Detection of Denial of Service Attack in Wireless Network using Dominance based Rough Set

N. Syed Siraj Ahmed
School of Computing science and Engineering
VIT University
Vellore, India 632014

D. P. Acharjya
School of Computing science and Engineering
VIT University
Vellore, India 632014

***Abstract*—Denial-of-service (DoS) attack is aim to block the services of victim system either temporarily or permanently by sending huge amount of garbage traffic data in various types of protocols such as transmission control protocol, user datagram protocol, internet connecting message protocol, and hypertext transfer protocol using single or multiple attacker nodes. Maintenance of uninterrupted service system is technically difficult as well as economically costly. With the invention of new vulnerabilities to system new techniques for determining these vulnerabilities have been implemented. In general, probabilistic packet marking (PPM) and deterministic packet marking (DPM) is used to identify DoS attacks. Later, intelligent decision prototype was proposed. The main advantage is that it can be used with both PPM and DPM. But it is observed that, data available in the wireless network information system contains uncertainties. Therefore, an effort has been made to detect DoS attack using dominance based rough set. The accuracy of the proposed model obtained over the KDD cup dataset is 99.76 and it is higher than the accuracy achieved by resilient back propagation (RBP) model.**

***Keywords*—Denial of service; Rough set; Lower and upper approximation; Dominance relation; Data analysis**

I. INTRODUCTION

Denial-of-service attack is one of the most threatening security issues in wireless networks. Over the past few years, it is observed that while surfing websites on the internet a computer in the network host may have been the target of denial-of-service attacks using various protocols such as TCP, UDP, ICMP, and HTTP. Among which TCP flooding is the most prevalent [1]. This results in disruption of services at high cost. The main objective of denial-of-service attack is to consume a large amount of resources, thus preventing legitimate users from receiving service with some minimum performance. TCP flooding [1] exploits TCPs three-way handshake procedure, and specifically its limitation in maintaining half-open connections. Denial of service attack is a technique to make a host or network resource block to its intended users. The attack temporarily or permanently interrupts or suspends services of a computer in the network host connected to the Internet. A permanent denial-of-service attack damages a system so badly that it requires replacement or reinstallation of hardware such as routers, printers, or other network hardware. Hence in general, detection is required before the spread of this attack. Detection of such an attack is often a part of information security [2, 3]. Therefore, it is essential to secure wireless networks from such an attack.

A distributed denial of service (DDoS) attack is a simultaneous network attack on a victim from a large number of compromised hosts, which may be distributed widely among different, independent networks [4]. By exploiting asymmetry between network wide resources, and local capacities of a victim a DDoS attack can build up an intended congestion very quickly. The Internet routing infrastructure, which is stateless and based mainly on destination addresses, appears extremely vulnerable to such coordinated attacks. It is a type of cyber attacks in which the victim will be overloaded and will not able to perform any normal functions. Many researchers have presented their work in various directions. Gavrilis and Dermatas uses radial basis function neural network and statistical features to achieve accurate classification of abnormal activity under DDoS attack without interfering normal traffic [5]. The advantage of this method is that it can block the traffic selectively based on the attack. Wang et al. introduced a queuing model for the evaluation of the denial of service attacks in computer networks. The network is characterized by a two-dimensional embedded Markov chain model. It helps in developing a memory-efficient algorithm for finding the stationary probability distribution which can be used to find other interesting performance metrics such as connection loss probability and buffer occupancy percentages of half-open connections [6]. Gelenbe and Lukes proposed a model to defense denial of service attack using cognitive packet network infrastructure. The technique uses smart packets to select paths based on quality of service [7].

Mell introduces resistant intrusion detection system architecture to counter denial of service attack. The components of intrusion detection system architecture are invisible to the attacker and also this architecture relocates intrusion detection system components from attacked hosts. This is achieved by using mobile agent technology [8]. Hamdi uses outbound and inbound demilitarized zone to detect denial of service attack. The major advantage is that it also identifies synchronize-flooding attack [9]. Later, Chen et al., applied targeted filtering method to identify a distributed denial of service attack. The advantage is that it can be deployed at a local firewall. But, it takes extra time to detect the attack [10]. Rajkumar and Selvakumar proposed a model using Resilient back propagation (RBP) algorithm as the base classifier for the detection of denial of service attack [11]. From the literature survey, it is understood that much research is carried out for the detection of denial of service attack and distributed denial of service attack.

Denial-of-service attacks commonly block the services of legitimate user in a wireless network either temporarily oe permanently by supplying either short term orlong term harmful artificial traffic. Additionally, it is observed that the information system pertaining to denial-of-service attack in wireless network contains uncertainties and the attributes involved in the information system have some specific order. To deal with such uncertainties, criteria, and specific order the concept of dominance based rough set can be used. This motivation help us to think a alternative approach using dominance based rough set.

In this paper, we propose an alternative method using dominance based rough set for the detection of denial of service attack. The rest of the paper is organized as follows: we discuss basic concepts of dominance based rough set in section 2. Section 3 discusses dominance principle. A case study is presented in section 4 to analyze and track denial of service attack using dominance based rough set. Finally, the paper is concluded with a conclusion.

II. Foundation of Information System

An information system provides an expedient to describe a finite set of objects called the universe with a finite set of attributes thereby represents all the available information and knowledge. Formally, it is defined as a four tuple $T = (U, A, V, f)$ where $U = \{x_1, x_2, \cdots, x_n\}$ is a non-empty finite set of objects called the universe, $A = \{a_1, a_2, \cdots, a_n\}$ is a nonempty finite set attributes. The component V is defined as $V = \cup_{a \in A} V_a$, where V_a is the set of attribute values that an attribute a may take. The component $f : (U \times A) \to V$ is an information function. The information system is said to be a decision system if $A = C \cup \{d\}, C \neq \phi, \{d\} \neq \phi$ and $C \cap \{d\} = \phi$ where C is a set of conditional attributes and d is the decision [12].

Let $B \subseteq A$. Two objects x_i and x_j are said to be B-indiscrinble if $f(x_i, a) = f(x_j, a)$ for all $a \in B$. Mathematically, we denote it as $IND(B)$ is defined as below and we write $x_i I_B x_j$.

$$IND(B) = \left\{(x_i, x_j) \in U^2 : f(x_i, a) = f(x_j, a) \forall a \in B\right\}$$

Object x_j dominates object x_i on criteria a if $V_a^{x_j} \leq V_a^{x_i}$, where $V_a^{x_j}$ is the attribute value of object x_j on criteria a. Let $Q \subseteq C$ be a criteria set. Let us define a dominance relation $dm(Q)$ on U as

$$dm(Q) = \left\{(x_i, x_j) \in U^2 : V_a^{x_j} \leq V_a^{x_i} \forall a \in Q\right\} \quad (1)$$

If $(x_i, x_j) \in dm(Q)$, then we write $x_j D_Q x_i$. Let $P \subseteq C$ is a criteria set. Let us define $D_P^+(x_i)$, P-dominating x_i as below.

$$D_P^+(x_i) = \{x_j \in U : x_j D_P x_i\} \quad (2)$$

Similarly, we define a set $D_P^-(x_i)$, P-dominated by $x_i$as below.

$$D_P^-(x_i) = \{x_j \in U : x_i D_P x_j\} \quad (3)$$

Two object x_i and x_j are said to be inconsistent, if their criterion do not satisfy dominance principle with ordered decision class [13].

III. Dominance based Rough Set

Rough set of Pawlak is a mathematical tool used in data analysis in particular to analyze uncertainties [14]. But it fails to analyze data containing preference order and may lead to loss of information. To overcome the limitations the concept of dominance based rough set is introduced [15, 16, 17]. In dominance based rough set, given a set of objects, there is a criterion at least among condition attributes. Additionally attributes like color, country may not be of preference ordered. Therefore, criteria attributes are divided into ordered decision classes based on decision attribute. Also criteria in condition attributes are correlated semantically with ordered decision attribute by means of dominance relation.

Formally, dominance based rough set (DRS) is based on the concept of dominance principle to extract knowledge from the information system. Here, the classification is carried out based on decision class (d). Therefore, the decision (d) divides the universe U into finite number of classes, CL, such as

$$CL = \{CL_i : i \in T\}; T = \{1, 2, 3, \cdots, m\}$$

Additionally, these classes are ordered. It means that, if $r, s \in T$ and $r > s$, then the objects of class Cl_r are preferred then the objects of class Cl_s. The upward and downward unions of every element Cl_i of CL is given as $Cl_i^{\geq}$ and $Cl_i^{\leq}$ respectively. Mathematically, it is defined as

$$Cl_i^{\geq} = \cup_{j \geq i} Cl_j; Cl_i^{\leq} = \cup_{j \leq i} Cl_j$$

Let $Q \subseteq C$, objects certainly belongs to $Cl_i^{\geq}$ and $Cl_i^{\leq}$ are in their lower approximations $\underline{Q}(Cl_i^{\geq})$ and $\underline{Q}(Cl_i^{\leq})$ respectively. The lower approximations are defined as below.

$$\underline{Q}(Cl_i^{\geq}) = \left\{x \in U : D_Q^+ \subseteq Cl_i^{\geq}\right\} \quad (4)$$

$$\underline{Q}(Cl_i^{\leq}) = \left\{x \in U : D_Q^- \subseteq Cl_i^{\leq}\right\} \quad (5)$$

Similarly, objects possibly belong to $Cl_i^{\geq}$ and $Cl_i^{\leq}$ are in their upper approximations $\overline{Q}(Cl_i^{\geq})$ and $\overline{Q}(Cl_i^{\leq})$ respectively. It is defined as below.

$$\overline{Q}(Cl_i^{\geq}) = \bigcup_{x \in Cl_i^{\geq}} D_Q^+(x) \quad (6)$$

$$\overline{Q}(Cl_i^{\leq}) = \bigcup_{x \in Cl_i^{\leq}} D_Q^-(x) \quad (7)$$

The boundary region of $Cl_i^{\geq}$ and $Cl_i^{\geq}$, which contains ambiguous elements are defined as below

$$BN_Q(Cl_i^{\geq}) = \overline{Q}(Cl_i^{\geq}) - \underline{Q}(Cl_i^{\geq})$$

$$BN_Q(Cl_i^{\leq}) = \overline{Q}(Cl_i^{\leq}) - \underline{Q}(Cl_i^{\leq})$$

A. Dominance Relation Based Rule Formation

For a given information system, the dominance principle is capable of deducing more generalized description of objects. This can be done by means of upward and downward union of rough approximation. This is a fundamental concept in a knowledge discovery.

Let $Q \subseteq C$ be a conditional attributes. Based on the rough approximation, the Q-lower and Q-upper approximations are computed on criterion attribute to extract the knowledge. The rules generated from criterion attribute using upward and downward union of Q-lower, Q-upper approximations are of the form "If Condition then Decision".

In real life situation, the data collected may be uncertain, vague and imprecise which may leads to inconsistency. The inconsistency data are identified in rough set by means of indiscernible relation. Likewise the inconsistency presents in the collected data are identified in dominance based rough set on employing dominance relation. The two objects are said to be inconsistent when the criteria attributes do not satisfy dominance principle with decision attribute. Further such inconsistency exists in logic must be removed try as it leads to error decision. The simplest way to remove such inconsistency is to omit the inconsistent objects. The five kinds of determinate rules associated with dominance based rough set are defined as follows [13].

1) For all criteria $a_i \in Q \subseteq C$; if $f(x, a_1) \geq V^x_{a_1}$ and $f(x, a_2) \geq V^x_{a_2}$ and $\cdots$ $f(x, a_i) \geq V^x_{a_i}$, then $x \in Cl^{\geq}_t$ where $t \in \{2, 3, \cdots, n\}$. Rules generated in such way called as certain $D_{\geq}$ decision rules. These rules are obtained from $\underline{Q}(Cl^{\geq}_t)$.
2) For all criteria $a_i \in Q \subseteq C$; if $f(x, a_1) \geq V^x_{a_1}$ and $f(x, a_2) \geq V^x_{a_2}$ and $\cdots$ $f(x, a_i) \geq V^x_{a_i}$, then $x \in Cl^{\geq}_t$ where $t \in \{2, 3, \cdots, n\}$. Rules generated in such way called as possible $D_{\geq}$ decision rules. These rules are obtained from $\overline{Q}(Cl^{\geq}_t)$.
3) For all criteria $a_i \in Q \subseteq C$; if $f(x, a_1) \leq V^x_{a_1}$ and $f(x, a_2) \leq V^x_{a_2}$ and $\cdots$ $f(x, a_i) \leq V^x_{a_i}$, then $x \in Cl^{\leq}_t$ where $t \in \{1, 2, \cdots, (n-1)\}$. Rules generated in such way called as certain $D_{\leq}$ decision rules. These rules are obtained from $\underline{Q}(Cl^{\leq}_t)$.
4) For all criteria $a_i \in Q \subseteq C$; if $f(x, a_1) \leq V^x_{a_1}$ and $f(x, a_2) \leq V^x_{a_2}$ and $\cdots$ $f(x, a_i) \leq V^x_{a_i}$, then $x \in Cl^{\leq}_t$ where $t \in \{1, 2, \cdots, (n-1)\}$. Rules generated in such way called as possible $D_{\leq}$ decision rules. These rules are obtained from $\overline{Q}(Cl^{\leq}_t)$.
5) Let $O_1 = \{a_1, a_2, \cdots, a_k\} \subseteq C$; $O_2 = \{a_{k+1}, a_{k+2}, \cdots, a_i\} \subseteq C$; $Q = (O_1 \cup O_2)$; O_1 and O_2 are not necessarily disjoint. If $f(x, a_1) \geq V^x_{a_1}$ and $f(x, a_2) \geq V^x_{a_2}$, $\cdots$, and $f(x, a_k) \geq V^x_{a_k}$ and $f(x, a_{k+1}) \leq V^x_{a_{k+1}}$ and $f(x, a_{k+2}) \leq V^x_{a_{k+2}}$, $\cdots$ and $f(x, a_i) \leq V^x_{a_i}$, then $x \in Cl_u \cup Cl_{u+1} \cup \cdots \cup Cl_v$, where $r \leq u \leq v \leq t$ and $r, u, v, t \in T$. Rules generated in such way called as approximate $D_{\geq\leq}$ decision rules. These rules are obtained from $\overline{Q}(Cl^{\leq}_r) \cap \overline{Q}(Cl^{\geq}_t)$.

The rules 1 and 3 represent certain knowledge whereas rules 2 and 4 represent possible knowledge that can be extracted from the information system. The rules 5 represent ambiguous knowledge. If $y \in \underline{Q}(Cl^{\geq}_t)$ such that $f(y, a_1) = V^y_{a_1}$, $f(y, a_2) = V^y_{a_2}$, $\cdots$, $f(y, a_i) = V^y_{a_i}$, then y is called as basis of the rule. An object which matches both condition and decision parts of a rule supports the decision rule. An object which meets only condition part of a rule is covered by a decision rule. Decision rules either certain or approximate is said to be complete if it satisfies following conditions.

1) Each $x \in \underline{Q}(Cl^{\geq}_t)$ must support at least one certain $D_{\geq}$ decision rule whose consequent is $x \in Cl^{\geq}_r$ where $r, t \in \{2, 3, \cdots, n\}$ and $r \geq t$.
2) Each $x \in \underline{Q}(Cl^{\leq}_t)$ must support at least one certain $D_{\leq}$ decision rule whose consequent is $x \in Cl^{\leq}_r$ where $r, t \in \{1, 2, \cdots, (n-1)\}$ and $r \leq t$.
3) Each $x \in (\overline{Q}(Cl^{\leq}_r) \cap \overline{Q}(Cl^{\geq}_t))$ must support at least one approximate $D_{\geq\leq}$ decision rule whose consequent is $x \in Cl_u \cup Cl_{u+1} \cup \cdots \cup Cl_v$ where $r \leq u \leq v \leq t$ and $r, u, v, t \in T$.

It means that, the set of rules must cover all objects of the information system. Additionally, it assigns consistent objects to their original classes and inconsistent objects to clusters of classes pertaining to this inconsistency.

IV. Proposed Research Design

A common type of attack used to block the service of the wireless network in recent years is denial of service attack. Therefore, recognizing such an attack is of great challenge. To this end, in this section, we purpose our research design for detecting dos attack. The following Figure 1 depicts an abstract view of the model. The initial step of any model development is problem identification that includes basic knowledge of the problem undertaken. The data collected initially preprocessed. The main objective is to transform the raw input data into an appropriate format for subsequent analysis. The various steps involved are merging of data from data repositories, data cleaning which removes noise and duplicate observations and then selecting relevant observations as per the requirement of the problem undertaken. The selection of observations is done in order to analyze only one decision denial-of-service. The processed data is partitioned into two categories such as training data of 55% and testing data of 45%. The training data is analyzed using dominance based rough set to identify the decision class that effects the decision. We apply DOMLEM algorithm to obtain the rules. algorithm:

A. DOMLEM Algorithm

In rough set theory several algorithms are proposed for induction of decision rules [18, 19, 20]. Some of these algorithms also generate minimum number of rules. Generally, we use heuristic approach to deduce rules because of NP-hard nature [18]. In this paper we use DOMLEM algorithm as proposed by Greco et al [13] for the detection of denial-of-service attack. The algorithm is repeatedly applied for all lower or upper approximations of the upward (downward) unions of decision classes. Considering preference order of decision classes and of getting minimum rules, the algorithm is applied repeatedly starting from the strongest union of classes. Therefore, decision rules of the lower approximations of upward unions of classes

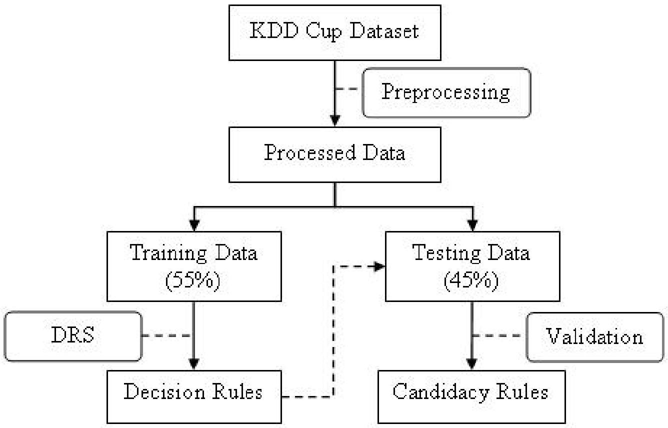

Fig. 1: Abstract View of Research Design

should be taken into consideration in decreasing order. The following notations are used in the DOMLEM algorithm.
C: Denotes set of conditional attributes
Q: Denotes set of criteria, $a_i \in Q \subseteq C$
E: Denotes conjunction of elimentary conditions $e = \{f(x, a_i) \geq V^x_{a_i}\}$
$[E]$: Denotes set of objects in E; $[E] = \{x : f(x, a_i) \geq V^x_{a_i}\}$
FM_k: Denotes the first measure
SM_k: Denotes the second measure

Algorithm 1: (DOMLEM)
Input: Lower approximation of upward union; $\underline{Q}(Cl^{\geq}_i), i = m, (m-1), \cdots, 2$
Output: Set of $D_{\geq}$ decision rules

Begin
- $D_{\geq} = \phi$
- for each $\underline{Q}(Cl^{\geq}_i)$, do
  - $\mathbf{E}$ = Find_Rules ($\underline{Q}(Cl^{\geq}_i)$)
  - for each rule $r \in \mathbf{E}$, do
    - **if** r is a minimal rule, **then** $D_{\geq} = D_{\geq} \cup \{r\}$

End

Function Find_Rules

Begin
- $G = \underline{Q}(Cl^{\geq}_i)$
- $\mathbf{E} = \phi$
- while $G \neq \phi$, do
  - $E = \phi$
  - $S = G$
  - while $E = \phi$ or not ($[E] \subseteq \underline{Q}(Cl^{\geq}_i)$), do
    - $best = \phi$
    - for each criteria $a_i \in Q$ do
      - $Cond = \{f(x, a_i) \geq V^x_{a_i} : \exists x \in S, f(x, a_i) = V^x_{a_i}\}$
      - for each $e_k \in Cond$, do
        - $FM_k = |[e_k] \cap G| / |[e_k]|$
        - $SM_k = |[e_k] \cap G|$
        - find e_k for which FM_k and SM_k is maximum
        - $best = best \cup \{e_k\}$
      - end for
    - end for
    - $E = E \cup \{best\}$
    - $S = S \cap [best]$
  - end while
  - for each $e_k \in E$, do
    - if $[E - \{e_k\}] \subseteq \underline{Q}(Cl^{\geq}_i)$, then $E = E - \{e_k\}$
  - $\mathbf{E} = \mathbf{E} \cup \{E\}$
  - $G = \underline{Q}(Cl^{\geq}_i) - \cup_{e \in E}[E]$
- end while

End

B. An Illustration of DOMLEM Algorithm

This section explains how the above concepts can be applied in analyzing denial-of-service attack in a wireless network. To analyze the above concepts, we have considered the dataset discussed by various authors in their papers [15, 21, 22, 23]. We present the dataset in the following Table 1. The various attributes considered are packets received or sent per seconds (Mbps), number of attacker nodes, types of protocol, service block period, and damage. We denote these attributes as a_1, a_2, a_3, a_4, and a_5 respectively. The attribute a_3 may take values TCP, UDP, or ICMP. Similarly, different values the attribute a_4 may take are zero (Zo), short (So), long (Lo), or permanent (Pt). Finally, the different values that the attribute a_5 may take are hardware fail (HF), software fail (SF), system hang (SH), system reset (SR), time waste (TW), or no damage (ND). The decision attribute (d) describes category of denial of service attack such as permanent denial of service attack (PDA), distributed denial of service attack (DDA), simple denial of service attack (SDA), and no attack (NA). Consider the attributes $Q = \{a_1, a_2, a_4\}$ as criteria among all conditional attributes a_1, a_2, a_3, a_4, a_5.

The above table contains 13 objects of denial-of-service attack in a wireless network and its various conditional attribute values, where U denotes node number. For analysis purpose, the dataset is divided into two training dataset of 7 objects (55%) and testing dataset of 6 objects (45%). We employ dominance based rough set data analysis on training dataset to obtain candidacy classes. The testing dataset is used to detect over fitting of the decision classes based on the predefined threshold value 70%. The decision divides the training dataset of universe into finite number of classes, CL, as below.

$$CL = \{Cl_1, Cl_2, Cl_3, Cl_4\}$$

where Cl_1 = $\{x_1, x_7\}$; Cl_2 = $\{x_2, x_6\}$; Cl_3 = $\{x_3\}$ and Cl_4 = $\{x_4, x_5\}$. It is also observed that the class Cl_4 has more

TABLE I: An information system of denial-of-service attack in a wireless network

U	a_1	a_2	a_3	a_4	a_5	d
x_1	1.3	2	TCP	Zo	ND	NA
x_2	2.67	1	UDP	So	TW	SDA
x_3	2.5	4	ICMP	Lo	SF	DDA
x_4	3.0	5	UDP	Lo	SH	PDA
x_5	2.4	2	TCP	Pt	SF	PDA
x_6	2.6	7	TCP	Lo	SF	SDA
x_7	2.68	1	ICMP	So	ND	NA
x_8	3.1	4	UDP	Lo	SR	DDA
x_9	2.68	1	ICMP	So	TW	SDA
x_{10}	2.5	6	UDP	Pt	HF	PDA
x_{11}	2.7	2	TCP	So	TW	SDA
x_{12}	3.2	3	ICMP	Lo	SH	NA
x_{13}	1.5	0	UDP	Zo	ND	NA

delay than Cl_3; Cl_3 has more delay than Cl_2; and Cl_2 has more delay than Cl_1. The downward unions of every element Cl_i, $i = 1, 2, 3$ of CL are given below.

$$Cl_1^{\leq} = \{x_1, x_7\}$$
$$Cl_2^{\leq} = \cup_{j\leq 2} Cl_j = Cl_1 \cup Cl_2 = \{x_1, x_2, x_6, x_7\}$$
$$Cl_3^{\leq} = \{x_1, x_2, x_3, x_6, x_7\}$$

Similarly, the upward unions of training dataset element Cl_i, $i = 4, 3, 2$ of CL are given below.

$$Cl_4^{\geq} = \cup_{j\geq 4} Cl_j = Cl_4 = \{x_4, x_5\}$$
$$Cl_3^{\geq} = \cup_{j\geq 3} Cl_j = Cl_3 \cup Cl_4 = \{x_3, x_4, x_5\}$$
$$Cl_2^{\geq} = \{x_2, x_3, x_4, x_5, x_6\}$$

Let us consider the downward union $Cl_1^{\leq} = \{x_1, x_7\}$ on considering the criteria $Q = \{a_1, a_2, a_4\} \subseteq C$, the lower and upper approximations are given as $\underline{Q}(Cl_1^{\leq}) = \{x_1\}$ and $\overline{Q}(Cl_1^{\leq}) = \{x_1, x_2, x_7\}$ respectively. Therefore, the boundary objects are $BN_Q(Cl_1^{\leq}) = \{x_2, x_7\}$. It is because the objects x_2 and x_7 violates the dominance principle. This can be seen from the information system presented in Table I. From Table 1, it is clear that object x_7 dominates object x_2 on criteria Q, but the decision corresponding to the object x_7 is finer then the decision corresponding to the object x_2. Hence, they are inconsistent. Also, it can be shown that objects x_3 and x_6 are also inconsistent. Similarly the lower, upper approximations, and boundary of downward and upward unions of other classes are presented below.

$$\underline{Q}(Cl_2^{\leq}) = \{x_1, x_2, x_7\}, \overline{Q}(Cl_2^{\leq}) = \{x_1, x_2, x_3, x_6, x_7\}$$
$$BN_Q(Cl_2^{\leq}) = \{x_3, x_6\}$$
$$\underline{Q}(Cl_3^{\leq}) = \{x_1, x_2, x_3, x_6, x_7\},$$
$$\overline{Q}(Cl_3^{\leq}) = \{x_1, x_2, x_3, x_6, x_7\}, BN_Q(Cl_3^{\leq}) = \{\phi\}$$
$$\underline{Q}(Cl_4^{\geq}) = \{x_4, x_5\}, \overline{Q}(Cl_4^{\geq}) = \{x_4, x_5\}$$
$$BN_Q(Cl_4^{\geq}) = \{\phi\}$$
$$\underline{Q}(Cl_3^{\geq}) = \{x_4, x_5\}, \overline{Q}(Cl_3^{\geq}) = \{x_3, x_4, x_5, x_6\}$$
$$BN_Q(Cl_3^{\geq}) = \{x_3, x_6\}$$
$$\underline{Q}(Cl_2^{\geq}) = \{x_3, x_4, x_5, x_6\}$$
$$\overline{Q}(Cl_2^{\geq}) = \{x_2, x_3, x_4, x_5, x_6, x_7\}, BN_Q(Cl_2^{\geq}) = \{x_2, x_7\}$$

Now, we explain how certain $D_{\geq}$ decision rules are induced for the upward union. Let us consider the class $Cl_4^{\geq}$ and the lower approximation $\underline{Q}(Cl_4^{\geq}) = \{x_4, x_5\}$ for obtaining $D_{\geq}$ decision rules. Employing the DOMLEM algorithm on $\underline{Q}(Cl_4^{\geq})$, we get the elimentary conditions as below.

$$e_1 = \{f(x, a_1) \geq 3.0\} = \{x_4\}; 1/1; 1$$
$$e_2 = \{f(x, a_1) \geq 2.4\} = \{x_2, x_3, x_4, x_5, x_6, x_7\}; 2/6; 2$$
$$e_3 = \{f(x, a_2) \geq 2.0\} = \{x_1, x_3, x_4, x_5, x_6\}; 2/5; 2$$
$$e_4 = \{f(x, a_2) \geq 5.0\} = \{x_4, x_6\}; 1/2; 1$$
$$e_5 = \{f(x, a_4) \geq Lo\} = \{x_3, x_4, x_5, x_6\}; 2/4; 2$$
$$e_6 = \{f(x, a_4) \geq Pt\} = \{x_5\}; 1/1; 1$$

The elementary conditions e_1, e_6 produce the highest first measure and second measure. But, both elementary conditions covers only one distinct positive example. Further both $[e_1]$, $[e_6]$ are the subsets of $\underline{Q}(Cl_4^{\geq})$. We choose elementary condition e_1 initially which covers the object x_4 and is used to introduce the rule. However, we can also choose the elementary condition e_6. Further, the object x_4 is removed from G and the remaining object is to be covered is x_5. Thus, we have 4 elimentary conditions as below to cover the object x_5.

$$e_7 = \{f(x, a_2) \geq 2.0\} = \{x_1, x_3, x_5, x_6\}; 1/4; 1$$
$$e_8 = \{f(x, a_1) \geq 2.4\} = \{x_2, x_3, x_5, x_6, x_7\}; 1/5; 1$$
$$e_9 = \{f(x, a_4) \geq Lo\} = \{x_3, x_5, x_6\}; 1/3; 1$$
$$e_{10} = \{f(x, a_4) \geq Pt\} = \{x_5\}; 1/1; 1$$

Next, we can pick the elementary condition e_{10} because of the highest first and second measure which covers the object x_5. Thus no need to proceed further and the rule can be written as:

if $f(x, a_1) \geq$ 3.0, then $x \in Cl_4^{\geq}$

if $f(x, a_4) \geq$ Pt, then $x \in Cl_4^{\geq}$

Similarly, consider $\underline{Q}(Cl_2^{\geq})$ to obtain the rules for the class $x \in Cl_2^{\geq}$. On employing the DOMLEM algorithm we get the following elimentary conditions.

$$e_1 = \{f(x, a_1) \geq 2.5\} = \{x_2, x_3, x_4, x_6, x_7\}; 3/5; 3$$
$$e_2 = \{f(x, a_1) \geq 3\} = \{x_4\}; 1/1; 1$$
$$e_3 = \{f(x, a_1) \geq 2.4\} = \{x_2, x_3, x_4, x_5, x_6, x_7\}; 4/6; 4$$
$$e_4 = \{f(x, a_1) \geq 2.6\} = \{x_2, x_6, x_7\}; 1/3; 1$$
$$e_5 = \{f(x, a_2) \geq 4\} = \{x_3, x_4, x_6\}; 3/3; 1$$
$$e_6 = \{f(x, a_2) \geq 5\} = \{x_4, x_6\}; 2/2; 2$$
$$e_7 = \{f(x, a_2) \geq 2\} = \{x_1, x_3, x_4, x_5, x_6\}; 4/5; 4$$
$$e_8 = \{f(x, a_2) \geq 7\} = \{x_6\}; 1/1; 1$$
$$e_9 = \{f(x, a_4) \geq Lo\} = \{x_3, x_4, x_5, x_6\}; 4/4; 4$$
$$e_{10} = \{f(x, a_4) \geq Pt\} = \{x_5\}; 1/1; 1$$

The elementary conditions $e_2, e_5, e_6, , e_8$, and e_9 have the highest first measure but the elimentary condition e_9 has the highest second measure and so we choose the elementary condition e_9. Further $[e_9]$ is subset of $\underline{Q}(Cl_2^{\geq})$ and covers all positive examples. Thus the process terminates and the rule can be written as:

if $f(x, a_4) \geq$ Lo, then $x \in Cl_2^{\geq}$

Likewise, we explain how certain $D_{\leq}$ decision rules are induced for the downward union. Let us consider the class $Cl_1^{\leq}$ and the lower approximation $\underline{Q}(Cl_1^{\leq}) = \{x_1\}$ for obtaining $D_{\leq}$ decision rules. The elementary conditions obtained are given below.

$e_1 = \{f(x, a_1) \leq 1.3\} = \{x_1\}\,; 1/1; 1$

$e_2 = \{f(x, a_4) \leq Zo\} = \{x_1\}\,; 1/1; 1$

$e_3 = \{f(x, a_2) \leq 2\} = \{x_1, x_2, x_7\}\,; 1/3; 1$

The elementary conditions e_1, and e_2 have the highest first measure and covers all the positive examples. Further both $[e_1], [e_2]$ are subsets of $\underline{Q}(Cl_1^{\leq})$. Therefore, the process terminates and the rules can be stated as:

if $f(x, a_4) \leq$ Zo, then $x \in Cl_1^{\leq}$

if $f(x, a_1) \leq$ 1.3, then $x \in Cl_1^{\leq}$

Similarly, we consider $\underline{Q}(Cl_2^{\leq}) = \{x_1, x_2, x_7\}$ to obtain the rules for the class $Cl_2^{\leq}$. The elementary conditions obtained are listed below.

$e_1 = \{f(x, a_1) \leq 1.3\} = \{x_1\}\,; 1/1; 1$

$e_2 = \{f(x, a_1) \leq 2.67\} = \{x_1, x_2, x_3, x_5, x_6\}\,; 2/5; 2$

$e_3 = \{f(x, a_1) \leq 2.68\} = \{x_1, x_2, x_3, x_5, x_6, x_7\}\,; 3/6; 3$

$e_4 = \{f(x, a_2) \leq 2\} = \{x_1, x_2, x_7\}\,; 3/3; 3$

$e_5 = \{f(x, a_2) \leq 1\} = \{x_2, x_7\}\,; 2/2; 2$

$e_6 = \{f(x, a_4) \leq Zo\} = \{x_1\}\,; 1/1; 1$

$e_7 = \{f(x, a_4) \leq So\} = \{x_1, x_2, x_7\}\,; 3/3; 3$

The elementary conditions e_1, e_4, and e_7 have the highest first measure and the condition e_1 covers only one positive example. Alternatively, the conditions e_4, and e_7 have the highest second measure and covers all the positive examples. Further, both $[e_4]$, and $[e_7]$ are subsets of $\underline{Q}(Cl_2^{\leq})$. Therefore, the process terminates and the rule can be stated as:

if $f(x, a_2) \leq$ 2, then $x \in Cl_2^{\leq}$

if $f(x, a_4) \leq$ So, then $x \in Cl_2^{\leq}$

Likewise, consider $\underline{Q}(Cl_3^{\leq}) = \{x_1, x_2, x_3, x_6, x_7\}$ to obtain the decision rules for the class $Cl_3^{\leq}$. The elementary conditions obtained are listed below.

$e_1 = \{f(x, a_1) \leq 1.3\} = \{x_1\}\,; 1/1; 1$

$e_2 = \{f(x, a_1) \leq 2.67\} = \{x_1, x_2, x_3, x_5, x_6\}\,; 4/5; 4$

$e_3 = \{f(x, a_1) \leq 2.6\} = \{x_1, x_3, x_5, x_6\}\,; 3/4; 3$

$e_4 = \{f(x, a_1) \leq 2.68\} = \{x_1, x_2, x_3, x_5, x_6, x_7\}\,; 5/6; 5$

$e_5 = \{f(x, a_1) \leq 2.5\} = \{x_1, x_3, x_5\}\,; 2/3; 2$

$e_6 = \{f(x, a_2) \leq 1\} = \{x_2, x_7\}\,; 2/2; 2$

$e_7 = \{f(x, a_2) \leq 2\} = \{x_1, x_2, x_7\}\,; 3/3; 3$

$e_8 = \{f(x, a_2) \leq 4\} = \{x_1, x_2, x_3, x_5, x_7\}\,; 4/5; 4$

$e_9 = \{f(x, a_2) \leq 7\} = \{x_1, x_2, x_3, x_4, x_5, x_6, x_7\}\,; 5/7; 5$

$e_{10} = \{f(x, a_4) \leq Zo\} = \{x_1\}\,; 1/1; 1$

$e_{11} = \{f(x, a_4) \leq So\} = \{x_1, x_2, x_7\}\,; 3/3; 3$

$e_{12} = \{f(x, a_4) \leq Lo\} = \{x_1, x_2, x_3, x_4, x_6, x_7\}\,; 5/6; 5$

The elementary conditions e_1, e_6, e_7, e_{10}, and e_{11} have highest first measure whereas e_7 and e_{11} have highest second measure. But, both elementary conditions e_7 and e_{11} covers same positive examples. Further both $[e_7]$, and $[e_{11}]$ are the subsets of $\underline{Q}(Cl_3^{\leq})$. Therefore, we can choose either of the elementary conditions e_7 and e_{11}. Let us choose the elementary condition e_7 that covers objects x_1, x_2, and x_7. To proceed further, the objects x_1, x_2, and x_7 are removed from G and the process is repeated. The remaining objects are to be covered are x_3, and x_6. Therefore, the above elementry conditions leads to 7 elementary conditions as below.

$e_{13} = \{f(x, a_1) \leq 2.67\} = \{x_3, x_5, x_6\}\,; 2/3; 2$

$e_{14} = \{f(x, a_1) \leq 2.6\} = \{x_3, x_5, x_6\}\,; 2/3; 2$

$e_{15} = \{f(x, a_1) \leq 2.68\} = \{x_3, x_5, x_6\}\,; 2/3; 2$

$e_{16} = \{f(x, a_1) \leq 2.5\} = \{x_3, x_5\}\,; 1/2; 1$

$e_{17} = \{f(x, a_2) \leq 4\} = \{x_3, x_5\}\,; 1/2; 1$

$e_{18} = \{f(x, a_2) \leq 7\} = \{x_3, x_4, x_5, x_6\}\,; 2/4; 2$

$e_{19} = \{f(x, a_4) \leq Lo\} = \{x_3, x_4, x_6\}\,; 2/3; 2$

The elementary conditions e_{13}, e_{14}, e_{15}, and e_{19} have the highest first measure. Also, the second measure of these conditions are same. But, it is not sufficient to create decision rules using any of the conditions because all these conditions cover objects either x_5 or x_4 which is a negative example. Therefore, one has to consider complexes $(e_{13} \cap e_{19})$, $(e_{14} \cap e_{19})$, and $(e_{15} \cap e_{19})$. All the complexes have highest first measure and covers positive examples. Therefore, we get the following decision rules.

if $f(x, a_2) \leq$ 2, then $x \in Cl_3^{\leq}$

if $f(x, a_4) \leq$ So, then $x \in Cl_3^{\leq}$

if $f(x, a_1) \leq$ 2.67 and $f(x, a_4) \leq$ Lo, then $x \in Cl_3^{\leq}$

if $f(x, a_1) \leq$ 2.6 and $f(x, a_4) \leq$ Lo, then $x \in Cl_3^{\leq}$

if $f(x, a_1) \leq$ 2.68 and $f(x, a_4) \leq$ Lo, then $x \in Cl_3^{\leq}$

Now we explain how approximate $D_{\geq\leq}$ approximate decision rules are induced form $\overline{Q}(Cl_1^{\leq}) \cap \overline{Q}(Cl_2^{\geq}) = \{x_2, x_7\}$. Let $O_1 = \{a_1, a_2\}$ and $O_2 = \{a_1, a_4\}$. The elementary conditions obtained are listed below.

$e_1 = \{f(x, a_1) \geq 2.67\} = \{x_2, x_4, x_7\}\,; 2/3; 2$

$e_2 = \{f(x, a_1) \geq 2.68\} = \{x_4, x_7\}\,; 1/2; 1$

$e_3 = \{f(x, a_2) \geq 1\} = \{x_1, x_2, x_3, x_4, x_5, x_6, x_7\}\,; 2/7; 2$

$e_4 = \{f(x, a_1) \leq 2.67\} = \{x_1, x_2, x_3, x_5, x_6\}\,; 1/5; 1$

$e_5 = \{f(x, a_1) \leq 2.68\} = \{x_1, x_2, x_3, x_5, x_6, x_7\}\,; 2/6; 2$

$e_6 = \{f(x, a_4) \leq So\} = \{x_1, x_2, x_7\}\,; 2/3; 2$

The elementary conditions e_1, and e_6 produces the highest first measure. But, both elementary conditions e_1 and e_6 covers the positive and negative example. Further both $[e_1]$, $[e_6]$ are not the subsets of $\overline{Q}(Cl_1^{\leq})\cap\overline{Q}(Cl_2^{\geq})$. Thus one has to consider complex $(e_1 \cap e_6)$. It is also a subset of $\overline{Q}(Cl_1^{\leq}) \cap \overline{Q}(Cl_1^{\leq})$. Additionally, it produces the highest first and second measure. Therefore, the rule can be stated as below:

if ($f(x, a_1) \geq 2.67$ and $f(x, a_4) \leq$ So) then $x \in Cl_1 \cup Cl_2$

Similarly, on considering $\overline{Q}(Cl_2^{\leq}) \cap \overline{Q}(Cl_3^{\geq}) = \{x_3, x_6\}$ and O_1, O_2 as stated above, the approximate $D_{\geq\leq}$ rules are computed. The elementary conditions obtained are listed below.

$e_1 = \{f(x, a_1) \geq 2.5\} = \{x_2, x_3, x_4, x_6, x_7\}\,; 2/5; 2$

$e_2 = \{f(x, a_1) \geq 2.6\} = \{x_2, x_4, x_6, x_7\}\,; 1/4; 1$

$e_3 = \{f(x, a_2) \geq 4\} = \{x_3, x_4, x_6\}\,; 2/3; 2$

$e_4 = \{f(x, a_2) \geq 7\} = \{x_6\}\,; 1/1; 1$

The elementary condition e_4 produces the highest first measure, covers positive example, and $[e_4]$ is a subsets of $\overline{Q}(Cl_2^{\leq}) \cap \overline{Q}(Cl_3^{\geq})$. Therefore, the elementary condition e_4 is considered to generate rule. Further, the object x_6 is removed and elementary conditions are obtained to include the object x_3.

$e_5 = \{f(x, a_1) \geq 2.5\} = \{x_2, x_3, x_4, x_7\}\,; 1/4; 1$

$e_6 = \{f(x, a_2) \geq 4\} = \{x_3, x_4\}\,; 1/2; 1$

$e_7 = \{f(x, a_1) \leq 2.5\} = \{x_1, x_3, x_5\}\,; 1/3; 1$

$e_8 = \{f(x, a_1) \leq 2.6\} = \{x_1, x_2, x_3, x_5\}\,; 1/4; 1$

$e_9 = \{f(x, a_4) \leq Lo\} = \{x_1, x_2, x_3, x_4, x_7\}\,; 1/5; 1$

The elementary conditions e_6 produces the highest first measure, covers both positive and negative example, and is not a subset of $\overline{Q}(Cl_2^{\leq}) \cap \overline{Q}(Cl_3^{\geq})$. Thus we have to consider the complex $(e_6 \cap e_7)$ to cover the positive example x_3. The rules generated in this way are listed below.

if ($f(x, a_2) \geq 7$) then $x \in Cl_2 \cup Cl_3$

if ($f(x, a_1) \leq 2.5$ and $f(x, a_2) \geq 4$) then $x \in Cl_2 \cup Cl_3$

Now, collectively we write the decision rules obtained as below.

1) if $f(x, a_1) \geq 3.0$, then $x \in Cl_4^{\geq}$
2) if $f(x, a_4) \geq$ Pt, then $x \in Cl_4^{\geq}$
3) if $f(x, a_4) \geq$ Lo, then $x \in Cl_2^{\geq}$
4) if $f(x, a_4) \leq$ Zo, then $x \in Cl_1^{\leq}$
5) if $f(x, a_1) \leq 1.3$, then $x \in Cl_1^{\leq}$
6) if $f(x, a_2) \leq 2$, then $x \in Cl_2^{\leq}$
7) if $f(x, a_4) \leq$ So, then $x \in Cl_2^{\leq}$
8) if $f(x, a_2) \leq 2$, then $x \in Cl_3^{\leq}$
9) if $f(x, a_4) \leq$ So, then $x \in Cl_3^{\leq}$
10) if $f(x, a_1) \leq 2.67$ and $f(x, a_4) \leq$ Lo, then $x \in Cl_3^{\leq}$
11) if $f(x, a_1) \leq 2.6$ and $f(x, a_4) \leq$ Lo, then $x \in Cl_3^{\leq}$
12) if $f(x, a_1) \leq 2.68$ and $f(x, a_4) \leq$ Lo, then $x \in Cl_3^{\leq}$
13) if $f(x, a_1) \geq 2.67$ and $f(x, a_4) \leq$ So, then $x \in (Cl_1 \cup Cl_2)$
14) if $f(x, a_2) \geq 7$, then $x \in (Cl_2 \cup Cl_3)$
15) if $f(x, a_1) \leq 2.5$ and $f(x, a_2) \geq 4$, then $x \in (Cl_2 \cup Cl_3)$

Finally, the rules obtained are validated with the testing dataset on computing the accuracy (Acc.) basing on precision (Prec.) and recall (Rec.). The precision, recall, and accuracy are computed using the equations (8), (9), and (10). The notation T_P is used for correct classification of cases to decisions whereas F_P is used for incorrect classification of cases to decisions. The notation T_N is the number of cases which correctly classified as negative whereas F_N is the number of incorrect cases classified as positive. Additionally a rule is also discarded if the accuracy falls less than the predefined threshold value 70%.

$$Prec. = \frac{|T_P|}{|T_P + F_P|} \tag{8}$$

$$Rec. = \frac{|T_P|}{|T_P + F_N|} \tag{9}$$

$$Acc. = \frac{|T_P + T_N|}{|T_P + F_P + T_N + F_N|} \tag{10}$$

The computation of precision, recall, and accuracy for the testing objects is presented in Table II. It is clear that the accuracy of rules 1, 5, 8, 9, 10, 11, 12, 14, and 15 are less than the predefined threshold value and hence discarded.

TABLE II: Rule validation of denial-of-service attacks in a wireless network

Rule	Sup. Obj.	T_P	F_N	F_P	T_N	Prec.	Rec.	Acc.
1	-	0	1	2	3	0	0	50
2	x_{10}	1	0	0	5	1	1	100
3	x_8,x_{10}	2	0	1	3	1	0.5	83.33
4	x_{13}	1	1	0	4	1	0.5	83.33
5	-	0	2	0	4	0	0	66.67
6,7	x_9,x_{11}, x_{13}	3	1	0	2	1	0.75	83.33
8,9	x_9,x_{11}, x_{13}	3	2	0	1	1	0.67	66.67
10	x_{13}	1	4	0	1	1	0.2	33.33
11	x_{13}	1	4	0	1	1	0.2	33.33
12	x_9,x_{13}	2	3	0	1	1	0.4	50
13	x_9,x_{11}, x_{13}	3	1	0	2	1	0.75	83.33
14	-	0	0	3	3	0	0	50
15	-	0	1	3	2	0	0	33.33

V. Empirical Study of DoS Attack

This section describes how the proposed technique is used for a dataset. The dataset is preprocessed so that it may be able to give as an input to our developed system. Collection of data is a critical problem. This can be done by three ways as by using real traffic, by using sanitized traffic, and by using simulated traffic. However difficulties exist in using these approaches. Real traffic approach is very costly while sanitized approach is risky. The creating of simulation is also a difficult task. Further, in order to model various wireless networks, different types of traffic is needed. In order to avoid dealing with these difficulties, Knowledge Discovery Dataset (KDD)-cup dataset is considered for experimental analysis.

The dataset contains 11,160 records in which decisions for 3,260 records are normal whereas for 7,900 recorrds are various dos attacks such as neptune, udp storm, smurf, ping of death (PoD), back, teardrop, land, mailbomb, process table. Each sample of the dataset represents a connection between two wireless network hosts according to network protocols. It is described by 42 features as depicted in Table III. Out of 42 features, 41 are conditional features and one is decision. The set of 41 features are divided into four subsets such as basic feature set, data flow feature set, host based feature set, and content feature set. The basic feature set, a_1 to a_9, is used to check the status of the flags, number of source bytes, number of destination bytes, types of protocols used, and duration of the period while information is communicated. The content feature set, a_{10} to a_{22}, is used to check the number of logins failed, number of compromised, number of logged-in, and number of guest login etc. Likewise the data flow feature set, a_{23} to a_{31}, is used to verify the sending and receiving errors during communication between source and destination. Similarly, the host based feature set, a_{32} to a_{41}, is used to get the information of receiving host and sending host errors while communication. From 41 features, 38 features are continuous or discrete (quantitative) and remaining 3 features are qualitative or categorical.

Each sample of decision feature is labeled as either normal or various dos attack. The dataset contains 10 class labels out of which one class is normal and remaining classes are different dos attacks such as neptune, udp storm, smurf, pod, back, teardrop, land, mail bomb, process table respectively. Some dos attacks such as mail bomb, neptune, or smurf abuse a perfectly legitimate feature.The teardrop, pod create malformed packets that confuse the TCP/IP stack of the machine that is trying to reconstruct the packet. The other dos attacks such as back, land takes the advantage of bugs in a particular network daemon.

TABLE III: Features set of denial-of-dervice attack

S. No.	Features	Notation	Type
I	**Basic Feature**		
1	duration	a_1	continuous
2	protocol-type	a_2	symbolic
3	service	a_3	symbolic
4	flag	a_4	symbolic
5	src-bytes	a_5	continuous
6	dst-bytes	a_6	continuous
7	land	a_7	discrete
8	wrong-fragment	a_8	continuous
9	urgent	a_9	continuous
II	**Content Feature**		
10	hot	a_{10}	discrete
11	num-failed-logins	a_{11}	continuous
12	logged-in	a_{12}	discrete
13	num-compromised	a_{13}	continuous
14	root-shell	a_{14}	discrete
15	su-attempted	a_{15}	discrete
16	num-root	a_{16}	continuous
17	num-file-creations	a_{17}	continuous
18	num-shells	a_{18}	continuous
19	num-access-files	a_{19}	continuous
20	num-outbound-cmds	a_{20}	continuous
21	is-host-login	a_{21}	discrete
22	is-guest-login	a_{22}	discrete
III	**Data Flow Feature**		
23	count	a_{23}	continuous
24	srv-count	a_{24}	continuous
25	serror-rate	a_{25}	continuous
26	srv-serror-rate	a_{26}	continuous
27	rerror-rate	a_{27}	continuous
28	srv-rerror-rate	a_{28}	continuous
29	same-srv-rate	a_{29}	continuous
30	diff-srv-rate	a_{30}	continuous
31	srv-diff-host-rate	a_{31}	continuous
IV	**Host Based Feature**		
32	dst-host-count	a_{32}	continuous
33	dst-host-srv-count	a_{33}	continuous
34	dst-host-same-srv-rate	a_{34}	continuous
35	dst-host-diff-srv-rate	a_{35}	continuous
36	dst-host-same-src-port-rate	a_{36}	continuous
37	dst-host-srv-diff-host-rate	a_{37}	continuous
38	dst-host-serror-rate	a_{38}	continuous
39	dst-host-srv-serror-rate	a_{39}	continuous
40	dst-host-rerror-rate	a_{40}	continuous
41	dst-host-srv-rerror-rate	a_{41}	continuous
V	**Decision**		
42	decision	d	symbolic

A. Experimental Analysis

We implement wireless network dos detection system with C programming language and perform experiments in a computer with 2.67 GHz Intel core i3 processor, and 2 GB RAM. Total 11,600 records are divided into two categories such as training dataset of 6,138 (55%) records and testing dataset of 5022 (45%) records. The details of training, testing, total dataset and its various classifications are given in Table IV. Out of 41 conditional features 18 features such as a_1, a_3, a_4, a_6, a_{13}, a_{14}, a_{16}, a_{17}, a_{19}, a_{20}, a_{21}, a_{22}, a_{33}, a_{34}, a_{35}, a_{37}, a_{40}, a_{41} are considered as criterion as suggested by various authors [24, 25]. For better visualization of the dataset, a graphical representation is shown in Figure 2.

Experimental analysis is carried out on each class of training dataset. Initially, we employed DOMLEM algorithm on 1887 records that are falling under the category normal. The total number of rules generated are 23. The rules generated are presented on Table V. These rules are further validated with 1373 records of testing dataset and found that rules 6, 9, 10, 16 and 18 are having accuracy less than the predefined threshold value. Hence, these rules are discarded. A graphical representation is shown in Figure 3. Likewise 740 records of data that are falling under the category of neptune, 767 records of data of udp storm, 762 records of data of smurf, 1042 records of data of pod, 188 records of data of back, 285 records of data of tear-drop, 155 records of data of land, 162 records of data of mail-bomb, and 150 records of data of process table are passed to DOMLEM algorithm. The total number of rules generated are 146. The category neptune generated 30 rules, category udp storm generated 18 rules, category smurf generated 17 rules, category pod generated 20 rules, category back generated 15 rules, category tear-drop generated 12 rules, category land generated 10 rules, category mail bomb generated 13 rules, and the category process table generated 11 rules. These rules are further validated with the testing dataset as mentioned in Table IV. The number of rules discarded for the categories naptune, udp storm, smurf, pod,

TABLE IV: Training, testing classification of datasets

S. No.	Description	Training Set	Testing Set	Total Set
1	normal	1,887	1,373	3,260
2	neptune	740	1,270	2,010
3	udp-strom	767	478	1,245
4	smurf	762	579	1,341
5	pod	1,042	680	1722
6	back	188	24	212
7	tear-drop	285	29	314
8	land	155	150	305
9	mail-bomb	162	250	412
10	process-table	150	189	339
	Total	6,138 (55%)	5,022 (45%)	11,160

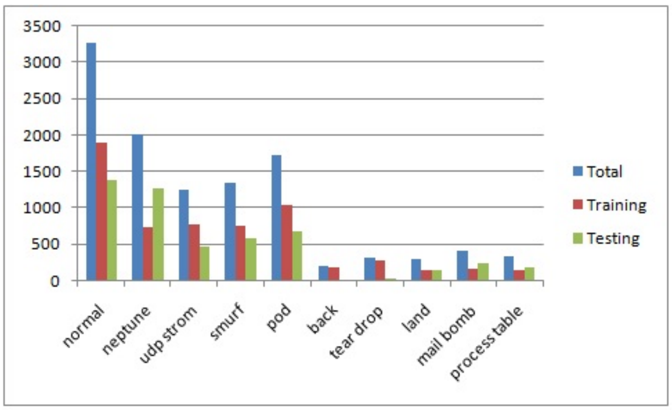

Fig. 2: Characteristics of Dataset

TABLE V: Selected list of normal rules

Rule No.	Description	Acc.
1	If $a_{34} \geq 0.34$ then d=Normal	100
2	If $a_{35} \geq 0.32$ then d=Normal	99
3	If $a_{37} \geq 0.34$ then d=Normal	100
4	If $a_{40} \geq 1$ then d=Normal	99
5	If $a_{19} \leq 0$ then d=Normal	100
6	If $a_{21} \geq 1$ then d=Normal	63
7	If $a_{1} \leq 0$ then d=Normal	100
8	If $a_{34} \geq 0.34$ and $a_{19} \leq 0$ then d=Normal	100
9	If $a_{22} \leq 1$ then d=Normal	33.33
10	If $a_{20} \leq 0$ then d=Normal	67.66
11	If $a_{34} \geq 0.34$ and $a_{1} \leq 0$ then d=Normal	99
12	If $a_{37} \geq 0.34$ and $a_{19} \leq 0$ then d=Normal	99
13	If $a_{37} \geq 0.34$ and $a_{1} \leq 0$ then d=Normal	100
14	If $a_{22} \geq 1$ and $a_{20} \leq 0$ then d=Normal	100
15	If $a_{21} \geq 1$ and $a_{1} \leq 0$ then d=Normal	99
16	If $a_{21} \geq 1$ and $a_{20} \leq 0$ then d=Normal	57.67
17	If $a_{34} \geq 0.34$ and $a_{21} \leq 1$ then d=Normal	100
18	If $a_{21} \geq 1$ and $a_{22} \leq 1$ then d=Normal	63.15
19	If $a_{34} \geq 0.34$ and $a_{37} \leq 0.34$ then d=Normal	98
20	If $a_{35} \geq 0.32$ and $a_{1} \leq 0$ then d=Normal	100
21	If $a_{35} \geq 0.32$ and $a_{20} \leq 0$ then d=Normal	100
22	If $a_{37} \geq 0.34$ and $a_{22} \leq 1$ then d=Normal	98
23	If $a_{37} \geq 0.34$ and $a_{21} \geq 1$ then d=Normal	100

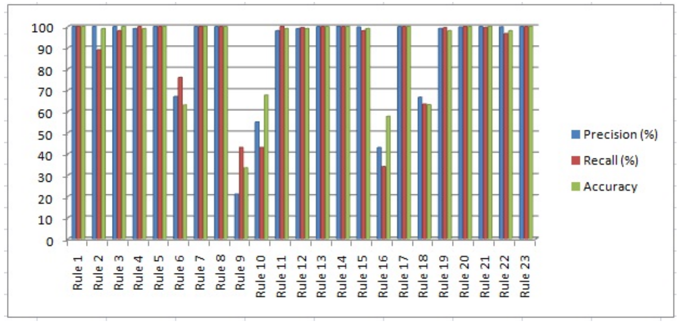

Fig. 3: Graphical view of precision, recall, accuracy

back, tear-drop, land, mail bomb, and process table are 6, 3, 2, 2, 3, 2, 2, 3, and 3 respectively. The final rules selected for various categories naptune, udp storm, smurf, pod, back, tear-drop, land, mail bomb, and process table are presented in Table VI, Table VII, Table VIII, Table IX, Table X, Table XI, Table XII, Table XIII, and Table XIV respectively.

TABLE VI: Selected list of neptune rules

Rule No.	Description	Acc.
1	If $a_{33} \geq 304$ then d=Neptune, Smurf	97.77
2	If $a_{34} \geq 0.36$ then d=Neptune, UDP Storm	99.65
3	If $a_{35} \geq 0.76$ then d=Neptune, Smurf, POD	100
4	If $a_{37} \geq 0.25$ then d=Neptune, Smurf	98.78
5	If $a_{40} \geq 0.24$ then d=Neptune	100
6	If $a_{41} \geq 0.15$ then d=Neptune	99.65
7	If $a_{13} \geq 4$ then d=Neptune, Smurf	99.65
8	If $a_{14} \geq 255$ then d=Neptune, POD	100
9	If $a_{16} \leq 531$ then d=Neptune	100
10	If $a_{17} \geq 8854$ then d=Neptune	99.65
11	If $a_{19} \geq 104$ then d=Neptune	100
12	If $a_{20} \leq 148$ then d=Neptune	100
13	If $a_{13} \geq 1$ then d=Neptune, POD	100
14	If $a_{14} \geq 1$ then d=Neptune, Smurf	100
15	If $a_{34} \leq 0.10$ then d=Neptune	100
16	If $a_{37} \geq 0.61$ then d=Neptune	99.65
17	If $a_{1} \geq 31$ and $a_{20} \leq 104$ then d=Neptune	100
18	If $a_{6} \geq 2252$ and $a_{22} \geq 1$ then d=Neptune	100
19	If $a_{6} \geq 8854$ and $a_{33} \geq 255$ then d=Neptune, Smurf	100
20	If $a_{6} \geq 1461$ and $a_{33} \leq 148$ then d=Neptune	100
21	If $a_{17} \geq 304$ and $a_{34} \leq 0.04$ then d=Neptune, POD	100
22	If $a_{16} \geq 245$ and $a_{6} \geq 3634$ then d=Neptune, UDP Storm	100
23	If $a_{14} \geq 76$ and $a_{22} \geq 1$ then d=Neptune	100
24	If $a_{34} \geq 0.61$ and $a_{19} \leq 148$ then d=Neptune	100

B. Comparison with different approach

In this section, we compare results of proposed model with five different models such as resilient back propagation (RBP) [11], markov chain model (MCM) [6], radial basis function (RBF) [5], resistant architecture model (RAM) [8], and wavelet transform model (WTM) [9]. Unlike Table XV, the computation is carried out for each case across each technique. The following TABLE XVI presents the comparative analysis of all the techniques mentioned above. The accuracy of the proposed model over the KDD cup dataset is 99.76 whereas

TABLE VII: Selected list of UDP strom rules

Rule No.	Description	Acc.
1	If $a_{33} \geq 42$ then d=UDP Strom, Smurf	100
2	If $a_6 \geq 42$ then d=UDP Strom, POD	99.87
3	If $a_{35} \geq 0.28$ then d=UDP Strom, Neptune	100
4	If $a_{37} \geq 1$ then d=UDP Strom	100
5	If $a_{40} \geq 0$ then d=UDP Strom, Back	100
6	If $a_{41} \leq 0.25$ then d=UDP Strom, Smurf, Back	99.87
7	If $a_{14} \geq 7$ then d=UDP Strom, Back	100
8	If $a_{16} \geq 40$ then d=UDP Strom, Land	100
9	If $a_{17} \leq 40$ then d=UDP Strom, POD	99.87
10	If $a_{20} \geq 1$ then d=UDP Strom, Teardrop	100
11	If $a_{33} \geq 253$ then d=UDP Strom	100
12	If $a_6 \geq 40$ and $a_{14} \leq 40$ then d=UDP Strom	100
13	If $a_{21} \geq 1$ and $a_{33} \leq 255$ then d=UDP Strom	100
14	If $a_6 \geq 40$ and $a_{33} \geq 7$ then d=UDP Strom	100
15	If $a_{33} \geq 77$ and $a_6 \geq 0$ then d=UDP Strom	100

TABLE VIII: Selected list of smurf rules

Rule No.	Description	Acc.
1	If $a_{40} \geq 0.31$ then d=Smurf, UDP Storm	100
2	If $a_{41} \geq 0.14$ then d=Smurf, POD	100
3	If $a_{13} \geq 23$ then d=Smurf	100
4	If $a_{14} \geq 30$ then d=Smurf, Neptune	100
5	If $a_{16} \geq 93$ then d=Smurf, Back	100
6	If $a_{17} \geq 64$ then d=Smurf, Teardrop	100
7	If $a_{19} \geq 185$ then d=Smurf	100
8	If $a_{40} \geq 0.31$ and $a_{41} \geq 0.14$ then d=Smurf	100
9	If $a_{41} \leq 0.14$ and $a_{13} \geq 23$ then d=Smurf	100
10	If $a_{14} \geq 30$ and $a_{16} \geq 93$ then d=Smurf	100
11	If $a_{14} \geq 30$ and $a_{19} \geq 185$ then d=Smurf	100
12	If $a_{17} \geq 64$ and $a_{19} \geq 185$ then d=Smurf	100
13	If $a_{16} \leq 93$ and $a_{13} \geq 23$ then d=Smurf	100
14	If $a_{19} \geq 185$ and $a_{16} \geq 93$ then d=Smurf	100
15	If $a_{41} \leq 0.14$ and $a_{19} \geq 185$ then d=Smurf	100

TABLE IX: Selected list of POD rules

Rule No.	Description	Acc.
1	If $a_{33} \geq 829$ then d=POD, Smurf	100
2	If $a_{34} \geq 0.32$ then d=POD, Back	100
3	If $a_{35} \geq 0.08$ then d=POD, Neptune	100
4	If $a_{37} \geq 0.11$ then d=POD	100
5	If $a_{40} \geq 0.47$ then d=POD, Land	100
6	If $a_{41} \geq 0.03$ then d=POD	100
7	If $a_{33} \geq 829$ and $a_{34} \leq 0.32$ then d=POD	99.45
8	If $a_{33} \geq 829$ and $a_{35} \geq 0.08$ then d=POD	100
9	If $a_{40} \geq 0$ and $a_{34} \leq 0.32$ then d=POD	100
10	If $a_{40} \geq 0$ and $a_{35} \geq 0.08$ then d=POD	100
11	If $a_{37} \geq 0.11$ and $a_{34} \leq 0.32$ then d=POD	100
12	If $a_{37} \geq 0.11$ and $a_{35} \geq 0.08$ then d=POD	100
13	If $a_{33} \leq 829$ and $a_{40} \geq 0$ then d=POD	100
14	If $a_{37} \leq 0.11$ and $a_{34} \leq 0.32$ then d=POD	100
15	If $a_{37} \leq 0.11$ and $a_{35} \geq 0.08$ then d=POD	100
16	If $a_{34} \geq 0.32$ and $a_{41} \geq 0.03$ then d=POD	100
17	If $a_{35} \geq 0.08$ and $a_{34} \geq 0.32$ then d=POD	100
18	If $a_{34} \leq 0.32$ and $a_{35} \geq 0.08$ and $a_{33} \geq 829$ then d=POD	99.45

TABLE X: Selected list of back rules

Rule No.	Description	Acc.
1	If $a_{13} \geq 105$ then d=Back, POD	100
2	If $a_{14} \geq 146$ then d=Back, Land	100
3	If $a_{16} \geq 6$ then d=Back, Process table	100
4	If $a_{17} \geq 20$ then d=Back, Mailbomb	100
5	If $a_{19} \geq 1032$ then d=Back	100
6	If $a_{20} \geq 7$ then d=Back, Land	100
7	If $a_{13} \geq 105$ and $a_{14} \geq 146$ then d=Back	100
8	If $a_{16} \geq 6$ and $a_{13} \geq 105$ then d=Back	100
9	If $a_{17} \geq 20$ and $a_{14} \geq 146$ then d=Back	100
10	If $a_{19} \geq 1032$ and $a_{20} \leq 7$ then d=Back	100
11	If $a_{20} \geq 7$ and $a_{14} \geq 146$ then d=Back	100
12	If $a_{17} \leq 20$ and $a_{13} \geq 105$ then d=Back	100

TABLE XI: Selected list of teardrop rules

Rule No.	Description	Acc.
1	If $a_{40} \geq 0.52$ then d=Teardrop, Back	100
2	If $a_{41} \geq 0.51$ then d=Teardrop, Neptune	100
3	If $a_{33} \geq 20$ then d=Teardrop	100
4	If $a_{37} \geq 0.17$ then d=Teardrop, Land	100
5	If $a_{40} \geq 0.52$ and $a_{33} \geq 20$ then d=Teardrop, Land	100
6	If $a_{40} \geq 0.52$ and $a_{37} \geq 0.17$ then d=Teardrop, POD	100
7	If $a_{41} \geq 0.51$ and $a_{33} \geq 20$ then d=Teardrop	100
8	If $a_{41} \geq 0.51$ and $a_{37} \leq 0.17$ then d=Teardrop	100
9	If $a_6 \leq 520$ and $a_{35} \leq 0.20$ then d=Teardrop	100
10	If $a_6 \geq 520$ and $a_{34} \leq 0.17$ then d=Teardrop	100

TABLE XII: Selected list of land rules

Rule No.	Description	Acc.
1	If $a_{22} \geq 1$ then d=Land, Teardrop	100
2	If $a_{21} \geq 1$ then d=Land, Back	100
3	If $a_{20} \geq 79$ then d=Land, POD	99.97
4	If $a_{16} \geq 18$ then d=Land, Smurf	100
5	If $a_6 \geq 511$ and $a_6 \geq 145$ then d=Land	100
6	If $a_{40} \geq 0.51$ and $a_{41} \leq 0.79$ then d=Land, Mailbomb	100
7	If $a_6 \geq 145$ and $a_{34} \geq 0.18$ then d=Land	99.97
8	If $a_{22} \geq 1$ and $a_{33} \leq 18$ then d=Land	100

TABLE XIII: Selected list of mailbomb rules

Rule No.	Description	Acc.
1	If $a_{16} \geq 1000$ then d=Mailbomb, Land	100
2	If $a_6 \geq 1024$ then d=Mailbomb, Process table	100
3	If $a_{33} \geq 7$ then d=Mailbomb, Back, Land	100
4	If $a_{34} \geq 0.25$ then d=Mailbomb, Smurf	100
5	If $a_{33} \geq 114$ then d=Mailbomb, Neptune	100
6	If $a_{16} \geq 1000$ and $a_{33} \geq 7$ then d=Mailbomb	100
7	If $a_{16} \leq 1000$ and $a_{34} \geq 0.25$ then d=Mailbomb	100
8	If $a_6 \geq 1024$ and $a_{33} \geq 114$ then d=Mailbomb, Process table	100
9	If $a_6 \geq 1024$ and $a_{34} \geq 0.25$ then d=Mailbomb	100
10	If $a_{16} \leq 1000$ and $a_{33} \geq 114$ then d=Mailbomb, Land	100

TABLE XIV: Selected list of process table rules

Rule No.	Description	Acc.
1	If $a_{37} \geq 0.10$ then d=Process table, Mailbomb	100
2	If $a_{33} \geq 224$ then d=Process table, Back	100
3	If $a_{37} \geq 0.10$ and $a_4 \geq 0$ then d=Process table, POD	100
4	If $a_{33} \geq 224$ and $a_4 \geq 0$ then d=Process table, Smurf	100
5	If $a_{22} \geq 0$ and $a_6 \geq 1024$ then d=Process table	100
6	If $a_{13} \geq 224$ and $a_{19} \geq 1024$ then d=Process table, Mailbomb	100
7	If $a_{37} \leq 0.10$ and $a_{35} \geq 0.10$ then d=Process table	100
8	If $a_{33} \leq 224$ and $a_{37} \geq 0.10$ then d=Process table	100

TABLE XV: Precision, recall, accuracy of denial-of-service attack

S. No.	Descr.	T_P	F_N	F_P	T_N	Prec.	Rec.	Acc.
1	normal	1360	3	10	3649	0.99	1	99.74
2	neptune	1,258	2	10	3752	0.99	1	99.76
3	udp strom	465	5	8	4544	0.98	0.99	99.74
4	smurf	556	8	15	4443	0.97	0.99	99.54
5	pod	605	6	9	4342	0.99	0.99	99.70
6	back	21	2	1	4998	0.95	0.91	99.94
7	tear drop	26	1	2	4993	0.92	0.96	99.94
8	land	139	8	3	4872	0.99	0.95	99.78
9	mail bomb	238	7	5	4772	0.98	0.97	99.76
10	process table	175	7	7	4833	0.96	0.96	99.72
	Total	4903	49	70	45198	0.99	0.99	99.76

the accuracy of the RBP model over the same dataset is 99.35. It indicates that the accuracy of the proposed model is 0.41 higher than the RBP model. For better visualization, a graphical representation of the comparative analysis is shown in Figure 4. Figure 5 depicts the number of rules generated, number of rules discarded, and the number of rules finally selected for each class. The total number of rules generated are 169, and 18% number of rules are discarded through validation. This results the number of rules minimized to 82%.

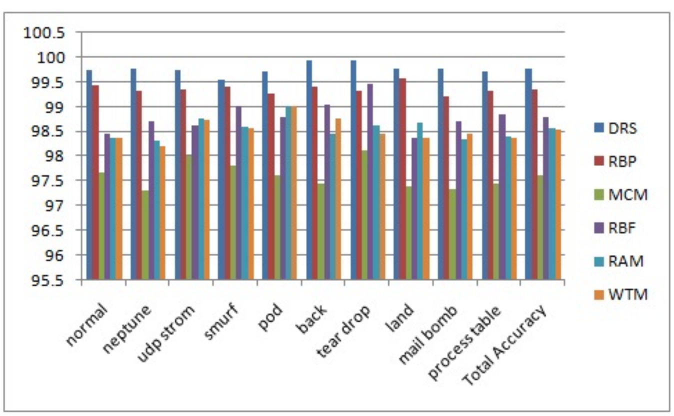

Fig. 4: Graphical Presentation of Comparative Analysis

TABLE XVI: Comparative analysis

S. No.	Descr.	DRS Acc.	RBP Acc.	MCM Acc.	RBF Acc.	RAM Acc.	WTM Acc.
1	normal	99.74	99.42	97.67	98.45	98.37	98.36
2	neptune	99.76	99.32	97.30	98.71	98.31	98.21
3	udp strom	99.74	99.35	98.00	98.63	98.75	98.74
4	smurf	99.54	99.41	97.79	99.01	98.59	98.57
5	pod	99.70	99.27	97.61	98.79	99.01	99.00
6	back	99.94	99.40	97.45	99.04	98.45	98.77
7	tear drop	99.94	99.31	98.11	99.45	98.63	98.45
8	land	99.78	99.56	97.38	98.37	98.67	98.37
9	mail bomb	99.76	99.22	97.32	98.71	98.35	98.44
10	process table	99.72	99.32	97.43	98.84	98.38	98.37
	Total Acc.	99.76	99.35	97.60	98.80	98.55	98.53

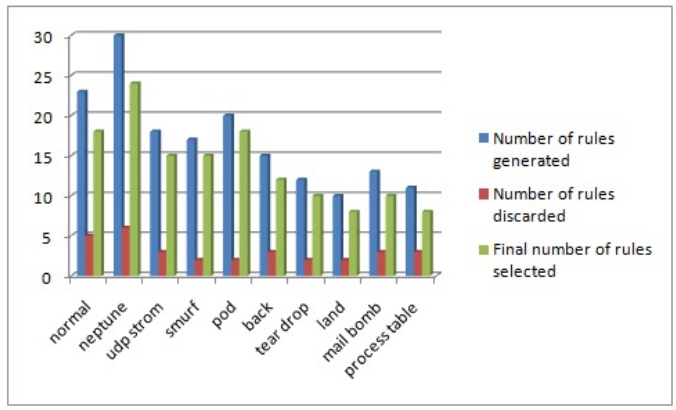

Fig. 5: Graphical view of numbers of rules selected

VI. Conclusion

Denial-of-service attack is one of the key security threats in wireless networks. Defending against DoS attack is of prime importance for industries, and internet service providers. To overcome this attack many techniques are proposed by various researchers [5, 6, 8, 9, 11]. In this paper, we propose a model for the detection of denial of service attack in wireless networks using dominance based rough set. The proposed model is analyzed with the help of KDD cup dataset. The total number of rules generated are 169, and 18% number of rules are discarded through validation. This results the number of rules minimized to 82%. Additionally, it is compared with existing techniques and found better accuracy. The accuracy of the proposed model is 99.76 whereas the accuracy of the RBP model is 99.35. This shows that the proposed model is 0.41 higher than the RBP model.

References

[1] X. Chuiyi, Z. Yizhi, B. Yuan, L. Shuoshan and X. Qin, *A distributed intrusion detection system against flooding denial-of-services attacks*, International Conference on Artificial Computing Technology, 2011, pp. 878-881.

[2] X. Ren, *(2009)Intrusion detection method using protocol classification and rough set based support vector machine*, Computer and Information Science, 2 (2009), pp. 100-108.

[3] R. C. Chen, K. F. Cheng and C. F. Hsieh, *Using rough set and support vector machine for network intrusion detection*, International Journal of Network Security and Its Applications, 1 (2009), pp. 1-13.

[4] Y. Wang, *Analysis of a distributed denial of service*, Computer Network, Elsevier, 41 (2009), pp. 1200-1210.

[5] D. Gavrilis and E. Dermatas, *Real-time detection of distributed denial-of-service attacks using RBF networks and statistical features*, Computer Network, Elsevier, 48 (2005), pp. 235-245.

[6] Y. Wang, C. Lin, Q. L. Li and Y. Fang, *A queueing analysis for the denial of service (DoS) attacks in computer networks*, Computer Networks, Elsevier, 51 (2007), pp. 3564-3573.

[7] E. Gelenbe and G. Loukas, *A self-aware approach to denial of service defence*, Computer Networks, Elsevier, 51 (2007), pp. 1299-1314.

[8] P. Mell, D. Marks and M. McLarnon, *A denial-of-service resistant intrusion detection architecture*, Computer Networks, Elsevier, 34 (2000), pp. 641-658.

[9] M. Hamdi and N. Boudriga, *Detecting denial-of-service attacks using the wavelet transform*, Computer Networks, Elsevier, 30 (2007), pp. 3203-3213.

[10] S. Chen, Y. Tang and W. Du, *Stateful DDoS attacks and targeted filtering*, Journal of Network and Computer Applications, Elsevier, 30 (2007), pp. 823-840.

[11] P. A. R. Kumar and S. Selvakumar, *Distributed denial of service attack detection using an ensemble of neural classifier*, Computer Communication, Elsevier, 30 (2011), pp. 1328-1341.

[12] Z. Pawlak, *Rough set: theoretical aspects of reasoning about data*, Springer+Business Media, Springer, 1 (1991).

[13] S. Greco, B. Matarazzo and R. Slowinski, *The use of rough sets and fuzzy sets in MCDM*, In T. Gal, T. Stewart and T. Hanne (eds.) Advances in Multiple Criteria Decision Making, chapter 14, Kluwer Academic Publishers, 1999, pp. 14.1-14.59.

[14] Z. Pawlak, *Rough Sets*, International Journal of Computer and Information Sciences, 11 (1982), pp. 341-356.

[15] D. B. Parker, *Demonstrating the elements of information security with treats*, Proceeding of the 17th National Computer Security Conference, 1994, pp. 421-430.

[16] S. Greco, B. Matarazzo and R. Slowinski, J. Stefanowski, *An algorithm for induction of decision rules consistent with the dominance principle*, European Journalof Operational Research, 117 (1999), pp. 63-83.

[17] J. Blaszczynski, S. Greco, B. Matarazzo, R. Slowinski and M. Szelag, *Dominance-based rough set data analysis framework*, Users Guide, pp. 1-19.

[18] J. W. Grzymala-Busse, *LERS - a system for learning from examples based on rough sets*. In R. Slowinski (eds.) Intelligent Decision Support. Handbook of Applications and Advances of the Rough Sets Theory, Kluwer Academic Publishers, Dordrecht, 1992, pp. 3-18.

[19] J. Komorowski, Z. Pawlak, L. Polkowski and A. Skowron, *Rough Sets: tutorial*. In A. Skowron (eds.) Rough Fuzzy Hybridization. A new trend in decision making, Springer Verlag, Singapore, 1999, pp. 3-98.

[20] J. Komorowski, *On rough set based approachesto induction of decision rules*. In L. Polkowski, A. Skowron (eds.) Rough Sets in Data Mining and Knowledge Discovery, Physica-Verlag, 1 (1998), pp. 500-529.

[21] A. Chonka, W. Zhou, J. Singh and Y. Xiang, *Detecting and tracing distributed denial-of-service attacks by intelligent decision prototype*, International Conference on Pervasive Computing and Communications, 1994, pp. 421-430.

[22] J. Yuan and K. Mills, *Monitoring the macroscopic effect of distributed denial-of-service flooding attack*, IEEE Transactions on Dependable and Secure Computing, 2 (2005), pp. 1-12.

[23] T. Peng, C. Leckie and K. Ramamohanarao, *Survey of network-based defense mechanisms countering the DoS and DDoS problems*, ACM Computing Surveys, 39 (2007), pp. 370-373.

[24] Y. Qing, W. Xiaoping, L. Yongqing and H. Gaofeng, *A hybrid model of RST and DST with its applications in intrusion detection*. International Symposium on Intelligent Information Technology and Security Information, 2010, pp. 202-205.

[25] R. Shanmugavadivu and N. Nagarajan, *Network intrusion detection system using fuzzy logic*. Indian Journal of Computer Science and Engineering, 2 (2011), pp. 101-111.