Journal of Computer Security 13 (2005) 877–903
IOS Press

Experimental analysis of attacks against intradomain routing protocols

Antonio Pescapè * and Giorgio Ventre
Dipartimento di Informatica e Sistemistica, University of Napoli "Federico II" (Italy), Via Claudio 21, 80125 Naples, Italy
E-mail: {pescape, giorgio}@unina.it

Nowadays attacks against the routing infrastructure are gaining an impressive importance. Therefore, approaches to network security and reliability must take into account effects of routing protocols attacks and consequently must consider techniques to protect the network infrastructure. However, in spite of an increasing attention by scientists and practitioners to this issue, there is still a lack of experimental quantitative studies on the effects of routing attacks. To cope with these deficiencies, in this work we present a framework to conduct experimental analysis of routing attacks, and to prove its usefulness we study three attacks against routing protocols: *route flapping on RIP*, *Denial of Service on OSPF* by means of the *Max Age attack* and, finally, *route forcing on RIP*. We present a qualitative analysis and a performance analysis that aims to quantify the effects of routing protocol attacks with respect to routers resources and network traffic over controlled test beds.

Keywords: Attacks against intradomain routing protocols, router and network resources, performance analysis, networking tools

1. Introduction

Four top-level requirements for Internet security have been recently identified: end-system security, end-to-end security, secure Quality of Service (QoS), and secure network infrastructure [25]. In this paper, we point our attention to issues concerning the security of the network infrastructure and in particular to routing protocols attacks.

Routing protocols implement mechanisms to discover the optimal route between end points, describing peering relationships, methods of exchanging information, and other kind of policies. Since network connectivity depends on proper routing, it follows that routing security is a critical issue for the entire network infrastructure. In spite of this, while other aspects of computer and communication security, as network applications and system security, are subjects of many studies and of widespread interest, attacks against routing protocols are less known. For example, in [21]

*Corresponding author, Dipartimento di Informatica e Sistemistica, University of Napoli "Federico II", Via Claudio 21, 80125 Napoli, Italy. Tel.: +39 081 7683908, Fax: +39 081 7683816, E-mail: pescape@unina.it.

it is reported that *Denial of Services* (DoS) attacks currently found on the Internet can be divided into three categories: (i) attacks against websites, (ii) attacks against DNS root servers and (iii) attacks directly against the routing infrastructure; and is reported that the last category is largely less known than the other ones.

As Bellovin states in [32], this is probably due to two fundamental reasons: effective protection of the routing infrastructure is a really hard problem and it is outside the scope of traditional communications security communities. Moreover, most communications security failures happen because of buggy code or broken protocols whereas routing security failures happen despite good code and functioning protocols. For instance, in the case of the routing infrastructure one or more dishonest or compromised routers can alter the routing process, and a hop-by-hop authentication is not sufficient.

Attacks against routing protocols can be categorized in two main classes: insider and outsider attacks [27]. Outsider attacks are conducted by impersonating legitimate routers and injecting false routing information into the network or by making links or routers temporarily unavailable. Insider attacks, instead, are mounted from one or more compromised routers. Besides this classification, several kinds of attacks can be performed [9]:

- By finding the address of a router in a significant location, and then attacking the router using Denial of Service methods that are based on resource exhaustion, it is possible to disrupt communications among end-systems. Such attack requires the ability to send large quantities of packets to the router under attack, possibly through distributed sources.
- Another attack is to force a session between peers to failure sending malformed packets or through violations of the protocol's state machine. For example, by sending a TCP reset packet it is possible to force the closing of a TCP connection. This attack requires the ability to send packets to the router under attack as well.
- A different technique is to act on the information carried within the protocol packets, rather than the protocol itself, and participate to the routing distributed process. For instance, the attackers may attempt to divert traffic along a monitored path, thus gaining access to the information in the data stream, or into a black hole, causing a Denial of Service. The attackers may even force a routing loop in the system, causing wide scale network outages. Attacking the protocol in any of these ways requires the attackers to be able to access the link between two routers, and to inject false data, or modify data on the fly.
- The attackers can compromise a router attached to the network, and use it to send forged routing messages. This requires being able to manipulate the configuration of the router, possibly by gaining access to the console, or setting configuration through Simple Network Management Protocol (SNMP), or exploiting vulnerabilities in the router's operating system, etc.

When implementing routing protocols it is good practice to provide protection against both insider and outsider attacks. Common countermeasures that have been proposed to defeat or mitigate these attacks include the use of digital signatures to verify authenticity of routing messages, the addition of sequence numbers and timestamps in each routing message to protect against outsiders re-ordering or delaying legitimate routing information and, finally, strong origin authentication using symmetric or asymmetric cryptographic primitives. Often, in despite of protection need, in real networks there are no implemented countermeasures. This is due to several motivations: (i) cost of security in routing protocols in terms of complexity, processing, storage and bandwidth overhead [27,39]; (ii) lack of security expertise of routing administrators. Also, Bellovin states that once a protocol has been standardized or even just stabilized, it is technically and politically hard to add new mechanisms to enhance the protocol's security.

In this paper we are not focused on particular countermeasure mechanisms, but rather, on the effects of some routing protocols attacks. This work presents an approach to conduct experimental studies of attacks against the routing infrastructure observing the effects and quantifying their impact with respect to network and devices resources. More precisely, we studied several attacks against Interior Gateway Protocols (IGP) focusing our attention on an experimental study of three types of attacks:

- The *Max Age attack* on Open Shortest Path First (OSPF): the target of this attack against OSPF routing protocol is a Denial of Service. Implementing the Max Age Attack we cause subnets inaccessibility.
- *Route Flapping* on Routing Information Protocol (RIP): sending false information, we force the RIPv2 to use different routes at different time intervals.
- *Route Forcing* on RIP: we force the traffic to follow a longer path.

In order to have an evaluation of these attacks, we used a controlled and fully configurable open test bed. In this way we were able to control as much variables as possible, as well as to configure several network topologies. This also allowed repeatability of experiments, obtaining numerical results, which show the degradation of the network configurations under test, with high confidence intervals. Pointing our attention on *router resources*, *network traffic*, *convergence time* and, in general, on *network behavior* before, during and after the attacks, we found interesting results for all of the above attack scenarios.

The rest of the paper is organized as follows. After the introduction, in Section 2 some related works are presented. Section 3 gives a rapid background on attacks against routing protocols. Section 4 shows our experimental scenario in terms of hardware and software architectures whereas in Section 5 we expose our analysis and experimentation. Finally, Section 6 presents some conclusion remarks and issues for future research.

2. Related works

The first work to report security vulnerabilities in the Internet routing infrastructure was [33], where attacks against IP Source Routing, RIP, Exterior Gateway Protocol and ICMP Route Redirects have been depicted. The author denounced that Internet Protocols were intrinsically insecure because the Internet architecture was not designed with security in mind. Indeed, most of the vulnerabilities still found in current networks derive from such deficiency.

A comprehensive talk on router security in modern infrastructures is given in [32], with particular focus on OSPF and Border Gateway Protocol (BGP), and explanation of link-cutting attacks to divert traffic, which are described in deeper detail in [34]. Also, in [34] a demonstration of how to calculate what links to cut in complex but realistic topologies is given. In [10] and [6] a complete view on known attacks against routing protocols is presented. In [22] some of the well-known vulnerabilities in inter-domain routing protocols are discussed. Vetter et al. [5] present an experimental study of insider attacks for the OSPF routing protocol, analyzing its weaknesses and reporting on how an implementation of the *max sequence number* attack was capable to block routing updates for more than an hour in their experimental test bed. In [1] and [2] Kumar and Crowcroft discuss security threats to routing protocols, both distance vector and link state based, and then propose various counter measures to make these protocols more secure against external threats.

Most of current research works focus on security enhancement or design for a single specific routing protocol or a class of routing protocols. In 1996 Smith et al. [3] analyzed security vulnerabilities in the BGP routing protocol and presented a set of modifications to the protocol to minimize or eliminate the most significant threats. Later, an extension of BGP, denominated S-BGP [35], was proposed as an architectural solution to BGP's security problems. S-BGP makes use of IPsec [28] to protect all BGP traffic between neighbor routers, enforcing data authentication, data integrity and anti-replay features. Unfortunately the implementation of such architecture requires high memory and CPU resources and training of operational staff of network service providers [36]. For these reasons, currently, S-BGP is not widely deployed. In [7] and [23] the authors use a simulation of AS models derived from the Internet to evaluate the impact that signatures, verification and certificate handling have on convergence time, message size and storage for the principal approaches to securing BGP. Also in [20] an evaluation of the costs of commonly proposed techniques for *origin authentication* is presented.

In [4] authors discuss security improvements to distance vector-based routing protocols, proposing the use of timestamps, signatures and sequence numbers to protect legitimate routing update messages. Also in [39] the authors show mechanisms to secure distance vector and path vector routing protocols. Murphy [31] proposes the introduction of digital signatures to secure OSPF, a link state-based algorithm, in order to prevent tampered *link-state advertisements* (LSAs). In [40] the authors present their experimental observations of persistent route flaps on OSPF and show a

methodology to solve such problems. They propose a scheme for damping the route flaps which is based on a figure of merit, named *penalty*, that helps in categorizing paths into "well behaved" or "ill behaved".

In [27] Hauser, Przygienda and Tsudik examine the cost of security in link state routing and develop two interesting techniques for efficient and secure processing of link state updates.

Being that, for the reasons exposed, the above-mentioned security improvements are difficult to be rapidly deployed, recent research efforts have also been focused on detection of routing attacks rather than only on their prevention. In [30] a cooperative intrusion detection approach to protect routing infrastructures from Denial of Service attacks is proposed. At the basis of such approach is the development of failure models that characterize the behavior of misbehaving routers. Based on these models it is possible to design distributed diagnosis protocols that detect and logically remove misbehaving routers from the routing system. In [8,13,37] the authors present scalable intrusion detection architectures for real-time detection of link-state routing protocol attacks. In their works the authors show that their approach is effective in detecting attacks to the OSPF routing protocol.

Despite of many contributions to computer and network security, to the best of our knowledge and due to the already cited reasons, no previous works exist on effects of attacks against routing protocols in terms of (i) hardware resources of routers (i.e., CPU, RAM, ...); (ii) network traffic; and (iii) routing algorithms convergence time (i.e., in the case of OSPF, as for the routers workload, we also measure the recalculation time of Dijkstra algorithm).

This *quantitative* information is of paramount importance in order to assert the real impact of such attacks and to devise possible techniques to make computer networks inherently more secure and robust. Indeed, our work steps from the assumption that analysis of Internet traffic while OSPF and RIP are under attack is beneficial in understanding how the attacks can be prevented and recovered. Our results can be used as a reference scenario for (i) studying the effects of attacks against routing protocols and for (ii) planning useful mitigation strategies and restoring policies.

3. Attacks against interior gateway routing protocols

This section represents a succinct survey and discussion of attacks against IGP routing in network infrastructures. Along with an attacks tutorial, this section presents a sort of reference taxonomy with particular focus to router and routing protocol vulnerabilities in RIP and OSPF. RIP and OSPF were first taken into consideration, as they are the most commonly deployed intra-domain routing protocols. Both these protocols describe methods for exchanging routing information (network topology and routing tables) between routers of an Autonomous System. Both RIP and OSPF are mainly affected by the lack of mechanisms to guarantee integrity and authentication of the information exchanged.

- RIPv1 is intrinsically insecure since there is no authentication system and it uses the unreliable UDP (*User Datagram Protocol*). RIPv2 includes an option to set an up to 16-character clear text password string or an MD5 signature. The use of an MD5 signature would obviously make spoofing a much more complex task, although RIP packets can be easily spoofed. Despite the relative ease of spoofing, several very large networks rely on RIP for some of their routing functions. In particular the final parts (last router of the tree) of an AS (*Autonomous System*) are generally based on RIP.
- OSPF is more secure than RIP, featuring several built in security mechanisms. In [11] it is pointed out that there are characteristics of link-state routing protocols, with respect to distance vector protocols, which make them more robust to some attacks. The use of flooding to propagate information should guarantee that if a malicious router would modify other router's information, as long as there is an alternate path, good routers should always receive the correct messages. Also, the fact that every router independently possesses entire topology information and is responsible only for its own local portion of the topology leads to a sort of information independency that could allow finding which router is lying. However, various elements of an LSA (*Link State Advertisement*) can be altered by intercepting and re-injecting OSPF packets. OSPF can be configured to use no authentication, text-based password authentication, or MD5. If an intruder gains the correct level of access, he could use a tool such as *dsniff* [17] to capture OSPF packets and obtain the clear text password.

In the following, there is a list of a number of common attacks against routing protocols. In this section, first, we present attacks against RIP. Second, we describe some attacks against OSPF and, finally, we illustrate the effect of a dated *gated* bug.

In the RIP scenario, with the term *Black Hole* attack, the following situation is depicted. All the traffic to a network is redirected to a specific router, the attacker, and will never reach the proper destination. With the term *Route Flapping* we mean an attack consisting in the advertisement and withdrawal of a route (or withdrawal and re-advertisement) alternating in rapid succession, thereby causing a route oscillation. Finally, with *Route Forcing* we mean to force a different path with respect to the optimal path indicated by the routing protocol, causing service degradation.

In the OSPF scenario, when an attacker continually interjects legitimate advertisement packets of a given routing entity with spoofed packets in which the age is set to the maximum value, he causes network confusion and may contribute to a DoS condition. Such an attack not only consumes network bandwidth, but also makes the routing information database inconsistent, disrupting correct routing. In this case we have the *Max Age* attack.

The *Sequence++* attack is based on a similar mechanism, with the difference that injected packets have a sequence number always larger than the legitimate ones, indicating a fresher route. The original router contests this, in a process called "fight back", by sending a refresh message with its own LSA and an even newer sequence

number. This attack leads to network instability, thereby contributing to a DoS condition.

When the maximum sequence number 0x7FFFFFFF is injected by an attacker/intruder we have the *Max Sequence* attack. Attacker router then appears to be the freshest route. Theoretically this should create the cited "fight back" condition from the original router. In practice, it has been found that in some cases the Max Seq LSA is not removed and remains in the link state database for more than an hour, giving control to the attacker for that time period. Indeed new legitimate LSAs are discarded because they are considered obsolete with respect to that injected from the attacker.

When a compromised AS edge router generates a deluge of external LSAs, and each of them will be flooded to every router in the AS we have the *Table Overflow* attack. There is no mechanism to verify that these external LSAs are valid. Eventually, every router will be filled in these LSAs, preventing the routing protocol from successfully installing any new network entries. This could also be catastrophic if the implementation does not have proper protections against overflows.

Finally, as regards the *gated* [14] routing software, the *Bogus LSA* attack was referred to a bug in an implementation of *gated*. This attack made *gated* crash when a specially crafted LSA packet was sent and required all *gated* processes to be stopped and restarted to purge the bad LSA, thereby causing a DoS condition. This attack may not affect hardware routers and is most likely fixed in more recent versions of *gated*.

The previous selected attacks are quite different in nature. One has to do with the way that routers propagate information, another one with the utilization of the age field, two of them are tied to sequencing and/or overflow depending on how they are looked at, and another one is tied to a coding error. It is clear, that such different nature could pose several limitations to the way to fix them in a unified fashion. This work provides just a methodology to qualitatively and quantitatively analyze the impact that routing attacks have on the network infrastructure, leaving room to propose innovative mitigation strategies and restoring policies.

4. Experimental scenario

In this section we present the network scenario and the tools used for the experimental characterization.

In order to have an evaluation of the above-mentioned attacks, we used as a proof-of-concept a controlled and fully configurable open test bed. In this way we were able to control as much variables as possible, as well as to configure on our own several network topologies. We preferred to use a controlled test bed also to have a full control of network devices and network traffic. This aspect is important when several tests must be performed in order to obtain numerical results. Indeed, if we want to

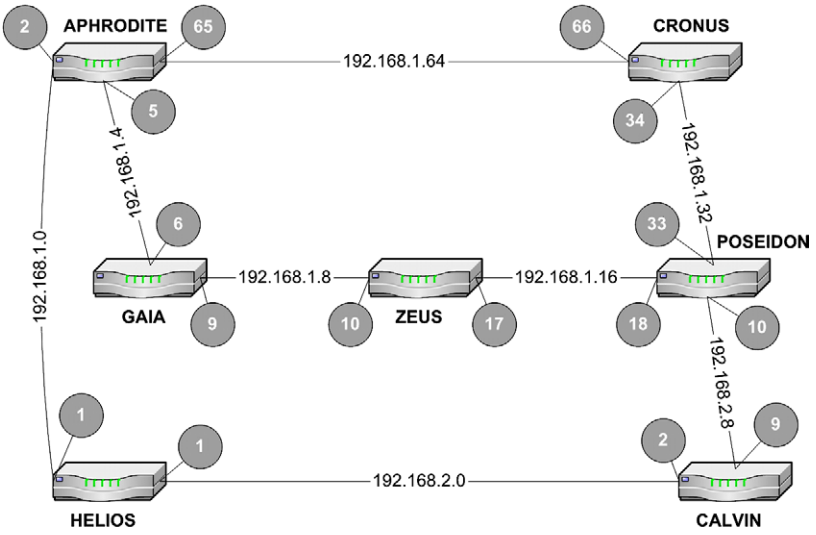


Fig. 1. The reference network.

Table 1
Hardware and OS description

CPU	Intel PII 850 MHz
Memory	RAM 128 Mb
	Cache 256 kb
Operating system	Linux Red Hat 7.1–kernel 2.4.2-2
Network cards	Ethernet Card 10/100 Mbps

draw a general behavior and we want to guarantee the experiment repeatability we have to know and control the dependent variables.

Our objective, indeed, is to demonstrate how it is possible to show and analyze what happens to a network under attack and in particular how to evaluate the quantitative impact on network resources. Thus, while the presented methodology is of general validity, the numerical results obtained in the following experiments are important to show the degradation of the specific network configurations under test.

In Fig. 1 the experimental test bed used in our practical analysis is depicted, whereas in Table 1 a complete description of hardware and software characteristics is shown. The experimental test bed is composed of eight networks and seven routers (with back-to-back connections). We called the routers Aphrodite, Cronus, Poseidon, Gaia, Zeus, Helios, and Calvin. In Fig. 1, the numbers in the bullets represent the last field of the IP address related to the indicated network address (placed on the link between two routers). It is worth noting that, by changing the position of the links between each pair of routers and by consequently modifying (when needed) the addressing plane, we are able to produce a large number of network topologies.

Table 2
Software tools

Tool	Description
GNU Zebra	Routing Protocols Demon
Spoof	Routing packets forger
D-ITG	Packet level Traffic Generator
Ethereal	Network Sniffer
Snmpd	SNMP Agent

We used Linux Red Hat 7.1–kernel 2.4.2-2 as operating system and GNU Zebra [19] to implement routing protocols on our Linux routers. As regards traffic generation, we used D-ITG (*Distributed Internet Traffic Generator*) [16,29], a traffic generator developed at Dipartimento di Informatica e Sistemistica of the University of Napoli "Federico II". To passively capture transferred packets, without influencing the systems constituting the network configurations under test, we worked with the *Ethereal* [15] network sniffer.

With the terms *packet forgers*, instead, we mean tools able to create and send known protocols packets, filling the various fields with information chosen by the user, usually an attacker. Such software also allow to indicate IP source and destination addresses in a totally arbitrary way and attend to the calculation of the control fields that are built from the data inserted in the packet. To perform "routing packets forgery" and conduct our simulated attacks, among several available tools as *Spoof* [26], *Nemesis* [24], *IRPAS* [12], and *srip* [18], we chose *Spoof*. Theoretically the most flexible *packet forger* should produce packets in the format of any protocol. In real cases every *packet forger* is limited to few protocols and, often, it has several restrictions. *Spoof* is not an exception in this sense: it is able to produce packets of the RIPv1, RIPv2, OSPF and BGP format, with some restrictions regarding OSPF. We chose it because of its bias towards routing protocols, in respect to other packet forgers, and because it was previously used in other works reported in literature that are related to security of routing protocols [38].

In Table 2 a summary of used "open source" tools is reported. Using common PCs as well as open source and freeware tools guarantees that our experiments can be repeated in a more simple way by other practitioners or researchers in the field of computer network security.

5. Experimental analysis

As already said, we take into account only IGP and, in particular, we present our practical analysis of the following attacks against routing protocols:

- Route Flapping on RIP.
- Denial of Service on OSPF (*Max Age Attack*).
- Route forcing on RIP.

Furthermore, out of the scope of routing attacks we show how to use the route forcing technique as a rudimental traffic engineering mechanism.

Before to step into experimental details, it is worth mentioning that we repeated each test several times. In the following subsections, in the case of numerical values, we present the average value over several experiment repetitions. Thanks to the use of a controlled test bed we had a confidence interval greater than or equal to 96%. Therefore, for each experiment, the numerical results found in this work can be considered very reliable in representing how an attack impacts the network configuration under test. This is an important aspect of the proposed methodology.

5.1. Route Flapping on RIPv2

In Fig. 2 the network scenario and the actors (subnets and routers) of this attack are depicted.

In this test, by sending false information from **POSEIDON** to **CRONUS**, we forced the RIPv2 to use different routes at different time intervals. By using *spoof* we announced to **APHRODITE** a longer distance for the **192.168.2.0** subnet with respect to the real path, this was repeated at regular intervals. Therefore, the effect of the attack was to make believe **CRONUS** that the shortest path to reach subnet **192.168.2.0** had **POSEIDON** as first hop instead of **APHRODITE**. RIPv2 reacted with a *path oscillation* (*route flapping*) between **CRONUS** and subnet **192.168.2.0**: one time the best path had **POSEIDON** as first hop, the other time it had **APHRODITE**.

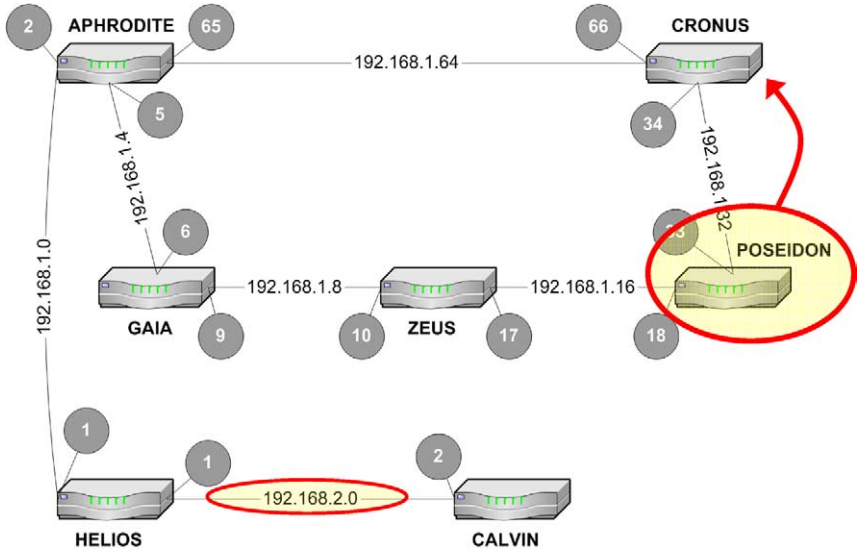


Fig. 2. Route Flapping on RIP: network test bed.

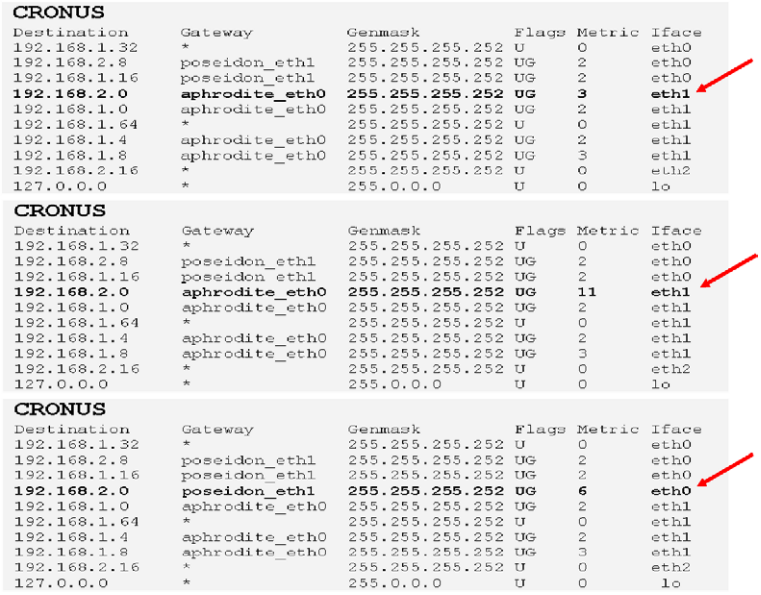
We carried out a high number of trials by varying the time interval between two malicious packets. Also, all trials have been repeated in two different situations. In the first one, the false information (malicious packets) sent from **POSEIDON** to **CRONUS** specified a distance between **APHRODITE** and **192.168.2.0** subnet equal to 10 hops. In the second one, the false information reported that the **192.168.2.0** subnet was unreachable from **APHRODITE** (in RIP this is obtained by sending packets announcing a distance equal to 16 hops).

In Fig. 3, the routing table of **CRONUS**, before, during and after the attacks, is shown. The normal situation is established when **CRONUS** receives the **POSEIDON** DV (*Distance Vector*) and it refreshes its routing table.

In Fig. 4, results of the *traceroute* command between **CRONUS** and **CALVIN** (192.168.2.2) during the attack are presented.

Before analyzing experimental results we would like to underline that, in a normal situation (where the attack is not present), RIP indicates a path between **CRONUS** and **192.168.2.0** subnet made by 3 hops, whereas during route flapping the path length oscillates between 3 and 6 hops.

Our test lasted for 200 seconds and we repeated it for different time intervals between two successive malicious packets containing false information (3 s, 6 s, 9 s, . . . up to 30 s). For an interval of 200 s we evaluated the number of route flappings and the length of a stable path.



CRONUS

Destination	Gateway	Genmask	Flags	Metric	Iface
192.168.1.32	*	255.255.255.252	U	0	eth0
192.168.2.8	poseidon_eth1	255.255.255.252	UG	2	eth0
192.168.1.16	poseidon_eth1	255.255.255.252	UG	2	eth0
192.168.2.0	**aphrodite_eth0**	**255.255.255.252**	**UG**	**3**	**eth1**
192.168.1.0	aphrodite_eth0	255.255.255.252	UG	2	eth1
192.168.1.64	*	255.255.255.252	U	0	eth1
192.168.1.4	aphrodite_eth0	255.255.255.252	UG	2	eth1
192.168.1.8	aphrodite_eth0	255.255.255.252	UG	3	eth1
192.168.2.16	*	255.255.255.252	U	0	eth2
127.0.0.0	*	255.0.0.0	U	0	lo

CRONUS

Destination	Gateway	Genmask	Flags	Metric	Iface
192.168.1.32	*	255.255.255.252	U	0	eth0
192.168.2.8	poseidon_eth1	255.255.255.252	UG	2	eth0
192.168.1.16	poseidon_eth1	255.255.255.252	UG	2	eth0
192.168.2.0	**aphrodite_eth0**	**255.255.255.252**	**UG**	**11**	**eth1**
192.168.1.0	aphrodite_eth0	255.255.255.252	UG	2	eth1
192.168.1.64	*	255.255.255.252	U	0	eth1
192.168.1.4	aphrodite_eth0	255.255.255.252	UG	2	eth1
192.168.1.8	aphrodite_eth0	255.255.255.252	UG	3	eth1
192.168.2.16	*	255.255.255.252	U	0	eth2
127.0.0.0	*	255.0.0.0	U	0	lo

CRONUS

Destination	Gateway	Genmask	Flags	Metric	Iface
192.168.1.32	*	255.255.255.252	U	0	eth0
192.168.2.8	poseidon_eth1	255.255.255.252	UG	2	eth0
192.168.1.16	poseidon_eth1	255.255.255.252	UG	2	eth0
192.168.2.0	**poseidon_eth1**	**255.255.255.252**	**UG**	**6**	**eth0**
192.168.1.0	aphrodite_eth0	255.255.255.252	UG	2	eth1
192.168.1.64	*	255.255.255.252	U	0	eth1
192.168.1.4	aphrodite_eth0	255.255.255.252	UG	2	eth1
192.168.1.8	aphrodite_eth0	255.255.255.252	UG	3	eth1
192.168.2.16	*	255.255.255.252	U	0	eth2
127.0.0.0	*	255.0.0.0	U	0	lo

Fig. 3. Route Flapping on RIP: routing tables during the tests.

```
...
...
ven ott  3 09:54:59 CEST 2003
traceroute to 192.168.2.2 (192.168.2.2), 30 hops max, 38 byte packets
 1  aphrodite_eth0 (192.168.1.65)  0.509 ms  0.304 ms  0.238 ms
 2  helios_eth2 (192.168.1.1)  0.493 ms  0.398 ms  0.356 ms
 3  calvin_eth0 (192.168.2.2)  0.577 ms  0.589 ms  0.503 ms

ven ott  3 09:55:01 CEST 2003
traceroute to 192.168.2.2 (192.168.2.2), 30 hops max, 38 byte packets
 1  poseidon_eth1 (192.168.1.33)  0.420 ms  0.295 ms  0.245 ms
 2  zeus_eth1 (192.168.1.17)  0.675 ms  0.375 ms  0.347 ms
 3  gaia_eth0 (192.168.1.6)  1.120 ms  0.461 ms  0.387 ms
 4  aphrodite_eth0 (192.168.1.65)  0.465 ms  0.413 ms  0.389 ms
 5  helios_eth2 (192.168.1.1)  0.590 ms  0.512 ms  0.536 ms
 6  calvin_eth0 (192.168.2.2)  0.676 ms  0.762 ms  0.710 ms
...
...
```

```
...
...
ven ott  3 09:55:07 CEST 2003
traceroute to 192.168.2.2 (192.168.2.2), 30 hops max, 38 byte packets
 1  poseidon_eth1 (192.168.1.33)  0.421 ms  0.299 ms  0.240 ms
 2  zeus_eth1 (192.168.1.17)  0.452 ms  0.404 ms  0.335 ms
 3  gaia_eth0 (192.168.1.6)  0.593 ms  0.492 ms  0.394 ms
 4  aphrodite_eth0 (192.168.1.65)  0.445 ms  0.428 ms *
 5  helios_eth2 (192.168.1.1)  0.802 ms  0.757 ms  0.550 ms
 6  calvin_eth0 (192.168.2.2)  0.634 ms  0.740 ms  0.603 ms

ven ott  3 09:55:14 CEST 2003
traceroute to 192.168.2.2 (192.168.2.2), 30 hops max, 38 byte packets
 1  aphrodite_eth0 (192.168.1.65)  0.464 ms  0.290 ms  0.239 ms
 2  helios_eth2 (192.168.1.1)  0.468 ms  0.384 ms  0.348 ms
 3  calvin_eth0 (192.168.2.2)  0.493 ms  0.533 ms  0.475 ms
...
...
```

Fig. 4. Route Flapping on RIP: traceroute command during the tests.

	3	**6**	**9**	**12**	**15**	**18**	**21**	**24**	**27**	**30**
1ST CYCLE	*8*	*12*	*10*	*12*	*10*	*10*	*8*	*10*	*8*	*6*
2ND CYCLE	*2*	*6*	*8*	*8*	*6*	*6*	*8*	*2*	*4*	*6*

Fig. 5. Route Flapping on RIP: number of route changing during the tests.

In Fig. 5, the number of the route changes (route flapping) inducted is reported. The first cycle is related to the attack that sends information on distance between **APHRODITE** and **192.168.2.0** subnet equal to 10 hops. The second cycle is related to the attack that sends information on the distance between **APHRODITE** and **192.168.2.0** subnet equal to 16 hops (unreachable). We noticed that in the second cycle there is a lower number of route changes than in the first cycle. In addition, we observed that in both cycles, the maximum number of route changes is obtained far from the interval bounds (as you can see the highest value is obtained in the experiment with a time interval, between two successive packets, of around 12 s). Close to the interval bounds a smaller number of oscillations is experienced.

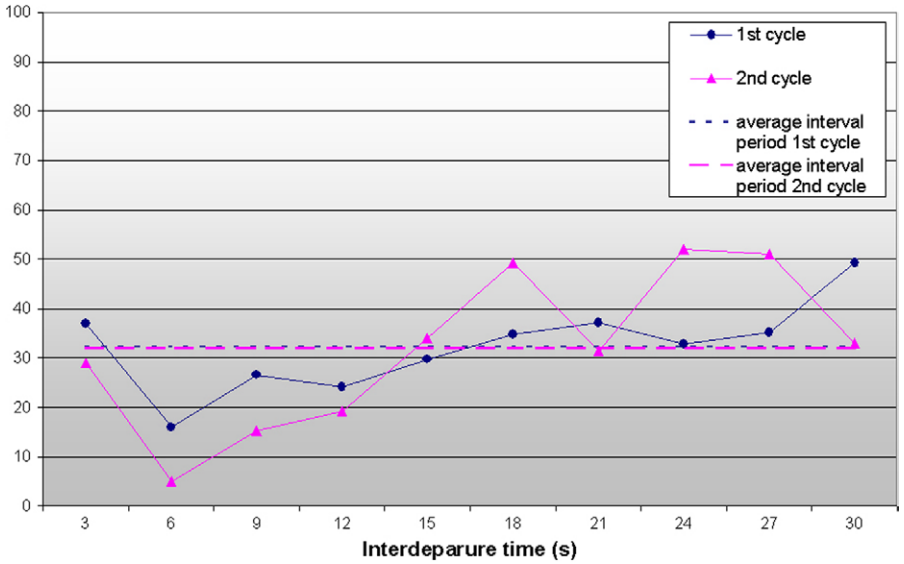


Fig. 6. Route Flapping on RIP: average time of a stable path (3 hops).

In Fig. 6 and in Fig. 7, the average durations of stable paths (measured in seconds) in function of the malicious packets inter-departure time, are depicted.

We noticed that in the first cycle the average duration of stable paths was independent from malicious packets inter-departure time.

Finally, using SNMP (Simple Network Management Protocol), for each router in the test bed and for each experiment, the total amount of input and output traffic was calculated.

In Figs 8–11 the traffic amount is indicated as a percentage of the minimum number of bytes for each interface of each router (in both input and output directions). The figures contain a single line for each router.

In Figs 8–11 it is possible to observe that there is a similar envelope for **CALVIN**, **HELIOS**, **APHRODITE**, and **CRONUS** plots. We measured a bigger amount of traffic on the path **POSEIDON-ZEUS-GAIA**. This behavior is suitable with our target. Indeed, our attack has the result to force this last path rather than the one through **CRONUS-APHRODITE** to reach the **192.168.2.0** subnet.

5.2. Denial of Service on OSPF

In this case the experimental test bed is depicted in Fig. 12. The target of OSPF *Max Age Attack* is to cause a *Denial of Service*. Using *spoof* we sent false LSAs from **POSEIDON** to **CALVIN**. These LSAs contained information on **ZEUS** and **CRONUS** and had the *Age* field equal to the maximum value. The result of these actions was the loss of contact between **ZEUS/CRONUS** and **CALVIN/HELIOS**.

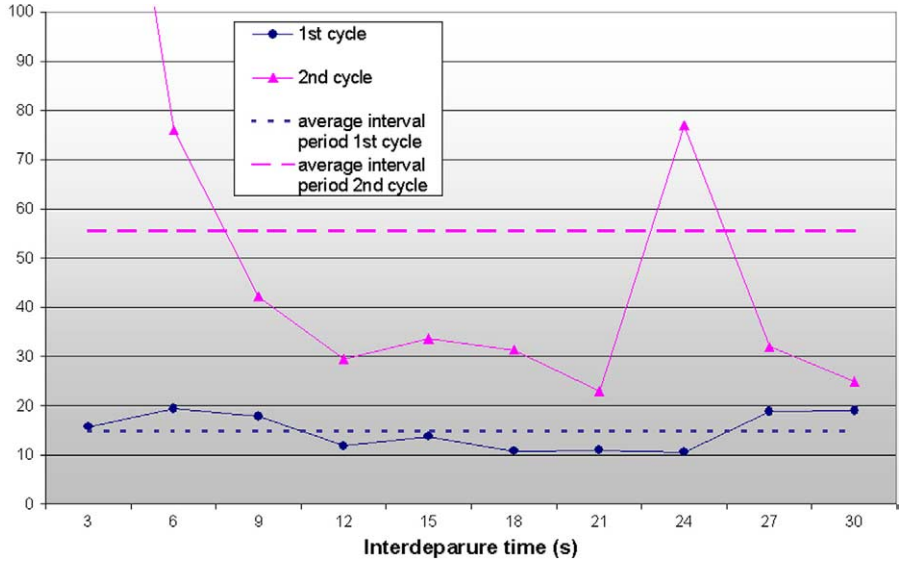


Fig. 7. Route Flapping on RIP: average time of a stable path (6 hops).

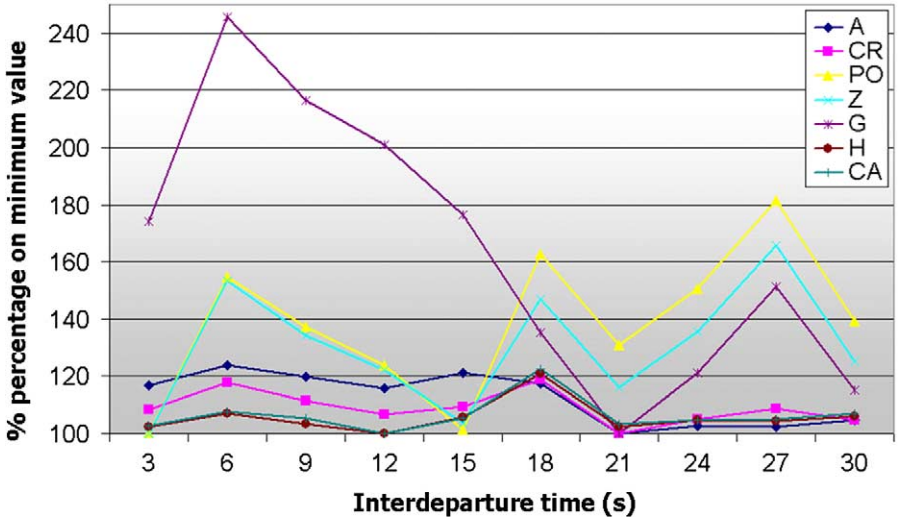


Fig. 8. Route Flapping on RIP: IN Traffic (first cycle).

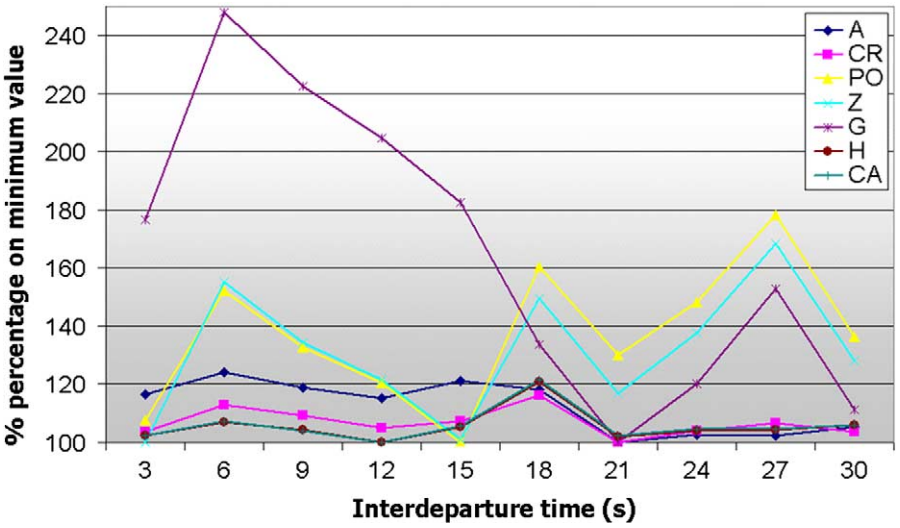


Fig. 9. Route Flapping on RIP: OUT Traffic (first cycle).

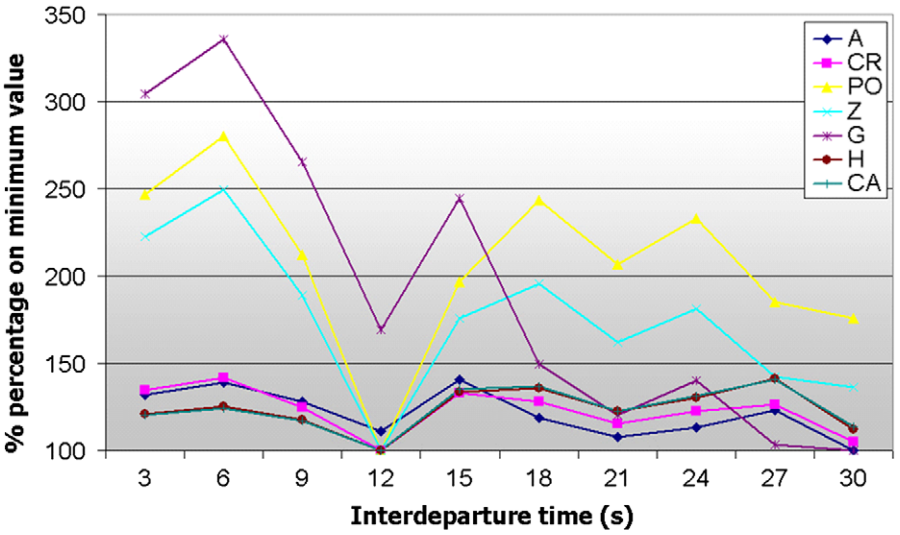


Fig. 10. Route Flapping on RIP: IN Traffic (second cycle).

Indeed, because of the information that is believed to be sent by **POSEIDON**, the routers named **CALVIN** and **HELIOS** deleted the entries for **ZEUS** and **CRONUS** in their LSA databases.

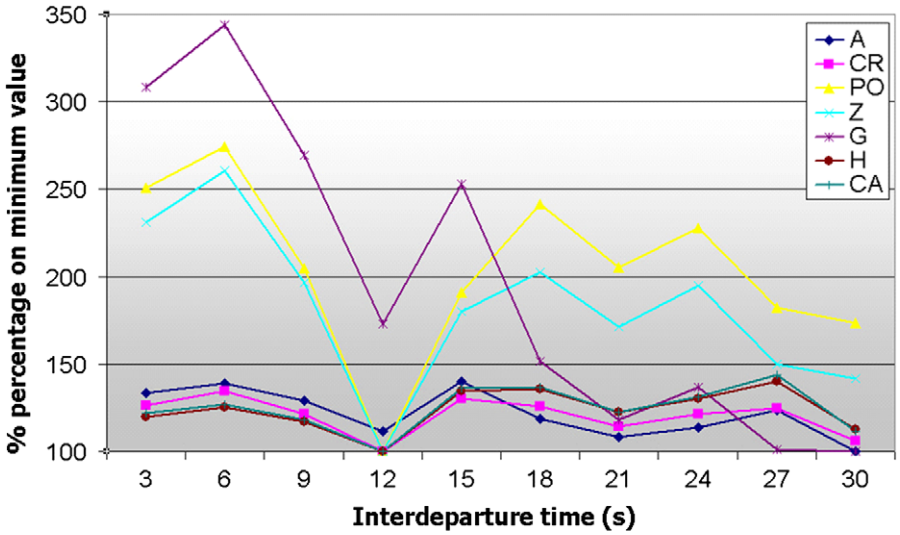


Fig. 11. Route Flapping on RIP: OUT Traffic (second cycle).

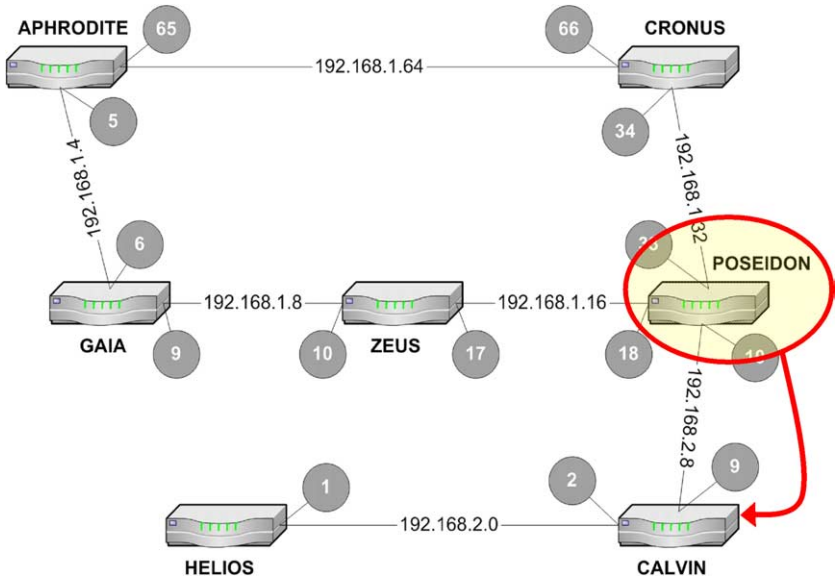


Fig. 12. Denial of Service on OSPF: experimental test-bed.

Attack consequences are very simple to understand. In Fig. 13 it is shown how, with the ping program, we verified the success of the attack. In particular, we used the *ping* command from **CALVIN** and **HELIOS** directed to **subnets 192.168.1.4**, **192.168.1.8,** and **192.168.1.64**.

The **HELIOS** *LSA database* before the attack (Fig. 14) and during the attack (Fig. 15) confirmed the depicted behavior.

During this attack, by using ***OspfSpfRuns*** SNMP variable we counted the number of executions of the Dijkstra algorithm (in a single specific area) for each router of our real test bed. By taking into account the number of recalculations during the attack we can evaluate the computational load for each router. It is worth nothing that in our test we use a PC with a RAM equal to 128 Mb. In a real router we may find a smaller amount of memory, hence to control the number of recalculations – when a large number of routing entries is present – is important. This is even more relevant when we have a large network with a large number of routers.

It is important to underline that during our first experiment (50 seconds) there were six executions of the Dijkstra algorithm for each router. This means that, besides the depicted *DoS* effect, we were able to increase also the computational load of each router in the test bed. Finally, it is interesting to note that if we increase the duration of the experiment (100 s, 150 s, ...) the number of recalculations per second is basically constant.

After this first measure we repeated the experiment eliminating all other traffic sources (*ping*, ...) from our experimental network. In this way we were able to measure the precise amount of traffic on the network during the attack.

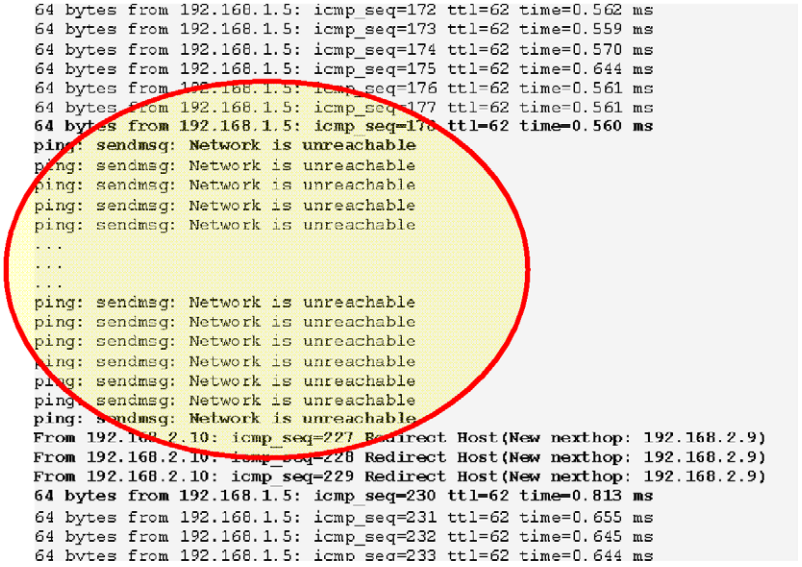


Fig. 13. Denial of Service on OSPF: attack result.

```
                Router Link States (Area 0.0.0.0)

Link ID         ADV Router      Age  Seq#       CkSum  Link count
192.168.1.9     192.168.1.9     1777 0x80000007 0x6f0d 2
192.168.1.10    192.168.1.10    1774 0x80000006 0x4d15 2
192.168.1.33    192.168.1.33    1776 0x80000008 0xd22c 3
192.168.1.34    192.168.1.34    1791 0x80000007 0x1987 2
192.168.1.65    192.168.1.65    1790 0x80000007 0x8e8d 3
192.168.2.1     192.168.2.1       65 0x80000009 0x56b7 2
192.168.2.9     192.168.2.9      100 0x80000009 0x314b 2
```

Fig. 14. Denial of Service on OSPF: LSA database before the attack.

```
                Router Link States (Area 0.0.0.0)

Link ID         ADV Router      Age  Seq#       CkSum  Link count
192.168.1.9     192.168.1.9       55 0x80000008 0x6d0e 2
192.168.1.33    192.168.1.33      53 0x80000009 0xd02d 3
192.168.1.65    192.168.1.65      68 0x80000008 0x8c8e 3
192.168.2.1     192.168.2.1      143 0x80000009 0x56b7 2
192.168.2.9     192.168.2.9      178 0x80000009 0x314b 2
```

Fig. 15. Denial of Service on OSPF: LSA database during the attack.

	Experiments duration (s)		
	50 s	100 s	150 s
Number of recalculations	6	11	18
Number of recalculations per second	0.12	0.11	0.12

Fig. 16. Denial of Service on OSPF: recalculations number.

In order to have a reference value, during a time interval equal to 60 s, we measured the traffic load with and without the *Max Age Attack* on OSPF.

In Fig. 17 comparisons among data related to network load when the systems were under attack, expressed in percentages with respect to the values found without the attack, are depicted. The most important result is that the maximum increase of traffic happens on the router interfaces that rely on the subnet that was unreachable during the attack.

5.3. Route Forcing on RIP

In Fig. 18 the network test bed used in the case of *Route Forcing* on RIPv2 is shown. Between **APHRODITE** and **CRONUS** there is one single hop. Using the

		in (%)	*out* (%)	*total in* (%)	*total out* (%)
Aphrodite	**eth_0**	150	118	152	141
	eth_2	153	687		
Cronus	**eth_0**	110	107	113	115
	eth_1	119	150		
Poseidon	**eth_0**	122	134	115	119
	eth_1	108	111		
	eth_2	135	121		
Zeus	**eth_0**	573	143	141	139
	eth_1	122	136		
Gaia	**eth_0**	656	152	183	185
	eth_1	144	692		
Helios	**eth_1**	129	120	129	120
Calvin	**eth_0**	119	129	125	124
	eth_1	131	120		

Fig. 17. Denial of Service on OSPF: Traffic analysis.

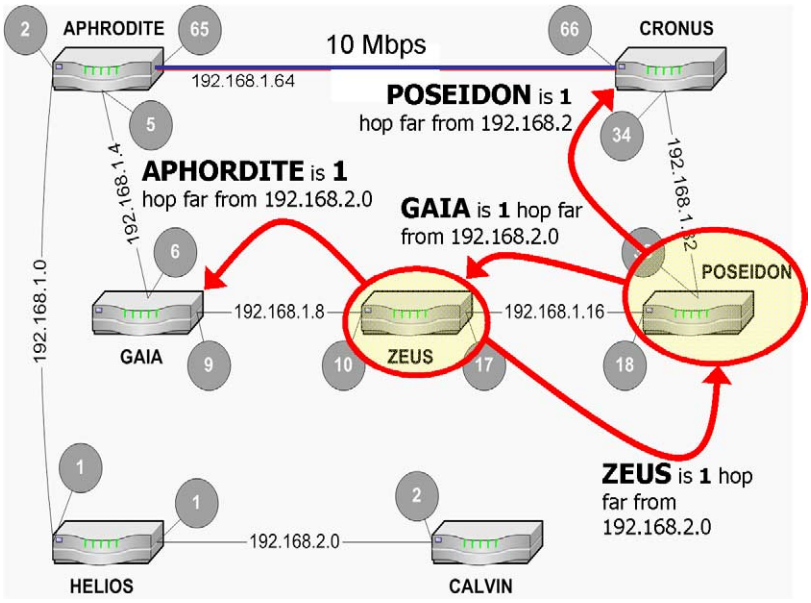


Fig. 18. Route Forcing on RIP: network test bed.

Route Forcing attack, we forced the traffic to follow a longer path from **CRONUS** to **APHRODITE,** to finally reach subnet **192.168.2.0**.

In order to obtain such result we used *spoof* on both **ZEUS** and **POSEIDON**. For each one we sent two kinds of malicious false packets:

- **POSEIDON** announced to **CRONUS** that the distance between **POSEIDON** and subnet **192.168.2.0** is equal to 1 hop; after that **POSEIDON** sent to **ZEUS** a packet containing the information that the distance between **GAIA** and subnet **192.168.2.0** is equal to 1 hop.
- **ZEUS** announced to **POSEIDON** that the distance between **ZEUS** and subnet **192.168.2.0** is equal to 1 hop; after that **ZEUS** sent to **GAIA** a packet containing the information that the distance between **APHRODITE** and subnet **192.168.2.0** is equal to 1 hop.

Repeating this packet sending activity every 3 s we forced the artificial route for the entire experimental time interval. The change we obtained in the path is depicted in Fig. 19.

5.3.1. Route Forcing on RIPv2: a positive perspective

In our research activities on QoS we used this last attack also as a rudimental traffic engineering mechanism. For instance, when we move from a path with a bottleneck equal to 10 Mbps to a path made of 100 Mbps links we can see this attack as a simple traffic engineering system.

In this section we present this attack also as a traffic engineering tool. Obviously, at the opposite side if we move from a path with 100 Mbps links to a path with a bottleneck we can consider it as an attack. But, in general, as previously said – apart of link capacity – we consider this as an attack because we move from a shorter to a longer path.

We show the results of our *Route Forcing* on RIPv2 in terms of throughput when we move from a path with a link bandwidth equal to 10 Mbps to a path made of links of 100 Mbps. Indeed, using D-ITG we measured the throughput on both paths (A and B) depicted in Fig. 19. In Figs 20–23 the TCP and UDP throughput are shown.

Thanks to the route changing we measured a traffic gain of a factor equal to ten in the B path with respect to the A path. In order to make clear the throughput variation we have also reported the throughput trend between **CRONUS** and **CALVIN** during the route forcing. Before the attack, TCP and UDP flows experienced a saturated path, whereas after the attack both flows relied on a non-saturated path (Figs 24 and 25).

6. Conclusion and issues for future research

In this work we presented a methodology to conduct experimental analysis of attacks against routing protocols. We showed how an attack can be performed and

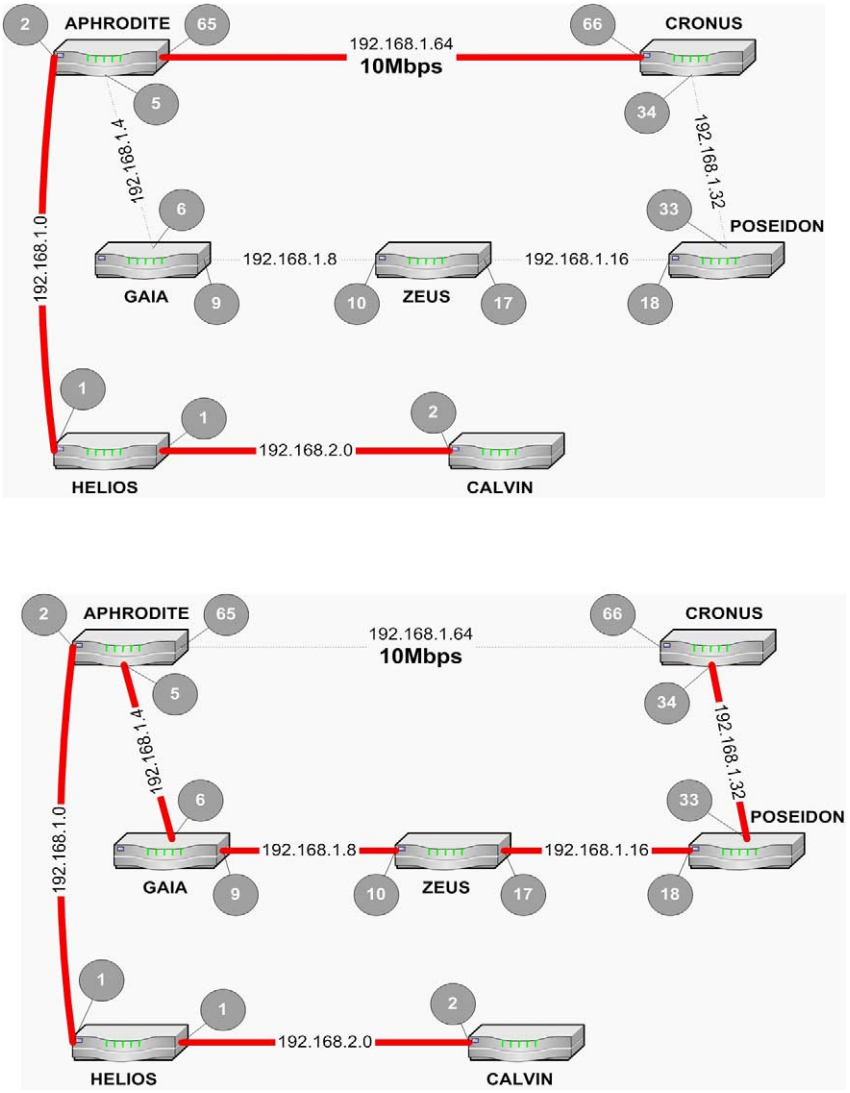


Fig. 19. Route Forcing on RIP: the path before (A path) and after (B path) the attack.

emulated, with which tools it is possible to carry out the emulation and analysis, and what are the impacts on the networks under attack. As for the effects, we pointed our attention on *router resources*, *network traffic*, *convergence time* and, in general, on *network behavior* before, during and after the attacks.

In particular, in this work we presented our practical analysis in the case of three distinct attacks against Interior Gateway Protocols that differ for the techniques in-

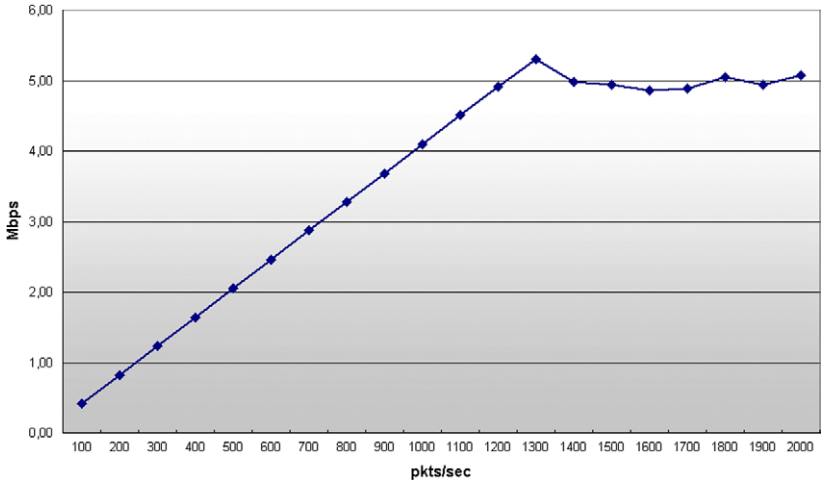


Fig. 20. Route Forcing on RIP: TCP throughput on A path.

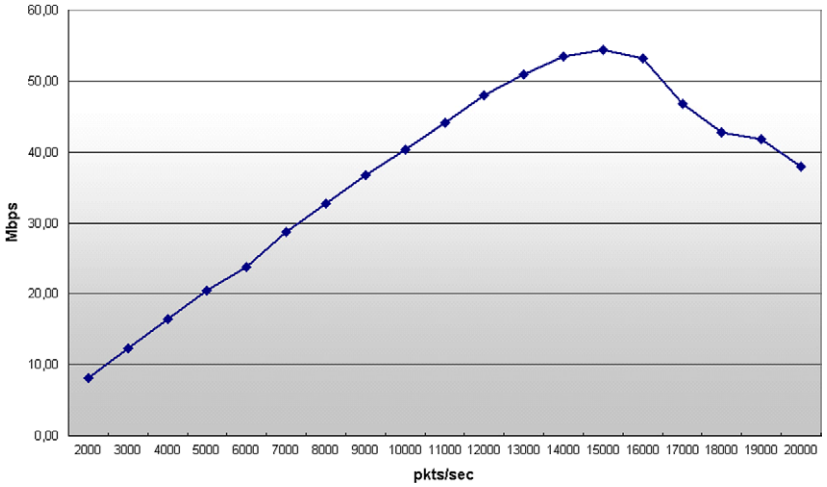


Fig. 21. Route Forcing on RIP: TCP throughput on B path.

volved and for their objectives. More precisely, we studied Route flapping on RIP, Denial of Service on OSPF through the Max Age Attack, and Route forcing on RIP.

We showed how it is possible to inject "malicious" packets into a network to cause service disruptions or network malfunctioning. We used a controlled test bed composed of common PCs and open source Operating Systems and networking tools. We showed results of various routing protocol attacks without taking into account issues related to the configuration of the packet forger (*Spoof*), routing demons (GNU *Zebra*) and the traffic generator (*D-ITG*): these are not straightforward tasks and rep-

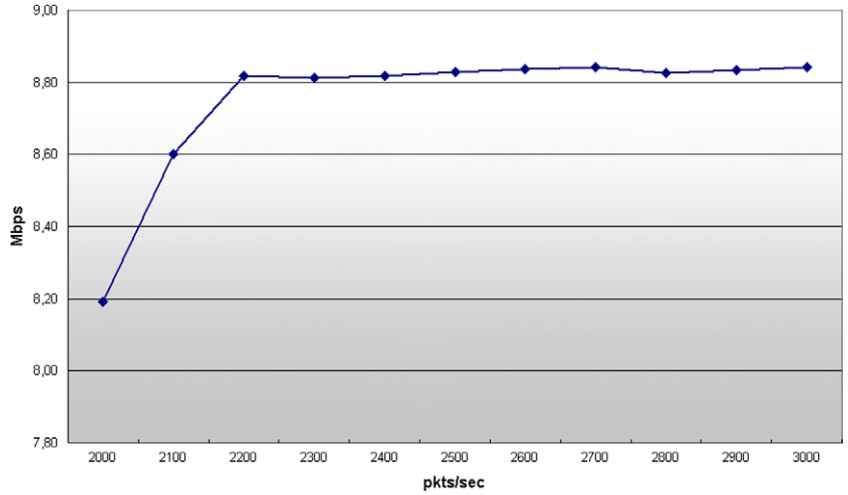


Fig. 22. Route Forcing on RIP: UDP throughput on A path.

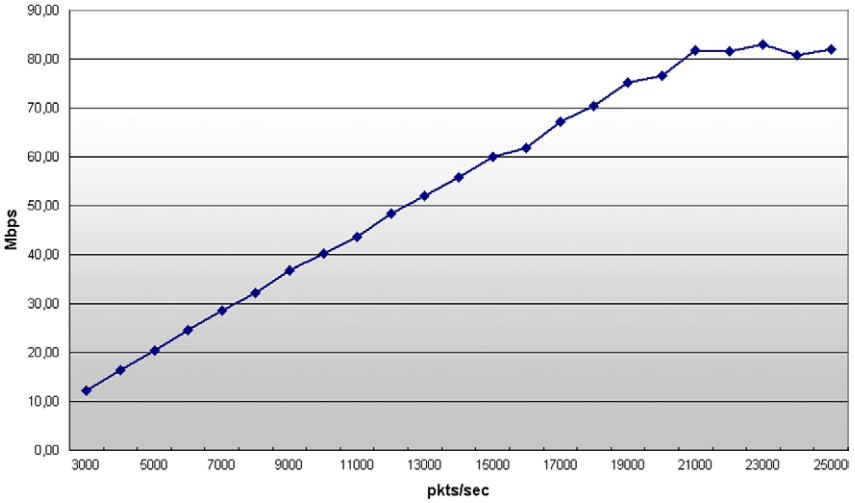


Fig. 23. Route Forcing on RIP: UDP throughput on B path.

resent an important aspect to take into account when this kind of experimental work is considered.

Thanks to the use of a controlled test bed in our lab, we configured a number of different network topologies where we performed the depicted experimentations. In this paper we showed results related to three different topologies, which allowed us to better understand the effects of the attacks to routing infrastructures. Along with

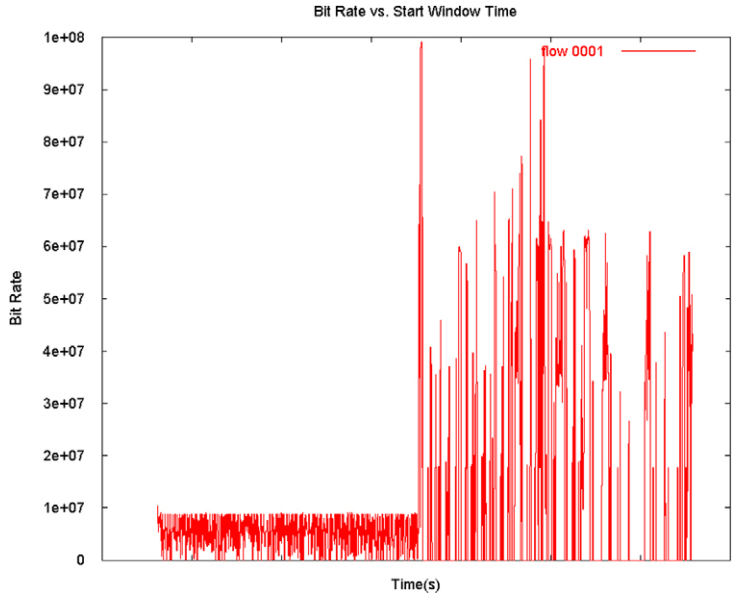


Fig. 24. Route Forcing on RIP: TCP throughput during the attack.

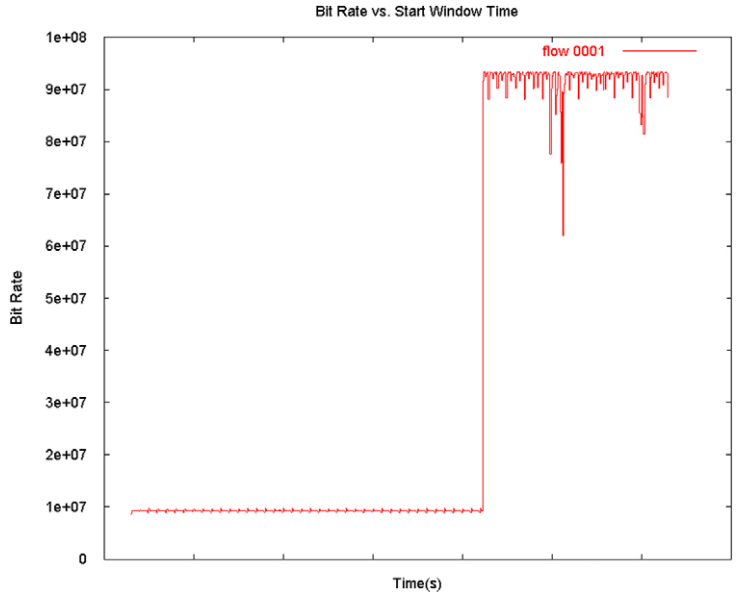


Fig. 25. Route Forcing on RIP: UDP throughput during the attack.

quantitative and numerical results we found in the presented scenarios, we think that the main contribution of this paper is a clear presentation of some of the effects, in terms of several different variables, of attacks against routing protocols. As side effect, in the case of similar networks topologies, results shown in this work can be used as a reference to develop a study framework on the effects of routing protocols attacks. Indeed we think that, a complete parametric characterization of such effects could be useful in the planning phase of innovative and resilient networks.

As said before, this work constitutes a proof of concept of how, with the presented approach, it is possible to experimentally analyze, qualitatively and quantitatively, the effects of routing attacks on the network infrastructure. Thus, while the presented methodology is of general validity, the numerical results obtained in the following experiments are important to give an idea of the performance degradation of the specific network configurations under test.

It is clear that changing network topologies (in terms of number of routers, links and their interconnections) the quantitative results will be different and we do not think that our numerical results can be extended to generic network topologies at much higher scale, but in our opinion we firmly believe that this is a first step in one of the fundamental analysis in the field of network security.

As regards future works, because our test-bed allows experiments on a small-scale, after this first stage, we will test the considered attacks on a realistic network of a much wider-scale (scalability analysis). In parallel, to cope with much wider-scale networks we also are working on a more complete, but less realistic, scenario in a simulation environment. Another issue for research is to enlarge our work to vendor-dependent routing architectures (Cisco, Juniper, . . .).

As already stated, our results are based on fixed topology configurations for specific attack scenarios (one configuration per attack). Out of the scope of this paper, our ongoing work deals with the investigation of the same attack scenarios for a number of different configurations and compare/contrast the results with the reference results shown in this paper. As for this last point, that we called the "test-bed dependencies", we are planning to perform more analysis over different network topologies in order to claim that results are configuration-independent or configuration-dependent. Also, keeping the same topology but changing the attacked routers would allow an interesting comparison.

Finally, as a side effect, it will be also interesting to investigate how the performance variations shown in this paper could be used as an input for innovative IDSs. More precisely, the output variables studied in this work could be considered like features for an anomaly based IDS working with soft-computing techniques. The experimental results of this kind of IDS could be compared with other existing IDSs [8,13,30,37] specifically designed for routing attacks.

Acknoledgements

A preliminary short version of this work has been published at the Workshop on Information Assurance WIA 2004. WIA 2004 was a workshop of IEEE IPCCC 2004.

This work has been carried out partially under the financial support of the "*Ministero dell'Istruzione, dell'Università e della Ricerca (MIUR)*" in the framework of the FIRB Project "*Middleware for advanced services over large-scale, wired-wireless distributed systems (WEB-MINDS)*", by Quasar PRIN project and finally by E-Next Network of Excellence. We would like to thank Giuseppe Giannini and Alberto Dainotti for their valuable support and anonymous reviewers for their suggestions.

References

[1] B. Kumar, Integration of security in network routing protocols, *SIGSAC Review* **11**(2) (1993).

[2] B. Kumar and J. Crowcroft, Integrating security in network routing protocols, *ACM SIGCOMM Computer Communication Review* **23**(5) (1993), 36–51.

[3] B.R. Smith and J.J. Garcia-Luna-Aceves, Securing the border gateway routing protocol, in: *Proc. Global Internet '96*, London, UK, 1996.

[4] B.R. Smith, S. Murthy and J.J. Garcia-Luna-Aceves, Securing Distance-Vector routing protocols, in: *Proceedings of IEEE ISOC Symposiums on Network and Distributed System Security*, 1997, pp. 85–92.

[5] B. Vetter, F. Wang and S.F. Wu, An experimental study of insider attacks for the OSPF routing protocol, in: *Proceedings of IEEE International Conference on Network Protocols (ICNP)*, 1997, pp. 293–300.

[6] C. Wilson, Protecting network infrastructure at the protocol level, *V1.0., Netw3.com Consulting*, December 15, 2000.

[7] D.M. Nicol, S.W. Smith and M. Zhao, Evaluation of efficient security of BGP route announcements using parallel simulation, *Elsevier Simulation Modelling Practice and Theory Journal, special issue on Modeling and Simulation of Distributed Systems and Networks* **12**(3-4) (2004), 187–216 (Revision of TR2).

[8] F. Jou, C. Sargor, F.S. Wu, H.Y. Chang and F. Wang, Design and implementation of a scalable intrusion detection system for the protection of network infrastructure, in: *Proceedings of DARPA Information Survivability Conference and Exposition*, Hilton Head, Island, SC, 2000.

[9] F. Majstor, Attack to routing protocols, in: *Talk at RSA Conference 2003.*

[10] F. Wang, B. Vetter and S.F. Wu, Secure routing protocols: Theory and practice, *Technical Report, North Carolina State University*, May 1997.

[11] F. Wang and S.F. Wu, On the vulnerability and protection of OSPF routing protocol, in: *Proceedings of IEEE Seventh International Conference on Computer Communications and Networks*, Lafayette, LA, 1998.

[12] FX, IRPAS – Internet work routing protocol attack suite, http://www.phenoelit.de/irpas/ (as of July 2005).

[13] H. Chang, S.F. Wu and Y.F. Jou, Real-time protocol analysis for detecting link-state routing protocol attacks, *ACM Transactions on Information and System Security (TISSEC)* **4**(1) (2001), 1–36, ISSN: 1094-9224.

[14] http://www.gated.org/ (as of July 2005).

[15] http://www.ethereal.com (as of July 2005).

[16] http://www.grid.unina.it/software/ITG (as of July 2005).

[17] http://www.monkey.org/∼dugsong/dsniff/ (as of July 2005).

[18] http://www.ouah.org/protocol_level.htm (as of July 2005).

[19] http://www.zebra.org/ (as of July 2005).

[20] J. Ioannidis, Origin authentication in interdomain routing, in: *Proceedings of the 10th ACM Conference on Computer and Communications Security*, 2003, pp. 165–178.

[21] L. Zhang, How to make the Internet more resilient? in: *Proc. NSF: The Workshop on Fundamental Research in Networking*, 2003.

[22] M. Baltatu, A. Lioy, F. Maino and D. Mazzocchi, Security issues in control, management and routing protocols, *Journal of Computer Networks* **34**(6) (2000), 881–894.

[23] M. Zhao, S.W. Smith and D.M. Nicol, Evaluating the performance impact of PKI on BGP security, in: *Proceedings of 4th Annual PKI Research Workshop (PKI 05)*, Gaithersburg, Maryland, April, 2005.

[24] The nemesis packet injection tool-suite, http://nemesis.sourceforge.net/ (as of July 2005).

[25] R. Braden, D. Clark, S. Crocker and C. Huitema, Report of IAB Workshop on Security in the Internet Architecture, RFC 1636.

[26] Reliable Software Group Website (University of California Santa Barbara), http://www.cs.ucsb.edu/∼rsg/ (as of July 2005).

[27] R. Hauser, T. Przygienda and G. Tsudik, Lowering security overhead in link state routing, *Journal of Computer Networks* **31** (2000), 885–894.

[28] R. Thayer, N. Doraswamy and R. Glenn, IP Security Document Roadmap, *RFC 2411*, November 1998.

[29] S. Avallone, D. Emma, A. Pescapè and G. Ventre, Performance evaluation of an open distributed platform for realistic traffic generation, *Performance Evaluation: An International Journal (Elsevier Journal)* **60**(1–4) (2005), 359–392, ISSN: 0166-5316.

[30] S. Cheung and K.N. Levitt, Protecting routing infrastructures from Denial of Service using cooperative intrusion detection, in: *Proceedings of New Security Paradigms Workshop*, Cumbria, UK, 1997.

[31] S.L. Murphy, Digital signature protection of the OSPF routing protocol, in: *Proceedings of IEEE ISOC Symposiums on Network and Distributed System Security*, 1996.

[32] S.M. Bellovin, Routing security, *Talk at British Columbia Institute of Technology*, June 2003.

[33] S.M. Bellovin, Security problems in the TCP/IP protocol suite, *ACM Computer Communications Review* **19**(2) (1989).

[34] S.M. Bellovin and E.R. Gansner, Using link cuts to attack Internet routing, *ATT Research, Technical Report*, 2004.

[35] S. Kent, C. Lynn, J. Mikkelson and K. Seo, Secure Border Gateway Protocol (S-BGP), in: *Proceedings of ISoc Network and Distributed Systems Security Symposium, Internet Society*, Reston, VA, 2000.

[36] S. Kent, Securing the border gateway protocol: a status update, in: *Proceedings of Seventh IFIP TC-6 TC-11 Conference on Communications and Multimedia Security*, 2003.

[37] S. Wu, H. Chang, D. Qu, F.W.F. Jou, F. Gong, C. Sargor and R. Cleaveland, JiNao: Design and implementation of a scalable intrusion detection system for the OSPF routing protocol, *Journal of Computer Networks and ISDN Systems*, 1999.

[38] V. Mittal and G. Vigna, Sensor-based intrusion detection for intra-domain distance-vector routing, in: *Proceedings of CCS 2002, 9th ACM Conference on Computer and Communications Security*, Washington, DC, 2002.

[39] Y. Hu, A. Perrig and D. Johnson, Efficient security mechanisms for routing protocols, in: *Proceedings of Network and Distributed Systems Security 2003.*

[40] Y. Ohara, M. Bhatia, N. Osamu and J. Murai, Route flapping effects on OSPF, in: *Proceedings of SAINT Workshops 2003*, 232–237.