ISSN 2319-345X
Vol. 4, No. 1, January 2015

IJMRBS

# International Journal of Management Research and Business Strategy

www.ijmrbs.com

MEGHANA PUBLICATIONS
www.meghanapublications.com

International Journal of Management Research and Business Strategy

ISSN 2319-345X www.ijmrbs.com
Vol. 4, No. 1, January 2015


# AN EFFECTIVE WAY OF PROTECTION AGAINST SHREW ATTACK IN COMPUTER NETWORKS

**Akshay Keshwatkar[1]*, Vishwa V[1] and Arvind Anand[1]**

*Corresponding Author:***Akshay Keshwatkar** ✉ *akshay.b.keshwatkar@gmail.com*

Shrew attack is a low rate burst which reduces the bandwidth of TCP flow and affects the performance of all TCP based protocols. A method has been suggested called as a shrew attack protection model that isolates attackers and identifies victims by keeping track of their drop rates and admit those packets which are having high priority for the queue. Shrew Attack Protection (SAP) has previously been demonstrated using network simulator ensuring the TCP sessions retain their bandwidth shares. In this paper the implementation of shrew attack model is done with router and switches and the results are analyzed to validate the performance of improvement of bandwidth after network attack.

***Keywords:*** Shrew Attack, TCP, Bandwidth, Priority

## INTRODUCTION

Denial-of-Service (DoS) attack tends to make the network resources available by flooding a huge volume of traffic. An attack that makes use of the Retransmission Time Out (RTO) mechanism in TCP is referred as a Shrew attack or a Reduction of Quality (RoQ) attack. Consecutive packet drops in the TCP flow is a result of the retransmission of packets just after expiration of the retransmission timer. Consequently, this synchronized attack results in the packet reaching the router that is already flooded with the synchronized burst. The attack can quite easily be designed to eat up the bandwidth both in wired as well as wireless networks and these attacks are detected by generally examining the frequency domain characteristics of incoming traffic flows to a server (Yu Chen and Kai Hwang, 2006). Moreover, their signals processing approach uses the properties of periodic and non-periodic signal and were successful in the detection of shrew attack within few seconds. However, they could not find a solution. Another approach went on to identify malicious shrew packets using small flow table and dropping such packets decisively to halt the attack such that well behaved TCP connection can regain their bandwidth (Kwok *et al*., 2005). But, their solution was insensitive to distributed shrew attack that occupies small bandwidth. Another approach that detects shrew attack flows hidden in legitimate TCP/UDP streams by spectral analysis against

[1] Department of Computer Science and Engineering, Velammal Engineering College, Chennai, Tamil Nadu, India.

pre-stored template of average attack spectral characteristics was suitable for either software or hardware implementation (Yu *et al.,* 2005) and this scheme is implemented using the NS-2 network simulator using real-life Internet background traffic mixed with attack datasets used by established research groups. Simulated results show high detection accuracy by merging alerts from cooperative routers. Both theoretical modelling and simulation experimental results were reported. The defence against shrew attack was investigated in two levels, viz., router, IP Tables and at the application layer level (Dilum *et al.,* 2007). However, they did not give a complete solution. A router-based technique was investigated to mitigate RoQ attacks and it is a scalable technique that passively detects the low rate DoS and RoQ attacks (Amey Shevtekar and Nirwan Ansari, 2008). After confirming the onset of a RoQ attack, their filtering algorithm is enabled to separate long-lived legitimate flows at the router, and subsequently to drop the RoQ attack packets. This filtering algorithm is best suited for RoQ attacks. In order to reduce the attack a simple protection mechanism, called Shrew Attack Protection (SAP), for defending against a shrew attack is developed which attempts to track and isolate SAP identifiers, Shrew Attackers, TCP victims by monitoring their drop rates and preferentially admits those packets from the victims with high drop rates to the output queue (Chia-Wei Chang *et al*., xxxx). This is to ensure that well-behaved TCP sessions can retain their bandwidth shares. SAP model requires only a small number of counters to find victims; this makes SAP ready to implement on the top of existing router mechanisms. In our paper the shrew attack is prevented by hardware implementation using a small number of counters which can handle huge number of packets in today's networks.

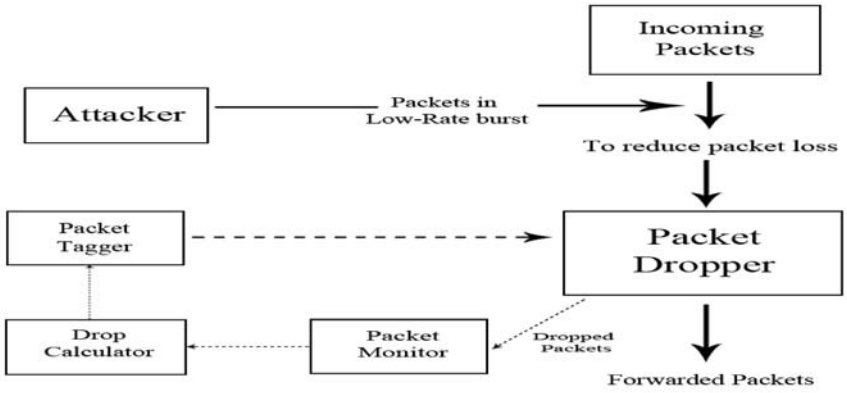
The rest of the paper is organized as follows. Section II presents discussion about Shrew Attack, Section III presents proposed work, Section IV presents implementation and Section V presents Results and Section VI represent Future Enhancement.

## SHREW ATTACK

The Internet is a global system of interconnected computer networks that use the standard Internet Protocol Suite (TCP/IP) to serve billions of users worldwide. It consists of different type topology to connect between various sources. Due to traffic or any other technical reason most of the time the data are lost in transmission. One of the error types in communication is packet loss which is due to dropping of packet by any means which may lead to attacks in TCP connection.

Packet drops are heavy in edges than in internal nodes that lead to shrew attack which uses a low-rate burst carefully designed to exploit TCP's retransmission time-out mechanism. It can throttle bandwidth of a TCP flow in a hidden manner (Amey Shevtekar and Nirwan Ansari, 2008). The Shrew attack is acts as low-rate attack which consists of short, maliciously chosen duration bursts of packets that repeat with a fixed, maliciously chosen, slow-time-scale frequency (Amey Shevtekar and Nirwan Ansari, 2008). Traffic pattern is carefully designed to exploit TCP retransmission timeout mechanism, when multiplex this with TCP and cross-traffic, this pattern will be able to throttle TCP's flow in internet.

**Figure 1: Shrew Attack Model**




## PROPOSED WORK

Protection against Shrew Attack Model nullifies a shrew attack by controlling the packet drop rates at application level by tagging them with priority. Figure 1 shows SAM as an architectural model which is used to defend against the shrew attack rather than attempting to detect and isolate. A SAM component includes Packet Tagger, Drop Calculator, Packet Monitor and Packet Dropper. First one is Packet Tagger which tagged the packet with either high or low priority. The role of the drop calculator component is to monitor drop rates. Given the historical drop rates of all ports collected, the role of the Packet Monitor is to determine the fair drop rate threshold *pf* that it wants to limit. Packet Tagger component perform tagging according to the determined fair drop rate.

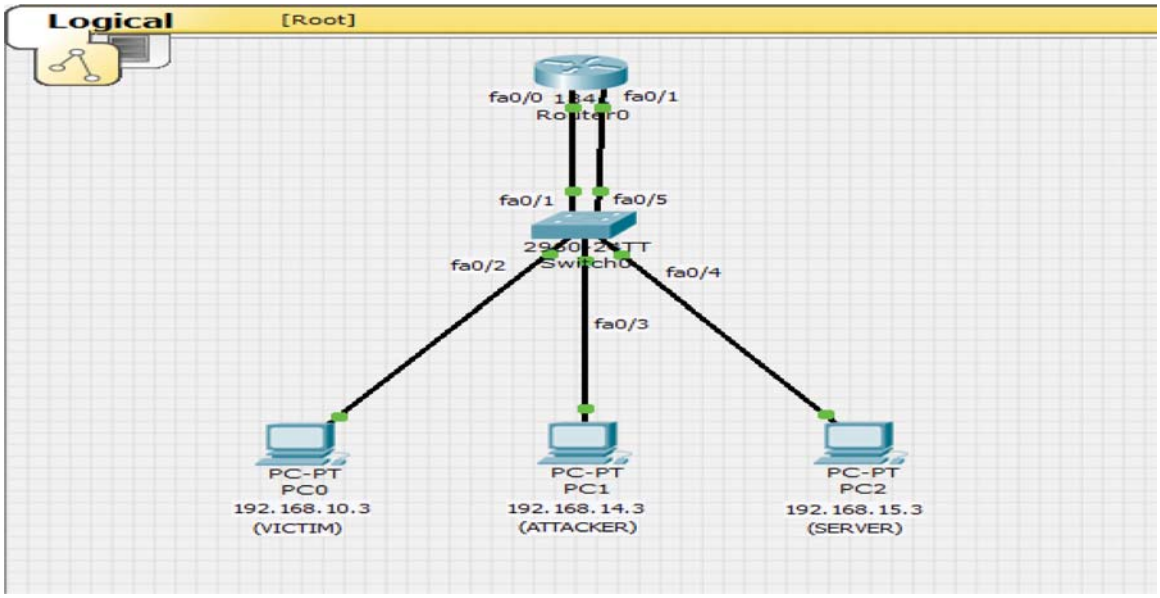
Till now researches have been shown mitigating the attack in inner router where the packet-loss is very less. In this paper SAM is implemented in edges to reduce the packet-loss and further stop the TCP connection from closing. SAM mechanism requires only a small number of counters to find potential victims, which makes SAM readily implementable on top of existing router mechanisms.

## IMPLEMENTATION

The modules incorporated and integrated for SAM model are: (1) Creation of Networks; (2) Implementing the Attack; (3) SAM components which includes Packet Tagger, Drop Calculator, Packet Monitor and Packet Dropper.

A Network is created with the required topology of having CISCO Router and Switch, PCs, VLANs and Trunk port. Virtual Lan consists of a group of hosts with a common set of requirements that communicate as if they were attached to the same broadcast domain, regardless of their physical location. Connection on a switch that is used to connect a guest virtual machine consisting of VLAN and it has same attributes as that of a physical local area network and it allows for end stations to be grouped together even if they are not located on the same network switch configured through software instead of physically relocating devices. VLAN information is never passed between the switches. T50 Sukhoi PAK FA Mixed Packet Injector (f.k.a. F22 Raptor) is used to implement the attack. The topology used is depicted in Figure 2. It consists of 3 VLANs and 2 interfaces.

**Figure 2: Network Topology**




## SAM AND ITS COMPONENTS

This module is the core of the project where the SAM Architecture is implemented which prevents the Shrew attack in any network and SAM consists of four components. First one is Packet Tagger which tags the packet with either high or low priority. The second component being packet dropper is used to drop the low priority packet on arrival. For this dropping scheme drop calculator is used to keep count of dropping and Packet Monitor which used to fix the threshold drop rate. Two sets of counters are available for each application port: one set for arrivals, and the other set for drops. The values of these counters are in terms of progressing bytes (Chia-Wei Chang *et al*., 2009).

**Drop Calculator:** The role of the drop Calculator component is to monitor drop rates. Counters are maintained for each TCP flow: one set for Arrivals, and the other set for Drops. The values of these counters are in terms of progressing bytes.

**Packet Monitor:** Given the historical drop rates of all ports collected, the role of the Packet Monitor is to determine the fair drop rate threshold pf that it wants to limit. This drop rate threshold calculation actually depends on the average total drop rate *pa*. This average drop rate *pa* will be updated every ts intervals and will consider the previous values by using the formula:

$$Pa = \frac{\sum_{i=1}^{i=F} \Delta di(t)}{\sum_{i=1}^{i=F} \Delta ai(t)} \quad ...(1)$$

where $\Delta di(t)$ and $\Delta ai(t)$ are the successive arrival and drop counts and F is the number of flows.

**Packet Tagger:** The packet monitor takes one more parameters, pmin. The parameter pmin specifies a minimum drop rate threshold, under which SAM does not intervene. More specifically, if pa ≥ pmin, then the SAM uses the current average drop rate to serve as the fair drop rate by fixing pf = pa. If pa ≤ pmin, then the SAM will fix pf = pmin. Given the drop rate limits determined

by the packet monitor, the role of the packet tagger component is to perform the tagging according to the determined fair drop rate also in particular the arrival packets are always given high priority if the instant drop rate is higher than the fair drop rate threshold set by the packet monitor. Otherwise, they are tagged as low priority

**Access Control List:** Access Control List (ACL's) are filters that controls the routing updates or packets that are to be permitted or denied that flows inside or outside of a network and they are solely used by network administrators to filter the traffic and to provide extra security for their networks. This can be applied on routers (Cisco). ACLs provide a powerful way to control traffic into and out of your network; this control can be as simple as permitting or denying network hosts or addresses. ACLs can be configured for all routed network protocols. The most important reason to configure ACLs is to provide security for network. However, ACLs can also be configured to control network traffic based on the TCP port being used. Drop rates of particular system are calculated using ACL (TCP/IP Illustrated, 1994; http://www.cisco.com/en/US/products/ps5875/index.html; http://www.cisco.com/en/US/products/ps6406/index.html; http://www.cisco.com/en/US/tech/tk389/tk815/technologies_configuration_example09186a00800949fd.shtml; http://www.tcl.tk/man/tcl/tutorial/tcltutorial.html)

**Process ACLs:** Traffic that comes into the router is compared to ACL entries based on the order that the entries occur in router and new statements are added generally to the end of list. The router continues to look until it has match and if no matches are found whenever the router will reach the end of the list, the traffic is always denied. This is the reason that the frequently hit entries are always added at the top of the entire list. There is generally a deny for traffic that is never permitted. A single entry ACL which has only one deny entry will hve the effect of denying all traffic. There should be at least one permit statement in an ACL or all traffic is blocked. These two ACLs 101 and 102 have the same effect.

!— This command is used to permit IP the traffic from 11.1.1.0

!— Network to 172.14.3.0 network. All the packets with a source

access-list 101 permit ip 11.1.1.0 0.0.0.255 172.14.3.0 0.0.0.255

!— This command is used to permit IP the traffic from 11.1.1.0

!— Network to 172.14.3.0 network. All the packets with a source

!— Address which is not in this range will be rejected always.

Access-list 102 permit the ip 11.1.1.0 0.0.0.255 172.14.3.0 0.0.0.255

Access-list 102 deny ip any

In this example, the last entry is absolutely sufficient moreover you do not need these first three entries because TCP also includes Telnet and the IP always includes TCP also the User Datagram Protocol (UDP) along with Internet Control Message. Protocol (ICMP).

!— This command is always used to permit the Telnet traffic

!— From the machine 11.1.1.2 to machine 172.14.3.1.

Access list 101 permit tcp host 11.1.1.2 host 172.14.3.1 eq telnet

!— This command is always used to permit the tcp traffic from this

!— 11.1.1.2 host machine to 172.14.3.1 host machine.

Access list 101 permit tcp host 11.1.1.2 host 172.14.3.1

!— This command always is used to permit the udp traffic from this

!— 11.1.1.2 host machine to 172.14.3.1 host machine.

Access list 101 permit udp host 11.1.1.2 host 172.14.3.1

!— This command is used to permit the IP traffic from this

!— 11.1.1.0 network to 172.14.3.0 network.

Access list 101 permit ip 11.1.1.0 0.0.0.255 172.14.3.0 0.0.0.255

**Apply ACLs:** ACL doesn't show any effect until they are applied to the router's interface. It will be good if we apply the ACL on the interface which is close to the source of the traffic.

As shown in Figure 3, when tried to block traffic from source to destination, we can apply an incoming ACL to C0 on the router A instead of an outgoing list to C1 on router C. If traffic is related to a DHCP request and if it is not permitted explicitly, then the traffic is dropped. When you have a close look at DHCP request in IP and destination address is 255.255.255.255 and source address is 0.0.0.0 (Ethernet1/0), len 604, rcvd 2 UDP dst=67, src=68. Note that the destination port is 68 and source 67. Hence, always permit this kind of traffic in the access-list otherwise the traffic will drop due to deny statement. Tool Command Language (TCL) is a Scripting language used for configuring switches and routers.

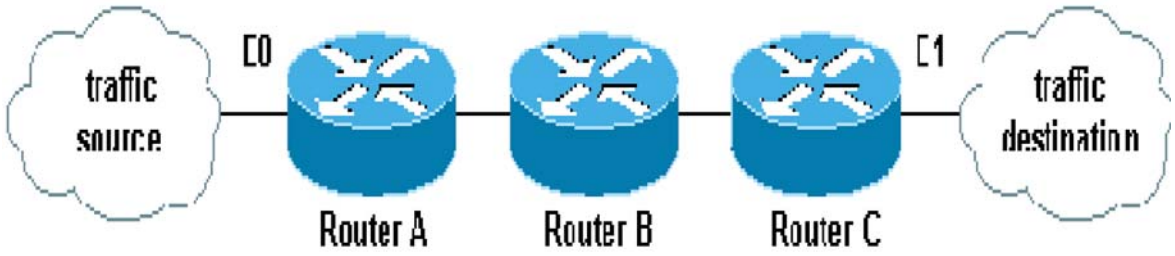
The input are:-

192.168.10.3

192.168.14.3

Server ip : 192.168.15.3

**Figure 3: Sample Network-To Explain ACL**




| Drop Rate Tabe | | | |
|---|---|---|---|
| IP Address | I/P Packet | O/P Packet | Drop Rate |
| 192.168.10.3 | 121 | 110 | 9.09090909091 |

| 192.168.14.3 0 .0 No Transmission | | |
|---|---|---|
| IP Address | Fair Drop Rate | Current Drop rate |
| 192.168.10.3 | 6.2466051059 | 9.0909090909 |
| 192.168.14.3 | 99.8643198928 | -100 |

Creating access control lists

Calculating drop rates

Logging the drop rates

File name: file_fdr.txt

Determining current drop rate

Dropping 192.168.14.3

Shut on work for 192.168.14.3

Revoking drops 192.168.14.3

Allowing packets from 192.168.14.3

## RESULTS

In this module performance result is used for comparison to show how the theoretical model (SAM) increases the throughput and performance of the TCP connection without closing the connection due to packet-loss. Figure 4 shows the network usage after attack in which the bandwidth utilization is reduced and Figure 5 displays the scenario of the attack with a Network utilization of 49% and the scenario of rise in bandwidth utilization is shown in Figure 6 after shrew attack model is implemented.

**Figure 4: Network Usage After Attack**

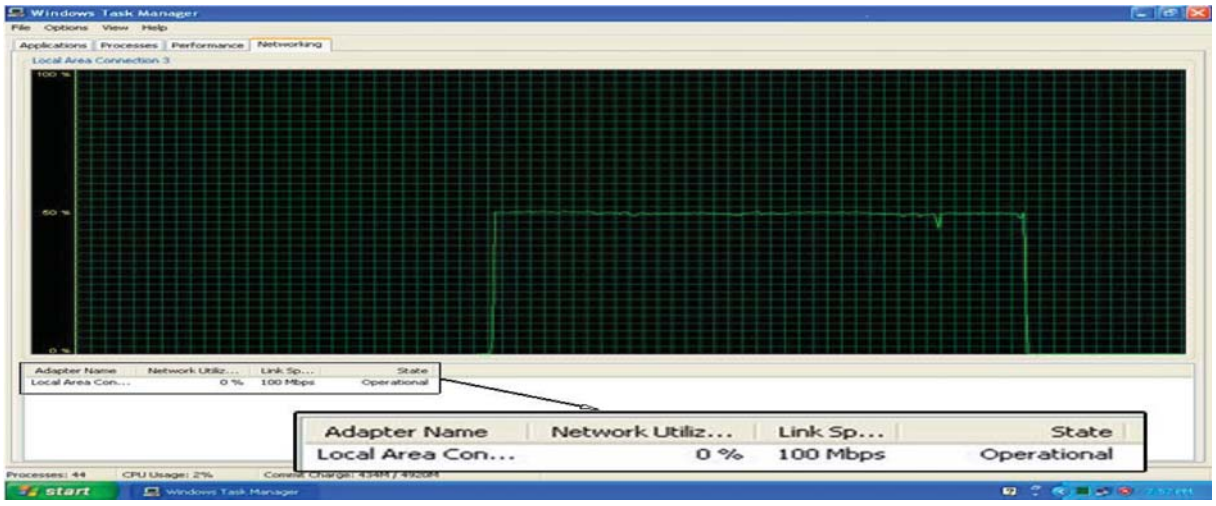



**Figure 5: Network Usage During Attack**

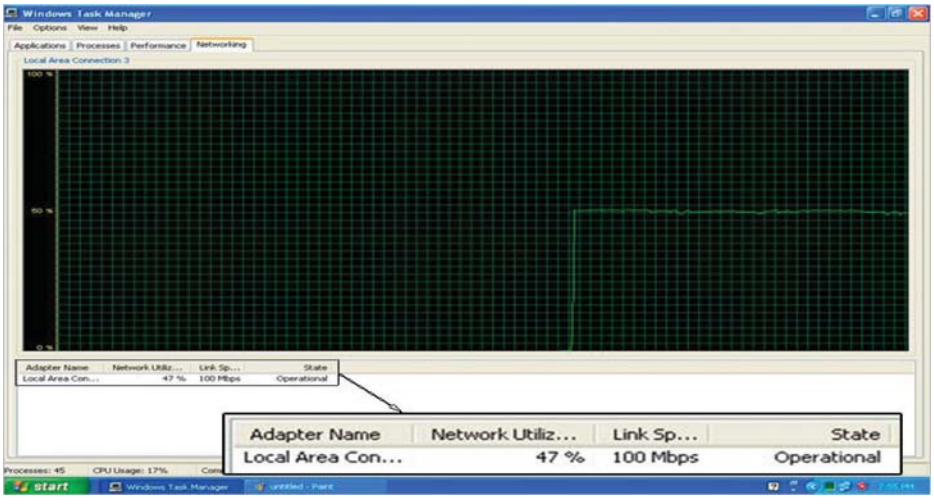
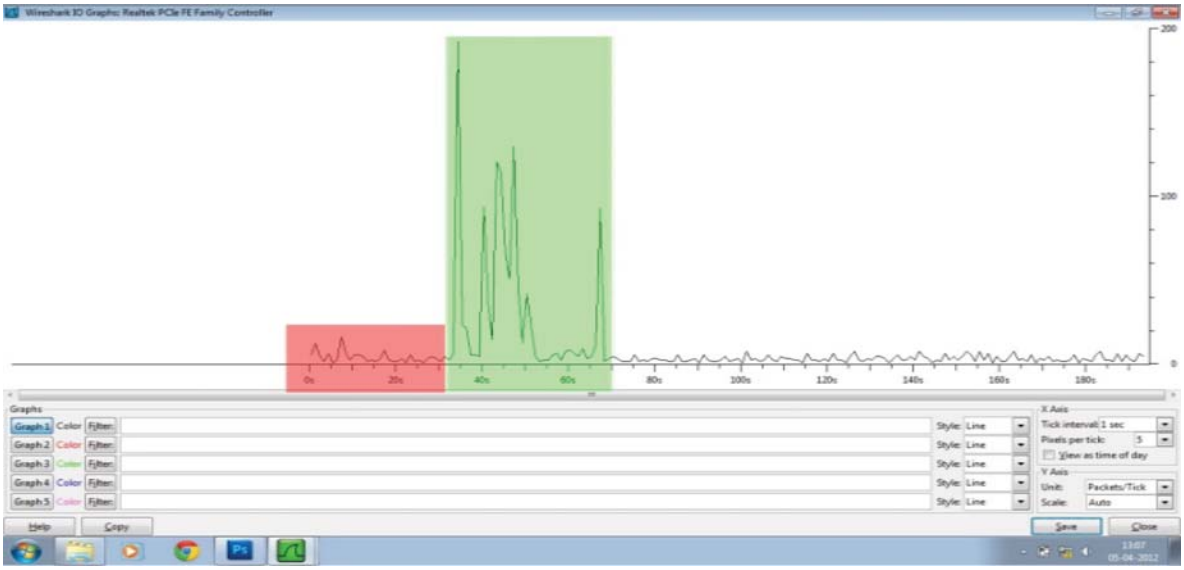


**Figure 6: Victim's Network Graph**



| Table 1: Performance Results | |
|---|---|
| Scenario | Drop rate (%) |
| Without attack(normal network conditions) | 2.00803212851 |
| During attack | 99.86541 |
| When SAM is active | 1.36054421769 |

Input: Using SAM, traffic pattern shows the attack.

Output: Performance comparison without and with using SAM.

The performance comparison is depicted in the Figure 6. The red area in the graph shows the victim's network graph during the attack. The green zone in the graph shows the rise in bandwidth used by the victim when SAM is active.

## CONCLUSION

SAM provides a mechanism to maintain the throughput by identifying the victim(s) and thereby preventing the TCP session termination. Thus the inefficient use of bandwidth due to shrew attack has been prevented. Since the script runs on router, it plays a major role in the implementation of SAM. Moreover, Routers have knowledge about LANs connected; it makes SAM easier and efficient. Best utilization of the existing network, router mechanism is guaranteed in SAM model as it does not require additional hardware/ software.

## FUTURE ENHANCEMENT

Network congestion is unpredictable and we do not know how the packets may arrive in future technology. The need to update the knowledge of the network pattern and restructure the model arises. Future research focuses on the worst case scenario when all the packets may be tagged with high priority paving way for more congestion in the queue management.

## REFERENCES

1. Amey Shevtekar, Nirwan Ansari (2008), "A router-based technique to mitigate reduction of quality (RoQ) attacks", *The International journal of Computer and Telecommunication Networking*, Vol. 52, No. 5, pp. 957-970.

2. Chia-Wei Chang, Seungjoon Lee, Bill Lin and Jia Wang (2009), "The Taming of The Shrew: Mitigating Low-Rate TCP-Targeted Attack", *IEEE transactions on network service management*, Vol. 7, No. 1.
3. Dilum Bandara H M N, Yan Chen and Douglas Tear (2007), "Towards A Secure Network: Defense vs. Attack", *Advanced topics in Networking-Network Security*.
4. http://www.cisco.com/en/US/products/ps5875/ index.html
5. http://www.cisco.com/en/US/products/ps6406/index.html
6. http://www.cisco.com/en/US/tech/tk389/tk815/technologies_configuration_example09186a 00800949fd.shtml
7. http://www.tcl.tk/man/tcl/tutorial/tcltutorial.html
8. Kwok Y-K, Tripathi R, Chen Y, and Hwang K (2005), "HAWK: Halting Anomaly with weighted ChoKing to Rescue Well-Behaved TCP Sessions from Shrew DoS Attacks".
9. TCP/IP Illustrated (1994), The Protocols, Vol. 1, Addison-Wesley,
10. Yu Chen and Kai Hwang (2006), "Collaborative detection and filtering of shrew DDoS attacks using spectral analysis".
11. Yu Chen, Yu Kwong-Kwok and Kai Hwang (2005), "Filtering shrew DDOS attack using a new frequency domain approach".

International Journal of Management Research and Business Strategy
Hyderabad, INDIA. Ph: +91-09441351700, 09059645577
E-mail: editorijmrbs@gmail.com or editor@ijmrbs.com
Website: www.ijmrbs.com