5.3.5 and 5.3.6 provide a detailed description about various modules of the proposed framework, namely, the Local Intrusion Detection System (LIDS) module, the Cluster Intrusion Detection System (CIDS) module and the Global Detection System (GDS) module, respectively. Section 5.4 provides the simulation results and comparison analysis of the proposed framework with other existing intrusion detection frameworks. Conclusion and future works are provided in Section 5.5.

5.2 Related Works

Many cryptography and authentication based protective mechanisms have been proposed in the literature to address the security threats in VANETs [125] [134]. A novel Authentication, Authorization and Accounting (AAA) access control scheme for application services in VANETs are proposed in [135] [136]. These schemes are based on IEEE 802.11i standards and use EAP-Kerberos model, wherein the vehicles willing to join the network send authentication request message through the intermediate vehicles and the RSU, until they reach a centralized authentication server that can grant access to the requesting mobile users. An efficient pre-authentication scheme to realize fast and secure handoff in IEEE 802.11 based vehicular network by reducing four-way handshake to two-way handshake between the RSU and requesting vehicles is proposed in [137]. Although, IEEE 802.11 AAA-based authentication mechanism provides a promising solution for authentication and authorization between vehicles and service providers in VANETs, a full 802.11 based authentication requires a long authentication delay between 750 to 1200 ms due to lengthy round trip

time between the AAA server and the RSU [138]. In addition, due to frequent change in associated RSUs, the frequency of authentication between vehicles and RSUs will be high. Therefore, it is not feasible to apply a full 802.11 based authentication in VANETs due to its heavy operations and long delays.

An accurate and lightweight intrusion detection framework, called AECFV, which aims to protect VANET against various attacks is proposed in [129]. Their framework uses a combination of specification rules and a Support Vector Machine (SVM) based classifier model to detect various types of attacks. However, the drawback of their framework is the overhead involved in training the complex SVM classifier model. A novel approach for detecting Wormhole attack in VANET is proposed in [139]. They showed that their scheme can be easily implemented in AODV routing protocol with low overhead and without requiring any special hardware. Their scheme uses authentication mechanism based on HEAP method [140]. They showed that their scheme is able to detect the malicious vehicles performing Wormhole attack with high accuracy. However, their framework can only detect Wormhole attacks.

A Learning Automata (LA) based IDS framework for VANET is proposed in [141], wherein the vehicles are equipped with LA to capture different activities and states of the vehicles. A Markov Chain Model is used to represent the states and their associated transitions in the network. The vehicle density determines the transition of vehicle from one state to the other. A classifier model based on parameter called the Collaborative Trust Index (CTI) is then used to detect any malicious activities and attacks in the vehicular network. However, the drawback of this framework is the overhead involved in developing a complex Markov Chain Model, which puts a severe limitation on their real time deployment. A framework to identify and evict misbehaving faulty vehicles from the vehicular network is proposed in [126]. The framework uses the revocation of certificate by the Certification Authority (CA) as the primarily tool to evict misbehaving vehicles from the vehicular network. However, such approach has a vulnerability window because of the latency involved in identifying misbehaving vehicles and distributing revocation information. To address this issue, the authors proposed two different protocols tailored for the identification and eviction of malicious vehicles from the network. They showed that their framework achieves a sufficiently high level of robustness by effectively identifying and evicting the faulty vehicles from the network with low latency. However, the drawback of their framework is that it requires a modification in the protocol stack to identify malicious vehicles.

A host based intrusion detection system for VANET is proposed in [142]. Their framework uses a statistical technique to determine whether the data being forwarded by the vehicle is genuine or fake without using any trust or reputation schemes. It works under the assumption that fake data being disseminated by the malicious vehicles are easy to detect since their parameter values differ greatly from the genuine data being forwarded by the normal vehicles. The main objective of the malicious rogue vehicles is to inflict damage to the network by either flooding or dropping data packets, which can be measured by the statistical mechanism proposed in their framework. However, the main issue of this framework is that it requires each vehicle to run its own intrusion detection system, which increases the computational overhead at each vehicle and also introduces a large volume of IDS traffic in the vehicular network. REST-Net, a novel intrusion detection system for mitigating the authenticity and integrity challenges in VANET is proposed in [143]. Their framework uses a dynamic rule-based IDS detection engine that analyzes and monitors data packets through plausibility checks to detect and prevent dissemination of fake messages in VANET. However, the drawback of this framework is the latency involved in identifying the malicious vehicles and distributing the revocation information across the network.

From our survey of related works, we found that there are many drawbacks associated with the existing intrusion detection frameworks proposed for VANET namely, use of complex Markov chain and SVM based models, high overhead involved in authentication process and high latency in dissemination of revocation information across the network. In addition, some of the frameworks introduce a significant volume of IDS traffic, which can cause congestion in a bandwidth constrained vehicular network. Moreover, some of the frameworks are geared toward detection of specific class of attacks and cannot be generalized for detecting other type of attacks. We aim to address these issues in existing intrusion detection frameworks by proposing a game theory-based multi layered intrusion detection framework for VANET.

5.3 Multi layered game theory-based hybrid intrusion detection framework

This section provides an overview of the proposed multi-layered game theory-based intrusion detection framework. The overall architecture of the proposed multi-layered game theory-based intrusion detection framework is shown in Fig. 5.5. As shown in the figure, the proposed framework comprises multiple clusters, with each cluster containing a unique CH. Vehicles communicate with their respective Cluster Heads (CHs) using the IEEE

802.11p wireless standard and the CHs communicate with the Road Side Units (RSUs) using the wireless Long-Term Evolution (LTE) standard. Two different wireless standards were chosen to maintain a fair trade-off between latency and operational cost. Vehicular networks employing only the IEEE 802.11p standard encounter high delay in dissemination and delivery of safety messages due to broadcast storm and disconnected network problems at high and low vehicular densities, respectively. Cellular technologies based on LTE can mitigate this problem, as they have low latency and wide-range communication. However, a pure cellular-based VANET communication is not feasible due to high cost of communication between the vehicles and the CHs. It also incurs a high number of hand-off occurrences at the base station because of high vehicular mobility. Therefore, a hybrid architecture that uses a combination of IEEE 802.11p and LTE based wireless standards provides the best trade-off between the latency and the operational cost in VANETs.

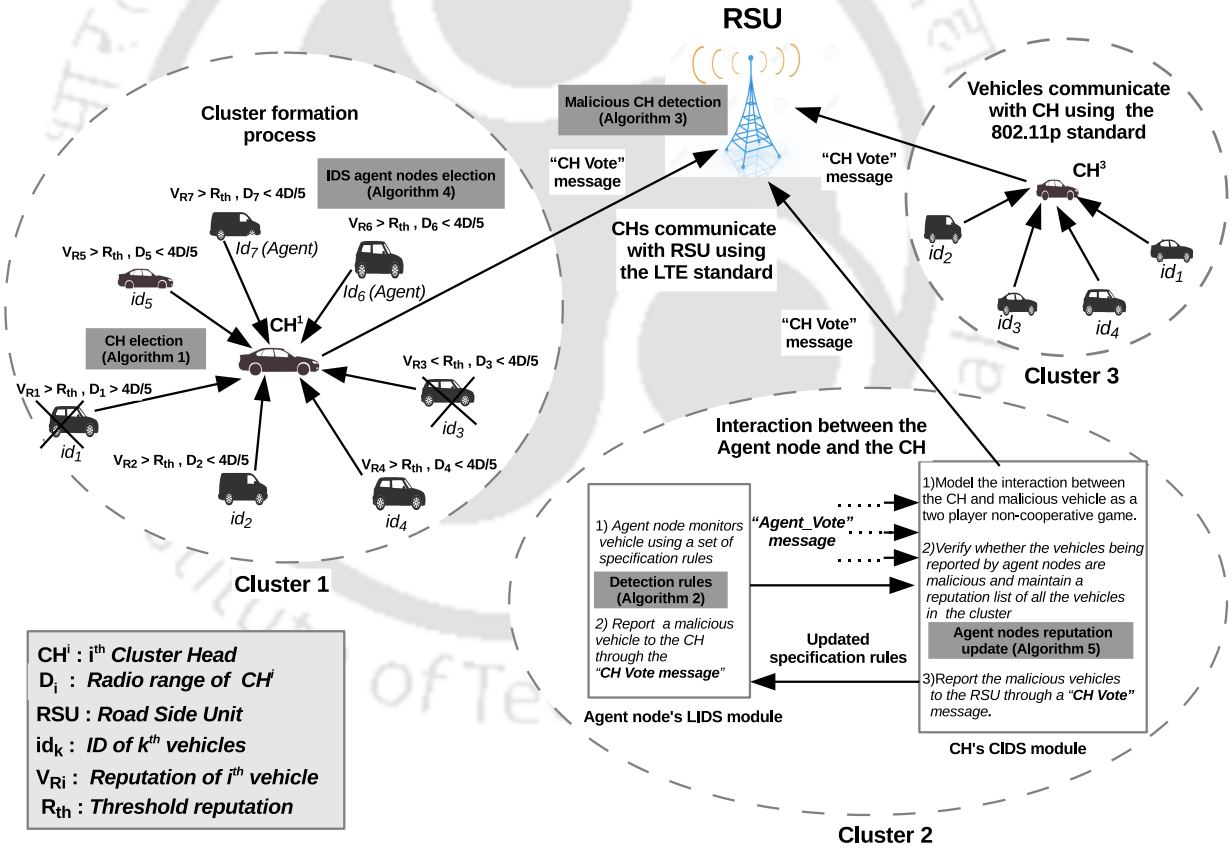

Figure 5.5: Proposed multi layered VANET intrusion detection framework's architecture

The proposed framework carries out the intrusion detection operation at three different levels. At the lowest level, the agent nodes operate the Local Intrusion Detection System

(LIDS) modules to monitor vehicles in their neighborhood. The agent node's LIDS module uses a set of specification rules based on the Received Signal Strength Indicator (RSSI), Packet Delivery Rate (PDR), Packet Forwarding Rate (PFR) and Duplicate Packet Rate (DPR) values of the vehicles to detect malicious vehicles in its neighborhood. In addition, the agent nodes also monitor their respective CHs for sign of maliciousness. If majority of the agent nodes find the CH to be malicious, a new CH is elected in its place. A detailed description about the agent node election algorithm and the agent node's LIDS module is provided in sub-section 5.3.4.

At the intermediate level, the CH operates the Cluster Intrusion Detection System (CIDS) module to monitor vehicles in its cluster. The CH's CIDS module uses a combination of specification rules and a lightweight neural network based anomaly detection module to detect malicious vehicles in the cluster. It uses the information received from the agent nodes in its cluster to device a game theory-based probabilistic monitoring strategy to minimize the overall volume of IDS traffic in the vehicular network. It also employs a mechanism to update the reputation values of the vehicles and the agent nodes within its cluster based on their observed behaviors. If the reputation of any vehicle or agent node falls below the threshold value then it is removed from the cluster by the CH. A detailed description about the CH's CIDS module is provided in sub-section 5.3.5.

At the highest level of the proposed framework, the RSU operates the Global Decision System (GDS) module, which receives input from multiple CHs within its radio range. The malicious vehicles reported by the CHs are assigned to the *Blacklist* table of the RSU. The RSUs periodically broadcast the identities of these malicious vehicles to prevent other normal vehicles in the network from communicating with them. A detailed description about the RUS's GDS module is provided in sub-section 5.3.6.

As shown in Fig. 5.5, various algorithms operate at different layers of the proposed IDS framework. Brief descriptions about these algorithms are provide below:

1. *CH election algorithm (Algorithm 1)*: This algorithm runs at every cluster of the vehicular network and elects the CH for the given cluster.

2. *Detection rule algorithm (Algorithm 2)*: This algorithm uses set of specification rules based on Packet Delivery Rate (PDR), Received Signal Strength Indicator (RSSI), Duplicate Packet Rate (DPR) and Packet Forwarding Rate (PFR) values to detect malicious vehicles in the cluster.

3. *Malicious CH detection algorithm (Algorithm 3)*: This algorithm is executed by the agent nodes to verify whether the CH is normal or malicious.

4. *IDS agent nodes election algorithm (Algorithm 4)*:This algorithm is executed at every cluster to elect a set of agent nodes, which are responsible for aiding the CH in monitoring and identifying malicious vehicles.

5. *Agent node reputation update algorithm (Algorithm 5)*: This algorithm updates the reputation values of the agent nodes in the cluster based on their observed behavior. The agent nodes found to be behaving maliciously are penalized with negative payment and removed from the cluster by the CH.

We make the following assumptions with respect to the proposed intrusion detection framework:

1. Vehicles are equipped with 802.11p enabled wireless DSRC radios, which enable them to communicate with each other. Vehicles use Global Positioning System (GPS) and digital maps to determine their coordinates and direction of movements at real time. Additionally, vehicles employ public key based cryptographic solutions to ensure communication privacy and source authentication.

2. The vehicular network is partitioned into multiple grid regions and each region is assigned a unique identity number (ID) as shown in the Fig. 5.6,. In the figure, these IDs are numbered **A** through **T**. In order to comply with the maximum transmission range under DSRC standard, each grid's dimension is set to 1000 m $\times$ 1000 m. The vehicles are grouped into clusters based on their grid IDs, velocities and direction of movement. Vehicles can only communicate with other vehicles in their own cluster. Any inter cluster communication has to be made via the CH. The CHs exchange their information containing the list of malicious vehicles when they come into each others' radio range.

3. Prior to participating in the vehicular network, vehicles must initially register with one of the RSUs in the network. The RSUs maintain a reputation list of all the registered vehicles in their radio range.

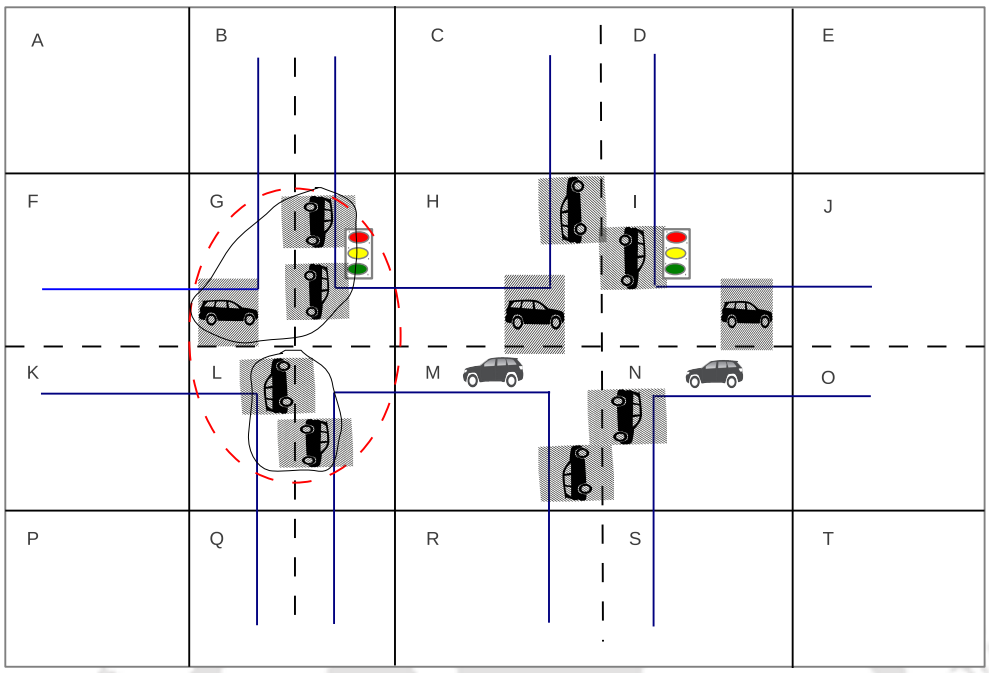

Figure 5.6: An illustration of cluster formation in the proposed framework

5.3.1 Attack types in VANET

Due to the wireless nature and broadcast medium of communication in VANET, a malicious vehicle can disseminate false alert messages for its own selfish gain and disrupt the normal functioning of the network. In our study we have considered the following class of attacks in VANETs:

1. **Selective forwarding and black hole attacks**: In the selective forwarding attack, the malicious vehicle selectively forwards the data packets while dropping others. On the other hand, in the black hole attack the malicious vehicle drops all the packets that it receives without forwarding them further. A malicious vehicle or a CH performing these attacks can be detected by computing their Packet Delivery Rate (PDR) and Received Signal Strength Indicator (RSSI) values and comparing them with a threshold PDR ($T_{pdr_{sf}}$, $T_{pdr_{bh}}$) and RSSI ($T_{rssi_{bh}}$) values.

2. **Denial of Service (DoS) attack**: In this attack, the malicious vehicle inundates the network with a large number of fake alert messages about road accident and congestion in order to consume the network's bandwidth. A malicious vehicle performing a DoS attack can be detected by computing its Duplicate Packet Rate (DPR) and Packet Forwarding Rate (PFR) values. If its DPR and PFR exceed the threshold values $T_{dpr_{dos}}$

and $T_{pfr_{dos}}$, respectively then it is assumed to be carrying out the DoS attack.

3. **Wormhole attack**: In this attack, two malicious vehicles located at different locations collude together to form a private tunnel. To execute this attack, the malicious vehicle generates a high RSSI value signal to convince other normal vehicles in its neighborhood that it has the shortest path to destination or the CH. Thereafter, the malicious vehicle forwards all the received packets to another malicious vehicle at the other end of the tunnel, which in turn either drops the packets or modifies them before forwarding them to the destination. If the RSSI and PDR values of the vehicle being monitored exceed the threshold values $T_{rssi_{wh}}$ and $T_{pdr_{wh}}$, respectively then it is assumed to be carrying out the wormhole attack.

4. **Sybil attack**: In this attack, the malicious vehicle creates a multiple fake identities of itself in order to prevent detection when launching various other attacks like black hole and DoS attacks. Sybil attack can be detected by computing the RSSI value of the vehicle and then verifying whether it follows a normal distribution. To detect this attack, the mean (μ) and the standard deviation (σ) corresponding to the RSSI values of all the vehicles in the cluster are calculated. The 'Z-score' of RSSI value of vehicle v_i ($RSSI_{v_i}$) is then calculated using the formula $\frac{RSSI_{v_i}-\mu}{\sigma}$. If the 'Z-score' of RSSI value of the vehicle being monitored exceeds the value 2.5 ($T_{rssi_{syb}}$) in the normal distribution curve, then the vehicle is assumed to be carrying out the Sybil attack.

Before delving into the detailed description of the proposed multi-layered game theory-based intrusion detection framework for VANET, we provide an elaborate discussion about the distributed clustering algorithm employed by the proposed framework for generating stable vehicular clusters. We also discuss a novel CH election algorithm along with a stimulus structure based on Vickrey-Clarke-Groves (VCG) mechanism [144] for motivating vehicles to actively participate in the CH election process.

5.3.2 Distributed cluster formation and CH election algorithms

The effectiveness of any cluster based VANET intrusion detection framework largely depends upon the stability of the clusters produced by the clustering algorithms. Stable clusters reduce the overhead involved in cluster formation process and provide vehicles with sufficient time frames to exchange their data. Towards this end, a distributed clustering algorithm that produces highly stable vehicular clusters with enhanced connectivity among

member vehicles is proposed as a part of the proposed intrusion detection framework.

VANETs are characterized by high vehicular mobility, which makes the clustering process in VANET difficult. However, vehicles in VANET are constrained by road topologies, which require them to follow traffic lights and road signs leading to a predictable mobility pattern with restricted movement along predefined directions [145] [146]. The proposed clustering algorithm exploits these constraints to produce stable vehicular clusters. It requires vehicles to periodically broadcast beacon messages to inform other vehicles in the neighborhood about their presence. The beacon message comprises various information about the vehicle namely, its identity, coordinates, velocity, direction of movement and cluster membership status. The details regarding the coordinates, velocity and direction of movement of the vehicle are obtained from the GPS device equipped in the vehicle.

Each vehicle in the proposed framework maintains a data structure in the form of velocity vectors that keep logs of its neighborhood vehicles' velocities over a specified period of time. Given these data, the vehicle can make prediction about the future relative position and the approximate moment when its neighborhood vehicles will be out of its range. This enables the vehicle to estimate the link quality of its neighborhood vehicles and prevent the re-clustering process when groups of vehicles moving in a different directions come together. In such case, the period of meeting between the groups of vehicles is usually very short and therefore, changing cluster structure will result in another re-clustering, once the groups move outside each others' transmission range. In the proposed clustering mechanism, due to better movement prediction, the estimated potential link quality will be poor in such case, which prevents the re-clustering and leads to increased cluster stability. On the other hand, in case of traffic jam, different set of clusters moving in different directions will usually stay in each others' range for longer period of time; this will be predicted by the link quality estimation procedure. In such case, re-clustering is carried out to merge the clusters.

The proposed clustering algorithm comprises two phases namely, the setup phase and the maintenance phase. In the cluster setup phase, vehicles in close proximity to each other are organized into clusters and CHs are selected for each individual cluster. In the cluster maintenance step, a secondary CH (SCH) is selected for each cluster. CH selected in the setup phase becomes the primary CH (PCH). When the PCH is no longer in the cluster, the SCH takes over. The cluster structure does not change but only the node playing the role of CH changes. This allows for stable cluster architecture, with low overhead, and better performance. In the subsequent sub-sections, we provide detailed descriptions about the

setup phase and the maintenance phase of the proposed clustering algorithm.

5.3.2.1 Setup Phase

In the proposed clustering algorithm, the vehicles can be in one of the following four states namely, Undecided (UD) state, Cluster Member (CM) state, Cluster Head (CH) state and Cluster Gateway (CG) state. Initially, all the vehicles in the network are in the UD state during which they are not part of any clusters. During this state, the vehicles broadcast beacon messages every t_j time unit and wait for beacon messages from other vehicles in their neighborhood. The beacon message comprises various vehicular information like vehicle's identity, velocity, coordinates, direction of movement and cluster membership status. When a vehicle v_i in the UD state receives beacon messages from the vehicle v_j at regular time interval t_j for a specified number of times (three in the proposed scheme), v_i adds v_j to its neighboring list.

In the proposed framework, the road network is divided into multiple grids and each grid is assigned a unique identity. Once the vehicles successfully exchange the beacon messages and create their neighboring list, they are grouped into multiple clusters. All the vehicles in the given cluster must have the same grid ID and direction of movement. In addition, the velocity of any vehicle in the given cluster must be within two and half standard deviation from the mean cluster velocity. This rule is based on the assumption that the vehicular velocities in the cluster follow the Normal distribution. Additionally, the member vehicles of the cluster must be within four-fifth radio range of the elected PCH. This avoids the frequent re-clustering process, as the member vehicles at the boundary radio range of the PCH are likely to exit the cluster and form a new cluster.

On the other hand, if a vehicle v_i in the UD state receives a CH Join Request (CJR) message, it indicates the presence of a CH in v_i's neighborhood. The CH broadcasts the CJR messages every t_j time units. Upon receiving the CJR message, v_i checks the Received Signal Strength Indicator (RSSI) value of the CJR message ($RSSI_{CJR}$). If the $RSSI_{CJR}$ value is greater than some predefined threshold $RSSI_{th}$, v_i sends a joining request (JRq) message containing its identity to the CH, which then grants the cluster membership to v_i. v_i then changes its status to CM state. However, if v_i stays in the UD state for more than $5t_j$ time units (i.e., does not receive CJR message during [t, t + $5t_j$]), then it initiates the CH election process by broadcasting the CH election message among its neighborhood vehicles to establish a new cluster.

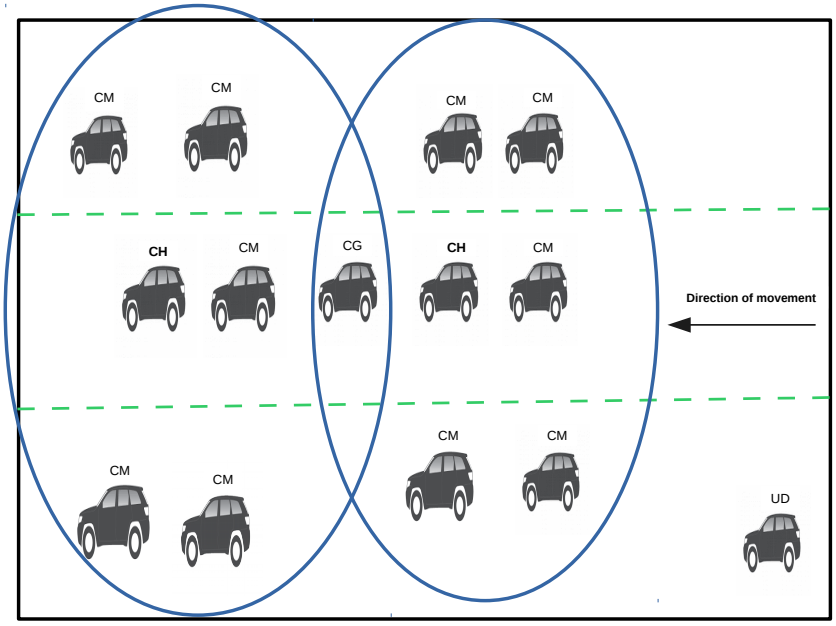

Figure 5.7: An illustration of various states of the vehicles

v_i remains a member of the cluster as long as it receives the CJR messages from its CH every t_j time units. If v_i does not receive CJR message during the interval [t, t + $3t_j$], it considers its association with its CH is lost and switches to the UD state. On the other hand, if v_i receives a CJR message from another CH whose RSSI value is greater than $RSSI_{th}$, while it is still a member of some existing cluster, then it switches to CG state after sending the JRq message to the new CH and receiving the corresponding ACK message. Therefore, a vehicle in the CG state is a member of two or more clusters. An illustrative example of various states of the vehicles is shown in Fig. 5.7

The CH broadcasts the CJR messages every t_j time units. Each member of the cluster is required to acknowledge the CJR message from the CH with an ACK message. If the CH does not hear from any of its member during the interval [t, t + $3t_j$] then the corresponding cluster member is assumed to have left the cluster and it is removed from the CH's membership list. The CH switches to UD state, if its cluster membership list becomes empty. In addition, when two neighboring CHs come close to each other and have a significant overlapping area (i.e., both CHs receive CJR messages from each other), cluster merging procedure is triggered. In such case, only one of them will keep its CH role, while the other will switch to a CM state. The members of the cluster (whose CH has just switched to a CM

state) will switch to UD states. They will then change their roles according to the procedure explained earlier (e.g., they can all become cluster members of the new CH if they are in the transmission range of the new CH). However, to avoid the overhead involved in frequent cluster reformation process, cluster merging procedure is deferred until certain criterion are fulfilled. The merging procedure is initiated only when two CHs have been within the threshold merging distance (D_{mrg}) for a T_{mrg} time period after coming into each other's transmission range. The decision of the CH to give up or retain its role after merging of clusters is based on the weighted factor CH^* (see Equation 5.1). This factor represents the minimum of the difference between the sum of velocity differences between the CH and its neighboring vehicles and the number of cluster member vehicles of the CH. The CH that will retain its role corresponds to the CH that produces the minimum value of this difference.

$$CH^* = Min_{i=1,2}\Big\{\alpha * \sum_{v_a \in CH_i}\Big(\mid V_{CH_i} - V_{v_a} \mid \; - \; (1-\alpha) * NeighCH_i\Big)\Big\} \tag{5.1}$$

where, $\alpha \in$ [0,1]. V_{CH_i} is the velocity of the i^{th} CH (CH_i). V_{v_a} is the velocity of the vehicle $v_a \in V_{CH_i}$ and $NeighCH_i$ is the number of cluster member vehicles of CH_i.

5.3.2.2 Maintenance Phase

The primary objective of the maintenance phase of the proposed clustering algorithm is to ensure high reliability and stability (less packet losses and better packet delivery) of the cluster structure produced during the setup phase. The basic idea is to use two CHs namely, the primary CH (PCH), which is elected during the setup phase and the secondary CH (SCH), which is elected in the maintenance phase. The PCH of the cluster C selects the SCH from among its cluster members. The cluster member vehicle v_i with the minimum sum of velocity difference between the PCH and cluster member vehicle, and the distance between PCH and cluster member vehicle (Equation 5.2) is selected as the SCH.

$$SCH = Min_{v_i}\Big\{\alpha * \mid V_{PCH} - V_{v_i} \mid \; + \; (1-\alpha) * D_{v_i}^{PCH}\Big\} \;\; \mid v_i \in \; C \;\; and \;\; v_i \neq \; PCH. \tag{5.2}$$

where, $\alpha \in$ [0,1]. V_{PCH} is the velocity of the PCH. V_{v_i} is the velocity of the member vehicle v_i of the cluster. $D_{v_i}^{PCH}$ is the distance between PCH and v_i.

When the PCH can no longer act as the CH (e.g., leaving the cluster by taking a highway exit), it will ask the SCH to take over the role of the PCH and change its own status to CM state. It will eventually change to UD state when it no longer receives CJR messages from the new PCH. The new PCH will keep the same identifier (Cluster_ID) as the previous PCH. Therefore, the cluster structure will remain intact (CMs of the cluster do not have to reorganize in new clusters) and thus no re-clustering overhead is generated. The new PCH will then select a new SCH from among its cluster members using Equation 5.2.

5.3.2.3 CH election algorithm

After generating and exchanging their SCF table details, the vehicles are grouped into different clusters based on their velocities, coordinates (cluster ID) and direction of movements. As discussed in the *Setup Phase* (sub-section 5.3.2.1), all the vehicles in a given cluster must have the same grid ID and direction of movement. Additionally, the velocities of any vehicle in the given cluster must be within two and half standard deviation of the mean cluster velocity. After the successful cluster formation process, the PCH election procedure is initiated at each cluster.

In this sub-section, we provide a detailed description about the PCH election process of the proposed clustering algorithm. Vehicles use the beacon messages that they receive from their neighborhood vehicles to create their Social Choice Function (SCF) tables. The SCF table of the vehicle comprises the identities of all the vehicles within its radio range along with their associated reputation values. Initially, each vehicle v_i requests the RSU to provide the reputation values of the vehicles in its SCF table. v_i later updates the reputation values of the vehicles in its SCF table (SCF_{v_i}) based on their observed behavior and updates received from the CH and the RSU. When the reputation of any vehicle $v_j \in SCF_{v_i}$ falls below the threshold value (R_{th}), it is removed from SCF_{v_i}. The SCF table of the vehicle also maintains the velocity log details of its neighborhood vehicles for the previous *t* time units in a velocity vector data structure. Velocity data obtained for the $(t+1)^{th}$ time unit from the vehicles are then averaged to eliminate short fluctuations using the following exponential smoothing function:

$$F_{v_{t+1}} = \gamma A_{v_t} + (1-\gamma) F_{v_t} \tag{5.3}$$

where, $\gamma \in [0,1]$ is the smoothing parameter. A_{v_t} and F_{v_t} are the vehicle's actual velocity

Table 5.1: *SCF* Table of vehicle v_a (SCF_{v_a}) with *n* neighbors

Neighbors	*Reputation*	*Velocity vector (miles per hour)*
v_1	0.59	<40, 35, 42, 38, 39 >
v_2	0.75	<40, 44, 40, 36, 42 >
v_3	0.91	<38, 43, 40, 37, 44 >
. . .	. . .	
v_n	0.83	<35, 39, 43, 39, 44 >

and the forecast velocity, respectively. A sample *SCF* table details of an arbitrary vehicle v_a (SCF_{v_a}) with *n* number of neighboring vehicles is shown in Table 5.1. The forecast velocity calculation of a vehicle v_b ($v_b \in SCF_{v_a}$) obtained using Equation 5.3 is shown in Table 5.2, wherein the initial forecast velocity of v_b is set equal to its actual initial velocity. The Mean Squared Error (MSE) of v_b is obtained by subtracting its forecast velocity values from its actual velocity values, squaring and summing them and then finally dividing the sum by the number of observations (6 in this case). The MSE of v_b corresponding to the observations in Table 5.2, with the value of γ set to 0.2 is 11.58.

Table 5.2: Forecast velocity and MSE calculation using exponential smoothing

Observation	*Actual velocity* (A_{v_b})	*Forecast velocity* (F_{v_b})	*Error*	$Error^2$
1	39	39.00	0.00	0.00
2	44	39.00	5.00	25.00
3	40	40.00	0.00	0.00
4	45	40.00	5.00	25.00
5	38	41.00	-3.00	9.00
6	43	40.40	2.60	6.16
7	39	40.92	-1.92	3.69

To elect the PCH, each vehicle v_i in the cluster C computes the utility function of every other vehicle $v_j \in SCF_{v_i}$ using the following rule:

$$U_{v_j}^{v_i} = \beta R_{v_j^i} + (1-\beta) \mid SCF_{v_j} \mid \; - \; MSE_{v_j}^{v_i} \tag{5.4}$$

where, v_j and $v_i \in C$. $R_{v_j^i}$ is the reputation of v_j in SCF_{v_i}. $\mid SCF_{v_j} \mid$ is the number of vehicles in the SCF table of v_j. $\beta \in$ [0,1] is the weight parameters used for specifying the significance of reputation and connectivity metrics of v_j in computation of the utility function $U_{v_j}^{v_i}$. $MSE_{v_j}^{v_i}$ is the MSE of v_j calculated by v_i using the exponential smoothing function given by Equation 5.3.

After computing the utility functions corresponding to every vehicle in their SCF lists, the

vehicles exchange their utility function lists ($Utility_{list}$). The $Utility_{list}$ of v_i is of the form $\{U^{v_i}_{v_a}, U^{v_i}_{v_b}, ..., U^{v_i}_{v_k}\}$, where $v_a, v_b .., v_k \in SCF_{v_i}$. In the next step, v_i computes the aggregated utility function of every vehicle $v_j \in SCF_{v_i}$ ($U_{v_{ij}}$) using the $Utility_{list}$ it received from other vehicles. Finally, the vehicle with the highest aggregated utility function is elected as the primary PCH. It is to be noted that the vehicles assign higher weights to their own utility functions compared to the utility functions received from other vehicles, while computing their aggregated utility functions. A detailed description of the proposed PCH election mechanism is given by Algorithm 3

Algorithm 3 Distributed PCH election algorithm

Input : *Utility function lists of vehicles in cluster C.*

Output : *PCH of the cluster C.*

$v_i \xleftrightarrow{SCF\ \ table} cluster^C_{-v_i}$ */* Each vehicle v_i in the cluster C exchange its SCF table details with every other vehicles in C */*

for *each* v_j, $v_i \in C$ **do**

  $U^{v_i}_{v_j} = \beta_1 R_{v^i_j} + \beta_2 |\ SCF_{v_j}\ | - MSE^{v_i}_{v_j}$, *where* $\beta_1 + \beta_2 = 1$ */* v_i computes the utility function of v_j. */*

end

$v_i \xleftrightarrow{Utility_{list}} cluster^C_{-v_i}$ */* vehicles in C exchange their utility function lists */*

for *each* $v_i \in C$ **do**

  if $v_j \in SCF_{v_i}$

  $U_{v_{ij}} = \alpha_1 U^{v_i}_{v_j} + \frac{\alpha_2 \sum_{k=1}^{N} U^{v_k}_{v_j}}{N}$, *where* $\alpha_1 + \alpha_2 = 1$ *with* $\alpha_1 > \alpha_2$ *and N is the number of vehicle from which* v_i *received the utility function lists of* v_j

end

if $U_{v_{ij}} > U_{v_{ik}} \quad \forall\ v_k \in C$ **then**

  $cluster^C_{-v_j} \xrightarrow{CH_{elect}} v_j$ */* vehicles in C informs v_j that it is the PCH */*

  $v_j \xrightarrow{ACK} cluster^C_{-v_j}$ */* v_j acknowledges that it is the PCH */*

else

  $v_j \xrightarrow{CH_{elect}} v_{j'}$, where $U_{v_{ij'}} > U_{v_{ij}} \quad \forall\ v_j \in C$

  $v_{j'} \xrightarrow{ACK} v_j$

end

To enhance the connectivity among vehicles within the cluster and to ensure that malicious vehicles are not provided cluster memberships, the clustering algorithm places additional constraints, which require vehicles in the cluster to be within four-fifth radio range of the elected PCH and also have an average reputation greater than the predefined threshold value (R_{th}) to be cluster members. As the vehicles on the boundary radio range of the PCH are more likely to exit the cluster, this process ensures the stability of the cluster and min-

imizes the frequency of the cluster formation process. The value of R_{th} is set to one-forth of the average reputation value of the agent nodes in the cluster. A detailed description about the agent nodes election process and their reputation update mechanism is provided in Section 5.3.4.

5.3.3 VCG mechanism based payment structure for CH

Since monitoring operation requires a substantial amount of computing resources therefore, vehicles do not have any profitable incentive to act as the CH unless they are provided with some form of stimulus. Towards this end, an incentive based structure for encouraging vehicles to participate in the CH election process is proposed. The payment is made to the elected CH in the form of enhanced reputation gain for carrying out the monitoring operations. Here CH refers to the PCH. Data packets of reputed vehicles are given higher priority compared to those with lower reputation values during traffic routing. Therefore, vehicles with higher reputation values maintain greater throughput even during the congestion period. The cost function of the vehicle v_i for performing the monitoring operation after being elected as the CH of the cluster C is given by the following equation:

$$Cst_{v_i} = \frac{R_{v^i_{avg}}}{\sum_{j=1}^{n} R_{v^j_{avg}}} * \frac{n}{U_{v^i_{avg}}} \tag{5.5}$$

where *n* is the total number of vehicles in C. $R_{v^i_{avg}} = \frac{\sum_{k=1}^{n-1} R_{v^k_i}}{n-1}$, $\forall\ v_k \in C$ and $v_k \neq v_i$ is the average reputation value of v_i in C. $U_{v^i_{avg}} = \frac{\sum_{k=1}^{n-1} U^{v_k}_{v_i}}{n-1}$ is the average aggregated utility function value of v_i in C, with $U^{v_k}_{v_i}$ calculated using Equation 5.4. Each vehicle v_i holds a private information about its type (Θ_{v_i}). The type Θ_{v_i} can be either *Normal* or *Malicious*. The reward function for vehicle v_i when it is elected as the CH by the mechanism is given by following equation:

$$Rwd_{v_i}(\Theta_{v_i}, \Theta_{-v_i}) = P_{v_i} - Cst_{v_i} \tag{5.6}$$

where Θ_{-v_i}, represents the type of all other vehicles except v_i. P_{v_i} is the payment made by the mechanism to v_i in the form of enhanced reputation gain. Every vehicle $v_k \in C$ would want to maximize its reward function (Rwd_{v_k}). It signifies the reward value for v_k if it chooses the type Θ_{v_k}. The vehicles might not truthfully reveal their cost function value by either over valuing or under valuing them, if doing so leads to higher reward. Therefore,

to address this issue, a payment structure based on VCG mechanism is proposed, wherein truthful revelation of the cost function value is the dominant strategy [144] [147]. The primary objective of the proposed CH election mechanism is to elect the vehicle $v_i \in C$ with the least cost function value (Cst_{v_i}) as the CH. Since, $Cst_{v_i} \propto \frac{1}{U_{v^i_{avg}}}$ therefore, electing vehicle with the least Cst_{v_i} as the CH is equivalent to electing vehicle with the highest average aggregated utility function value ($U_{v^i_{avg}}$) as the CH. We refer to this as the Social Choice Function (SCF) and define it as:

$$SCF = Min_{v_i \in C}\Big\{Cst_{v_i}\Big\}$$

If multiple vehicles have the same *SCF* value, then the vehicle with the highest reputation value amongst them is elected as the CH. However, if there is a tie even in the reputation values, then the vehicle with greatest number of vehicles in its *SCF* table is elected as the CH. The payment to the elected CH vehicle v_i is made using the VCG mechanism. The payment received by v_i (P_{v_i}) is equal to the second least cost function value of vehicle v_k (Cst_{v_k}) excluding the cost function of v_i.

$$P_{v_i} = Min_{v_k \neq v_i \in C}\Big\{Cst_{v_k}\Big\}$$

After v_i is elected as the CH and the payment (P_{v_i}) to be made to v_i is calculated, the agent nodes calculate the reward function for v_i (Rwd_{v_i}) using Equation 5.6 and inform the RSU to increment the reputation value of v_i by Rwd_{v_i} value.

In the subsequent sub-sections, we provide a detailed description about various components of the proposed VANET intrusion detection framework namely, the agent node's LIDS module, CH's CIDS module and the RSU's GDS module. These components interact with each other to identify the malicious vehicles and provide a comprehensive security to the vehicular network.

5.3.4 Agent node's Local Intrusion Detection System (LIDS) module

At the lowest level of the proposed IDS framework, a set of agent nodes are used to monitor the vehicles in a given cluster. The agent nodes use LIDS modules to monitor vehicles for sign of maliciousness. Each agent node maintains a Social Choice Function (*SCF*) table comprising the identities, velocity vectors and the reputation values of the vehicles in

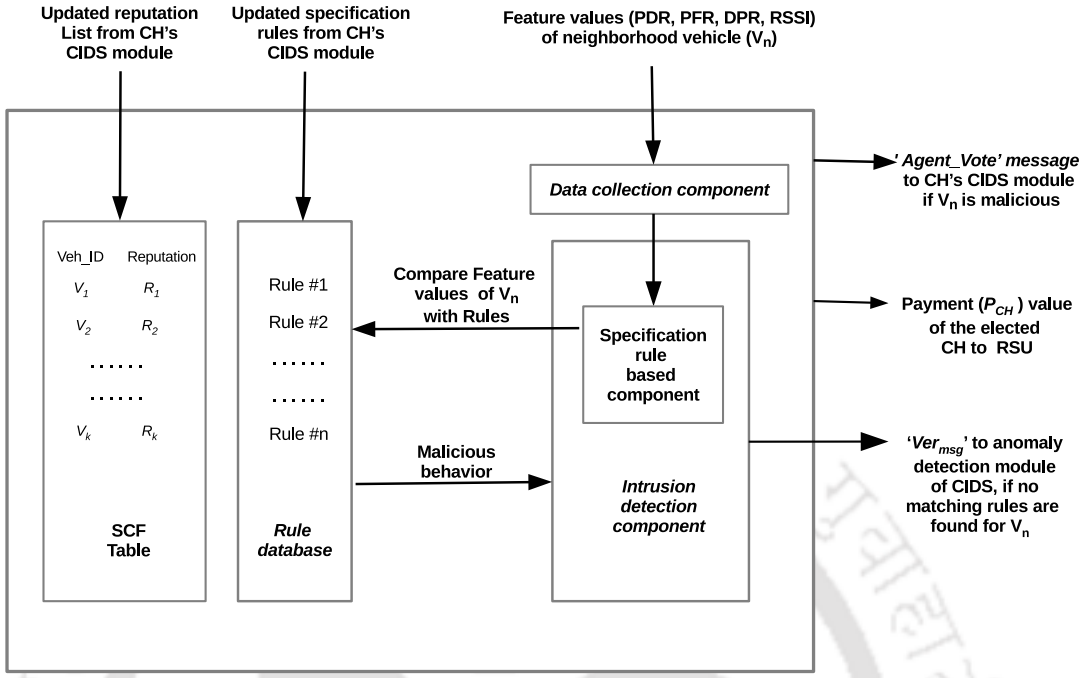

Figure 5.8: Agent node's LIDS module

its neighborhood. The agent node's *SCF* table also maintains the identities of the vehicles blacklisted by the RSU and the CH. The blacklisted vehicles have negative reputation values in the *SCF* table and are excluded from the vehicular communication.

The agent node's LIDS module uses a set of specification rules based on Received Signal Strength Indicator (RSSI), Packet Drop Rate (PDR), Packet Forwarding Rate (PFR) and Duplicate Packet Rate (DPR) values of the vehicles to detect malicious vehicles in the cluster. The overall architecture of the agent node's LIDS module is shown in Fig. 5.8. As shown in the figure, the '*Data collection component*' of the LIDS module computes the RSSI, PDR, PFR and DPR values of the vehicle being monitored. These information are then forwarded to the '*Intrusion detection component*', which uses a set of specification rules (Algorithm 4) stored in its '*Rule database*' to detect various type of attacks. When a malicious vehicle is identified by the agent node, it sends an '*Agent_Vote*' message comprising the identity of the malicious vehicle along with the attack type detected to the CH. The CH collects '*Agent_Vote*' messages from multiple agent nodes in its cluster to determine whether the vehicle being reported is indeed malicious. However, when there are no matching specification rules against the vehicle being monitored, a Ver_{msg} is sent to the CH's neural network based

'*Anomaly detection component*' for further analysis. Since the network's dynamics in VANET changes frequently, agent nodes receive updated specification rules from the CH at regular intervals.

Additionally, the agent nodes also monitor the CH for sign of maliciousness. Since CHs perform many vital tasks like data aggregation and monitoring, they are attractive targets for attacker as compromising them can provide the attacker with huge payoffs. The attacker can disrupt the vehicular network's operation through a compromised CH by propagating false information and ignoring to act against the malicious vehicles reported by the agent nodes. Therefore, a cooperative detection mechanism (Algorithm 5) is adopted by the agent nodes to identify the malicious CH. Each agent node maintains a binary variable called *CH_Status*, which is initially set to 0. However, when the agent node finds the CH to be malicious, it sets the *CH_Status* variable to 1. The agent node uses the set of specification rules given in Algorithm 4 to detect the malicious CH. When the agent node finds the CH to be malicious, it reports to the RSU. When more than one-half of the agent nodes in the cluster report the CH as malicious, the RSU blacklists the reported malicious CH and broadcasts a message asking the vehicles in the cluster to elect a new CH.

5.3.4.1 Distributed agent nodes election algorithm

The performance of the proposed intrusion detection framework largely depends on the agent nodes election algorithm. Electing few agent nodes degrade the detection rate of the IDS framework, while electing too many agent nodes introduce a large volume of intrusion detection related traffic, which can cause network congestion. Therefore, to maintain a good trade-off between the detection rate and the IDS traffic volume, a distributed agent nodes election algorithm (Algorithm 6) is proposed that elects an optimal number of highly reputed vehicles as the agent nodes. The agent nodes election process starts with a vehicle v_k broadcasting the *IDS_Agent_Elect ( )* message comprising its identity and its SFC table details. Upon receiving the *IDS_Agent_Elect ( )* message from v_k, every other vehicle in the cluster C broadcast their own *IDS_Agent_Elect ( )* messages. Each vehicle in C computes the average aggregated reputation ($Agg_{R_{v_i}}$) of every other vehicle $v_i \in C$ using the SCF table details it received in the *IDS_Agent_Elect ( )* messages from other vehicles. The algorithm then elects the top 'k' vehicles with the highest aggregated reputation values as the agent nodes. Through various round of simulations, it was observed that the best trade-off between the detection rate and the IDS traffic volume is obtained when 25% to 30% of the

Algorithm 4 Detection rules for various attacks

Input : *Identity (Node_ID), PDR, RSSI, DPR and PFR values of the vehicle.*

Output : *Prediction whether Node_ID is malicious or normal.*

if *(*$PDR_{Node\_ID} > T_{pdr_{sf}}$*)* **then**
  // node is performing selective forwarding attack
  send **Agent_Vote(Node_ID, selective forwarding)** message to the CH
end
if *(* $RSSI_{Node\_ID} > T_{rssi_{syb}}$*)* **then**
  // node is performing sybil attack
  send **Agent_Vote(Node_ID, sybil attack)** message to the CH
end
if *(*$PDR_{Node\_ID} > T_{pdr_{bh}}$ & $RSSI_{Node\_ID} > T_{rssi_{bh}}$*)* **then**
  // node is performing black hole attack
  send **Agent_Vote(Node_ID, black hole attack)** message to the CH
end
if *(*$DPR_{Node\_ID} > T_{dpr_{dos}}$ & $PFR_{Node\_ID} > T_{pfr_{dos}}$*)* **then**
  // node is performing DoS attack
  send **Agent_Vote(Node_ID, DoS attack)** message to the CH
end
if *(*$RSSI_{Node\_ID} > T_{rssi_{wh}}$ & $PDR_{Node\_ID} > T_{pdr_{wh}}$*)* **then**
  // node is performing worm hole attack
  send **Agent_Vote(Node_ID, worm hole attack)** message to the CH
end

Algorithm 5 Distributed cooperative mechanism for detecting malicious CH

Input : *'k' agent nodes' CH_Status variables.*

Output : *Prediction whether the CH is malicious or normal.*

$Agt_i \xleftrightarrow{CH\_Status} Agt_{k-i}$ */* k agent nodes exchange their CH_Status messages */*
if *Count(CH_Status == 1) < k/2;* **then**
  CH is normal
else
  CH is malicious
  Report the malicious CH to RSU
  RSU informs vehicles in the cluster to initiate a new CH election process
end

vehicles in the cluster are elected as the agent nodes.

Algorithm 6 Distributed election algorithm to elect *k* IDS agents of a cluster

Input : *Cluster C and IDS agent election messages* $\left(IDS\_Agent\_Elect(\,)\right)$.

Output : *'k' elected IDS agents of cluster C.*

$v_k \leftrightarrow Cluster^{C}_{-v_k}$: *IDS_Agent_Elect* $\left(ID_{v_k},\ SCF_{v_k}\right)$ */* Vehicles exchange the agent nodes election messages. */*

Each vehicle calculates the average aggregated reputation of every other vehicle $v_i \in C$ $(Agg_{R_{v_i}})$ *using the SCF list information obtained from the IDS_Agent_Elect ( ) messages.*

Let $\{R_{k_{th}}\}$ *be the set of 'k' number of vehicles in* C *with the highest aggregated reputation values.*

if $v_i \in \{R_{k_{th}}\}$ **then**

$Cluster^{C}_{\{-R_{k_{th}}\}} \xrightarrow{IDS_{agent}} v_i$ */* vehicles in C informs* v_i *that it is the agent node. */*

$v_i \xrightarrow{Confirm} Cluster^{C}_{\{-R_{k_{th}}\}}$ */* vehicles in C informs* v_i *that it is the agent node. */*

else

$v_i \xrightarrow{IDS_{agent}} v_j;\quad \forall \quad v_j \in \{R_{k_{th}}\}$

$v_j \xrightarrow{Confirm} v_i$

end

5.3.5 CH's Cluster Intrusion Detection System (CIDS) module

At the intermediate level of the proposed intrusion detection framework, the CH uses the *Cluster Intrusion Detection System (CIDS)* module to detect malicious vehicles. The overall architecture of the CH's CIDS module is shown in Fig. 5.9. As shown in the figure, the CIDS module comprises four different components namely, the '*Rule based detection component*', the neural network based '*Anomaly detection component*', the '*Update Rule component*' and the agent node '*Reputation update component*'. A detailed description about each of these components are provided in the subsequent sub-sections.

5.3.5.1 Rule based detection component

This component uses a set of specification rules based on RSSI, PDR, PFR and DPR values of the vehicles (Feature sets) to detect the malicious vehicles in the cluster. It uses the same

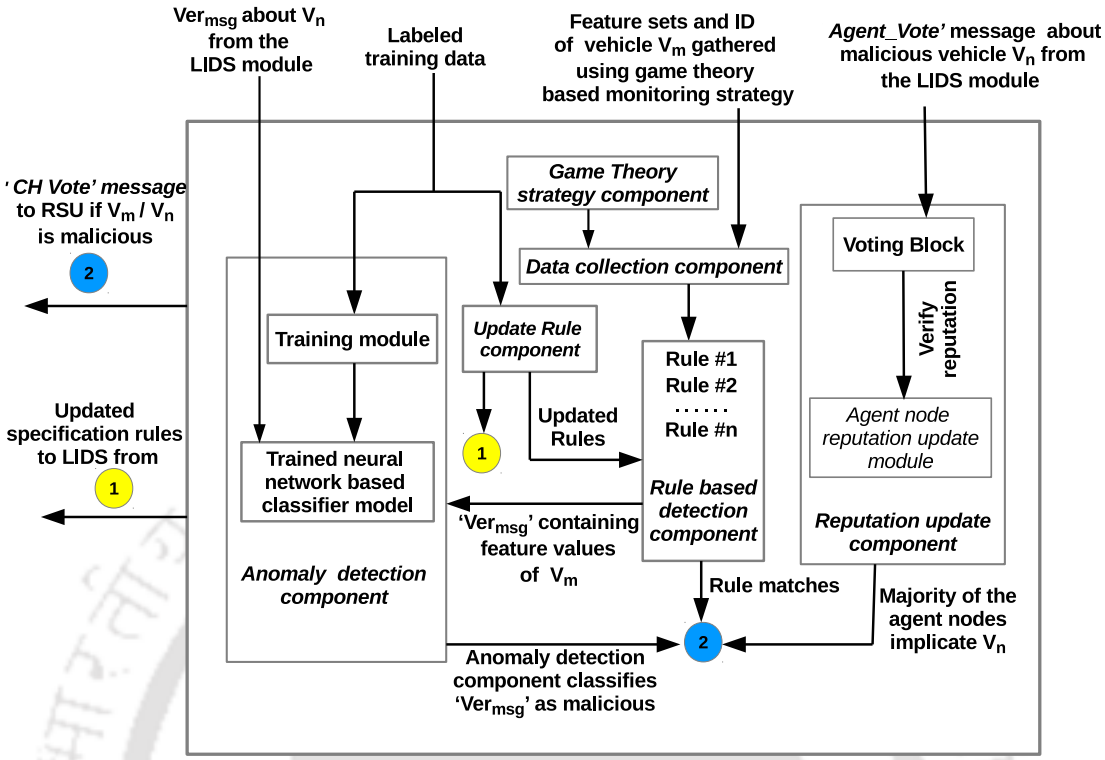

Figure 5.9: CH's CIDS module

set of specification rules as used by the agent node's LIDS module (Algorithm 4) to identify the malicious vehicles. When there are no matching specification rules, a verification message (Ver_{msg}) containing the Feature sets, reputation value and identity (ID) of the vehicle being monitored is sent to the '*Anomaly detection component*' for further analysis. The specification rules of the CH's CIDS module are updated more frequently compared to that of the agent node's LIDS module and therefore, the CIDS module contains the more updated version of the specification rules. When a malicious vehicle is detected by the CIDS module, it informs both the RSU and the agent nodes in its cluster about the malicious vehicle.

5.3.5.2 Neural network based anomaly detection component

This CH's CIDS component takes its input from the agent node's LIDS module and the CIDS's '*Rule based detection component*' in the form of verification messages (Ver_{msg}) and classify them as either normal or one of the attack types (malicious). The rationale behind choosing a neural network based classifier module is because of its ability to produce better classification model as compared to other classifier modules. The anomaly detection com-

ponent is initially trained with a labeled dataset comprising the Feature sets of the vehicles. Additional features like packet size, source and destination IP addresses, hop counts, packet sequence number, velocity and region ID are also used as parameters for training the classifier module. Since, the normal vehicular features in VANET vary over time due to change in several network parameters like topology, mobility, traffic conditions, congestion etc., therefore, anomaly detection component needs to re-trained periodically to incorporate the changes in the underlying network parameters into the classifier module.

5.3.5.3 Update rule component

This CH's CIDS component provides the updated specification rules to the '*Rule based detection component*' of both the LIDS and CIDS modules. However, the '*Rule based detection component*' of the CIDS module is updated more frequently compared to that of the LIDS module. The mean and the standard deviation values of the PDR, RSSI, PFR and DPR of the vehicles observed during the normal training period are used to derive new specification rules. Since, the specification rules of the LIDS module are updated less frequently, there is a possibility that the agent nodes might end up using outdated rules for monitoring vehicles. This can result in an increased false alarm rate and misclassification of some malicious vehicles as normal. However, the probability of such vulnerability is offset by the fact that all the malicious vehicles reported by the agent nodes are re-verified by the CH, which ensures that the overall performance of the IDS framework is maintained.

5.3.5.4 Agent node reputation update component

This CH's CIDS component maintains a reputation list of all the agent nodes in its cluster and updates their reputation values using the procedure described in Algorithm 7. In Algorithm 7, $\{Agent\}$ denotes the set of all the k agent nodes in the cluster and $\{Agent^*\} \subseteq \{Agent\}$ denotes the set of k^* ($k^* \leq k$) agent nodes that reported the vehicle v_m as malicious. The CH computes the average reputations R^k_{agg} and $R^{k^*}_{agg}$ of the agent nodes in the set $\{Agent\}$ and $\{Agent^*\}$, respectively. If $R^{k^*}_{agg}$ is greater than or equal to (R^k_{agg}), then the reputation of the agent nodes that reported v_m as malicious are incremented by one-fourth of their current reputation values. However, if $R^{k^*}_{agg}$ is less than R^k_{agg} but greater than one half of R^k_{agg}, then v_m is considered to be suspicious by the CH and a game theory-based probabilistic monitoring strategy is adopted by the CH to monitor v_m. Finally, if $R^{k^*}_{agg}$ is less

than one-half of R^k_{agg}, then the reputation values of the agent nodes that reported v_m as malicious are decremented by one-forth of their current values. Such a collaborative detection mechanism ensures that the malicious agent nodes can not collude together to falsely implicate a normal vehicle as malicious. In addition, when the reputation of any agent node falls below the threshold value ($0.3R^k_{agt}$), it is replaced with a new agent node.

Algorithm 7 Agent node reputation update mechanism

Input : *1) Cluster C with cluster head (CH)*

2) $\{Agent\} = \{agt_1, ..., agt_k\}$ *// Set of k agent nodes in C*

3) $\{Agent^*\} = \{agt_1, ..., agt_{k^*}\}$ *// Set of k^* ($k^* \leq k$) agent nodes that reported vehicle v_m as malicious*

Output: *Updated reputation values of agent nodes in $\{Agent^*\}$*

$R^{k^*}_{agg} = \sum_{j=1}^{k^*} \frac{R^{CH}_{agt_i}}{k^*}$ *// Aggregated reputation of the k^* agent nodes in $\{Agent^*\}$*

if *($R^{k^*}_{agg} \geq R^k_{agg}$)* **then**

  v_m is malicious

  $R^{CH}_{agt_i} = R^{CH}_{agt_i}$ + 0.25*$r^{CH}_{agt_i}$ $\forall$ $agt_i \in \{Agent^*\}$

else if *(0.5*$R^k_{agg} < R^{k^*}_{agg} < R^k_{agg}$)* **then**

  *Monitor v_m with monitoring probability determined by the '**Game Theory strategy component**'*

else

  $R^{CH}_{agt_i} = R^{CH}_{agt_i}$ - 0.25*$R^{CH}_{agt_i}$ $\forall$ $agt_i \in \{Agent^*\}$

end

if *($R^{CH}_{agt_j} < 0.3R^k_{agg}$)* **then**

  remove agt_j from the {Agent}

5.3.5.5 Game Theory strategy component

This CIDS component devices probabilistic monitoring strategies for the CH based on various parameters like CH's detection rate, false alarm rate and monitoring cost. Persistent CH monitoring operation produces a significant volume of intrusion detection related traffic, which can cause congestion in a bandwidth constrained vehicular network. To address this issue, a game theory-based probabilistic monitoring strategy is adopted by the CH to monitor the malicious vehicles reported by the agent nodes. The interaction between the

malicious vehicle and the CH is formulated as a two player non-cooperative game between the attacker and the defender. Without loss of generality, we make an assumption that both the CH and the malicious vehicle are rational players and their actions are based upon intelligent consideration of the possible consequences of their chosen strategy. In the said game, the malicious vehicle has two pure strategies : *Attack* or *Wait*. Similarly, the CH has two pure strategies : *Monitor* or *Not Monitor*. Each player choses a strategy that maximizes its overall payoffs. To develop the payoff matrix corresponding to the interaction between the CH and the malicious vehicle, we introduce the following terminologies:

- Let α, β and γ denote the detection rate, the false positive rate and the monitoring cost of the CH, respectively.
- Let δ be the average number of vehicles in the cluster accepting and forwarding information from a malicious vehicle.

Table 5.3: Strategic form of the game between the malicious vehicle (attacker) and the CH (defender)

	Attack	**Wait**
Monitor	(2α - γ + 1), (1 + δ - 2α)	-(β + δ), β
Not Monitor	-(1 - α), (1 - α + δ)	0, 0

Table 5.3 shows the strategic form of the non-cooperative game between the CH and the malicious vehicle. The strategy space of the CH and the malicious vehicle are S_D = {*Monitor, Not Monitor*} and S_A = {*Attack, Wait*}, respectively. A pure Nash Equilibrium (NE) of this non-cooperative game corresponds to the strategy pair (S_d^*, S_a^*) of the CH and the malicious vehicle that satisfies the following conditions:

$$U_A(S_d^*, S_a^*) \geq U_A(S_d^*, S_a) \quad \forall\, S_a \in S_A$$
$$U_D(S_d^*, S_a^*) \geq U_D(S_d, S_a^*) \quad \forall\, S_d \in S_D$$

where, $U_A(S_d^*, S_a^*)$ and $U_D(S_d^*, S_a^*)$ are the payoff utilities of the malicious vehicle and the CH when they choose their strategy S_d^* and S_a^*, respectively. Any unilateral deviation by either the CH or the malicious vehicle from their chosen NE strategy results in a reduced payoff for the deviating player. Clearly, there does not exist any pure strategy NE for this non-cooperative game. Therefore, we derive a mixed strategy NE. Let p and q denote the

probabilities of the malicious vehicle and the CH to play their pure strategies $Attack$ and $Monitor$, respectively. When the CH plays its strategy *Monitor* with probability *q*, the payoff utility of the malicious vehicle if it plays its pure strategies *Attack* and *Wait*, respectively are:

$$U_A(Attack) = (1 + \delta - 2\alpha)q + (1 - \alpha + \delta)(1 - q)$$
$$U_A(Wait) = \beta q$$

Similarly, when the malicious vehicle plays its strategy *Attack* with probability *p*, the payoff utility of the CH if it plays its pure strategies *Monitor* and *Not Monitor*, respectively are:

$$U_D(Monitor) = (2\alpha - \gamma + 1)p - (\beta + \delta)(1 - p)$$
$$U_D(Not\ \ monitor) = -(1 - \alpha)p$$

The malicious vehicle chooses to play its strategy *Attack* when U_A*(Attack)* > U_A*(Wait)*, i.e., when the monitoring probability of the CH (*q*) < $\frac{(1-\alpha-\delta)}{\alpha+\beta}$. Similarly, the CH choses to play its strategy *(Monitor)* when U_D*(Monitor)* > U_D*(Not monitor)*, i.e., when the malicious vehicle's attack probability (*p*) > $\frac{(\beta+\delta)}{(2+\alpha+\beta+\delta-\gamma)}$. Therefore, the mixed strategy NE of the non-cooperative game between the malicious vehicle and the CH corresponds to the strategy combination (p^*, q^*), where $p^* = \frac{(\beta+\delta)}{(2+\alpha+\beta+\delta-\gamma)}$ and $q^* = \frac{(1-\alpha-\delta)}{\alpha+\beta}$ are the probabilities of the malicious vehicle and the CH to play their strategy *Attack* and *Monitor*, respectively. It can be observed that both the attacking and the monitoring probabilities of the malicious vehicle and the CH are inversely proportional to the detection rate (α) of the CH i.e., $p^* \propto \frac{1}{\alpha}$ and $q^* \propto \frac{1}{\alpha}$. Therefore, a high value of α decreases both the attacking and monitoring probabilities at the NE. Adopting such a probabilistic game theory-based monitoring strategy significantly reduces the volume of intrusion detection related traffic in the vehicular network, without compromising the overall performance of the IDS framework.

5.3.6 RSU's Global Decision System (GDS) module

At the highest level of the proposed intrusion detection framework, the RSU maintains a blacklist of all the malicious vehicles being reported by the CHs. The CH uses the '*CH*

Vote' message to report the malicious vehicles to the RSU. We make an implicit assumption that RSUs are interconnected through secured connections and powerful firewalls, which prevent them from being compromised. Multiple CHs are associated with a given RSU. The i^{th} RSU (RSU^i) computes the aggregated reputation of the vehicle v_m being reported by the CHs using the following rule:

$$Agg_{v_m}^{RSU^i} = \frac{\sum_{k=1}^{n'} R_{CH_k}^{RSU^i}/n'}{\sum_{j=1}^{n} R_{CH_j}^{RSU^i}/n}$$

where n' and n are the number of the CHs that reported v_m as malicious and the total number of CHs within the radio range of RSU^i, respectively ($n' \subseteq n$). $R_{CH_j}^{RSU^i}$ is the reputation value of the j^{th} CH in RSU^i's reputation list. The RSUs exchange their computed aggregated reputation values for v_m. The global aggregated reputation value of v_m is then calculated using the following rule:

$$Glb_{v_m}^{RSU} = \frac{\sum_{i=1}^{l} Agg_{v_m}^{RSU^i}}{l}$$

where 'l' is the number of RSUs through which v_m has passed. Finally, v_m is categorized into one of the category class based on the following rules:

$$\begin{cases} Glb_{v_m}^{RSU} \leq 0.25, & v_m \ is \ normal \\ 0.25 < Glb_{v_m}^{RSU} \leq 0.6, & v_m \ is \ suspicious \\ 0.6 < Glb_{v_m}^{RSU} \leq 1, & v_m \ is \ malicious \end{cases}$$

The overall architecture of the RSU's GDS module is shown in Fig. 5.10. As shown in the figure, the RSU stores the identity of suspicious and malicious vehicles in its *Blacklist* table. It periodically broadcasts the identity of these vehicles to prevent other normal vehicles in its radio range from communicating with these malicious and suspicious vehicles. All the post crash notification and congestions messages received from malicious vehicles are ignored and discarded by the normal vehicles. In addition, the suspicious vehicles are debarred from participating in the CH and agent node election process. Therefore, the proposed intrusion detection framework ensures that only the trustworthy vehicles are elected as the CH and the agent nodes. The *Reputation List* table of the RSU receives the payment value (P_{CH}) to be made to the elected CH from the agent nodes. The RSU increments the reputation of the elected CH in its *Reputation List* by the P_{CH} value and broadcasts a message asking all the vehicles in its radio range to update the reputation value of the elected CH.

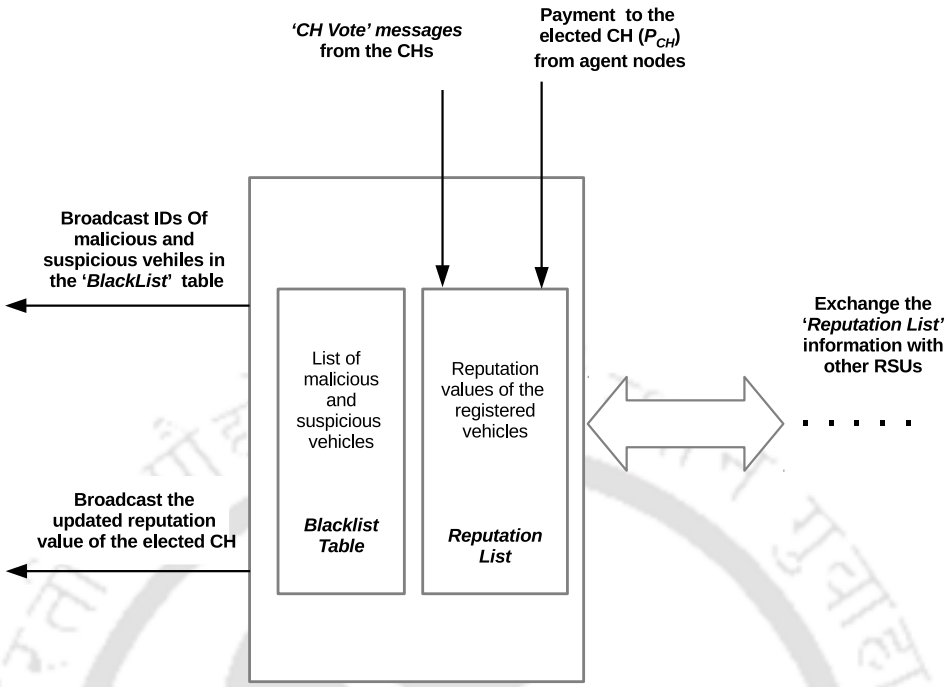

Figure 5.10: RSU's GDS module

Fig. 5.11 shows the overall interaction between various modules of the proposed IDS framework namely, the agent node's LIDS module, the CH's CIDS module and the RSU's GDS module. As shown in the figure, the agent node's LIDS module communicate with the CH's CIDS module using the '*Agent Vote*' and '*Verification*' messages. The CH use these messages from the agent nodes to detect malicious vehicles in its cluster. Additionally, the CH uses a combination of specification rules and a neural network based anomaly detection component to detect malicious vehicles and agent nodes in its neighborhood. Finally, the CH's CIDS module communicates with the RSU's GDS module using '*CH Vote*' messages. The RSU uses the vote messages received from its CHs to identify the malicious vehicles and agent nodes in its radio range. These malicious vehicles and agent nodes are then included in the RSU's Blacklist table. The CH then broadcasts the identities of the malicious vehicles in its Blacklist table to prevent other normal vehicles in the network from communicating with them.

5.4 Experimental Results

We have classified the experimental result section into two sub-sections namely, the *simulated vehicular network traffic* and the *real time vehicular network traffic*. The experimental

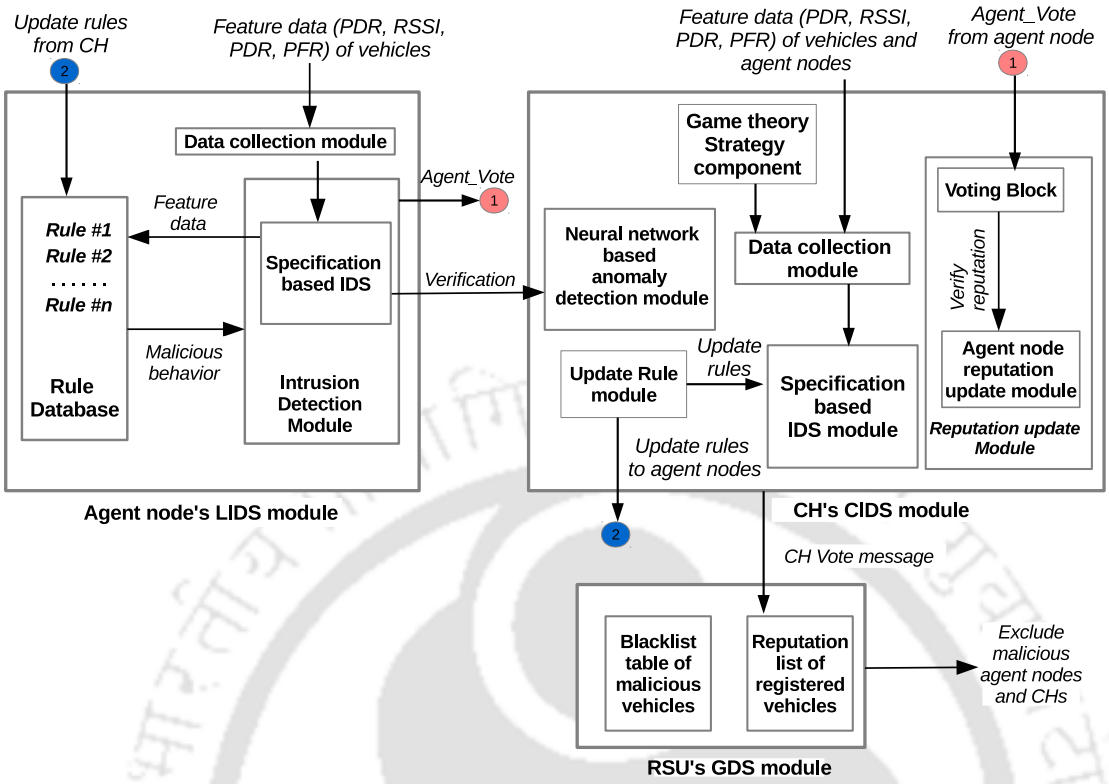

Figure 5.11: Interaction among various modules of the proposed IDS framework

setup and the results obtained on the simulated and the real time vehicular network traffic are provided in the sub-sequent subsections.

Following parameters were used to analyze the performance of different IDS frameworks: 1) Detection rate 2) False alarm rate 3) IDS traffic volume and 4) Average cluster membership duration of vehicles. We define the following terminologies prior to defining the detection rate and the false alarm rate of the IDS. *True positives (TPs)*: These are cases wherein the IDS correctly identifes the the attacks. *False positives (FPs)*: These are cases in which normal data traffic is incorrectly classified as attacks by the IDS. *False negatives (FNs)*: These are cases wherein the IDS fails to detect the attacks.

- **Detection Rate (DR)**: It is defined as the ratio of the actual number of attacks detected by the IDS to the total number of attacks in the network.

$$DR = \frac{TP}{TP + FN} \tag{5.7}$$

- **False Alarm Rate (FAR)**: It is defined as the ratio of number of normal data incorrectly

classified as attacks to the total number of attacks detected by the IDS.

$$FAR = \frac{FP}{FP + TP} \tag{5.8}$$

- **IDS Traffic Volume (ITV)**: It is defined as the ratio of volume of the intrusion detection related traffic to the total volume of traffic in the network (IDS and non IDS traffic) at any given instance of time.

$$ITV = \frac{IDS\ traffic}{IDS\ traffic + non\ IDS\ traffic} \tag{5.9}$$

- **Average cluster membership duration (ACMD)**: It is defined as the average period for which the vehicle remains associated with a cluster after it has been assigned to a particular cluster by the clustering algorithm.

5.4.1 Simulated vehicular network traffic

To evaluate the proposed IDS framework, simulations were performed in the NS3 [148] simulator with the realistic mobility of the vehicles generated by the open source traffic simulator, Simulation of Urban Mobility (SUMO) [149]. NS3 was chosen over NS2 for simulation in this chapter because NS2 does not support realistic vehicular mobility required for simulation of vehicular networks. A coordination mechanism was built to combine the traffic simulation capabilities of SUMO with the network simulation capabilities of NS3. As shown in Fig. 5.12, a square grid road topology of 5 $\times$ 5 km consisting of a two-lane roads and four intersection points in SUMO was considered for network traffic simulation. Each grid is identified by a unique ID (G1 through G25). The vehicles were injected into the road according to the Poisson process with rate equal to four vehicles per second. The total simulation time was 500 seconds. The clustering process started at the 60^{th} second when all the vehicles had entered the road. All the performance metrics were evaluated for the remaining 440 seconds. Two classes of vehicles with different maximum speed ranges were used in the simulation to create a realistic scenario with different types of vehicles on the road, such as passenger cars, buses, and trucks. The first class of vehicles had a maximum speed of 10 m/s, whereas the maximum speed of the second class of vehicles were varied between 10 m/s to 35 m/s.

We used the IEEE 1609 Wireless Access in Vehicular Environments (WAVE) protocol stack

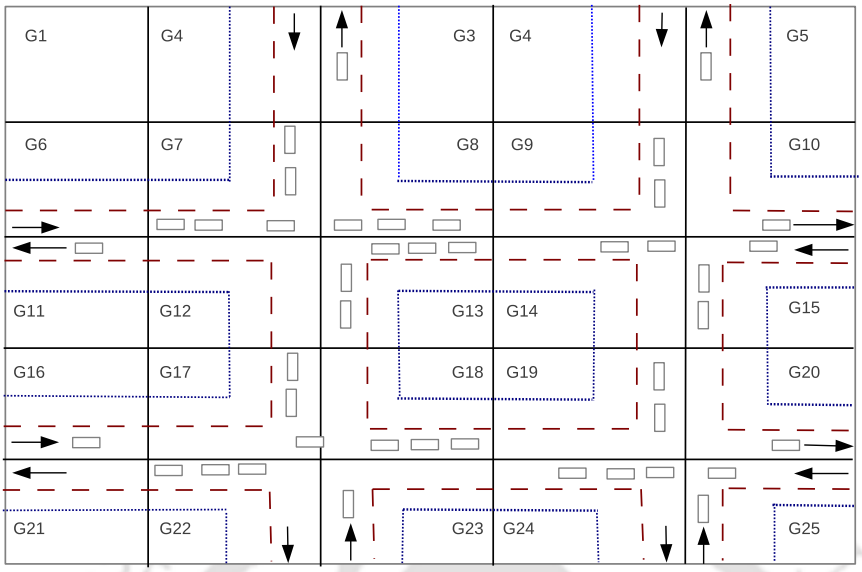

Figure 5.12: Simulation traffic scenario

[150] that builds on IEEE 802.11p WLAN standard and operates on seven reserved channels in the 5.9 GHz frequency band for our analysis. The vehicles use 802.11p WiFi with continuous access to a 10 MHz Control Channel (CCH) to transmit 300 byte safety message 12 times per second at 3 Mbps using WAVE Share Message Protocol (WSMP) packets. In addition, all vehicles attempt to randomly send 256 byte IP packets at an application rate of 6 Mbps using the Service Channels (SCHs) channels. Our measurements are based on averaging the results obtained from 10 simulations. The number of malicious vehicles was varied between 10% to 30% of the overall vehicles in the network. The key parameters used for simulation are shown in Table 5.4. PDR, PFR, DPR and RSSI values were calculated every 5 seconds. The set of specification rules used by the LIDS and the CIDS modules were updated every 30 and 15 seconds, respectively.

To make the SUMO and NS3 work together and to change traffic lights dynamically, a client was introduced. In order to get a meaningful data, SUMO was used to generate realistic road traffic with different type of vehicles and intelligent traffic lights. SUMO and NS3 were made to work in parallel by using Traffic Control Interface (TraCI) client, which is a generic interface that interlinks the road traffic in SUMO with network simulation of NS3. TraCI client makes it possible to control a running road traffic simulation in SUMO through commands from NS3. TraCI uses a TCP-based client/server architecture, wherein SUMO acts as a server and the external NS3 script (the “controller”) acts as a client. It helps simulate the real streets designed with lanes, traffic lights, turns and other traffic

Table 5.4: Simulation Parameters

Simulation Time	500 s
Simulation Area	$5 \times 5km^2$
Mobility	Car-following model
Propagation Model	Two-Ray Ground
No. of vehicle per cluster	15-20
No. of IDS agents per cluster	20-30%
Protocol Stack	IEEE 1609 WAVE
Routing Protocol	AODV
Radio range	200m
$T_{pdr_{sf}}$	60-65 %
$T_{pdr_{bh}}$	90-95 %
$T_{pdr_{wh}}$	80-85 %
$T_{rssi_{syb}}$	-40 - (-45) dBm
$T_{rssi_{bh}}$	-35 - (-40) dBm
$T_{rssi_{wh}}$	-50 - (-55) dBm %
$T_{dpr_{dos}}$	80-85 %
$T_{pfr_{dos}}$	90-95 %
C_m, C_a	0.15
Transmit power	30 dBm
CH's DR (α)	0.956
CH's FP rate (β)	0.085

entities. When any application in NS3 wants to change the vehicles' state in SUMO, it sends a message to the Traci client interface, which in turn generates commands according to applications and then send them to SUMO for execution followed by the retrieval of data back from the SUMO. Fig. 5.13 shows the interaction between NS3 and SUMO via the TraCI client.

Both the CH and the malicious vehicle adopt the game theory-based strategies discussed in sub-section 5.3.5.5 to maximize their overall payoff utilities. Fig. 5.14 shows the payoff utilities of the CH and the malicious vehicle under the Nash Equilibrium (NE) and the non NE strategies. The payoff utilities are calculated every three seconds into the simulation. It can be observed from the figure that if the player (CH or malicious vehicle) deviates from its NE strategy, while the opponent player continues to play the NE strategy then the payoff utility of the deviating player decreases. Therefore, the players do not have any profitable incentive to deviate from their NE strategy.

Figure 5.15 shows the Detection Rate (*DR*) of the proposed intrusion detection framework against four different type of attacks namely, selective forwarding, wormhole attack,

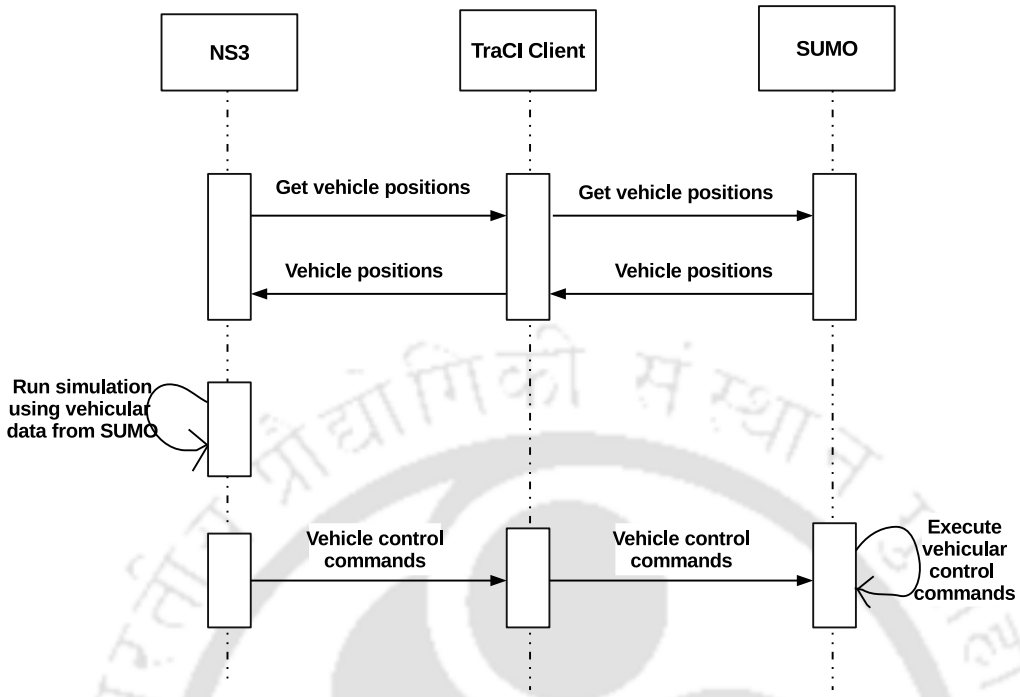

Figure 5.13: Interaction between NS3 and SUMO via TraCI client

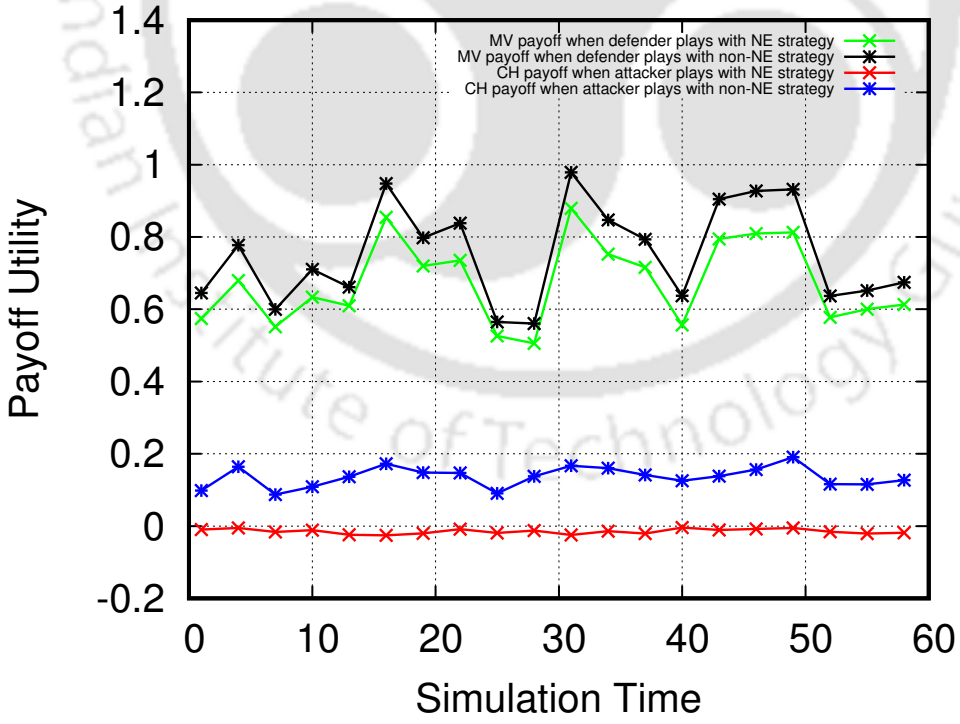

Figure 5.14: Payoff utility of the CH and the Malicious Vehicle (MV) under NE and non-NE strategies

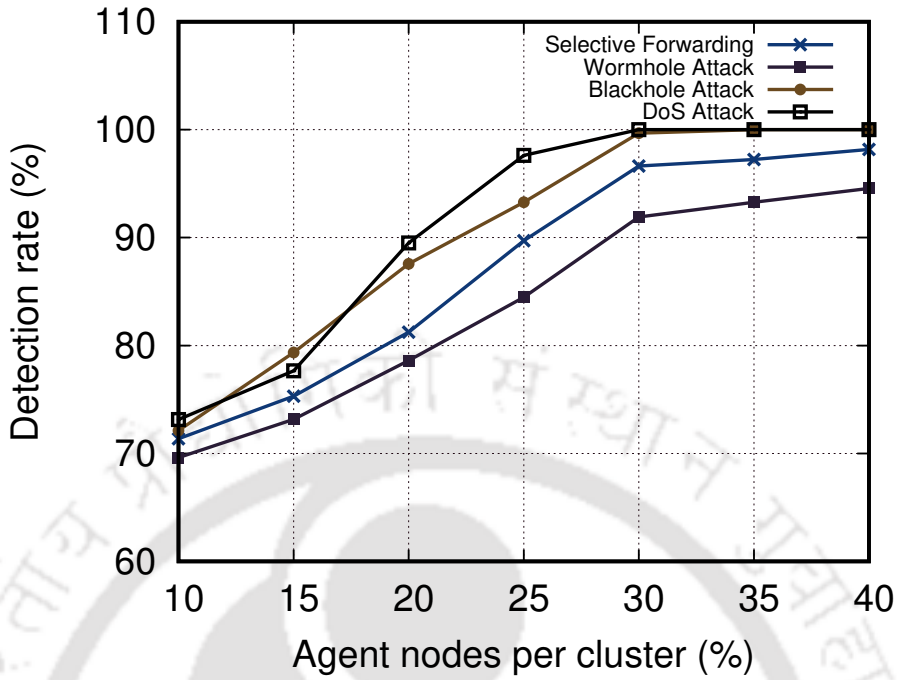

Figure 5.15: Detection rate of the proposed framework with varying percentage of agent nodes

blackhole attack and DoS attack. It can be observed from the figure that the *DR* of the proposed framework increases with the increase in the number of agent nodes per cluster. This can be attributed to the fact that as the number of agent nodes increase, more number of malicious vehicles are detected and reported to the CH. Fig. 5.16 shows the False Alarm Rate (*FAR*) of the proposed framework against various type of attacks. It can be observed from the figure that the *FAR* of the proposed framework increases with the increase in the number of agent nodes. This is because as the number of agent nodes increase, some of the malicious vehicles get elected as the agent nodes, which in turn provide false reports to the CHs.

From figures 5.15 and 5.16, it can be deduced that the best trade-off between high *DR* and low *FAR* is obtained when 25% to 30% of the vehicles in the cluster are elected as the agent nodes.

The performance of the proposed framework was evaluated against the frameworks proposed in *H.Sedjelmaci et al.* [129], *N.Kumar et al.* [141] and *A.Daeinabi et al.* [151]. The reason for choosing these frameworks for comparison is because of the similarity of the attack types considered in these frameworks with the proposed IDS framework. In addition, the framework described in [151] also proposes a clustering algorithm for VANETs, which

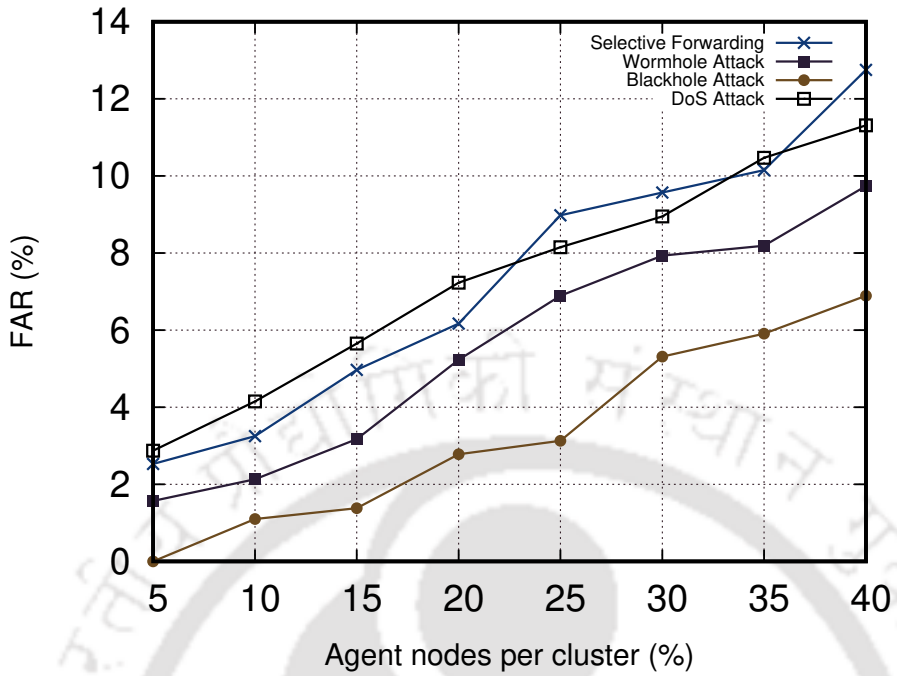

Figure 5.16: False alarm rate of the proposed framework with varying percentage of agent nodes

allows us to compare it with the proposed clustering algorithm. All these factors make them ideal candidates for comparison with the proposed framework.

Fig. 5.17 shows the volume of IDS traffic introduced into the vehicular network by various IDS frameworks (*H.Sedjelmaci et al.* [129], *N.Kumar et al.* [141] and *A.Daeinabi et al.* [151]). It can be observed from the figure that the volume of IDS traffic increases with the increase in the vehicular density for all the frameworks. However, the proposed framework introduces the least volume of IDS traffic compared to other frameworks. This can be attributed to the fact the proposed framework minimizes the amount of information exchanged between the agent node's LIDS module and the CH's CIDS module by electing optimum number of agent nodes for performing the monitoring task. In addition, the CH's CIDS module employs a game theory-based probabilistic monitoring strategy, which further reduces the volume of IDS traffic. On the other hand, the frameworks proposed in [129] [141] [151] require all the vehicles in the network to continuously perform the monitoring operation. This results in introduction of high volume of IDS traffic into the vehicular network, as more number of vehicles join the network.

Fig. 5.18 shows the *DR* and the *FAR* of various IDS frameworks against the black hole, worm hole, selective forwarding and DoS attacks. It can be observed from the figure that

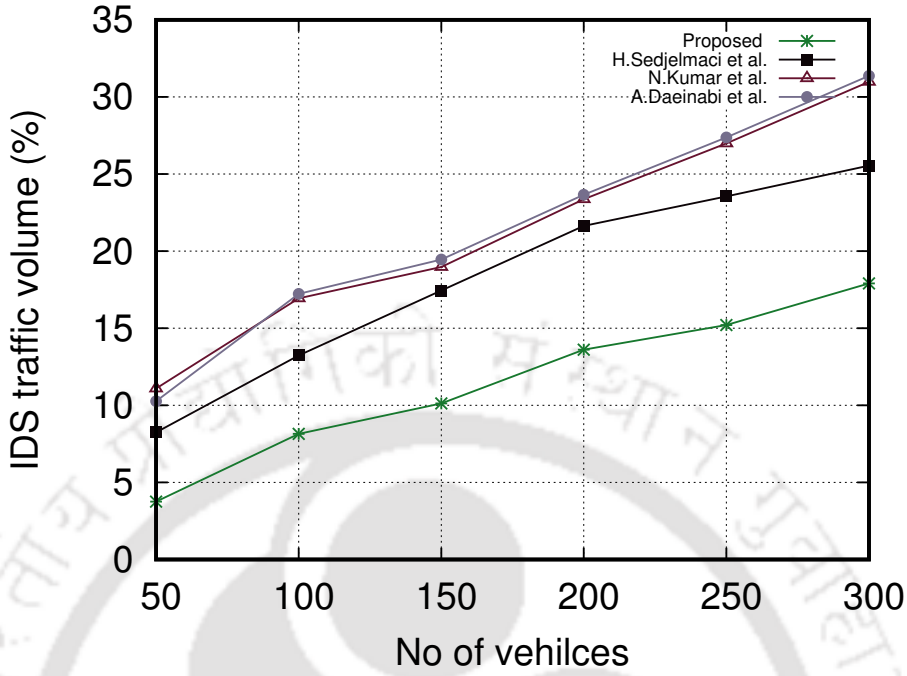

Figure 5.17: Volume of IDS traffic generated by different frameworks

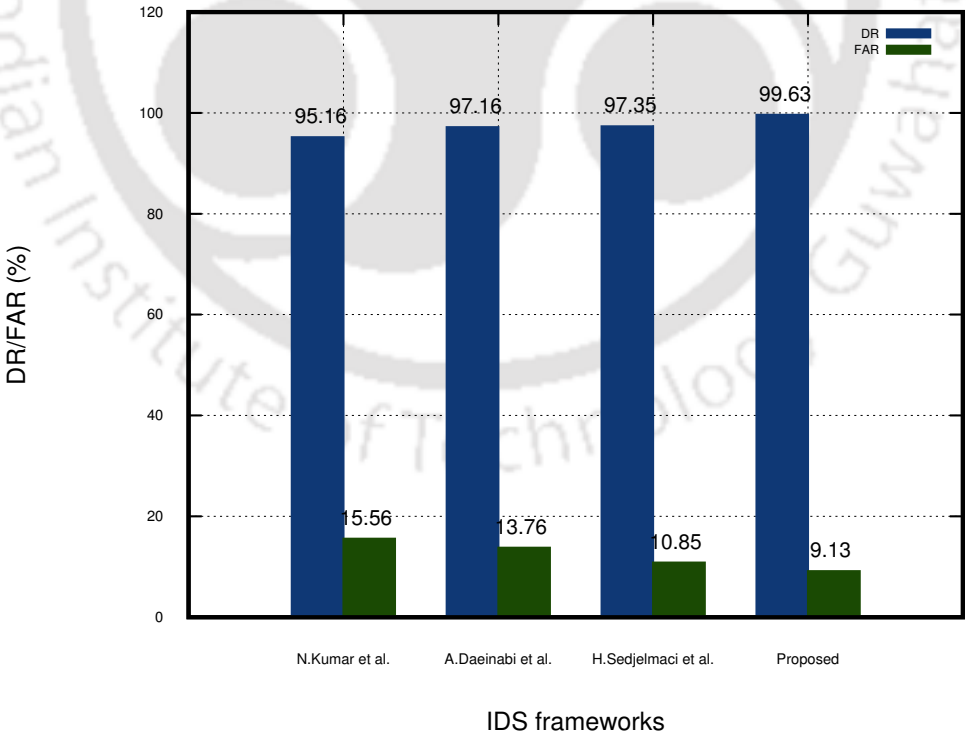

Figure 5.18: Detection rate and false alarm rate of different frameworks

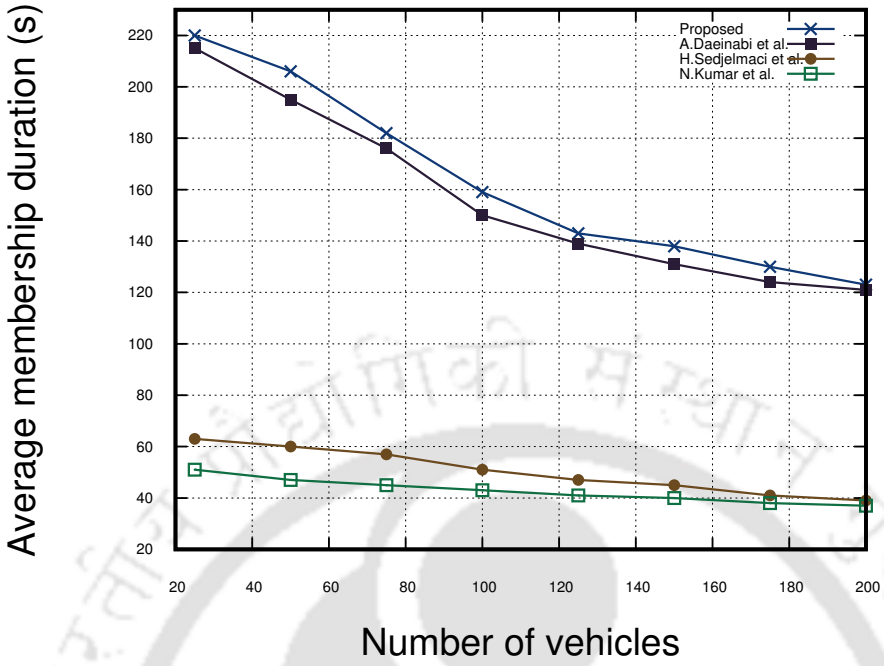

Figure 5.19: Average cluster membership duration of vehicles for various frameworks

the proposed framework achieves the highest *DR* and least *FAR* amongst all the frameworks against these attacks. The high *DR* of the proposed framework can be attributed to the fact that it uses a combination of specification rules and a lightweight neural network based classifier module to detect malicious vehicles, which greatly enhances its detection capabilities. Similarly, the proposed framework minimizes the *FAR* by electing an appropriate number of agent nodes for performing the monitoring operation. In addition, the proposed framework periodically updates the specification rules and retrains the neural network based classifier module, which further enhances its accuracy and minimizes the *FAR*.

Fig. 5.19 shows the average cluster membership duration of vehicles for various IDS frameworks. It can be observed from the figure that the proposed framework provides the highest cluster stability amongst all the frameworks by providing high cluster membership duration to vehicles in its clusters. Its performance is comparable to that of the framework proposed in [151], since both the frameworks use novel clustering algorithms to enhance the stability of the clusters and reduce the frequency of cluster formation process. On the other hand, the average cluster membership duration of the vehicles in [129] [141] are small, even at low vehicular densities, since they do not implement any mechanism to enhance the cluster stability. As a result the vehicular clusters in these frameworks are unstable.

5.4.2 Real time vehicular network traffic

In this subsection, we analyze the effectiveness of the proposed IDS framework on the real time road network of the German city Eichstätt obtained using the OpenStreetMap [152]. The road network was imported from the OpenStreetMap (OSM) to SUMO using an application called the NETCONVERT [149].

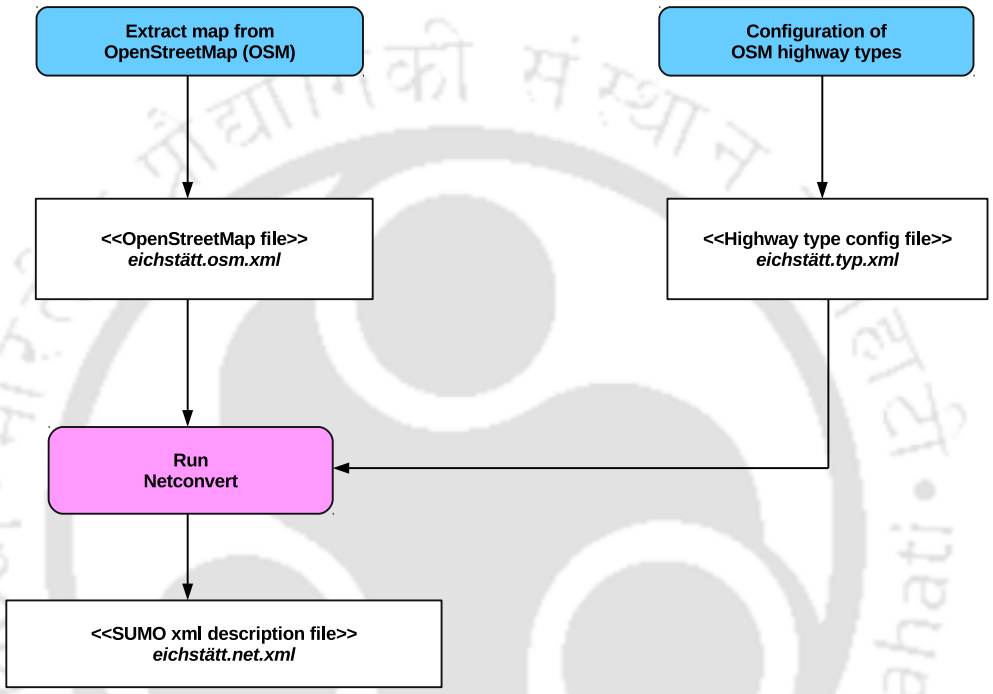

Figure 5.20: Overview of the steps involved in importing the traffic map of the German city Eichstätt from the OpenStreetMap into SUMO

Figure 5.20 shows the procedure involved in obtaining the road network of German city Eichstätt from the OSM into SUMO using NETCONVERT. The imported SUMO road network file was provided with default values of the road attributes like, speed limit, number of lanes, priority, one-way street and allowed vehicle classes depending on the highway types, using *SUMO edge type files* described in (http://sumo.dlr.de/wiki/SUMO_edge_type_file). Figures 5.21 and 5.22 show the OSM file and the corresponding SUMO network file of the Eichstätt city. Several types of the vehicles (cars, buses and emergency vehicles) with different priorities and maximum speeds were simulated in the road traffic. Different vehicles routes were set in the road traffic in SUMO. The total simulation time was 300 seconds. The results were obtained by averaging the output of 10 round of simulations.

Fig. 5.23 shows the volume of IDS traffic generated by different IDS frameworks on

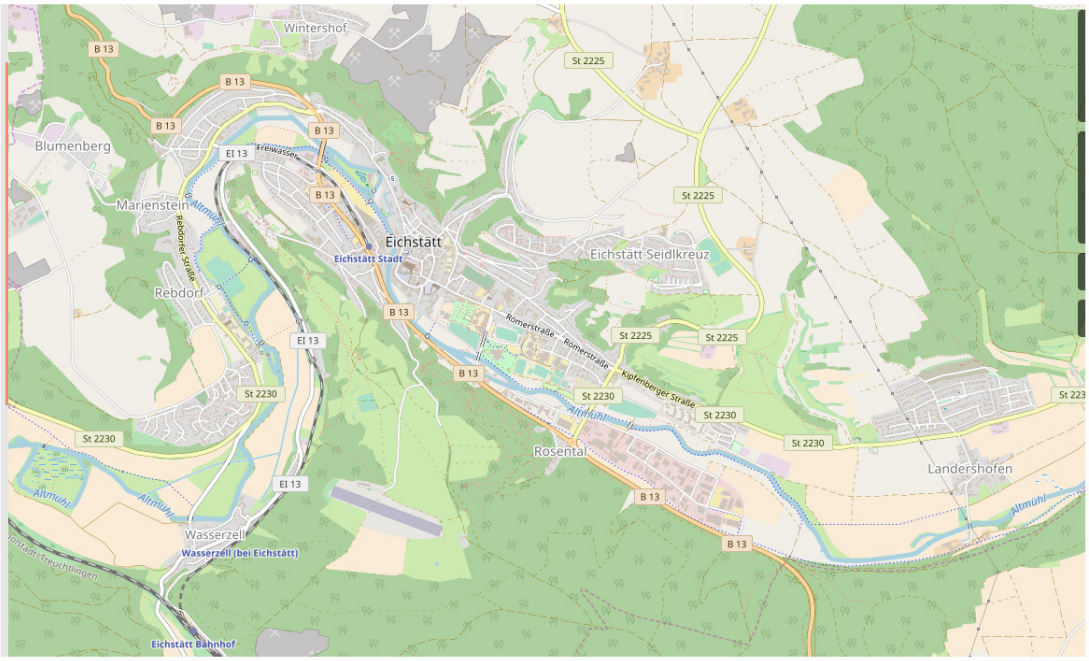

Figure 5.21: Map of German city Eichstätt obtained from OpenStreetMap

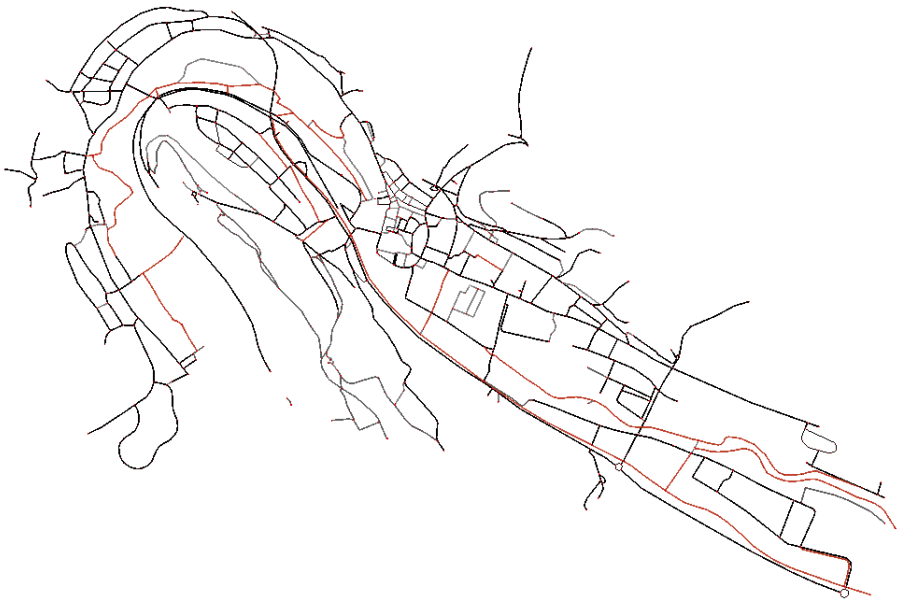

Figure 5.22: SUMO network file corresponding to OpenStreetMap map of German city Eichstätt

the Eichstätt road traffic network. It can be observed from the figure that the proposed framework introduces the least volume of the IDS traffic compared to other frameworks. Fig. 5.24 shows the *DR* and the *FAR* of various IDS frameworks on the Eichstätt road network traffic data against the black hole, worm hole, selective forwarding, DoS and sybil attacks. Again it can be observed that the proposed framework achieves the highest *DR* with least *FAR* amongst all the frameworks. These results vindicate that the proposed IDS framework significantly reduces the volume of IDS traffic in the vehicular network, while at the same time maintains a high *DR* and low *FAR* against wide range of attacks.

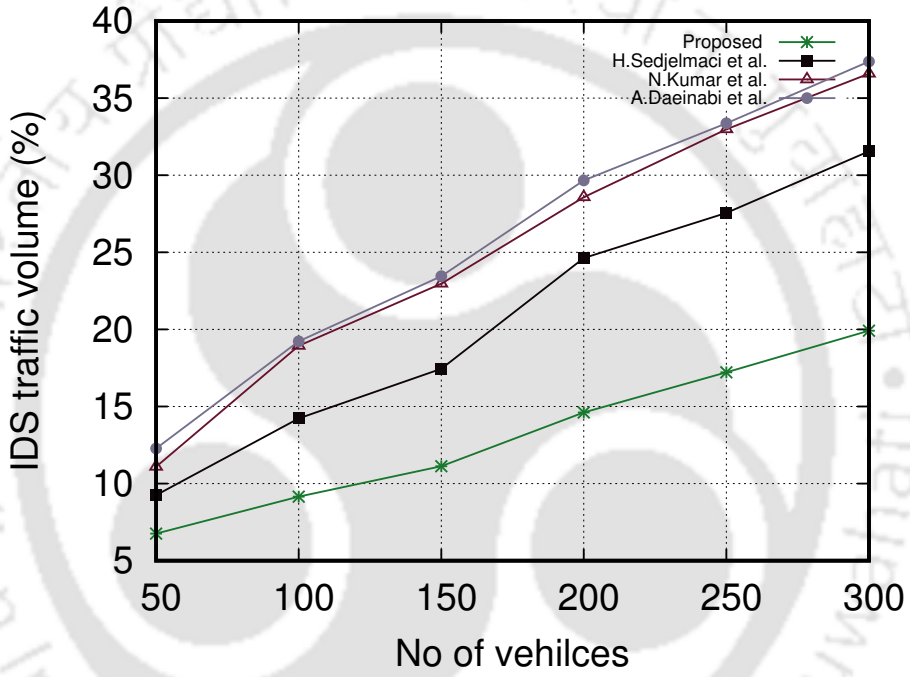

Figure 5.23: Volume of IDS traffic generated by different frameworks on the Eichstätt road traffic network

5.5 Conclusion

In this chapter, a novel clustering algorithm, a Cluster Head (CH) election algorithm and a game theory-based IDS framework for VANETs are proposed. The proposed clustering algorithm enhances the stability of the IDS framework by generating stable vehicular clusters with increased connectivity among member vehicles. CH and agent nodes election algorithms are then executed to elect the CH and a set of agent nodes for each cluster. The proposed IDS framework uses a set of agent nodes, CHs and Road Side Units (RSUs) oper-

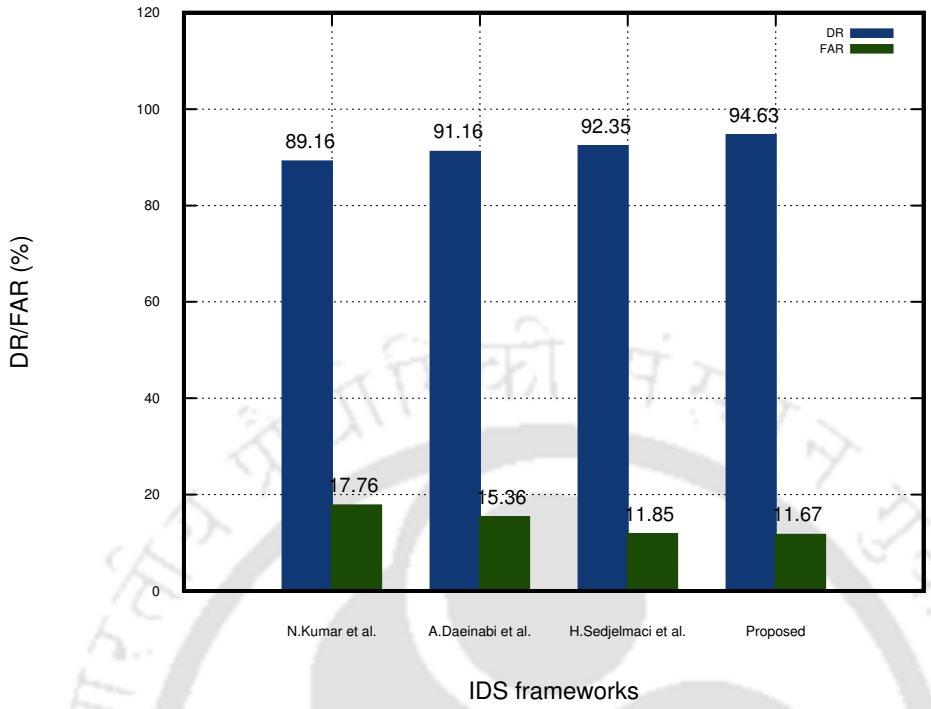

Figure 5.24: Detection rate and false alarm rate of different frameworks on the Eichstätt road traffic network

ating at three different levels of the vehicular network to carry out the intrusion detection operation in a distributed manner. The proposed IDS framework uses a set of specification rules based on the Packet Drop Rate (PDR), Packet Forwarding Rate (PFR), Receive Signal Strength Indicator (RSSI) and Duplicate Packet Rate (DPR) values of the vehicles, along with a lightweight neural network based classifier module for detecting malicious vehicles in the vehicular network. In addition, the proposed framework minimizes the volume of IDS traffic introduced into the vehicular network by modeling the interaction between the IDS and the malicious vehicle as a two player non-cooperative game, and by adopting a probabilistic IDS monitoring strategy based on the Nash Equilibrium of the game. The next chapter provides a summary of the thesis and scope for future work.

"Study how water flows in a valley stream, smoothly and freely between the rocks. Also learn from holy books and wise people. Everything - even mountains, rivers, plants and trees - should be your teacher."

Morihei Ueshiba

6

Conclusions and Future Work

The cost of cyber crime has seen an exponential growth in recent years with data theft and espionage related news being reported across wide range of domains like social networking sites, data servers, defense establishments, research and development centers, entertainment production house etc. Intrusion Detection Systems (IDSs) have been proposed in the literature to address these security threats. IDS is a hardware device or a software application that monitors the network traffic for sign of malicious activities and policy violations. When any intrusive network activities are detected, the IDS informs the administrator by raising alerts and generating log reports of the intrusions. The administrator can then take appropriate counter measures to contain the intrusions before any significant damage is inflicted to the network.

Based on their mode of operations, IDSs can broadly be classified into following three categories namely, signature based, event based and anomaly based. Signature based IDSs (also known as misuse based) correlate the header and payload information of the network data packets being monitored against a predefined set of attack signatures to identify the malicious network traffic. Event based IDSs basically act as state estimators and observe the sequence of events generated in the network to decide whether the states through which the system traverses corresponds to normal or compromised condition. Anomaly based IDSs use the normal behavior of the network traffic to build the baseline profile of the network. The real time network traffic is then correlated against the learned baseline profile to identify anomalous data traffic.

Although, all the three classes of IDSs are known to perform well against wide range of

network attacks, there are several drawbacks associated with them. Signature based IDSs are known to produce a large number of false positive alarms when deployed with default set of attack signatures, without considering the context of the underlying network. Sometimes more than 90% of the alerts being generated by them turns out to be false positives. Therefore, false alarm minimization of signature based IDSs is an important issue that needs to be addressed for enabling their wider acceptance. The number of states to be kept track of by the event based IDSs can grow exponentially large with the increase in network's size being monitored. This introduces a tremendous amount of computational overhead and limit their overall effectiveness in resource constrained networks. Event based IDSs also requires active techniques like sending out probe packets to identify the differences in sequencing of data packets under normal and attack conditions, which violate the standard operations of the protocol under consideration. Anomaly based IDSs are computation intensive and produce a significant volume of IDS traffic. This poses a significant challenge in the deployment of anomaly based IDSs in wireless networks like MANETs and VANETs, which are usually characterized by energy and resource constrained mobile nodes with limited bandwidth and finite memory storage. Therefore, anomaly based IDSs proposed for wireless networks must be tailored to meet their computational and energy constraints.

6.1 Summary of Contribution of the Thesis

In this thesis, various game theory-based IDS frameworks have been proposed with the primary objective of addressing various IDS related issues like, false alarm minimization, reduction of IDS traffic volume, minimization of energy consumption required for operating IDSs, adherence to network bandwidth constraints etc. The individual contribution of each chapter from Chapter 3 to Chapter 5 are as follows:

Contribution of Chapter 3: Signature based IDSs produce a large number of FP alarms that outnumbers the TP alarms by a ratio of almost 2:1. This puts a severe limitation on the IDS's accuracy and overwhelms the network administrator with deluge of false alerts. We proposed a novel game theory-based false alarm minimization scheme in this chapter to address this issue. The proposed scheme uses multiple vulnerability scanners to scan the network and create a vulnerability Threat profile of the network. The network's Threat profile comprises multiple vulnerability sets with each set containing one or more network vulnerabilities. Each vulnerability set is assigned a unique criticality weight based on the

severity of the vulnerabilities contained in it. The IDS alarms are initially correlated with vulnerabilities in the Threat profile to determine the potential TP alarms. Additionally, the proposed false alarm minimization scheme models the interaction between the attacker and the IDS (defender) as a two player non-cooperative game. Various attacking and monitoring strategies are examined to evaluate the Nash Equilibrium of the game and build the Sensible Vulnerability Set (SVS) of the network. The SVS consists of a subset of high criticality weight vulnerability sets from the network's Threat profile. The IDS alarms that pass the network's Threat profile correlation test are eventually correlated with vulnerabilities in the SVS to determine the final TP alarms. Experimental results on the benchmark IDEVAL dataset and the testbed dataset show that the proposed false alarm minimization scheme significantly reduces the number of false alarms generated by the signature based IDS, without causing any significant degradation in the detection rate of the IDS. The proposed scheme achieved the accuracy of 98.83% and 98.55% on the IDEVAL dataset and the testbed dataset, respectively.

Contribution of Chapter 4: MANETs are characterized by energy and resource constrained nodes. Therefore, any IDS framework proposed for MANET must take these constraints into consideration. In this chapter, we proposed a novel Bayesian game theory-based intrusion detection framework for MANETs. The proposed IDS framework models the intrusion detection process in MANET as a two player non-cooperative Bayesian game between the IDS and the node being monitored. Such Bayesian game theoretic modeling allows the IDS to adopt a probabilistic monitoring strategy based on the Bayesian Nash Equilibrium (BNE) of the game, which significantly reduces the energy consumption required for operating the IDS, without compromising the detection capabilities of the IDS. In addition, the proposed IDS framework uses a novel cluster leader node election algorithm based on VCG mechanism to ensure a uniform energy consumption across multiple MANET nodes for operating the IDS. This prevents the premature death of the nodes operating the IDS and hence avoids the network fragmentation problem. The proposed IDS framework uses a combination of lightweight and heavyweight IDS modules to achieve high detection rate and accuracy across wide range of attacks. The lightweight module uses a simple threshold based rules to detect malicious nodes, while the heavyweight module uses a powerful data mining based association rules to identify the malicious nodes. The proposed IDS framework achieved a detection rate of 91.78% with the false alarm rate of 0.5% on a simulated network implemented using the network simulator NS2.

Contribution of Chapter 5: In this chapter, a novel clustering algorithm, a CH election algorithm and a game theory-based IDS framework for VANETs are proposed. The proposed clustering algorithm ensures the stability of the IDS framework by generating stable vehicular clusters with enhanced connectivity among member vehicles. CH and agent nodes election algorithms are then executed to elect the CH and set of agent nodes for each cluster. The proposed IDS framework uses the the agent nodes, the CHs and the RSUs operating at three different levels of the vehicular network to carry out the intrusion detection operation in a distributed manner. The framework uses a set of specification rules based on the Packet Drop Rate (PDR), Packet Forwarding Rate (PFR), Receive Signal Strength Indicator (RSSI) and Duplicate Packet Rate (DPR) values of the vehicles, along with a lightweight neural network based classifier module for detecting malicious vehicles in the vehicular network. In addition, the proposed IDS framework minimizes the volume of IDS traffic introduced into the vehicular network by modeling the interaction between the IDS and the malicious vehicle as a two player non-cooperative game, and by adopting a probabilistic IDS monitoring strategy based on the Nash Equilibrium of the game. The proposed IDS framework achieved a detection rate of 99.63% with a false alarm rate of 9.13% on a simulated network implemented using the network simulator NS3 and open source traffic simulator SUMO. On the other hand, it achieved a detection rate of 94.63% with a false alarm rate of 11.67% on a real time road network of the German city Eichstätt implemented using NS3 simulator.

6.2 Scope of Future Work

The following are possible future research direction.

- Snort was used as a default signature based IDS in Chapter 3 to evaluate the proposed false alarm minimization scheme. The network's Vulnerability Threat Profile (VTP) was primarily populated based on the syntax and structures of alarms generated by Snort. The network's VTP can be made more heterogeneous by adopting alarm syntaxes from other signature based IDSs like BRO [13], EMERALD [12] etc.
- The performance of the false alarm minimization scheme proposed in Chapter 3 was primarily evaluated on wired network. Wireless networks based on IEEE 802.11 standards have more stringent set of constraints (in terms of energy and computational resources) as compared to wired networks. They are also more vulnerable to various security threats. Evaluation of the proposed false alarm minimization scheme on

these wireless networks is another possible direction for future work.

- In Chapter 4, three different type of attacks were considered to evaluate the performance of the proposed IDS framework namely, Route compromise, Traffic distortion and Black-hole attack. Evaluation of the proposed MANET IDS framework on other type of attacks like Denial of Service (DoS), Sybil attack, Worm hole attack etc., can be carried out as a future work. Moreover, various other equilibrium concepts like Pareto Equilibrium, Subgame Perfect Nash Equilibrium and Correlated Equilibrium can be explored to develop other variants of game theory-based IDS frameworks for MANETs.

- The refinement of the leader node and the IDS agent node election algorithms proposed in Chapter 4 and Chapter 5 to minimize their overall computational overhead and to reduce the volume of election related messages introduced into the network is another possible research direction.

- In Chapter 5, we used a single layered neural network to develop the anomaly detection classifier model. In future, we aim to improve the detection rate and minimize the false alarm rate of the IDS framework proposed for VANETs by analyzing the performances of various other classifier models using Support Vector Machine (SVM), Decision Tree, Logistic Regression, Multi-layered neural networks etc. Additionally, we also aim to extend and implement the proposed IDS framework to various other networks like Software Defined Network (SDN), Delay Tolerant Network (DTN) etc.

Bibliography

[1] Advanced Intrusion Detection Environment (AIDE), http://sourceforge.net/projects/aide/.

[2] Mtree, FreeBSD 5.3, http://www.freebsd.org/cgi/man.cgi?query=mtree, 2004.

[3] Simple watchdog (Swatch), https://sourceforge.net/projects/swatch/.

[4] John P. Rouillard. Refereed Papers: Real-time Log File Analysis Using the Simple Event Correlator (SEC). In *Proceedings of the 18th USENIX Conference on System Administration*, pages 133–150. USENIX Association, 2004.

[5] PortSentry, https://sourceforge.net/projects/sentrytools/.

[6] Scanlogd, https://directory.fsf.org/wiki/scanlogd.

[7] Bugtraq, https://www.securityfocus.com.

[8] CVE-Common Vulnerabilities and Exposures, https://cve.mitre.org/cgi-bin/cvename.cgi, 2013.

[9] Gordon Fyodor Lyon. *Nmap Network Scanning: The Official Nmap Project Guide to Network Discovery and Security Scanning*. 2009.

[10] Jay Beale, Renaud Deraison, Haroon Meer, Roelof Temmingh, and Charl Van Der Walt. *Nessus Network Auditing*. 2004.

[11] Martin Roesch. Snort-Lightweight Intrusion Detection for Networks. In *Proceedings of the 13th USENIX Conference on System Administration*, LISA '99, pages 229–238, 1999.

[12] Phillip A. Porras and Peter G. Neumann. EMERALD: Event Monitoring Enabling Responses to Anomalous Live Disturbances. In *20th National Information Systems Security Conference*, pages 353–365. IEEE, 1997.

[13] Vern Paxson. Bro: A System for Detecting Network Intruders in Real-time. In *Proceedings of the 7th Conference on USENIX Security Symposium - Volume 7*, SSYM'98, pages 3–3, 1998.

[14] J.E. Dickerson and J.A. Dickerson. Fuzzy network profiling for intrusion detection. In *19th International Conference of the North American Fuzzy Information Processing Society*, pages 301–306, 2000.

[15] Levent Ertoz, Eric Eilertson, Aleksandar Lazarevic, Pang ning Tan, Vipin Kumar, Jaideep Srivastava, and Paul Dokas. MINDS – Minnesota Intrusion Detection System. In *Next Generation Data Mining Boston*. MIT Press, 2004.

[16] Mohammad Sazzadul Hoque, Md. Abdul Mukit, and Md. Abu Naser Bikas. An Implementation of Intrusion Detection System Using Genetic Algorithm. *International Journal of Network Security & Its Applications*, 4(2), 2012.

[17] Paul Barford, Jeffery Kline, David Plonka, and Amos Ron. A signal analysis of network traffic anomalies. In *Proceedings of the 2Nd ACM SIGCOMM Workshop on Internet Measurment*, pages 71–82. ACM, 2002.

[18] Seong Soo Kim, A. L. Narasimha Reddy, and Marina Vannucci. *Detecting Traffic Anomalies through Aggregate Analysis of Packet Header Data*, pages 1047–1059. Springer Berlin Heidelberg, Berlin, Heidelberg, 2004.

[19] Eddie Kohler, Jinyang Li, Vern Paxson, and Scott Shenker. Observed Structure of Addresses in IP Traffic. In *Proceedings of the 2nd ACM SIGCOMM Workshop on Internet Measurment*, pages 253–266. ACM, 2002.

[20] Alfonso Valdes and Keith Skinner. Adaptive, Model-Based Monitoring for Cyber Attack Detection. In *Proceedings of the Third International Workshop on Recent Advances in Intrusion Detection*, volume 1907, pages 80–92. Springer-Verlag, 2000.

[21] Elidon Beqiri. *Neural Networks for Intrusion Detection Systems*, pages 156–165. Springer Berlin Heidelberg, Berlin, Heidelberg, 2009.

[22] Latifur Khan, Mamoun Awad, and Bhavani Thuraisingham. A new intrusion detection system using support vector machines and hierarchical clustering. *The VLDB Journal*, 16(4):507–521, October 2007.

[23] Gert DeLaet and Gert Schauwers. *Network Security Fundamentals*. Cisco Press, 2004.

[24] Roger B. Myerson. *Game Theory: Analysis of Conflict*. Harvard University Press, September 1997.

[25] Jianfeng Cai and U. Pooch. Allocate fair payoff for cooperation in wireless ad hoc networks using Shapley Value. In *Proceedings of the 18th International Parallel and Distributed Processing Symposium*, pages 219–, April 2004.

[26] A. Urpi, M. Bonuccelli, and S. Giodano. Modelling Cooperation in Mobile Ad Hoc Networks: A Formal Description of Selfishness. In *Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks*, pages 3–5, 2003.

[27] Yu Liu, Cristina Comaniciu, and Hong Man. A Bayesian Game Approach for Intrusion Detection in Wireless Ad Hoc Networks. In *Proceeding of the 2006 Workshop on Game Theory for Communications and Networks*. ACM, 2006.

[28] Yi-Ming Chen, Dachrahn Wu, and Cheng-Kuang Wu. A Game Theoretic Framework for Multi-agent Deployment in Intrusion Detection Systems. In *Security Informatics*, volume 9 of *Annals of Information Systems*, pages 117–133. Springer, 2010.

[29] M. Kodialam and T.V. Lakshman. Detecting network intrusions via sampling: A game theoretic approach. In *Twenty-Second Annual Joint Conference of the IEEE Computer and Communications*, volume 3, pages 1880–1889, March 2003.

[30] A. Agah, S.K. Das, K. Basu, and M. Asadi. Intrusion detection in sensor networks: a non-cooperative game approach. In *Network Computing and Applications, 2004. (NCA 2004). Proceedings. Third IEEE International Symposium on*, pages 343–346, Aug 2004.

[31] Tansu Alpcan and T. Basar. A game theoretic approach to decision and analysis in network intrusion detection. In *Proceedings of 42nd IEEE Conference on Decision and Control*, pages 2595–2600, Dec 2003.

[32] A. Agah and S. K. Das. Preventing DoS Attacks in Wireless Sensor Networks: A Repeated Game Theory Approach. *International Journal of Network Security*, 5(2):145–153, 2007.

[33] H. Sedjelmaci, T. Bouali, and S. M. Senouci. Detection and prevention from misbehaving intruders in vehicular networks. In *IEEE Global Communications Conference*, pages 39–44, Dec 2014.

[34] Doaa Al-Terri, Hadi Otrok, Hassan Barada, Mahmoud Al-Qutayri, and Yousof Al Hammadi. Cooperative based tit-for-tat strategies to retaliate against greedy behavior in vanets. *Computer Communications*, 104:108–118, May 2017.

[35] Hanaa Marshoud, Hadi Otrok, and Hassan Barada. Macrocell-femtocells resource allocation with hybrid access motivational model. *Physical Communication*, 11:3–14, 2014.

[36] Hadi Otrok; Clément Rousseau Stéphane Boyer, Jean-Marc Robert. An adaptive tit-for-tat strategy for IEEE 802.11 CSMA/CA protocol. *International Journal of Security and Networks*, 7(2):95–106, 2012.

[37] Omar Abdel Wahab, Hadi Otrok, and Azzam Mourad. A dempster–shafer based tit-for-tat strategy to regulate the cooperation in vanet using qos-olsr protocol. *Wireless Personal Communications*, 75(3):1635–1667, Apr 2014.

[38] Andreu Mas-Colell, Michael Whinston, and Jerry Green. *Microeconomic Theory*. Oxford University Press, 1995.

[39] Christos G. Cassandras and Stephane Lafortune. *Introduction to Discrete Event Systems*. Springer Publishing Company, Incorporated, 2010.

[40] R. Sekar, A. Gupta, J. Frullo, T. Shanbhag, A. Tiwari, H. Yang, and S. Zhou. Specification-based Anomaly Detection: A New Approach for Detecting Network Intrusions. In *Proceedings of the 9th ACM Conference on Computer and Communications Security*, pages 265–274, 2002.

[41] S. Whittaker, M. Zulkernine, and K. Rudie. Towards Incorporating Discrete-Event Systems in Secure Software Development. In *Third International Conference on Availability, Reliability and Security*, pages 1188–1195, March 2008.

[42] Mohammed J. Zaki. SPADE: An Efficient Algorithm for Mining Frequent Sequences. *Machine Learning*, 42(1-2):31–60, 2001.

[43] C.M. Kozierok. TCP/IP guide. No Starch Press, 2005.

[44] Mohamed G. Gouda and Chin-Tser Huang. A Secure Address Resolution Protocol. *Computer Networks*, 41(1):57–71, January 2003.

[45] Wesam Lootah, William Enck, and Patrick McDaniel. TARP: Ticket-based Address Resolution Protocol. *Computer Networks*, 51(15):4322–4337, October 2007.

[46] Prem Uppuluri and R. Sekar. Experiences with Specification-Based Intrusion Detection. In *Recent Advances in Intrusion Detection*, volume 2212, pages 172–189. Springer, 2001.

[47] S. E. Benaicha, L. Saoudi, S. E. B. Guermeche, and O. Lounis. Intrusion detection system using genetic algorithm. In *Science and Information Conference*, pages 564–568, Aug 2014.

[48] Sridhar Ramaswamy, Rajeev Rastogi, and Kyuseok Shim. Efficient Algorithms for Mining Outliers from Large Data Sets. *ACM SIGMOD Record*, 29(2):427–438, May 2000.

[49] Anukool Lakhina, Mark Crovella, and Christophe Diot. Mining Anomalies Using Traffic Feature Distributions. *ACM SIGCOMM Computer Communication Review*, 35(4):217–228, August 2005.

[50] B. Subba, S. Biswas, and S. Karmakar. A Neural Network based system for Intrusion Detection and attack classification. In *Twenty Second National Conference on Communication (NCC)*, pages 1–6, March 2016.

[51] Hongmei Deng, Qing-An Zeng, and D. P. Agrawal. Svm-based intrusion detection system for wireless ad hoc networks. In *2003 IEEE 58th Vehicular Technology Conference. VTC 2003-Fall (IEEE Cat. No.03CH37484)*, volume 3, pages 2147–2151 Vol.3, Oct 2003.

[52] B. Subba, S. Biswas, and S. Karmakar. Intrusion detection systems using linear discriminant analysis and logistic regression. In *2015 Annual IEEE India Conference (INDICON)*, pages 1–6, Dec 2015.

[53] M. Ali Aydin, A. Halim Zaim, and K. Gokhaon Ceylan. A hybrid intrusion detection system design for computer network security. *Computers & Electrical Engineering*, 35(3):517 – 526, 2009.

[54] Richard Lippmann, Joshua W. Haines, David J. Fried, Jonathan Korba, and Kumar Das. The 1999 DARPA Off-line Intrusion Detection Evaluation. *Computer Networks*, 34(4):579–595, October 2000.

[55] B. Subba, S. Biswas, and S. Karmakar. Enhancing effectiveness of intrusion detection systems: A hybrid approach. In *2016 IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS)*, pages 1–6, Nov 2016.

[56] Robert Mitchell and Ing-Ray Chen. A Survey of Intrusion Detection Techniques for Cyber-physical Systems. *ACM Computur Survey*, 46(4), March 2014.

[57] Neminath Hubballi and Vinoth Suryanarayanan. False alarm minimization techniques in signature-based intrusion detection systems: A survey. *Computer Communications*, 49:1 – 17, 2014.

[58] Tadeusz Pietraszek and Axel Tanner. Data Mining and Machine learning-Towards Reducing False Positives in Intrusion Detection. *Information Security Technical Report*, 10(3):169–183, Jan 2005.

[59] J.J. Treinen and R. Thurimella. Finding the Needle: Suppression of False Alarms in Large Intrusion Detection Data Sets . In *International Conference on Computational Science and Engineering*, pages 237–244, Aug 2009.

[60] Frederic Massicotte, Mathieu Couture, Lionel Briand, and Yvan Labiche. Model-Driven, Network-Context Sensitive Intrusion Detection. In *Model Driven Engineering Languages and Systems*, volume 4735, pages 61–75. 2007.

[61] Robin Sommer and Vern Paxson. Enhancing Byte-level Network Intrusion Detection Signatures with Context. In *Proceedings of the 10th ACM Conference on Computer and Communications Security*, pages 262–271, 2003.

[62] Tobias Chyssler, Stefan Burschka, Michael Semling, Tomas Lingvall, and Kalle Burbeck. Alarm Reduction and Correlation in Intrusion Detection Systems. In *DIMVA*, volume 46, pages 9–24, 2004.

[63] Damiano Bolzoni, Bruno Crispo, and Sandro Etalle. ATLANTIDES : An Architecture for Alert Verification in Network Intrusion Detection Systems. In *Proceedings of the 21st Conference on Large Installation System Administration Conference*, pages 1–12, 2007.

[64] Frederic Massicotte, Lionel C. Briand, Mathieu Couture, and Yvan Labiche. Context-Based Intrusion Detection Using Snort, Nessus and Bugtraq Databases. In *Proceedings of the 3rd International Conference on Privacy, Security and Trust*, pages 1–12, 2005.

[65] K. Alsubhi, E. Al-Shaer, and R. Boutaba. Alert prioritization in Intrusion Detection Systems. In *NOMS 2008 - 2008 IEEE Network Operations and Management Symposium*, pages 33–40. IEEE, April 2008.

[66] J. Yu, Y. V. R. Reddy, Sentil Selliah, Srinivas Kankanahalli, Sumitra Reddy, and Vijayanand Bharadwaj. TRINETR: an intrusion detection alert management systems. In *13th IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises*, pages 235–240. IEEE, June 2004.

[67] Alfonso Valdes and Keith Skinner. Probabilistic Alert Correlation. In *Proceedings of the 4th International Symposium on Recent Advances in Intrusion Detection*, pages 54–68. Springer-Verlag, 2001.

[68] Vivek Ramachandran and Sukumar Nandi. Detecting ARP Spoofing: An Active Technique. In *Proceedings of the First International Conference on Information Systems Security*, pages 239–250, Berlin, Heidelberg, 2005. Springer-Verlag.

[69] Neminath Hubballi, S. Roopa, Ritesh Ratti, F. A. Barbhuiya, Santosh Biswas, Arijit Sur, Sukumar Nandi, and Vivek Ramachandran. *An Active Intrusion Detection System for LAN Specific Attacks*, pages 129–142. Springer Berlin Heidelberg, Berlin, Heidelberg, 2010.

[70] A. Patcha and Jung-Min Park. A game theoretic approach to modeling intrusion detection in mobile ad hoc networks. In *Information Assurance Workshop, 2004. Proceedings from the Fifth Annual IEEE SMC*, pages 280–284, June 2004.

[71] S. Roy, C. Ellis, S. Shiva, D. Dasgupta, V. Shandilya, and Qishi Wu. A Survey of Game Theory as Applied to Network Security. In *International Conference on System Sciences (HICSS)*, pages 1–10, Jan 2010.

[72] Mohammad Hossein Manshaei, Quanyan Zhu, Tansu Alpcan, Tamer Bacşar, and Jean-Pierre Hubaux. Game Theory Meets Network Security and Privacy. *ACM Computing Surveys*, 45(3):25:1–25:39, July 2013.

[73] Constantinos Daskalakis, Paul W. Goldberg, and Christos H. Papadimitriou. The Complexity of Computing a Nash Equilibrium. *SIAM Journal on Computing*, 39(1):195–259, May 2009.

[74] Christos H. Papadimitriou. On the Complexity of the Parity Argument and Other Inefficient Proofs of Existence. *Journal of Computer and System Sciences*, 48(3):498–532, June 1994.

[75] Noam Nisan, Tim Roughgarden, Eva Tardos, and Vijay V. Vazirani. *Algorithmic Game Theory*. Cambridge University Press, 2007.

[76] John Nash. The bargaining problem. *Econometrica*, 18(2):155–162, 1950.

[77] John C. Harsanyi. Games with Incomplete Information Played by "Bayesian" Players, I-III. *Management Science*, 50:1804–1817, December 2004.

[78] Dimitris E. Charilas and Athanasios D. Panagopoulos. A survey on game theory applications in wireless networks. *Computer Networks*, 54(18):3421–3430, December 2010.

[79] Allen B. MacKenzie and Luiz A. DaSilva. *Game Theory for Wireless Engineers (Synthesis Lectures on Communications)*. Morgan & Claypool Publishers, 2006.

[80] Peng Liu. Incentive-based modeling and inference of attacker intent, objectives, and strategies. In *Proceeding of the 10th ACM Computer and Communications Security Conference*, pages 179–189, 2003.

[81] M. Estiri and A. Khademzadeh. A game-theoretical model for intrusion detection in wireless sensor networks. In *23rd Canadian Conference on Electrical and Computer Engineering*, pages 1–5. IEEE, 2010.

[82] Y. Qiu, Z. Chen, and L. Xu. Active Defense Model of Wireless Sensor Networks Based on Evolutionary Game Theory. In *6th International Conference on Wireless Communications Networking and Mobile Computing (WiCOM)*, pages 1–4. IEEE, 2010.

[83] N. Marchang, R. Datta, and S. K. Das. A Novel Approach for Efficient Usage of Intrusion Detection System in Mobile Ad Hoc Networks. *IEEE Transactions on Vehicular Technology*, 66(2):1684–1695, Feb 2017.

[84] Joseph S. Sherif, Rod Ayers, and Tommy G. Dearmond. Intrusion detection: The art and the practice. *Information Management & Compututer Security*, 11(4):175–186, 2003.

[85] M. Uddin, A.A. Rahman, J. Memon, and N. Uddin. Algorithm to detect intrusions using multi layer signature based model. *Journal of Applied Sciences Research*, 8(8):4457–4466, 2012.

[86] Tjhai, Gina C. and Papadaki, Maria and Furnell, Steven M. and Clarke, Nathan L. The Problem of False Alarms: Evaluation with Snort and DARPA 1999 Dataset. In *Proceedings of 5th International Conference on Trust, Privacy and Security in Digital Business, TrustBus 2008 Turin, Italy*, pages 139–150", 2008.

[87] Saeed Salah, Gabriel Maciá-Fernández, and Jesús E. Díaz-Verdejo. Survey a model-based survey of alert correlation techniques. *Compututer Networks*, 57(5):1289–1317, April 2013.

[88] Safaa O. Al-Mamory and Hong Li Zhang. A Survey on IDS Alerts Processing Techniques. In *Proceedings of the 6th WSEAS International Conference on Information Security and Privacy*, pages 69–78, 2007.

[89] Humphrey Waita Njogu, Luo Jiawei, and Jane Nduta Kiere. Network Specific Vulnerability Based Alert Reduction Approach. *Security and Communication Networks*, 6(1):15–27, January 2013.

[90] Pei-Te Chen and Chi-Sung Laih. IDSIC: An intrusion detection system with identification capability. *International Journal of Information Security*, 7(3):185–197, 2008.

[91] Benjamin Morin, Ludovic Mé, Hervé Debar, and Mireille Ducassé. M2D2: A Formal Data Model for IDS Alert Correlation. In *Recent Advances in Intrusion Detection*, volume 2516, pages 115–137. 2002.

[92] Weizhi Meng, Wenjuan Li, and Lam-For Kwok. EFM: Enhancing the performance of signature-based network intrusion detection systems using enhanced filter mechanism. *Computers & Security*, 43:189 – 204, 2014.

[93] Lin Chen and J. Leneutre. A Game Theoretical Framework on Intrusion Detection in Heterogeneous Networks. *Information Forensics and Security, IEEE Transactions on*, 4(2):165–178, June 2009.

[94] JOHN McHUGH. Testing Intrusion Detection Systems: A Critique of the 1998 and 1999 DARPA Intrusion Detection System Evaluations As Performed by Lincoln Laboratory. *ACM Transactions on Information and System Security*, 3(4):262–294, November 2000.

[95] John McHugh. The 1998 lincoln laboratory ids evaluation. In *Recent Advances in Intrusion Detection*, pages 145–161, Berlin, Heidelberg, 2000. Springer Berlin Heidelberg.

[96] David Kennedy, Jim O'Gorman, Devon Kearns, and Mati Aharoni. *Metasploit: The Penetration Tester's Guide*. 2011.

[97] Jingmin Zhou, A. J. Carlson, and M. Bishop. Verify results of network intrusion alerts using lightweight protocol analysis. In *21st Annual Computer Security Applications Conference*, pages 10 pp.–126. IEEE, Dec 2005.

[98] Neminath Hubballi, Santosh Biswas, and Sukumar Nandi. Towards reducing false alarms in network intrusion detection systems with data summarization technique. *Security and Communication Networks*, 6(3):275–285, 2013.

[99] Moon Sun Shin, Eun Hee Kim, and Keun Ho Ryu. False alarm classification model for network-based intrusion detection system. In *Proceedings of 5th International Conference on Intelligent Data Engineering and Automated Learning*, pages 259–265. Springer Berlin Heidelberg, 2004.

[100] A. Mishra, K. Nadkarni, and A. Patcha. Intrusion Detection in Wireless Ad-hoc Networks. *IEEE Wireless Communications*, 11(1):48–60, Feb 2004.

[101] Lei Zhang, Qianhong Wu, Bo Qin, Josep Domingo-Ferrer, and Bao Liu. Practical Secure and Privacy-preserving Scheme for Value-added Applications in VANETs. *Computer Communications*, 71(C):50–60, November 2015.

[102] M. La Polla, F. Martinelli, and D. Sgandurra. A survey on security for mobile devices. *IEEE Communications Surveys Tutorials*, 15(1):446–471, First 2013.

[103] Farooq Anjum and Petros Mouchtaris. *Intrusion Detection Systems*. John Wiley & Sons, Inc., 2006.

[104] P. Brutch and C. Ko. Challenges in Intrusion detection for wireless ad-hoc networks. In *Proceedings of Symposium on Applications and the Internet Workshops*, pages 368–373, Jan 2003.

[105] Yih-Chun Hu, Adrian Perrig, and DavidB. Johnson. Ariadne: A Secure On-Demand Routing Protocol for Ad-Hoc Networks. *Wireless Networks*, 11(1-2):21–38, 2005.

[106] Shengrong Bu, F.R. Yu, X.P. Liu, P. Mason, and H. Tang. Distributed Combined Authentication and Intrusion Detection With Data Fusion in High-Security Mobile Ad Hoc Networks. *IEEE Transactions on Vehicular Technology*, 60(3):1025–1036, March 2011.

[107] Z.M. Fadlullah, H. Nishiyama, N. Kato, and M.M. Fouda. Intrusion detection system (IDS) for combating attacks against cognitive radio networks. *IEEE Network*, 27(3):51–56, May 2013.

[108] Elhadi M. Shakshuki, Nan Kang, and Tarek R. Sheltami. EAACK - A Secure Intrusion-Detection System for MANETs. *IEEE Transactions on Industrial Electronics*, 60(3):1089–1098, 2013.

[109] M. Mohanapriya and Ilango Krishnamurthi. Modified DSR protocol for detection and removal of selective black hole attack in MANET. *Computers & Electrical Engineering*, 40(2):530 – 538, 2014.

[110] Sergio Marti, T. J. Giuli, Kevin Lai, and Mary Baker. Mitigating Routing Misbehavior in Mobile Ad Hoc Networks. In *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking*, pages 255–265. ACM, 2000.

[111] Kejun Liu, Jing Deng, Pramod K. Varshney, and Kashyap Balakrishnan. An Acknowledgment-Based Approach for the Detection of Routing Misbehavior in MANETs. *IEEE Transactions on Mobile Computing*, 6(5):536–550, May 2007.

[112] Sonja Buchegger and Jean-Yves Le Boudec. Performance Analysis of the CONFIDANT Protocol. In *Proceedings of the 3rd ACM international symposium on Mobile Ad Hoc Networking & Computing*, pages 226–236. ACM, 2002.

[113] P. Krishna, N. H. Vaidya, M. Chatterjee, and D. K. Pradhan. A Cluster-based Approach for Routing in Dynamic Networks. *SIGCOMM Comput. Commun. Rev.*, 27(2):49–64, April 1997.

[114] Hu. Yih-Chun and A. Perrig. A survey of secure wireless ad hoc routing. *IEEE Security & Privacy*, 2(3):28–39, May 2004.

[115] Yih chun Hu, Adrian Perrig, and David B. Johnson. Wormhole attacks in wireless networks. *IEEE Journal on Selected Areas in Communications*, 24:370–380, 2006.

[116] Yian Huang and Wenke Lee. A Cooperative Intrusion Detection System for Ad Hoc Networks. In *Proceedings of the 1st ACM Workshop on Security of Ad Hoc and Sensor Networks*, pages 135–147, 2003.

[117] Rakesh Agrawal and Ramakrishnan Srikant. Fast algorithms for mining association rules in large databases. In *Proceedings of the 20th International Conference on Very Large Data Bases*, pages 487–499. Morgan Kaufmann Publishers Inc., 1994.

[118] Ashoka Savasere, Edward Omiecinski, and Shamkant B. Navathe. An Efficient Algorithm for Mining Association Rules in Large Databases. In *Proceedings of the 21st International Conference on Very Large Data Bases*, pages 432–444. Morgan Kaufmann Publishers Inc., 1995.

[119] Teerawat Issariyakul and Ekram Hossain. *Introduction to Network Simulator NS2*. Springer Publishing Company, Incorporated, 1 edition, 2008.

[120] M. Kaliappan and B. Paramasivan. Enhancing secure routing in Mobile Ad Hoc Networks using a Dynamic Bayesian Signalling Game model. *Computers & Electrical Engineering*, 41:301 – 313, 2015.

[121] Rakesh Shrestha, Kyong-Heon Han, Dong-You Choi, and Seung Jo Han. A Novel Cross Layer Intrusion Detection System in MANET. In *24th IEEE International Conference on Advanced Information Networking and Applications (AINA)*, pages 647–654, 2010.

[122] Chin-Yang Tseng, Poornima Balasubramanyam, Calvin Ko, Rattapon Limprasittiporn, Jeff Rowe, and Karl Levitt. A Specification-based Intrusion Detection System for AODV. In *Proceedings of the 1st ACM Workshop on Security of Ad Hoc and Sensor Networks*, pages 125–134, 2003.

[123] Y. Toor, P. Muhlethaler, and A. Laouiti. Vehicle Ad Hoc Networks: Applications and Related Technical Issues. *IEEE Communications Surveys & Tutorials*, 10(3):74–88, July 2008.

[124] D. Jiang and L. Delgrossi. Ieee 802.11p: Towards an international standard for wireless access in vehicular environments. In *VTC Spring 2008 - IEEE Vehicular Technology Conference*, pages 2036–2040, May 2008.

[125] Mohamed Nidhal Mejri, Jalel Ben-Othman, and Mohamed Hamdi. Survey on VANET security challenges and possible cryptographic solutions. *Vehicular Communications*, 1(2):53 – 66, 2014.

[126] M. Raya, P. Papadimitratos, I. Aad, D. Jungels, and J. P. Hubaux. Eviction of Misbehaving and Faulty Nodes in Vehicular Networks. *IEEE Journal on Selected Areas in Communications*, 25(8):1557–1568, October 2007.

[127] Saira Gillani, Farrukh Shahzad, Amir Qayyum, and Rashid Mehmood. *A Survey on Security in Vehicular Ad Hoc Networks*, pages 59–74. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013.

[128] Panagiotis Papadimitratos, Virgil Gligor, and Jean-Pierre Hubaux. Securing Vehicular Communications - Assumptions,Requirements, and Principles. In *Workshop on Embedded Security in Cars (ESCAR)*, pages 5–14, 2006.

[129] Hichem Sedjelmaci and Sidi Mohammed Senouci. An accurate and efficient collaborative intrusion detection framework to secure vehicular networks. *Computers & Electrical Engineering*, 43:33 – 47, 2015.

[130] Shamsul Huda, Jemal Abawajy, Mamoun Alazab, Mali Abdollalihian, Rafiqul Islam, and John Yearwood. Hybrids of support vector machine wrapper and filter based framework for malware detection. *Future Generation Computer Systems*, 55:376 – 390, 2016.

[131] Farrukh Aslam Khan, Muhammad Imran, Haider Abbas, and Muhammad Hanif Durad. A detection and prevention system against collaborative attacks in mobile ad hoc networks. *Future Generation Computer Systems*, 68:416 – 427, 2017.

[132] Qi Guo, Xiaohong Li, Guangquan Xu, and Zhiyong Feng. MP-MID: Multi-Protocol Oriented Middleware-level Intrusion Detection method for wireless sensor networks. *Future Generation Computer Systems*, 70:42 – 47, 2017.

[133] Yulai Xie, Dan Feng, Zhipeng Tan, and Junzhe Zhou. Unifying intrusion detection and forensic analysis via provenance awareness. *Future Generation Computer Systems*, 61:26 – 36, 2016.

[134] José María de Fuentes, Lorena González-Manzano, Ana Isabel González-Tablas, and Jorge Blasco. Security Models in Vehicular Ad-hoc Networks: A Survey. *IETE Technical Review*, 31(1):47–64, 2014.

[135] Etienne S. Coronado and Soumaya Cherkaoui. An AAA Study for Service Provisioning in Vehicul Networks. In *Proceedings of 32nd Annual IEEE Conference on Local*

Computer Networks (LCN 2007), 15-18 October 2007, Clontarf Castle, Dublin, Ireland,, pages 669–676, 2007.

[136] Hasnaa Moustafa, Gilles Bourdon, and Yvon Gourhant. AAA in Vehicular Communication on Highways with Ad Hoc Networking Support: A Proposed Architecture. In *Proceedings of the 2nd ACM International Workshop on Vehicular Ad Hoc Networks*, pages 79–80, New York, NY, USA, 2005. ACM.

[137] Junbeom Hur, Chanil Park, and Hyunsoo Yoon. An Efficient Pre-authentication Scheme for IEEE 802.11-based Vehicular Networks. In *Proceedings of the Security 2nd International Conference on Advances in Information and Computer Security*, pages 121–136, Berlin, Heidelberg, 2007. Springer-Verlag.

[138] Haojin Zhu, Rongxing Lu, Xuemin Shen, and Xiaodong Lin. Security in Service-oriented Vehicular Networks. *Wireless Communications*, 16(4):16–22, August 2009.

[139] Seyed Mohammad Safi, Ali Movaghar, and Misagh Mohammadizadeh. A Novel Approach for Avoiding Wormhole Attacks in VANET. In *Proceedings of the 2009 Second International Workshop on Computer Science and Engineering - Volume 02*, pages 160–165. IEEE Computer Society, 2009.

[140] Akbani Rehan and Korkmaz Turgay. HEAP: A packet authentication scheme for mobile ad hoc networks. *Ad Hoc Networks*, 6(7):1134 – 1150, 2008.

[141] Neeraj Kumar and Naveen Chilamkurti. Collaborative trust aware intelligent intrusion detection in VANETs. *Computers & Electrical Engineering*, 40(6):1981 – 1996, 2014.

[142] K. Zaidi, M. B. Milojevic, V. Rakocevic, A. Nallanathan, and M. Rajarajan. Host-Based Intrusion Detection for VANETs: A Statistical Approach to Rogue Node Detection. *IEEE Transactions on Vehicular Technology*, 65(8):6703–6714, Aug 2016.

[143] A. Tomandl, K. P. Fuchs, and H. Federrath. REST-Net: A dynamic rule-based IDS for VANETs. In *7th IFIP Wireless and Mobile Networking Conference (WMNC)*, pages 1–8, May 2014.

[144] Noman Mohammed, Hadi Otrok, Lingyu Wang, Mourad Debbabi, and Prabir Bhattacharya. Mechanism design-based secure leader election model for MANET. *IEEE Transactions on Dependable and Secure Computing*, 8(1):89–103, January 2011.

[145] A. Ahizoune and A. Hafid. A new stability based clustering algorithm (SBCA) for VANETs. In *37th Annual IEEE Conference on Local Computer Networks - Workshops*, pages 843–847, Oct 2012.

[146] S. Kuklinski and G. Wolny. Density based clustering algorithm for VANETs. In *2009 5th International Conference on Testbeds and Research Infrastructures for the Development of Networks Communities and Workshops*, pages 1–6, April 2009.

[147] Basant Subba, Santosh Biswas, and Sushanta Karmakar. Intrusion detection in mobile ad-hoc networks: Bayesian game formulation. *Engineering Science and Technology, an International Journal*, 19(2):782 – 799, 2016.

[148] George F. Riley and Thomas R. Henderson. The NS-3 Network Simulator Modeling and Tools for Network Simulation. In *Modeling and Tools for Network Simulation*, pages 15–34. 2010.

[149] Krajzewicz Daniel, Erdmann Jakob, Behrisch Michael, and Bieker Laura. Recent Development and Applications of SUMO- Simulation of Urban Mobility. *International Journal On Advances in Systems and Measurements*, 5(3&4):128–138, December 2012.

[150] Roberto A. Uzcátegui and Guillermo Acosta-Marum. WAVE: A Tutorial. *IEEE Communications Magazine*, 47(5):126–133, May 2009.

[151] Ameneh Daeinabi, Akbar Ghaffar Pour Rahbar, and Ahmad Khademzadeh. VWCA: An efficient clustering algorithm in vehicular ad hoc networks. *Journal of Network and Computer Applications*, 34(1):207 – 222, 2011.

[152] Mordechai (Muki) Haklay and Patrick Weber. OpenStreetMap: User-Generated Street Maps. *IEEE Pervasive Computing*, 7(4):12–18, October 2008.

Vitae

Basant Subba was born in the Village - Aho-Yangtam, District- East Sikkim, Sikkim, India on 13^{th} August, 1986. He completed his schooling from Tashi Namgyal Senior Secondary School, Gangtok, Sikkim in the year 2005. He completed his Bachelor of Engineering (B.E.) degree from the Department of Computer Science and Engineering, JSS Academy of Technical Education, Bangalore, India in the year 2009. He completed his Master of Technology in the stream of Information Technology, Department of Computer Science and Engineering, National Institute Technology Durgapur, India in the year 2012. Following that, he joined the Ph.D programme at the Department of Computer Science and Engineering, Indian Institute of Technology Guwahati (IITG) in July 2012. Dr. Santosh Biswas and Dr. Sushanta Karmakar were his PhD Thesis supervisor at IITG. He has received GATE Scholarship for his Masters programme and MHRD Govt of India Fellowship for pursuing his PhD programme. His research interests include designing game theory and machine learning based intrusion detection frameworks for wired networks, Mobile Ad-hoc Networks (MANETs), Vehicular Ad-hoc Networks (VANETs) and Wireless Sensor Networks (WSNs)

Contact Information

Email	:	s.basant@iitg.ernet.in, basantsubba@gmail.com
Web	:	https://www.researchgate.net/profile/Basant_Subba2
Permanent Address	:	Basant Subba, S/o Mr. Suk Bahadur Subba, Gangtok, East Sikkim, Pin Code-737101. Sikkim, India