# Enhanced ICMP Traceback with Cumulative Path

Vrizlynn L. L. Thing[*,†] Henry C. J. Lee[†] Morris Sloman[*] Jianying Zhou[†]

[*] Department of Computing, Imperial College, London, United Kingdom
{vrizlynn.thing, m.sloman}@imperial.ac.uk

[†] Infocomm Security Department, Institute for Infocomm Research, Singapore
vriz@i2r.a-star.edu.sg, leecj@cet.st.com.sg, jyzhou@i2r.a-star.edu.sg

***Abstract*—Denial of Service (DoS) and Distributed DoS attacks constitute a major class of security threats today. As the attackers usually use IP spoofing to conceal their real location, several IP Traceback mechanisms have been proposed to trace the true source of the attackers to institute accountability. In wireless ad-hoc networks, where the nodes are typically devices with limited bandwidth, computational resource and battery power, and unpredictable routing topology, additional constraint is placed on these tracing techniques to locate the attack sources efficiently. In this paper, we proposed an enhancement scheme to ICMP Traceback with Cumulative Path (ITrace-CP) by performing dynamic probability adjustment against hop distance. Simulations were carried out on wired networks showing performance efficiency improvement of up to 190% and 143%, compared to ITrace-CP, for path lengths of 15 and 20 hops respectively. Further simulations on wireless ad-hoc network also showed significant performance improvement over previous work.**

***Keywords-Denial of Service; Distributed Denial of Service; IP Traceback; ICMP Traceback; Wireless Ad-Hoc Network***

## I. INTRODUCTION

Wireless ad-hoc networks are self-organizing systems formed by co-operating nodes within communication range of each other. Their topology is dynamic, decentralized, and ever changing with the ability and possibility of nodes moving arbitrarily. Their usage has become increasingly prevalent in emergencies (such as in the cases of disasters and wars) and also in daily life such as university campus and conference settings. The reason is mainly due to easy collaboration and efficient communication on the fly without the need for costly network infrastructure. These wireless networks are becoming an indispensable part of our life. However, users are concerned with the security vulnerabilities.

Denial of Service (DoS) [1] and Distributed DoS attacks constitute one of the major classes of security threats in the Internet today, depriving legitimate users of access to resources. Series of DDoS attacks that shut down some high-profile Web sites [2] have demonstrated their severe consequences. [3] shows the prevalence of DoS attacks in the Internet, whereby more than 12,000 attacks against over 5,000 distinct targets were observed in a 3-week long data collection period. As the attackers usually use IP spoofing to conceal their real location, several IP Traceback mechanisms [4-7] have been proposed to trace the true source of the attackers. This is needed to take further action against attackers or to try and stop the attack as close to the source as possible.

In the IP Logging approach [4], intermediate routers log the passage of all IP packets. The logging is based on the invariant portion of the IP header and the first 8 bytes of the payload. Hashing is then performed on the 28-byte information obtained above, followed by a Bloom filter processing to reduce the storage requirement. The logs are then retrieved from various routers when traceback for the path taken by any IP packet is initiated. In the IP Marking approach [5], intermediate routers along the path taken by the packets mark their information into the packet with certain probability. The victim of the attack can then examine the information found in the attack packets so as to construct the attack path. ICMP Traceback (ITrace) [6] is an Internet Draft proposed to the Internet Engineering Task Force (IETF). In this approach, intermediate routers generate an ITrace message probabilistically for each IP packet it processes, and send it to the same final destination of the IP packet. The victim of the attack can therefore use the ITrace messages to construct the attack path. In [7], an enhancement to ITrace, called the ICMP Traceback with Cumulative Path (ITrace-CP), was proposed to encode the entire attack path information in the ITrace-CP message.

However, in wireless ad-hoc networks, where the nodes are mainly devices with limited bandwidth, computational resource and battery power, and unpredictable routing behaviors, additional constraints are placed on these tracing techniques to locate the exact attack sources efficiently.

In this paper, we proposed an enhancement scheme to ITrace-CP by performing dynamic probability adjustment against hop distance. Simulations were carried out on wired and wireless networks to evaluate the scheme's performance, and the results are presented to show the efficiency improvement of the scheme over previous work. The rest of the paper is organized as follows. Section II provides the background on ITrace and ITrace-CP. Section III introduces the proposed Enhanced ITrace-CP scheme. Section IV gives details on the simulations, results and analysis. Section V concludes the paper.

## II. BACKGROUND

In the ICMP Traceback (ITrace) mechanism [6], a new ICMP message type is defined to carry information on routes that an IP packet has taken. As the IP Marking requires overloading some fields in the IP header, which raises backward compatibility problem, ITrace utilizes out-of-band messaging to achieve packet tracing. ITrace message is proposed to be generated at a low probability of 1/20000 at an intermediate router. This is assuming that the average diameter of the Internet to be 20 hops and therefore translates to a net increase in traffic of about 0.1% (which is 1/20000 x 20). The ITrace message generated contains information on the backward link, forward

link, or both links of the current router, which is then sent directly to the victim of the attack. An enhancement to ITrace, known as ICMP Traceback with Cumulative Path (ITrace-CP) [7], was proposed, whereby the ITrace-CP messages are made to carry entire attack path information so as to facilitate a faster attack path construction in the event of a DDoS attack. When a router receives an IP packet, an ITrace-CP message will be generated based on the probability set at the router. This message is then sent to the next hop router, instead of the destination address of the IP packet, as in the case of ITrace. This "next hop" should be as far as possible the same as the next hop for the corresponding IP packet. When the next hop router receives the ITrace-CP message for a corresponding IP packet forwarded to it, it will generate a new ITrace-CP message with the previous ITrace-CP message's contents and append its address to it. Therefore, the moment the furthest router from the victim (i.e. nearest to the attacker) generates an ITrace-CP message, the full attack path can be constructed.

## III. ENHANCED ITRACE-CP

In the ITrace proposal, it was recommended that ITrace messages be generated with a probability of 1/20000 at each intermediate router. However, in the ITrace-CP scheme, given that the path information is generated, it would increase tracing efficiency by generating ITrace-CP messages nearer to the attackers, or in other words, further from the victim, so as to be able to trace back to the attacker in a shorter time. Therefore, in this paper, we proposed an enhancement scheme to ITrace-CP by adjusting probability settings in proportion to the hop distance from the final destination. Distance information can be obtained through means such as traceroute (a widely available network tool). Here, we make the same assumption on the average Internet distance (i.e. 20 hops) and therefore, fix the upper bound limit on the additional traffic overhead to be 0.1%. In addition to increasing ITrace-CP messages generation probability in a linear manner, proportional to the hop distance, we consider distribution of the probability in an exponential manner so that a faster construction time is achievable within the same overhead constraint. The probability at each router is computed by:

$$p = d^x / c$$

where d is the distance from current router to the victim, x is the exponent and c is a constant for achieving the similar overhead as in ITrace and determined by the following formula,

$$\sum_{d=1}^{20} (d^x / c) = 1/1000$$

For values of x at 0.5, 1, 1.5, and 2, c was computed to be 61665.978, 210000, 760796.650, and 2870000 respectively. Figure 1 shows the probability used by each intermediate router for ITrace-CP messages generation with varying hop distance from the victim and exponent, x. With the enhanced scheme, routers with hop distance equal or greater than 10 and 11 from the victim will generate ITrace-CP messages with a higher probability for exponent, x, equals to 0.5 and 1 respectively. With exponent, x, equals to 1.5 or 2, higher probability is achievable only for hop distance equals or greater than 12. Therefore, for attack paths with the above-mentioned distance and their respective exponent values, the average construction time would be faster than ITrace or ITrace-CP. However, attack paths with shorter distance will require more time in detection, though with much lower traffic overhead (due to lower probability).

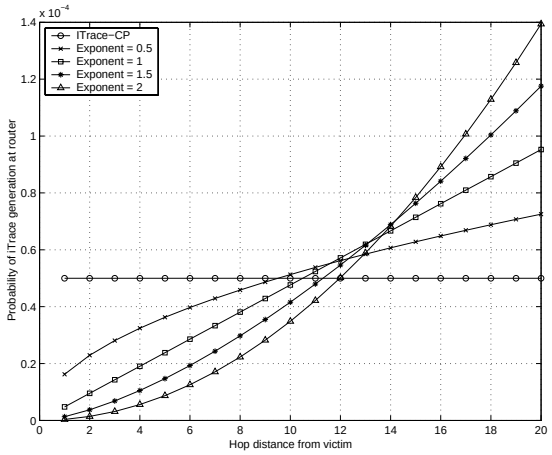


Fig. 1. Probability vs. Hop Distance

From [8], we see that the average path length in the Internet was found to be 15. Therefore, the speed of attack path detection will be greatly improved with the probability adjustment scheme in a real scenario of attack. Figure 1 also shows that the higher the exponent, the faster will be the attack path detection at higher hop distances. However, the curve will shift further to the right as the exponent increases and result in increasing hop distances to achieve the same probability as ITrace-CP messages generation. Furthermore, at the specific point of hop distance, the probability will drop with decreasing hop distances as the exponent is increased. Therefore, to use the enhanced scheme efficiently, a proper value of exponent will have to be chosen. Simulations were carried out to investigate the average time for attack path detection and traffic overhead incurred with varying hop distances and exponent values.

## IV. SIMULATIONS

We carried out simulations on wired and wireless ad-hoc networks using ns2. On the wired networks, we conducted simulations to evaluate the performance improvement of the enhanced ITrace-CP over the core ITrace-CP for different path lengths. We performed the simulation studies using the linear network topology, for attackers situated 5, 10, 15, and 20 hops away from the victim. 30 runs were carried out for each simulation scenario. The tree topology was not simulated as in the case of multiple attackers at the leaves of the tree, they would have been treated as independent attack paths. This would be similar to simulating multiple linear topologies. In all the simulation scenarios, the effective attack traffic arriving at the victim was 1 Mbits/s. However, as we are interested in the relative performances, this number is only indicative.

### *A. Time for Attack Path Detection*

The average times for the construction of various hops of the attack path for both schemes were obtained. The graphs were plotted and shown in Figure 2 to 5. In these figures, the x-axis represents the number of hops of the attack path discovered while the y-axis represents the time taken in seconds. The

figures correspond to attack paths of length 5, 10, 15 and 20 respectively.

It is shown in Figure 2 that enhanced ITrace-CP resulted in higher detection time due to the low number of hops in the attack path. While ITrace-CP was able to complete the detection of all hops in 23.8 secs, the enhanced scheme was able to detect only 4 hops within this time for exponent of 0.5 and 1, though with lower overhead due to lower probability of message generation (see Section III for theoretical analysis for shorter hop distances).

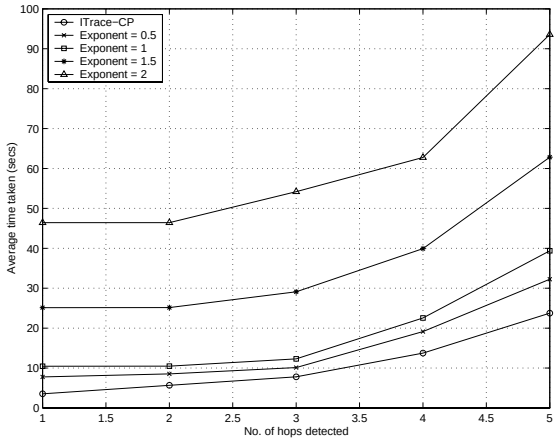


Fig. 2. Average time taken to detect various hops (5-hop attack path)

In Figure 3, we see the improvement in the detection time of the enhanced ITrace-CP at exponent value of 0.5. As we had mentioned in Section III, the enhanced scheme with exponent of 0.5 (for hop distance ≥ 10) and 1 (for hop distance ≥ 11) would result in faster detection. The average time taken for exponent value of 0.5 was shorter, and exponent value of 1 was slightly higher than that of ITrace-CP during the simulations.

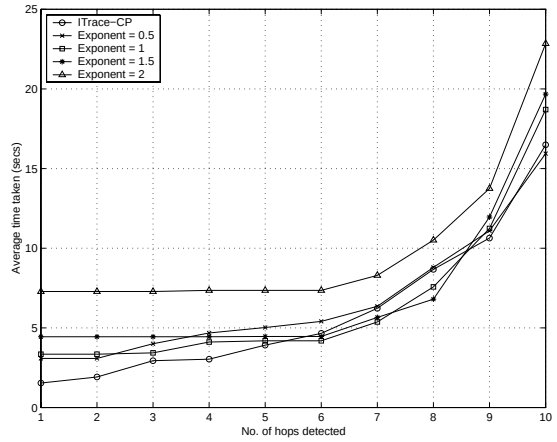


Fig. 3. Average time taken to detect various hops (10-hop attack path)

As shown in Figure 4, the performance of the enhanced scheme was better than the core ITrace-CP, for all values of exponent. Exponent values of 1.5 and 2 achieved the best performance with the same lowest average time incurred.

Figure 5 shows that the approximate average detection time was achieved for all exponent values with the probability adjustment scheme. The performance of the scheme, in terms of detection speed, was shown to be better than the core ITrace-CP from the 5th hop and onwards.

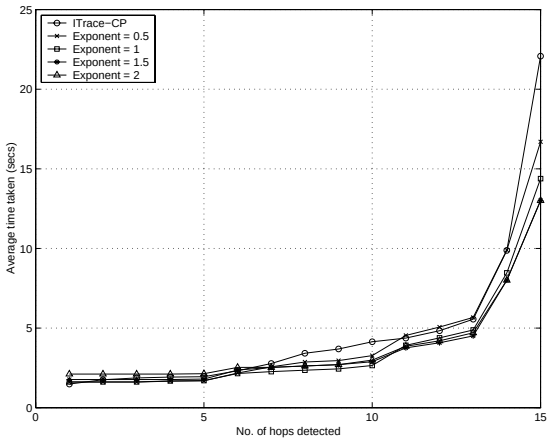


Fig. 4. Average time taken to detect various hops (15-hop attack path)

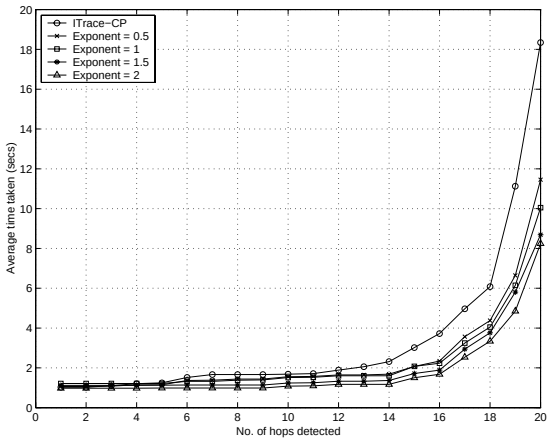


Fig. 5. Average time taken to detect various hops (20-hop attack path)

Table 1 shows the average time (in secs) taken to detect the full attack path for different path lengths and schemes.

| | 5-hop | 10-hop | 15-hop | 20-hop |
|---|---|---|---|---|
| ITrace-CP | 23.8 | 16.5 | 22.1 | 18.3 |
| Exponent 0.5 | 32.6 | 15.9 | 16.7 | 11.5 |
| Exponent 1 | 39.4 | 18.7 | 14.4 | 10 |
| Exponent 1.5 | 62.8 | 19.7 | 13 | 8.7 |
| Exponent 2 | 93.6 | 22.8 | 13 | 8.2 |

Table 1. Average attack path detection time in secs

### *B. Traffic Overhead*

The average traffic overhead (for 30 runs) incurred by the ITrace-CP messages generation for the construction of various hops of the attack path for both scheme were obtained. To make a fair comparison, the percentages of ITrace-CP messages generated over the data packets were computed and used for presentation. As the average time to detect the attack paths implicitly indicates the data packets sent, presenting just the number of ITrace-CP messages generated will result in misunderstanding of small traffic overhead when in actual fact, it was partly due to short detection time taken. The graphs were plotted and shown in Figure 6. The x-axis represents the length

of the attack path while the y-axis represents the percentages of traffic overhead incurred.

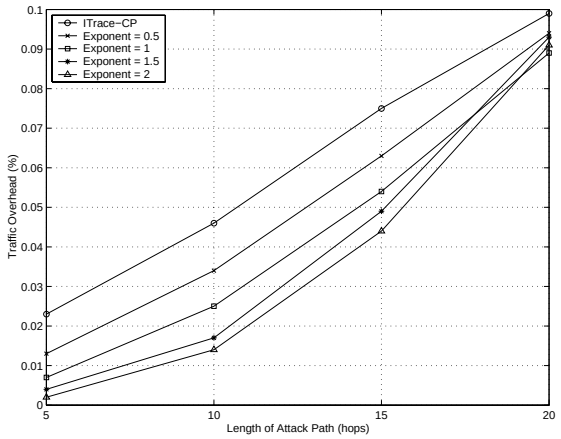


Fig. 6. Traffic Overhead incurred

As mentioned in Section III, the design of the enhanced scheme fixes the upper bound limit on the additional traffic overhead introduced to be 0.1%. This is similar to the ITrace and the ITrace-CP methods. Therefore, it can be seen from Figure 6 that the traffic overhead approaches 0.1% for attack path length of 20 hops. In the simulations, the performance of the enhanced scheme, in terms of traffic overhead incurred, was better than ITrace-CP, especially significant were those with attack path lengths less than 20 hops. This is due to decreasing ITrace-CP messages generation probability from the 20th hop to the 1st.

Impact on Infrastructure

In the above, we have only considered the impact of the traffic with respect to the victim as this is a significant aspect of DoS attacks. As in a real scenario, the link nearer to the victim will be more congested due to bandwidth availability. However, if the impact on the infrastructure was to be analyzed, the distance traversed by the packets will have to be taken into consideration. In the enhanced scheme, the chance of ITrace-CP message generation is higher at further hops away. There will be a higher impact on the Internet traffic if the same number of packets were to traverse more hops. Therefore, we performed an analytical study on the traffic overhead taking into account the distance traversed by the ITrace-CP packets, as follows.

$$\text{Adjusted Traffic Overhead} = \sum_{d=1}^{L} (d * p_d)$$

where $p_d$ is the probability of ITrace-CP message generation at a router that is d hops away from the victim, and L is the length of the attack path. In this case, we performed the analysis by adjusting the overhead varying linearly with the distance, which is the worst case, as the link bandwidth availability was not considered to simplify the analysis.

Taking attack path lengths of 5, 10, 15, and 20 hops, the traffic overhead is calculated and presented in Figure 7. For path lengths of 5 and 10, the traffic overhead was lower with the enhanced scheme. The improvement ranges from 39% to 89% for 5-hop length, and 16% to 62% for 10-hop length. However, improvement for 15-hop length (2% to 16%) was achievable for exponent values of 1 and above. The adjusted traffic overhead suffers deterioration of 2% for 15-hop path length with exponent value of 0.5, and from 18% to 46% at 20-hop path length for all exponent values.

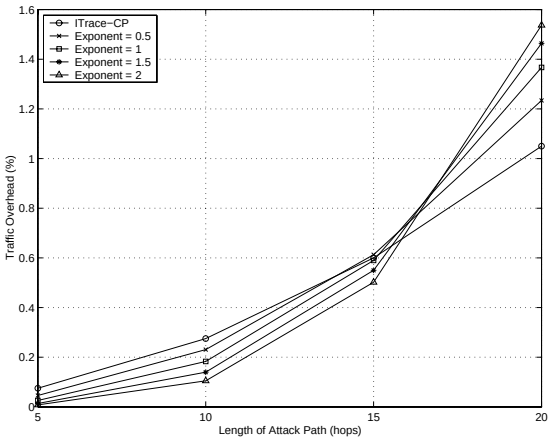


Fig. 7. Adjusted Traffic Overhead

### C. *Normalized Path Construction Efficiency*

To evaluate the overall performance, we computed the normalized path construction efficiency, η, defined as follows:

$$\eta = 1/(\text{Detection Time x Traffic Overhead})$$

Figure 8 and 9 show the normalized path construction efficiency versus the attack path length for the different schemes with and without adjusting the traffic overhead. It can be seen that the efficiency gap starts wider between the schemes at lower attack path length and converges towards the 20-hop distance. This was due to the much lower traffic overhead incurred for shorter path lengths.

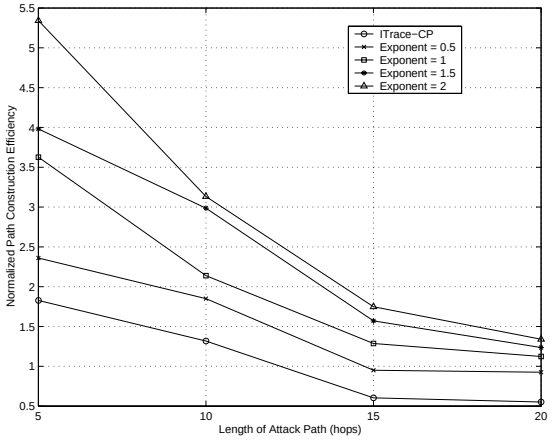


Fig. 8. Normalized Path Construction Efficiency with traffic overhead

For the 20-hop length in Figure 8, the overhead was set at the upper bound of 0.1% (see Section III), and therefore, the efficiency depended mainly on the average time savings. For the average Internet path length of 15 hops, the improvement in efficiency over ITrace-CP was in the range of 58% to 190%. The maximum performance efficiency improvement of the enhanced scheme over the core ITrace-CP was 192%, 138%, 190% and 143%, for attack paths of 5, 10, 15 and 20 hops respectively (achieved at exponent value of 2).

After making adjustment to the overhead traffic, the improvement in the efficiency over ITrace-CP for attack path

length of 15 hops, dropped to the range of 30% to 103%. The maximum performance efficiency improvement was 78%, 90%, 103% and 53%, for 5, 10, 15 and 20 hops respectively (achieved at exponent value of 2). We can also see that with the probability adjustment scheme, improvement in efficiency was achieved for all exponent values and path lengths.

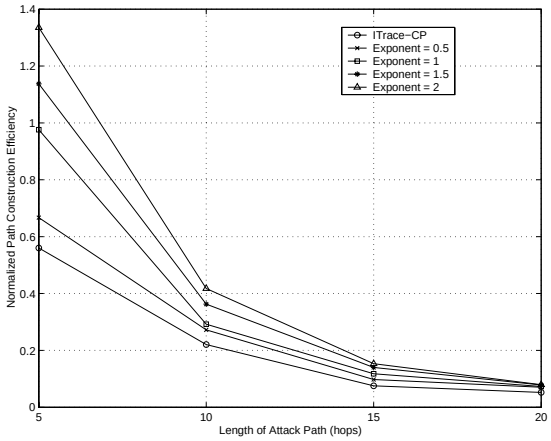


Fig. 9. Normalized Path Construction Efficiency with adjusted traffic overhead

We carried out further simulations on wireless ad-hoc network using the exact simulation scenario (i.e. using dynamic routing protocol in the 54-node wireless ad-hoc network), which has the highest difficulty of tracing, in [9]. The difficulty is due to the dynamic routing and high number of intermediate nodes, leading to unpredictable distribution of attack packets throughout the network, and resulting in high number of traversed paths.

In [9], using the Probabilistic Packet Marking (PPM) scheme, there were 10 paths traversed by the attack packets and 4 paths (carrying 62% of the attack traffic) were fully constructed. Using ITrace, there were 4753 attack paths and 176 paths (carrying 9% of the attack traffic) were fully constructed. Using our enhanced ITrace-CP scheme, 39 paths were traversed by the attack packets and 26 of them (carrying 84% of the attack traffic) were fully constructed. Therefore, application of this new traceback scheme resulted in a higher percentage of attack paths discovered, which were responsible for carrying most of the attack traffic. In this case, more efficient attack mitigation could be carried out to block the attack traffic at the network entrance points, nearer to the attacker.

## V. CONCLUSION

Maintaining total traffic overhead at 0.1% at the victim's end, we proposed adjustment of the probability of ITrace-CP message generation at each router, with respect to its distance from the victim. We have suggested that the probability of message generation should increase exponentially with distance in hops from the victim. An appropriate value of probability exponent had to be chosen as it influences the detection time to traceback attack paths of different lengths. Low exponent values resulted in performance either deteriorating or improving, depending on the path lengths. As the exponent value increases, detection time for short paths will increase significantly, and at the same time, the detection time for long paths will decrease. However, it was also observed that a saturation point will be reached for exponent values from 1.5 to 2, where increasing the power will no longer speed up the detection process significantly.

Using ns2, simulations were carried out for both wired and wireless networks, to evaluate the performance improvement of the enhanced scheme over the core ITrace-CP, based on detection time and traffic overhead. On the wired networks, performance efficiency improvement of up to 192%, 138%, 190% and 143%, for the 5, 10, 15 and 20-hop attack paths respectively, was achievable by applying the enhanced scheme. Theoretical analysis was then carried out to evaluate the impact of the traffic overhead on the infrastructure, taking into consideration the hops the packets had to traverse. Simplifying the analysis by extracting the link bandwidth component, this would be the worst case scenario. The maximum performance efficiency improvement was reduced to 78%, 90%, 103% and 53%, for the 5, 10, 15 and 20-hop attack paths respectively.

On the wireless ad-hoc network simulation, we used the scenario of 54-node network based on dynamic routing protocol, as in [9], which has the highest difficulty of tracing. Using the enhanced scheme, attack paths carrying 84% of the attack traffic were detected. This is a significant improvement compared to discovery of attack paths carrying 62% and 9% of the attack traffic, using PPM and ITrace respectively.

Therefore, we could conclude that the enhanced ITrace-CP scheme proposed was able to achieve a higher efficiency as compared to previous work in both wired (i.e. using ITrace and ITrace-CP) and wireless ad-hoc networks (i.e. using PPM and ITrace, as IP Logging was analysed to be infeasible for use in [9]).

## ACKNOWLEDGEMENTS

We would like to thank the Institute for Infocomm Research and the EU Diadem Firewall: FP6 IST-2002-002154 for their support in this research work.

## REFERENCES

[1] K. J. Houle, G. M. Weaver, "Trends in Denial of Service Attack Technology", CERT Coordination Center, Oct. 2001

[2] L. Garber, "Denial-of-Service attacks rip the Internet", IEEE Computer, Vol. 33, No. 4, pp. 12-17, Apr. 2000

[3] David Moore, Geoffrey M. Voelker, Stefan Savage, "Inferring Internet Denial-of-Service Activity", Usenix Security Symposium, Aug. 2001

[4] Alex C. Snoeren et al, "Hash-Based IP Traceback", ACM SIGCOMM 2001, Aug. 2001

[5] Stefan Savage et al, "Practical network support for IP traceback", ACM SIGCOMM 2000

[6] Steve Bellovin et al, "ICMP Traceback Messages", IETF Internet Draft, Version 4, Feb. 2003 (Work in progress)

[7] Henry C. J. Lee, Vrizlynn L. L. Thing, Yi Xu, Miao Ma, "ICMP Traceback with Cumulative Path, an Efficient Solution for IP Traceback", International Conference on Information and Communications Security, Oct. 2003, (Springer Lecture Notes in Computer Science, Vol. 2836, pp. 124-135, Sept. 2003)

[8] Abraham Yaar, Adrian Perrig, Dawn Song, "Pi: A Path Identification Mechanism to Defend against DDoS Attacks", IEEE Symposium on Security and Privacy, May 2003

[9] Vrizlynn L. L. Thing, Henry C. J. Lee, "IP Traceback for Wireless Ad-hoc Networks", 60th IEEE Vehicular Technology Conference, Sept. 2004