Securing Hybrid Wired/Mobile IP Networks from TCP-Flooding Based Denial-of-Service Attacks

Tarik Taleb, Hiroki Nishiyama, Nei Kato, and Yoshiaki Nemoto
Graduate School of Information Sciences - Tohoku University, Japan
Email: taleb@nemoto.ecei.tohoku.ac.jp,{bigtree,kato}@it.ecei.tohoku.ac.jp, nemoto@nemoto.ecei.tohoku.ac.jp

Abstract— **Protection of Mobile IP networks from Denial-of-Service (DoS) attacks, a serious security threat in today's Internet, is a one major step toward making this paradigm a reality. The paper proposes a method to detect DoS attacks, issued from mobile users, in the vicinity of flooding sources and in early stages before they cripple the targeted system. The fundamental challenge in attack detection consists in distinguishing between simple flash events and DoS attacks so as not to deprive innocent users from having legitimate accesses. In the proposed mechanism, this distinction is based on the fact that legitimate TCP flows obey the congestion control protocol, whereas misbehaving sources remain unresponsive. Suspicious flows are sent a test feedback and are required to decrease their sending rates. Legitimacy of such flows is decided based on their responsiveness. The scheme performance is evaluated through a set of simulations and encouraging results are obtained: short detection latency and high detection accuracy.**

I. Introduction

Along with the rapid globalization of the mobile telecommunications industry, the traffic amount of mobile computing will experience a dramatic increase. Given the constant threat of crackers and the diverse nature of data to be transported over Mobile IP networks, unwavering vigilance presents itself with respect to security threats.

Recent events have illustrated that the underlying Internet infrastructure is exposed to a high risk of denial-of-service (DoS) attacks. While DoS has been studied extensively for the wired IP networks [1] [2] [3], there is lack of research for preventing such attacks in Mobile IP networks. In the mobile context, DoS attacks are more difficult to protect against. Indeed, without a physical infrastructure, DoS mobile attackers are offered considerable flexibility in deciding where and when to attack. In addition, mobility guarantees them anonymity as the attacking sources can not be trivially tracked down.

Although the full impact of DoS attacks on the security of Mobile IP networks is yet to be felt, the seeds of these security concerns need to be considered. This challenging task underpins the research work outlined in this paper.

The remainder of this paper is structured as follows. Section II presents the complex landscape of security mechanisms in the mobile context. It highlights also the relevance of this work to the state-of-art of DoS attacks detection techniques. The proposed scheme is described in Section III. Section IV portrays the simulation environment and reports the simulation results. The paper concludes in Section V.

II. Related Work

Mobile IP offers great flexibility and potential mobility. It, however, introduces many new opportunities for launching more complex DoS attacks. Generally speaking, vulnerabilities of mobile IP networks can be classified into two broad categories: Spoofing-based attacks and flooding-based attacks.

1) Spoofing-Based Attacks: During a spoofing-based attack, an adverse node spoofs the identity of a legitimate mobile node and redirects the packets destined for the mobile node to other network locations. These attacks attempt either to deprive legitimate nodes from having access to Access Points (APs) or to grant attackers access privileges of valid clients.

In current wireless networks, spoofing-based attacks form the basis for most DoS attacks. Notable examples are AirJack [4] and Man-in-the-Middle [5]. In the latter, a rogue device authenticates with an AP as a valid user and then presents itself to other users as a viable AP. When a legitimate user attempts to authenticate with the corporate network via the rogue AP, the rogue node will tunnel the authentication session to the real network and steal the session encryption keys when they are passed between the client and the authentication server. These keys can then be used to authenticate anywhere in the network. To cope with such kind of attacks, a large body of prior research work has been done. Most of these research work focused on increasing the security level of authentication schemes. Different architectures have been considered for authentication in wireless networks (e.g. Choice [6], IEEE 802.11i [7]). Authentication is performed by using a shared secret key between mobile nodes and APs. Among authentication protocols, the Wireless Encryption Protocol (WEP) is the most widely used protocol and is specifically designed for IEEE 802.11 [8]. Despite its wide acceptance, the protocol is still insufficient as it is vulnerable to various cryptographic attacks that reveal the shared key used to encrypt and authenticate data. A wide arsenal of publicly available tools have automated these cryptographic attacks. Airsnort [9] and WEPCrack [10] are few examples.

In the case of Mobile IP networks, a spoofing-based attack occurs when an unauthorized user manages to do a bogus registration of a new Care-of-Address (CoA) for a particular mobile node. Such a bogus registration gives rise to two problems: disconnection of the legitimate mobile node and monitoring of its traffic. This type of vulnerability can be surmounted by strong authentication on all registration messages

0-7803-9415-1/05/$20.00 (C) 2005 IEEE

that are exchanged during the registration process, built onto both basic Mobile IP [11] and route optimized Mobile IP [12]. With the aid of a public key infrastructure [13], a scalable countermeasure against the spoofing attack can readily be deployed.

To further strengthen the security of authentication procedures, the IP Security protocol (IPSec) should be implemented on the IP-IP encapsulation used by Mobile IP to redirect IP datagrams to and from the mobile nodes [14]. IPSec supports a set of interoperable and cryptographically based security services such as integrity, confidentiality, and non-repudiation. In addition to these basic security properties, the IPSec protocol defines also a framework for key exchange and negotiation of security associations (SA), called Internet Security Association and Key Management Protocol (ISAKMP) [15]. Upon a handoff, the Foreign Agent (FA) should verify the identity of visiting mobile nodes either directly or indirectly via their home agents (HAs) before issuing a care-of address and permitting a successful completion of the registration. Protecting the shared secret key by applying IPSec on an end-to-end basis renders the Spoofing-based attacks almost impossible in Mobile IP networks.

2) Flooding-Based Attacks: Traditional flooding-based DoS attacks are designed to inhibit legitimate users from accessing a particular host or disrupt/degrade the performance of a network link by generating an excessive amount of data traffic (e.g. UDP Flood [16] and TCP Flood [17]). In Mobile IP networks, due to their hidden identity, hostile nodes, coming from different network administrative domains, can easily launch a flooding DoS attack against a server without being tracked down. While there has been a large body of prior work on the authentication issues in Mobile IP networks, Flooding-based DoS attacks remain largely unattended.

For wired networks, several approaches have been proposed to counter DoS attacks in the recent literature. They can be classified into two types: traceback and prevention techniques. The former commences their search for attackers upon the collapse of a victim system or a sharp degradation in performance. Most DoS traceback techniques are based on comparisons among traffic patterns or on packet marking [2]. Whilst these techniques may be efficient in terrestrial networks, they may run into difficulty in the case of Mobile IP networks. The main reason behind this limitation lies beneath the motion characteristic of end-users. Consider a scenario where a DoS attack source is roaming among different APs while flooding a victim with malicious traffic. In such a scenario, applying traceback techniques to pin the real attacker down would result in unsuccessful monitoring of traffic coming from the traversed APs. This would ultimately lead to confusing results.

Prevention techniques, on the other hand, attempts to throttle attacks before they severely harm the system. Recently proposed prevention techniques rely on monitoring changes in the internal characteristics of the network, such as the traffic volume, loss ratio, and queuing delays. Other prevention techniques watch for the behavior of specific packet types to detect DoS attacks and alert the victim. [3] is a notable example. Upon detection of a change in network characteristics, most prevention techniques take the harsh measure of shutting off the traffic destined to the victim. Such a draconian measure is unfair toward some legitimate packets that may be contained in the blocked traffic. In Mobile IP networks, given the fact that a single AP may have an extensive coverage area, such unfair event may easily occur when a potential number of legitimate users, from the same coverage area, access the same server simultaneously. Having multiple users accessing the same server at nearly the same time is referred to as "flash crowd" event throughout this paper. The challenge in this research is how to distinguish between traffic increase due to DoS attacks and that due to flash crowd events. To the author's best knowledge, this work is one of the first attempts to cope with Flooding-based DoS attacks in Mobile IP networks.

III. Feedback-Based DoS Attack Prevention Scheme

Based on a backscatter analysis, [18] has indicated that over 94% of DoS attacks use TCP packets. It has been shown also that a potential number of networks were victims of DoS attacks and had their vital links overloaded with unnecessary traffic. Knowing that flooding packets destined to a notorious UDP port can be easily identified as a DoS attack, this paper focuses on thwarting TCP-based attacks that attempt to overload servers or networks with useless traffic. The paper excludes the case of TCP-based attacks that consist of multiple TCP connections with less than three packets. The anomaly or legitimacy of such flows can be easily judged and there is a substantial set of mechanisms to cope with such ill-behaved connections in the recent literature [3] [19].

Detection is performed by monitoring agents located at Access Routers (AR). Each AR monitors traffic issued from users within the coverage areas of Base Stations (BSs) that are linked to the AR (Fig. 1). Throughout this paper, throughput refers to the total bandwidth consumed by traffic coming from end-users within the service area of the considered AR. The throughput is computed over a detection resolution, Δ time interval[1]. The choice of Δ is a compromise between the detection latency and the required computational load.

In terrestrial networks, a large-scale distribution of a DoS attack may make detection accuracy poor in the vicinity of the flooding sources. This phenomenon, known as Distributed-DoS (DDoS) attacks, is possible to occur in Mobile IP networks as well. To tackle such an issue, the Auto-Regressive model developed in [20] is used to predict the network traffic bandwidth over the detection resolution Δ. A comparison is then performed between the actually measured value of the traffic bandwidth and the predicted one. Throughout this paper, Θ_m and Θ_p denote the measured and predicted values of traffic bandwidth over each Δ time interval, respectively.

For each AR, three running states are defined: normal, alert, and action. Under normal conditions, the AR resides in normal

[1] In this paper, the monitoring procedure is based on only the bandwidth consumption. History of the loss rate and queuing delay may, however, be used as detection features to improve further the detection accuracy.

state, watching for abnormal traffic behavior. When the ratio of the measured traffic bandwidth to the predicted value exceeds a pre-defined threshold, Θ, as follows:

$$\partial_{\Re} > \Theta \tag{1}$$

where $(\partial_{\Re} = \frac{\Theta_m}{\Theta_p})$, the monitor considers it a possible DoS attack. In [20], the authors applied the Auto-Regressive model to a real DDoS attack traffic and the simulation results indicate that good performance can be obtained when the parameter θ is set to 1.4. Unless otherwise specified, Θ is given the value 1.4.

A high variation in bandwidth consumption (i.e. large values of $\partial_{\Re}$) is an indication of abnormal behavior inside the service area of the AR. This argument is based on the fact that a DoS attack should inject a significant amount of traffic into the AR to clog the targeted victim. Upon a noteworthy variation in bandwidth consumption, the AR switches to alert state. In the alert state, the AR clusters flows coming from users within its service area into a number of groups. Flows are defined as streams of packets sharing the quintuple: source and destination addresses, source and destination port numbers, and protocol field. Clustering of flows can be performed according to different elements. IP source and destination addresses are useful in forming aggregates of requests that are issued to access a particular server. Application type can be considered in case of a virulent worm that propagates by email and is overwhelming other traffic. IP destination prefix can be used in case of detection of flooding attacks targeting a site or a particular network link.

This paper considers the most common case; DoS attacks targeting a particular host or a set of web servers within the same domain. Clustering is performed thus according to the IP destination prefix. Once the clustering procedure is done, the AR sorts the clusters according to their aggregate traffic rate. Having a high-rate cluster of numerous flows addressed to the same IP destination prefix, the system can infer that this is either an ordinary flash crowd event or a DoS attack. The challenge consists in making distinction between the two cases. In the proposed mechanism, this distinction is based on the fact that legitimate TCP flows obey the Additive Increase Multiplicative Decrease (AIMD) concept of the TCP congestion control, whereas misbehaving sources remain unresponsive.

From the monitoring results of the cluster, the AR can compute the actual average rate of the flows. Let μ_i denote the measured rate of the i^{th} flow in the cluster. In the case of misbehaving flows with significantly higher measured rates, the AR sends them a test feedback requiring them to decrease their sending rates to a particular value, say $(\alpha \cdot \mu_i)$ where $(0 < \alpha < 1)$. Since at this stage, all flows, legitimate as well as malicious ones, will be requested to reduce their sending rates, α aims to minimize the damage that the scheme may cause to legitimate users. With no specific purpose in mind, α is set to 0.5 throughout the paper. After the test feedback transmission, if a flow does not react in a single RTT then it is unresponsive and its packets are discarded. Since flows are unaware of when ARs are monitoring their behavior, they have to always follow the test feedback. Information on unresponsive flows is stored in a *Black List* and is multicast to the neighboring ARs. Upon reception of such a list, ARs verify whether any of the mobile users in the *Black List* are within their service areas. If any are found, they will be sent a test feedback and their legitimacy will be based on their responsiveness as mentioned above.

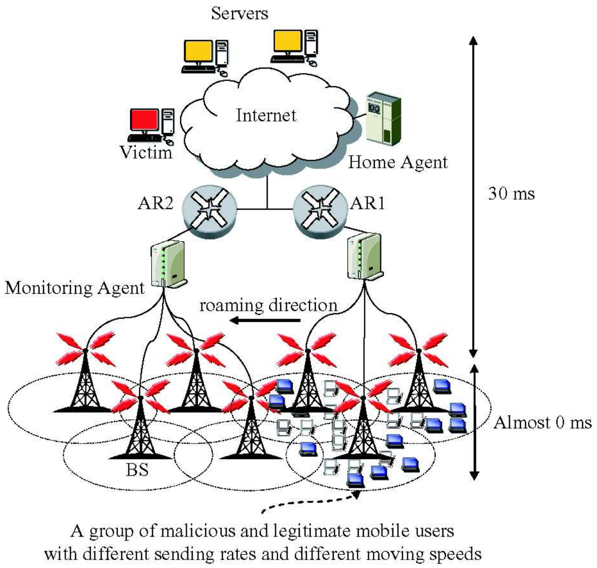
Servers
Internet
Victim
Home Agent
AR2
AR1
30 ms
Monitoring Agent
roaming direction
Almost 0 ms
BS
A group of malicious and legitimate mobile users
with different sending rates and different moving speeds

Fig. 1. Simulation environment

The test feedback is written in the receiver's advertised window (RWND) field carried by the TCP header of acknowledgment (ACK) packets. When the feedback reaches the sender, a legitimate user should react to the message and accordingly modify its current rate. It should be emphasized that during the monitoring process, TCP flows that have no ACK packets can be considered as part of a DDoS attack. Retrieval of such flows can be used as a strong indication to increase the level of attack likelihood and to stimulate the AR to enter the alert state. Depending on the size of the suspicious cluster, transmission of the test feedback can be performed in either a deterministic or probabilistic manner. In the deterministic case, all flows in the suspect cluster are sent the test feedback. The obvious drawback of the deterministic manner is that it incurs a significant processing overhead. In case of a dense cluster, transmission of the test feedback can be performed in a probabilistic manner. The pitfall of the probabilistic approach is that if the selected flows turn out to be all unresponsive, all the flows of the cluster would be blocked and may consequently frustrate some legitimate flows that may exist in the cluster. Finally, it should be notified that by implementing the proposed method as a background task for ARs (Monitoring agent in Fig. 1), the processing power required to identify suspect clusters and to send their flows a test feedback should not be an issue.

IV. Performance Evaluation

A. Simulation Setup

This section verifies how the proposed system is efficient in protecting hybrid wired/Mobile IP networks from DoS attacks while minimizing damage to innocent flows. The performance evaluation relies on computer simulation, using Network Simulator (NS) [21]. Fig. 1 depicts the used network configuration. The figure considers the case of two adjacent ARs, AR_1 and AR_2. AR_1 serves a population of (N_{total}=250) mobile users scattered randomly over three neighboring wireless cells. Mobile nodes move at speeds chosen randomly from the interval $[5m/s, 15m/s]$ toward destinations selected randomly within the service areas of the two ARs. The N_{total} users form (M=3) clusters based on the prefix of their IP destination addresses. For the sake of simplicity, the DoS attack is assumed to target a single victim located along with (M-1) servers, as shown in the figure. While (M-1) clusters represent no danger to the network and contain only legitimate flows that send data to destinations other than the victim, one cluster is assumed to contain flooding sources as well as legitimate ones. Let (N_s=150) denote the size of this suspicious cluster. Throughout this paper, χ denotes the percentage of attacking sources among the flows of the suspicious cluster. In the simulation, χ is varied to model different levels of aggressiveness of the attacking aggregate.

In the wireless part of the network, the coverage radius of wireless cells is set to 350 meters. To have the longest distance across the overlapping area equal to 100 meters, the distance between the two neighboring base stations is fixed to 600 meters. The wireless domain is connected to the wired network through ARs. Links connecting servers to the Internet are given capacities equal to $200Mbps$. The bandwidth of other links, wired and wireless, is set to $500Mbps$. As for the link delays, they are chosen in such a way that the round trip time of connections between mobile nodes and servers is $60ms$ when queuing delays are null.

Legitimate users implement the TCP NewReno version [22]. The DoS attack is modeled as several ON/OFF TCP-Flooding sources whose On/Off periods are of equal times and are chosen randomly between $0.5s$ and $1.5s$. Each attacking flow sends malicious packets at a rate derived from a uniform distribution with a mean μ_{mean} and a variance of σ^2. In NS implementation, the maximum and minimum values of the distribution are set to $(\mu_{mean} + \sigma)$ and $(\mu_{mean} - \sigma)$, respectively ($max\_$ = 6Mbps, $min\_$ = 4Mbps). At the beginning of the simulation, we start the legitimate flows and let them stabilize. At time $t = 10s$, the flooding sources are randomly activated over a time interval of $5s$. Simulations are all run for 210 s, a duration long enough to ensure that the system has reached a consistent behavior. The data packet size is fixed to $1kB$. In all simulations, the detection resolution Δ is set to $1s$. All results are an average of several simulation runs.

B. Simulation Results

To evaluate the detection accuracy of the proposed scheme, the following two quantifying parameters are used:

- False negatives ratio (R_{FN}): This measure involves the number of malicious flows that go undetected by the system (False Negatives N_{FN}). It should be always maintained in the vicinity of zero. R_{FN} is defined as ($R_{FN} = \frac{N_{FN}}{\chi \cdot N_s}$).
- False positives ratio (R_{FP}): This measure is defined as the ratio of innocent flows that are unfairly punished (False Positives N_{FP}) to the total number of legitimate flows ($R_{FP} = \frac{N_{FP}}{(1-\chi) \cdot N_s}$). This index should be always minimized to zero.

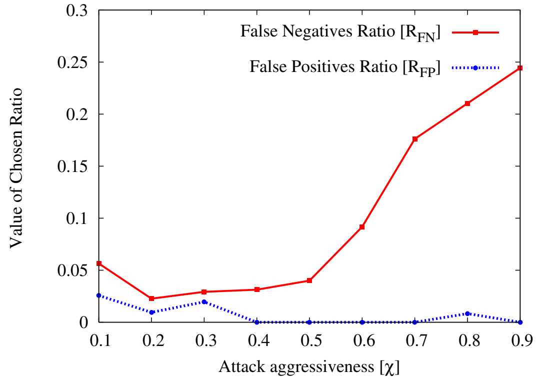

Fig. 2. False positives and negatives ratios for different levels of attack aggressiveness

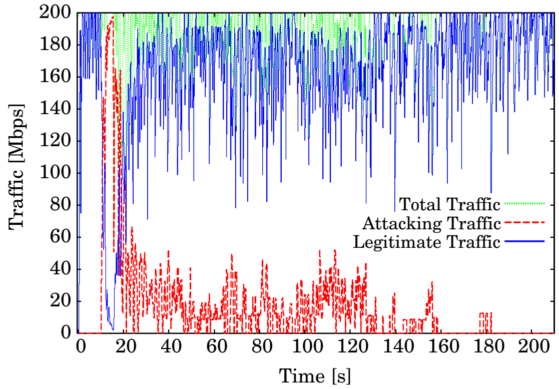

Fig. 3. Traffic of the suspicious cluster measured every $100ms$ ($\chi = 0.5$)

Fig. 2 graphs the false positives and false negatives ratios for different levels of attack aggressiveness. For all the considered values of χ, simulation results show that the false positives ratio remains in the vicinity of zero. This result is encouraging as the system does not cause the blocking of many legitimate users. The obtained false positives ratio is mainly due to the probabilistic method used in the test feedback transmission. Indeed, in the suspicious cluster, flows with high sending rates (compared to the average sending rate of the cluster flows) are grouped in a sub-cluster. Some flows of this sub-cluster are then randomly selected and sent each a test feedback. In case all the selected flows turn out to be unresponsive, all the flows of the sub-cluster are blocked. The obtained false positives ratio represents the legitimate flows that existed in the sub-cluster. On the other hand, the figure demonstrates that the false negatives ratio gets higher values as the attack becomes more aggressive. This is due also to the probabilistic transmission of the test feedback. Effectively, when the system forms the above-mentioned sub-cluster, some misbehaving flows go undetected as their sending rates are less than the average sending rate of the suspicious cluster flows. Despite of this fact, the simulation results indicate that even in case

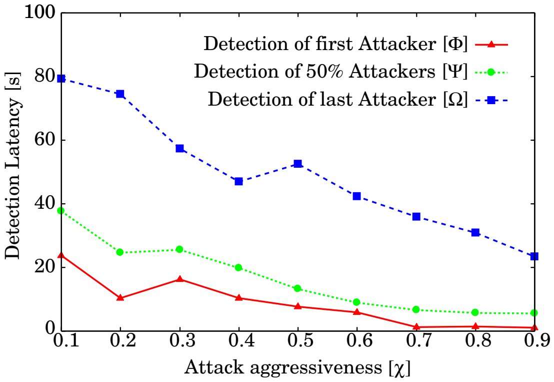

Fig. 4. Detection latency for different levels of attack aggressiveness

of significantly aggressive attacks ($\chi > 0.7$), more than 75% of the malicious flows were successfully blocked. It should be noted that guaranteeing smaller ratios of false positives is worthwhile even at the cost of a slight increase in the false negatives. Indeed, by blocking a set of flooding attacks, the undetected flooding sources (the false negatives) have to significantly increase their flooding rates to bring down the victim under protection. This increased flooding traffic makes it easier to detect the flooding attack and block again its sources by the proposed system. To illustrate the idea with more clarity, Fig. 3 plots the traffic variation of the malicious and legitimate flows in case of ($\chi = 0.5$). The traffic is measured every $100ms$. The figure demonstrates that at the beginning of the attack, a significant number of legitimate flows got their packets dropped and consequently backed off their sending rates as a large portion of the bandwidth was consumed by the attackers. After the detection and blocking of some flooding sources, legitimate users start increasing their sending rates.

To investigate the detection latency of the system, the attack detection latency is computed as:

- Φ: The sum of the detection delay from the start of the attack till the detection of the first malicious flow and the time required to judge whether the flow was responsive to the submitted test feedback or not.
- Ψ: The elapsed time from the start of the attack till the detection of 50% of the malicious flows.
- Ω: The elapsed time from the start of the attack till the detection of the last malicious flow.

Fig. 4 plots the values of Φ, Ψ, and Ω for different levels of attack aggressiveness. For all the considered values of χ, the presented results indicate that a large portion of misbehaving flows were isolated within a short time interval. From the above results, it can be concluded that the proposed system is able to detect attacks fast and accordingly minimize or eliminate the attack damage without depriving legitimate users from the network service.

V. Conclusion

In this paper, we proposed a prevention technique to throttle TCP-based bandwidth attacks over hybrid wired/Mobile IP networks. The proposed method uses an explicit feedback to test sources and judges accordingly their legitimacy. Indeed, suspicious flows are sent a test feedback and are required to decrease their sending rates. Unresponsive flows are blocked. Performance evaluation relied on computer simulation and a set of scenarios was considered. The obtained results elucidated both the short latency and high accuracy of the detection scheme.

Finally, it should be noted that while this paper focuses solely on the case of Mobile IP networks, with few modifications, this work can be also applied to shut down DDoS attacks within wireless metropolitan networks (e.g Worldwide Interoperability for Microwave Access (WiMax), IEEE 802.16).

References

[1] R.K.C. Chang, *"Defending against flooding-based distributed denial-of-service attacks: A tutorial"*, IEEE Communication Magazine, Vol. 40, No. 10, Oct. 2002, pp. 42-51.
[2] K. Park and H. Lee, *"On the effectiveness of probabilistic packet marking for IP traceback under Denial of Service Attack"*, In Proc. IEEE Infocom, Anchorage, Alaska, Apr. 2001.
[3] H. Wang, D. Zhang, and K.G. Shin, *"Detecting SYN flooding attacks"*, In Proc. of IEEE Infocom, New York, USA, Jun. 2002.
[4] M. Lynn and R. Baird, *"Advanced 802.11 Attack"*, Black Hat Briefings, Las Vegas, NV, USA, Jul. 2002.
[5] N. Asokan, V. Niemi, and K. Nyberg, *"Man-in-the-Middle in tunneled authentication protocols"*, In Proc. 11^{th} Int. Workshop on Security Protocols, Cambridge, UK, Apr. 2003.
[6] A. Miu and P.V. Bahl, *"Dynamic host configuration for managing mobility between public and private networks"*, In Proc. 3^{rd} Annual USENIX Symposium on Internet Technologies and Systems, San Francisco, Ca., USA, Mar. 2001.
[7] N.C. Winget, R. Housley, D. Wagner, and J. Walker, *"Security flaws in 802.11 data link protocols"*, Communications of the ACM, Vol. 46, No. 5, May 2003, pp. 35-39.
[8] S.H. Park, A. Ganz, and Z. Ganz, *"Security protocol for IEEE802.11 wireless local area network"*, Mobile Networks and Applications, Vol. 3 No. 3, Sep. 1998, pp.237-246.
[9] Airsnort http://airsnort.shmoo.com
[10] S. Fluhrer, I. Mantin, and A. Shamir, *"Weaknesses in the key scheduling algorithm of RC4"*, In Proc. 8^{th} Annual Workshop on Selected Areas in Cryptography, Toronto, Canada, Aug. 2001.
[11] C. Perkins, *"IP mobility support"*, IETF RFC 2002, Oct. 1996.
[12] A. Myles, D.B. Johnson, and C. Perkins, *"A Mobile host protocol supporting route optimization and authentication"*, IEEE J. Select. Areas Commun., Vol. 13, No. 5, Jun. 1995, pp. 839-849.
[13] S. Glass, T. Hiller, S. Jacobs, and C. Perkins, *"Mobile IP authentication, authorization, and accounting requirements"*, IETF RFC 2977, Oct. 2000
[14] J. Arkko, V. Devarapalli, and F. Dupont, *"Using IPsec to protect mobile IPv6 signaling between mobile nodes and home agents"*, IETF RFC 3776, Jun. 2004.
[15] D. Maughan, M. Schertler, M. Schneider, and J. Turner, *"Internet security association and key management protocol (ISAKMP)"*, IETF RFC 2408, Nov. 1998.
[16] CERT Advisory CA-96.01, *"UDP port denial-of-service attacks"*, Sep. 1997.
[17] CERT Advisory CA-96.21, *"TCP SYN flooding and IP spoofing attacks"*, Nov. 2000.
[18] D. Moore, G.M. Voelker, and S. Savage, *"Inferring Internet Denial-of-Service Activity"*, In Proc. 10^{th} USENIX Security Symposium, Washington DC, USA, Aug. 2001.
[19] A. Habib, M.M. Hefeeda, and B.K. Bhargava, *"Detecting Service Violations and DoS Attacks"*, In Proc. Network and Distributed System Security Symposium, San Diego, Ca., Feb. 2003.
[20] Y. Uchiyama, Y. Waizumi, N. Kato, Y. Nemoto, *"Detecting and tracing DDoS attacks in the traffic analysis using auto regressive model"*, IEICE Trans. Inf. & Syst., Vol. E87-D, No. 12, Dec. 2004, pp. 2635-2643.
[21] UCB/LBNL/VINT, *"Network Simulator - ns (version 2)"*, http://www.isi.edu/nsnam/ns/
[22] S. Floyd, T. Henderson and A. Gurtov, *"The NewReno Modifications to TCP's Fast Recovery Algorithm"*, IETF RFC 3782, Apr. 2004.