Security Issues in Control, Management and Routing Protocols

Madalina Baltatu, Antonio Lioy, Fabio Maino, Daniele Mazzocchi
Dipartimento di Automatica e Informatica
Politecnico di Torino
Torino (Italy)

Abstract— **The TCP/IP suite, the basis for today's Internet, lacks even the most basic mechanisms of authentication. As usage of the Internet increases, its scarcity of built-in security becomes more and more problematic. This paper describes serious attacks against IP control and management protocols with an accent on the ICMP protocol, as well as some of the well-known vulnerabilities of the inter-domain routing protocols. All the presented attacks have at least one common feature: they exploit intrinsic IP security flaws. The paper also discusses various solutions to these security breaches, including the use of IPsec, which currently offers cryptographic security services for the Internet infrastructure.**

Keywords—**Denial of Service, authentication, IP Security, routing security, public key infrastructure.**

I. Introduction

The concept of security is traditionally connected to the exigence of protecting sensitive data from unauthorized access, but nowadays security is frequently approached from a different perspective. With the growing use of the Internet infrastructure for commercial applications, the demand for quality of service has rapidly increased. Quality of Service (QoS) is one of the emerging paradigms in Internet, and seems to be the corner stone for more and more network services. An increasing number of applications need complex, reliable control protocols for guaranteeing QoS. As a consequence, the need for security in network infrastructure is stronger than ever. The main requirements are data origin authentication and data integrity for IP and for control and routing protocols.

As long as Internet is based on TCP/IP, its "insecurity" is inherent. IP was not designed with security in mind, and neither were its routing, control and management protocols. Some of the most serious security flaws of the TCP/IP protocol suite exist because hosts rely on IP source address for authentication. Others exist because network control mechanisms and routing protocols have minimal or non-existent authentication. During the last years, more working groups of the Internet Engineering Task Force made considerable efforts for introducing security mechanisms based on cryptography at different layers of the TCP/IP stack. One of the most significant work is the definition of a security architecture for the Internet Protocol, shortly IPsec [1]. This paper analyzes the use of IPsec as a possible solution to various attacks at the network infrastructure.

The goal of the paper is to present important security aspects inherent to protocols which play fundamental roles in the Internet architecture. Section 2 provides an in-depth analysis of the ICMP protocol together with an updated list of protocol attacks, and presents possible solutions to hinder these attacks. Section 3 gives a brief description of IGMP, together with its potential security risks. An analysis of various security mechanisms for routing protocols follows in Section 4. Basically, we discuss the security extensions defined for two commonly used intra-domain routing protocols, RIP and OSPF. Finally, the state-of-the-art of the network security mechanisms is presented.

II. Attacks Using ICMP Messages

ICMP, the Internet Control Message Protocol [2] is an integral part of any IP implementation. ICMP messages typically report errors encountered while processing IP datagrams. They are sent in several situations: when a datagram cannot reach its destination, when any of the gateways on the datagram's path does not have the buffering capacity to forward the datagram, or when a primary gateway can direct the host to send traffic on a shorter route. ICMP informational messages are also useful for network monitoring: `ping` and `traceroute` use this type of messages.

ICMP messages are sent using the basic IP header (see Figure 1). The first octet of the IP payload is the ICMP type field. The value of this field determines the format of the remaining data. Figure 2 summarizes the types currently defined. The RFCs that have an experimental status are marked with an asterisk. The code field depends on the message type, and is used to create an additional level of message granularity. The ICMP payload contains the IP header plus 64 bits of the original datagram's data. This

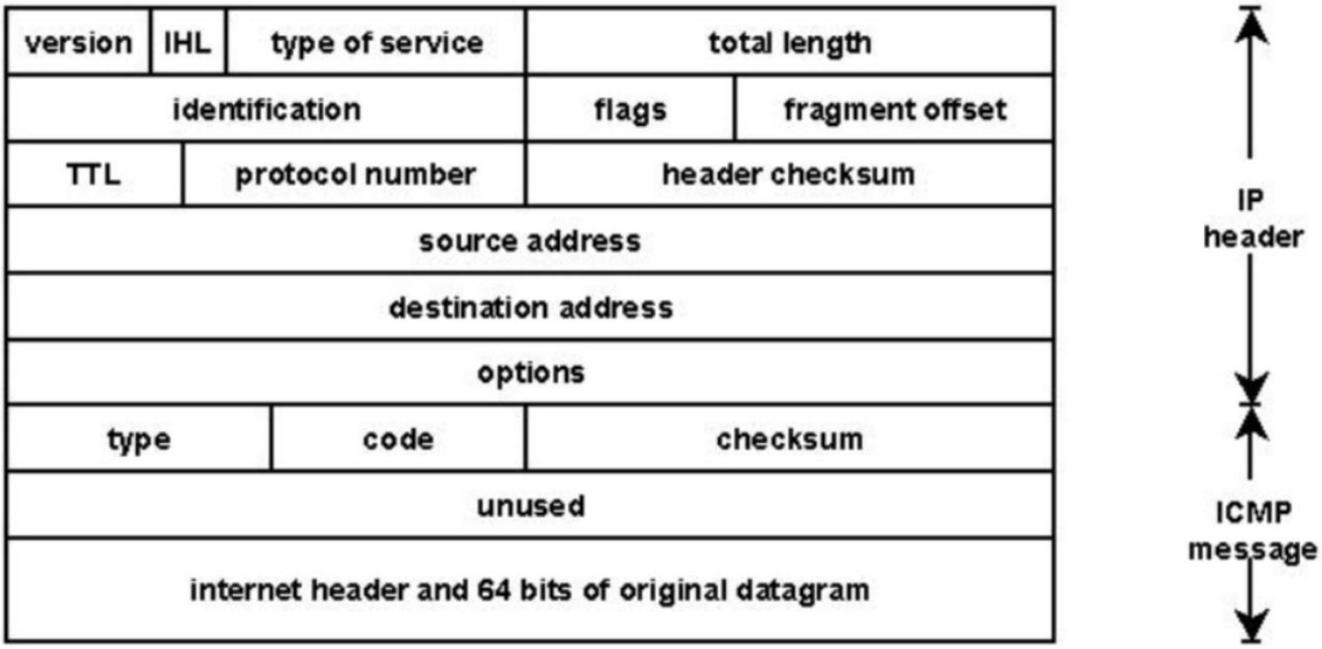

Figure 1. The ICMPv4 packet.

data is used by the receiving host to match the ICMP message to the appropriate process/application.

The next sections describe the most frequently used ICMP messages together with the security issues they arise. Malicious use of almost all of these messages results in DoS attacks: the communication between the victim system and its intended destination is blocked or at least slowed down.

A. Denial of Service via ICMP Messages

The *destination unreachable* message is sent by a gateway to the source host of a datagram if the network specified as destination by the incoming datagram is not reachable. Also, if at the destination host the IP module cannot deliver the datagram because the indicated protocol module or process port is not active, the destination host sends a *destination unreachable* message to the source host. When a datagram must be fragmented to be forwarded by a gateway, but the "Don't Fragment" flag is set, the gateway discards the datagram and returns a *destination unreachable* message.

The *time exceeded message* is usually sent to a host when a gateway processing a datagram originated by that host finds that the "time to live" field is zero. The datagram is discarded. Also, if a host reassembling a fragmented datagram cannot complete the reassembly within its time limit due to missing fragments, it discards the datagram, and sends a *time exceeded message* to the source host of the datagram.

If a gateway or a host processing a datagram finds a problem with the header parameters such that it cannot complete the processing, it discards the datagram, and can notify the source host via the *parameter problem* message.

Gateways discard IP datagrams if they do not have the buffer space needed to queue the datagrams for output to the next network on the route to the destination network. In this situation the gateway may send a *source quench* message to the source host. A destination host may also send a *source quench* as a request to the source host to cut back its transmission rate.

All these messages can hinder or slow down the communication if they are forged. In most of the cases, the application that receives such messages stops and returns an error code, or reduces its transmission rate. DoS attacks exploit either the *time exceeded* or *destination unreachable* messages. The attacker need to know only the local and remote port numbers of a TCP connection. Any of the above error messages aimed at that connection may be then forged and sent to one end of the connection, pretending to come from the other end. The connection will be broken. Information about established TCP connections is sometimes available through the `netstat` utility.

These attacks can be partially defended against if a host is careful about checking that an error message really refers to a particular connection. For applications that use TCP, this implies verifying that the ICMP packet con-

message type	*message name*	*defined in*
0	Echo Reply	RFC-792
3	Destination Unreachable	RFC-792
4	Source Quench	RFC-792
5	Redirect	RFC-792
8	Echo Request	RFC-792
9	Router Advertisement	RFC-1256
10	Router Solicitation	RFC-1256
11	Time Exceeded	RFC-792
12	Parameter Problem	RFC-792
13	Timestamp	RFC-792
14	Timestamp Reply	RFC-792
15	Information Request	RFC-792
16	Information Reply	RFC-792
37	Domain Name Request	RFC-1788*
38	Domain Name Replay	RFC-1788*
40	Security Failures	RFC-2521*

Figure 2. ICMPv4 message types.

tains a plausible sequence number in the returned-packet portion. These checks are less applicable to UDP, though. Furthermore, if the attacker has previously sniffed the traffic associated with A TCP connection, he/she can include in the forged *destination unreachable* message an “original IP header and 64 bit of data” which will resist to all checks. In this case the DoS cannot be avoided.

The only viable solution against such attacks is to authenticate the source of the error message. The authentication mechanism has to be implemented at the IP layer, for the ICMP to make use of it. One possible approach is to offer IPsec cryptographic authentication to ICMP. This is not a trivial task to accomplish, mainly because of the sporadic and unpredictable nature of the ICMP traffic. A detailed discussion of the IPsec solution is provided in Section II-G.

B. Re-routing with ICMP Route Redirect

The ICMP *redirect* message is sent by a gateway in the situation illustrated in Figure 3. The gateway G1 receives an IP datagram from a host on its attached network NET1. G1 checks its routing table and obtains the address of the next gateway (G2 in Figure 3), on the route to the datagram’s destination network, NET2. If G2 and the host identified by the source address of the datagram are on the same network, a *redirect* message is sent to the host. The message advises the host to send its traffic for network NET2 directly to G2 as this is a shorter path to the destination. G1 then forwards the original datagram to its Internet destination.

ICMP *redirects* are a destructive instrument of attack. Since these messages are used by gateways to advise hosts of better routes, they can be abused in the same way in which a routing protocol is. However, *redirects* are harder to abuse than other messages because they must be tied to a particular, existing connection. They cannot be used to make an unsolicited change to the host’s routing tables. Furthermore, *redirects* can be taken into consideration within a limited network topology: they must only be sent by the first gateway in the path to the originating host.

Suppose, though, that an intruder has penetrated a secondary gateway available to a target host T, but not the primary one. Assume further that the intruder sets up a false route to destination D through that compromised secondary gateway. The following attack scenario can be imagined. First, the intruder sends a false TCP open packet to host T, claiming to be from D. T will respond with its own open packet, routing it through the secure primary gateway. While this is in transit, a false *redirect* may be sent, claiming to be from the primary gateway, and referring to the bogus connection. This packet will appear to be a legitimate control message, hence the route change it contains will be accepted. If the target host makes this change to its global routing table, rather than just to the per connection cached route, the intruder may proceed with spoofing host D.

Malicious use of *redirects* results in Denial of Service. If hosts do not perform enough validity checks on such messages, the impact of the attack is quite serious, since

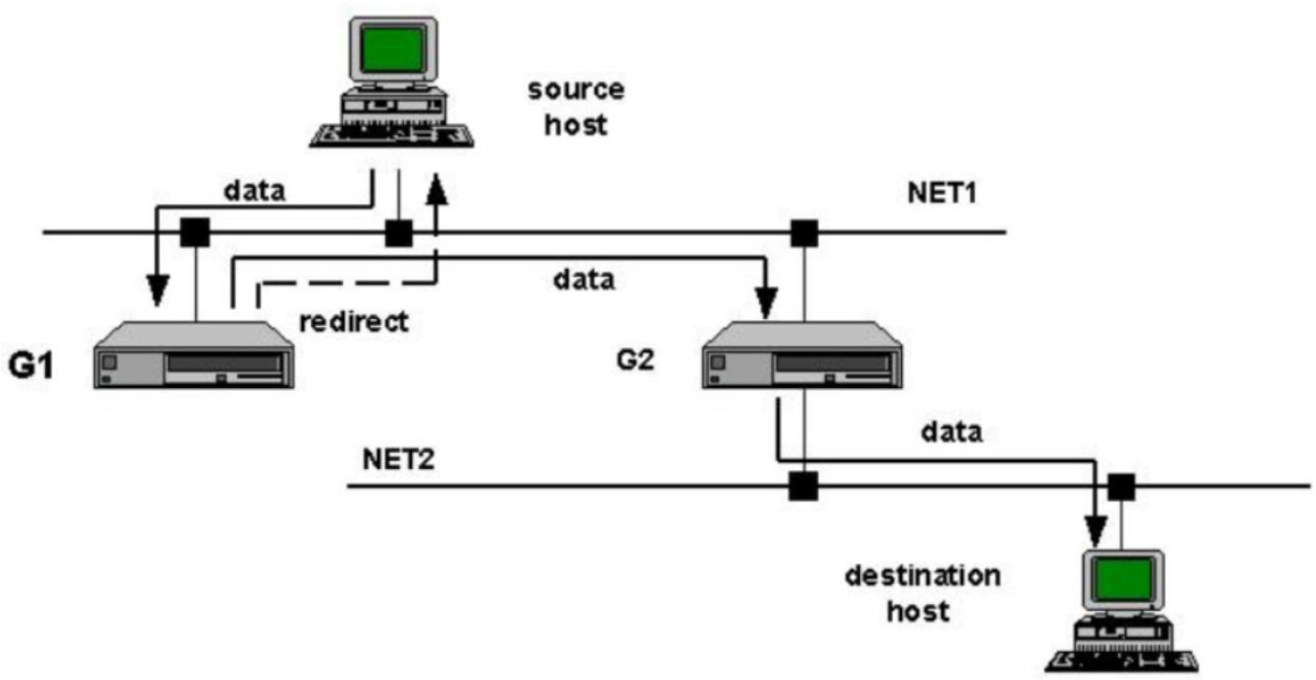

Figure 3. Use of redirect messages.

route entries created after receiving a *redirect* will not expire in time. Besides, the attack can be launched from anywhere, no access to local network is required. If the target system accepts *redirects*, it can be stopped from communicating with any particular address that is not on the same subnet as the target. Another plausible scenario for an attacker is to determine the communications between the victim host and a particular destination system to pass trough the attacker's system. The subverted traffic can then be easily sniffed.

Defense against these attacks is simple: if dynamic routing is not really needed, it is a legitimate option to disable *redirects*, even though this will make the systems less compliant with [3] ("systems MUST follow ICMP redirects unless they are routers"). A sensible option is to restrict route changes to the specified connection. The global routing table should not be modified in response to *redirects*. A better solution is to accept only IPsec authenticated *redirects*, and is up to the local network administrators to decide if the overhead introduced by IPsec processing is acceptable (see Section II-G).

C. Attacks Using ICMP Router Discovery Messages

The ICMP *router discovery* messages are an ICMP extension to enable hosts attached to multicast or broadcast networks to discover the IP addresses of their neighboring routers.

Learning the operational router IP address via the *router discovery* messages is an alternative to reading it from a configuration file maintained manually (a method which can be a significant administrative burden, and which has the disadvantage that cannot track dynamic changes in router availability), or to discover it by listening to routing protocol traffic (in this case the hosts have to recognize the particular routing protocols in use, which vary from subnet to subnet and which are subject to change at any time).

The messages used by the ICMP Router Discovery Protocol (IRDP) are:

- *router advertisement*: each router periodically multicasts a *router advertisement* from each of its multicast interfaces, announcing IP address(es) of that interface. Hosts discover the address of their neighboring routers simply by listening for advertisements.
- *router solicitation*: when a host attached to a multicast link starts up, it may multicast a *router solicitation* to ask for immediate advertisements, rather then waiting for the next periodic one to arrive.

These messages do not constitute a routing protocol because they do not inform hosts which of their neighboring routers is best to reach a particular destination. The IRDP protocol does not have any form of authentication, making it impossible for end hosts to verify whether or not the information they receive is valid. Therefore, it is possible for any system attached to a link to masquerade as a default router for hosts attached to that link. Any traffic sent to such an impostor is vulnerable to eavesdropping, to denial of forwarding service, and to modification by insertion, deletion, or alteration of packets. As such, the malicious use of IRDP might result in a number of common attacks like:

- passive monitoring: an attacker on the same network as the victim can re-route the outbound traffic of vulnerable systems through them, which will allow them to monitor or record one side of the conversation.
- man-in-the-middle: the attacker can act as a proxy between the victim and the end host. The victim, while thinking that it is connected directly to the end host, it is actually connected to the attacker, which is connected to the end host and is feeding the information through.
- denial of service: remote attackers spoofing IRDP packets can add bad default-route entries into a victim's routing table. Because the victim's system would be forwarding data to the wrong address, it will be unable to reach other networks.

The ICMP Router Discovery Advisory document [4] describes DoS attacks recently performed on Windows systems acting as DHCP clients, on which IRDP comes enabled by default. By spoofing *router advertisements*, an attacker can remotely add default route entries on a system. The default route entry added by the attacker will be preferred over the default route obtained from the DHCP server. This attack is documented in [5], which states that the attack succeeds if launched by an attacker on the same network as the victim. This is not true when the *router advertisement* contains two or more router addresses [4]. If a Windows system is configured as a DHCP client, any *router advertisement* will be accepted and processed. Once an advertisement is received, Windows checks to see how many gateway entries the packet contains. If the packet contains only one entry, it checks to make sure the IP source address of the advertisement is inside its own subnet. If it is, the router address entry inside the advertisement is checked to see that it also is within the subnet. If so, a new default route entry is added. If the address is outside the subnet, the advertisement is silently ignored. However, if the *router advertisement* contains two or more router addresses, the host will process the packet even though the IP source address is not local. If the host finds a router address inside the advertisement that is inside the host's subnet, it will add a default route entry for it. Because the host does not care about the IP source address of the advertisement as long as it has more than one entry, attackers can create bogus IRDP packets that will bypass anti-spoofing filters.

Before the host can add a new default route entry, it has to determine the route metric. On Windows95/98, normal default route entries obtained from a DHCP server have a metric of 1. In order to determine the metric for the default route entry obtained via IRDP, the Windows host subtracts the advertisement's preference value from 1000. By creating a *router advertisement* with a preference of 1000, the default gateway route added will have a metric of 0, making it the preferred default route.

SunOS systems will also intentionally use IRDP under specific conditions. For Solaris 2.6, the IRDP daemon, `in.rdisc`, will be started if the following conditions are met:

- the system is a host, not a router
- the system did not learn a default gateway from a DHCP server
- the system does not have any static routes
- the system does not have a valid `/etc/defaultrouter` file.

The immediate fix to these problems is to block at the external router/firewall all ICMP type 9 and type 10 packets. This should protect against remote DoS attacks. Another possibility is to provide authentication to these messages. [5] specifies that the *router advertisement* message format is defined so as to allow additional information to be carried within the message. Therefore, digital signatures or some other form of authentication information could be attached to these messages.

D. Attacks via ICMP Informational Messages

The *echo* and *echo reply* are ICMP messages that implement the colloquially known `ping` service, mainly used for network monitoring and diagnosis purpose. The following sections present two common attacks that make use of ICMP `echoes`. Additionally, the first attack exploits a failure in the TCP/IP implementation on some systems.

"Ping" as a vehicle of attack

Experience shows that it is possible to crash or reboot a large number of systems by sending a "ping" of a certain size from a remote machine. This is a serious problem, mainly because the attacker needs to know nothing about the machine other than its IP address.

The TCP/IP implementation allows for a maximum packet size of up to 65536 octets, containing a minimum of 20 octets of IP header information, and zero or more octets of optional information, with the rest of the packet being data. It is known that some systems will react in an unpredictable manner when receiving oversized IP packets. Reports indicate a range of reactions including crashing, freezing, and rebooting.

In particular, the reports received by [6] indicate that ICMP packets issued via the `ping` command have been used to trigger this behavior. Most implementations of `ping` will not allow an invalid datagram like this to be sent. Among the exceptions are Windows '95 and Windows NT, although they are certainly not the only ones. An *echo* message is sent inside the IP packet [2], consisting of eight octets of ICMP header information followed by the number of data octets in the ping request. Hence, the maximum allowable size of the data area is 65535 - 20 - 8 = 65507 octets.

It is possible to send an "illegal" *echo* with more than 65507 octets of data, due to the way the fragmentation is done. The fragmentation relies on an offset value in each fragment to determine where the individual fragment goes upon reassembly. Thus, on the last fragment it is possible to combine a valid offset with a suitable fragment size such that (offset + size) > 65535. Since typical machines do not process the packet until they have all fragments and have tried to reassemble it, there is the possibility for overflow of 16 bit internal variables, which can lead to system crashes, reboots, or kernel dumps.

If no OS patch is available, and the main concern is `ping` from outside the network, the best quick-fix solution is to block `ping` at the firewall. A better solution than blocking all ICMP `echoes` is to block only fragmented ones. This will allow the common 64 byte `pings` through on almost all systems, while blocking any bigger than the MTU size of a link.

"Smurf" attacks

The "smurf" attack (documented in [7]) is the most recent in the category of network-level attacks against Internet hosts. An aggressor sends a large amount of ICMP *echo* traffic at broadcast addresses, all of it having a spoofed source address of the victim. The situation is illustrated in Figure 4. If the routing device delivering traffic to those broadcast addresses performs the "IP broadcast to layer 2 broadcast" function, most hosts on that IP network will take the ICMP *echo* and reply to it with an *echo reply* each, multiplying the traffic by the number of responding hosts. On a multi-access broadcast network, there could potentially be hundreds of machines to reply to each packet.

There are two parties which are hurt by this attack: the intermediary (broadcast) devices and the spoofed address target, the victim machine. The victim is the target of a large amount of traffic the broadcast devices generate. The initiators of these attacks rely on the ability to "source spoof" traffic to the intermediary broadcast networks in order to generate the traffic which causes denial of service.

To stop this, all networks should perform source address checks either at the edge of the network where users connect or at the edge of the network with connections to the Internet. These checks will defeat the possibility of source spoofed packets from entering from leaf networks, or leaving for Internet (see Figure 4). To stop being an intermediary we have to take into account the fact that this attack relies on the ability of the router serving a large multi-access broadcast network to frame an IP broadcast address into a layer two broadcast address. The router may have an option to disable receiving traffic directed to network-prefix addresses and must have an option to disable forwarding broadcasts directed to network-prefix addresses.

Hosts can be patched to refuse to respond to broadcast ICMP *echoes*. [3] specifies that ICMP *echoes* for an IP broadcast or an IP multicast address may be silently discarded. This neutral stipulation results from a passionate debate between those who feel that *echoes* to a broadcast address provides a valuable diagnostic capability and those who feel that misuse of this feature can too easily create packet storms.

E. Security Failure Messages

These messages indicate failures when using IP Security Protocols (AH and ESP). As [8] states, for a statically configured Security Association (SA), these messages indicate that the related SA has to be manually reconfigured, or that an unauthorized operation is attempted. *Security failure* messages may also be used to trigger automated negotiation of session-keys.

The DoS attacks performed using ICMP messages usually succeed because the receiver of such messages does not maintain enough information on the communication the messages should be related to. Therefore, *security failure* messages have to be carefully verified to ascertain that they include information that matches a previously sent datagram. Besides, [8] advises that, when a prior SA between the parties has not expired, these messages should be sent with authentication. A dynamic SA must not be established, though for the only purpose to authenticate *security failures*, since this could be used for a very serious DoS attack. A target host may be flooded with forged IPsec packets from random IP Sources and have it start up numerous useless key management session to authentically inform the presumed senders of the error.

Security failures provide sufficient data to determine that they are in response to previously sent messages. Therefore, a recipient can accept all authenticated and unauthenticated *security failure* messages, since accurate check of the message content gives enough information to validate the message. This is due to the fact that *security failures* are slightly different from other ICMP messages: besides the IP header of the original packet, they also contain all IPsec headers that were present in the original packet. These headers would give enough information

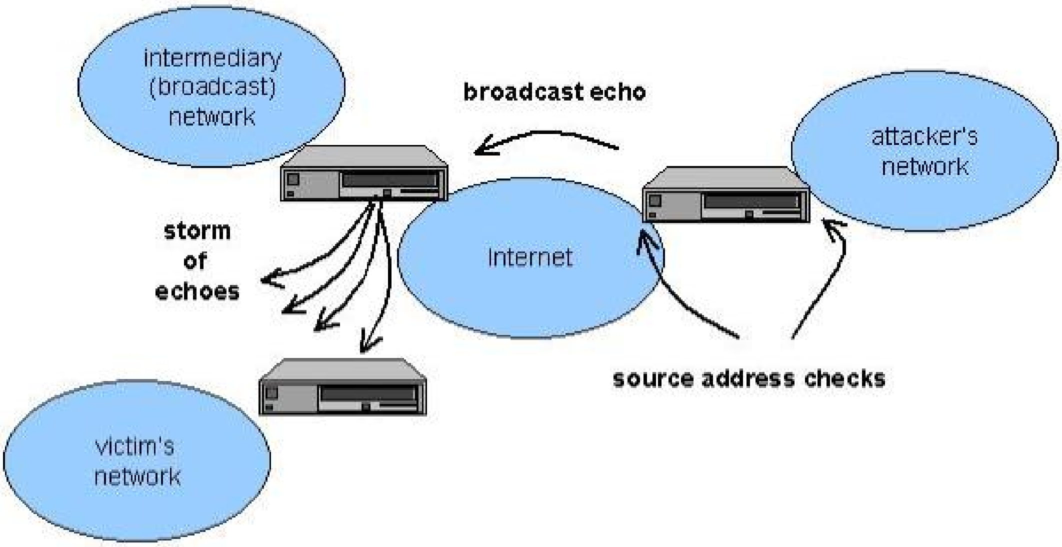

Figure 4. Checking IP source address.

to identify the original, offending packet.

F. ICMP DNS Messages for Reverse Look-up

These messages are intended to be used for learning the Fully Qualified Domain Name associated with an IP address. Even though a mechanism to perform address to domain name resolution exists (the IN-ADDR domain of the DNS), [9] sustains the utility of having a more efficient one. The proposed scheme suggests that each unicast address be queried directly for its corresponding domain name, by means of the ICMP *domain name request* message. The queried destination should respond with its domain name included in a *domain name replay*. The advantages of using this mechanism are that the naming is under the same administration as the address assignment, and that the queries are distributed in the same manner as IP routing. On the other hand, the security risks are too important because of the lack of authentication of the parties involved.

We consider ICMP-based domain name resolution completely unacceptable from the security point of view, unless additional authentication mechanisms are provided. [9] suggests the use of IPsec for authentication.

G. ICMP Protected with IPsec

In order to offer IPsec protection to ICMP, the source and the destination systems of the ICMP traffic have to establish the required IPsec SA. For an application which does not use IPsec services, it can be an unacceptable overhead to negotiate SAs only for the purpose of transmitting ICMP messages.

For the network topology shown in Figure 5, suppose that G2 receives an IP datagram from H1, for the destination H2. The link which connect G2 to the H2's network is temporary out of use, so G2 has to send a *destination unreachable* message to H1. If no SA exists for the communication between H1 and H2, G2 has to establish one with G1 or directly with H1, for the purpose of sending an authenticated *destination unreachable*. In IPsec the only way to do this is to start an IKE (Internet Key Exchange [10]) negotiation, but there are still few chances for this to succeed, unless additional conditions are fulfilled. Poten-

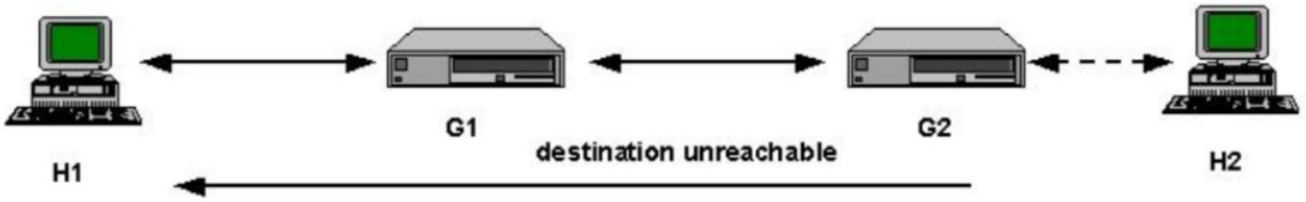

Figure 5. SPP operation example.

tial problems are the G1 admission policies, the existence of an IKE authentication method on which all part will agree, and the list can continue.

Even if IPsec is used for the communication between H1 and H2, it is still difficult to handle ICMP messages that might be generated by some of the parties involved. The IPsec SPD (Security Policy Database [1]) provides for a set of selectors for choosing the appropriate IPsec SA to process an incoming IP datagram. There is currently no standard way for these selectors to handle ICMP types and codes. Recent debates in [11] discuss the overloading of source and destination port selectors, for specifying ICMP type and code. Once these SPD selectors are defined, there are four possible methods of handling ICMP messages:

- Discard any ICMP message.
- Explicit ICMP SA: when G2 receives an ICMP error message for H1, G2 forwards it using an SA established to accept ICMP messages of this type and code. If such an SA does not exist, and if G2 and G1 policies permit, an SA is negotiated with IKE. The proposal parameters for this SA have to be at least as strong as any other SA that is used between the set of end-points to which the ICMP message is related.
- Implicit ICMP SA: G2 forwards an ICMP error message for H1 using the SA that was used to send the offending IP packet.In this case the main difficulty is finding the right SA to associate with the ICMP message. The data contained in the ICMP payload (“IP header and 64 bits or the original datagram”) should be used to discover the incoming SA on which the offending packet arrived. The corresponding outgoing SA has to be used to send the ICMP packet.
- IKE ICMP: instead of forwarding the ICMP packet, G2 sends this message to the key management system. The key management daemon will then send an IKE Notify message to the other end. Appropriate Notify messages have to be defined for each ICMP type.

III. IGMP RISKS

IGMP (Internet Group Membership Protocol) is used by IP hosts to report their multicast group membership to any neighboring multicast router. In order to outline the security flaws of IGMP, we consider only the messages exchanged between hosts and routers. The IGMP protocol is described in detail in [12].

There are three types of IGMP messages of concern to the host-router interaction:

- *membership query*: is issued by multicast routers, and has two further sub-types. The *general query* is used to learn which groups have members on an attached network, while the
- *group-specific query*: is used to learn if a particular group has any members on an attached network. The *general query* is addressed to all-systems multicast group (224.0.0.1), while the *group-specific query* message is sent to the particular group multicast address. With respect to its attached networks, a multicast router can assume one of two roles: querier or non-querier. There is normally only one querier per physical network, the one that has the lowest IP address.
- *membership report*: hosts multicast this type of message to the group to report their membership to that particular group. The report may be unsolicited (when a host joins a group), or in response to a *group query*.
- *leave group*: this message is issued by a host which leaves a multicast group. If the host was the last to reply to a *query* with a *membership report* for that group, the host sends the *leave group* message to all-routers multicast group (224.0.0.2).

[12] studies the effects that forged IGMP messages may have on multicast hosts and network traffic. A forged *query* from a machine with lower IP than the current querier will cause querier duties to be assigned to the forger. This might probably result in an DoS attack for some members of the multicast group if some additional conditions are fulfilled. A forged query message sent to a

group with members will cause the hosts which are members of the group to report their membership. This causes a small amount of extra traffic on the LAN, but causes no protocol problems.

A forged *report* message may cause multicast routers to think there are members of a group on a subnet when there are not. Forged *report* messages are troublesome if the source address of the *report* is spoofed.

A forged *leave* message will cause the querier to send out *group-specific queries* for the group in question. This causes extra processing on each router and on each member of the group, but cannot cause loss of desired traffic.

IV. Security Issues in Routing Protocols

Since routing protocols are responsible for maintaining network connectivity for all TCP/IP traffic, we can surely say that routing security is an essential issue for the entire network infrastructure.

The most secure protection for the routing mechanism is to adopt a static routing scheme. This scales well in a local network, of medium size, with no special QoS requirements. On the other hand, static routing is inappropriate for applications that need dynamic variation of packet flows, like it happens in high QoS services. In these cases, the use of a suitable and reliable routing protocol is mandatory. QoS is the main reason for which special attention has been lately accorded to defining authentication mechanisms for the routing protocols. 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, routing tables) between routers of an Autonomous System (AS). An AS is a group of routers exchanging routing information via a common routing protocol. Both RIP and OSPF are mainly affected by the lack of a mechanism for guaranteeing integrity and authentication of the information exchanged.

Before examining the recently proposed security mechanism for these two routing protocols, we will summarize the security threats the routing protocols are commonly subject to (a detailed presentation can be found in [13]). Attacks at routing protocols are usually divided in two types: insider and outsider attacks. Outsider attacks involve an intruder masquerading as a router who distributes fabricated, delayed or incorrect routing information. Insider attacks are mounted by a subverted or compromised router. The latter type of attacks is more dangerous since there is one of the insider routers to distribute false routing information.

Such attacks may have serious consequences on the network infrastructure and on the end-to-end communications. Feeding false routing information into an AS may compromise the routing table of some of the AS routers, which will result in DoS on the hosts which trust that router. This means that some hosts may not be able to reach some legitimate destinations, or the traffic flows for some particular destinations are deviated through suboptimal routes. The packets which follow routes that subverted routers indicate may be subject to eavesdropping and modification.

The common countermeasures [14] for these attacks are:

- cryptographic checksums to protect against outsiders generating fraudulent routing messages,
- per-message sequence numbers and timestamps to protect against outsiders re-ordering or delaying genuine routing information,
- strong origin authentication, using shared-key or public key cryptography.

A. Shared Key Authentication for Routing Protocols

One of the first proposals for providing security services to routing protocols was to use a shared key-based authentication scheme. The next sections provide an overview of this mechanism defined for both RIP and OSPF.

RIP-2 Authentication Scheme

RIP is one of the routing protocols based on the Bellman-Ford (or distance vector) algorithm. RIP is extensively used for exchanging routing information among gateways. This protocol is intended to be used as an interior gateway protocol (its scope is an AS), in networks of moderate size. For moderate/small networks, RIP has very little overhead in terms of bandwidth used, and in terms of configuration and management time.

The basic RIP protocol is not a secure protocol. An extensible authentication mechanism has been recently incorporated into the protocol enhancements. [15] proposes that RIP-2 use an authentication algorithm similar with the one proposed for SNMP Version 2, augmented by a sequence number. The mechanism is intended to be algorithm-independent, but for the moment keyed MD5 is the standard authentication algorithm for RIP-2.

While this mechanism is not unbreakable, it provides an enhanced probability that a system being attacked will detect and ignore hostile messages. Authenticated RIP-2 messages contain the output of an one-way function of the message and a secret RIP-2 authentication key. Authentication affords protection against forgery or message modification. It is possible to replay a message until the sequence number changes. The sequence number makes replay in the long term less likely to succeed. The mechanism does not afford confidentiality, since messages travel in clear. However, the mechanism is also exportable from most countries, which test, a privacy algorithm could fail.

The authenticated RIP packets contain, apart from the message digest data, a field which describes the authentication type, a field which identifies the key used to create the authentication data, and a sequence number. The authentication key is selected by the sender based on the outgoing interface. Each key has a lifetime associated with it, and is never used outside this interval. The value used in the sequence number is arbitrary, the suggestions are the classical two ones: time of the message's creation or a simple message counter.

An important aspect for any routing protocol is maintaining routing stability during the change of an authentication key. On this purpose, more than one authentication keys are stored and used on a given interface at the same time. It is recommended that a mechanism is provided for smooth authentication key switchovers. This mechanism prevents losing legitimate routing messages because the stored key is incorrect, and does not require routers to change all the keys at once.

This authentication scheme does not offer complete protection from insiders' attacks. There is no way to protect against a subverted router providing incorrect routing information.

Authentication for OSPF

Open Shortest Path First (OSPF) is used for distributing routing information between routers belonging to an AS. OSPF is a link-state protocol based on the Dijkstra algorithm.

The OSPF version 2 definition [16] states that only trusted routers can participate in the Autonomous System's routing. A large variety of authentication schemes can be configured for each IP subnet. For these purpose, there are two fields reserved inside the OSPF packet header: an authentication type field, and 64-bits of data whose use is determined by the previous field. Currently, two authentication schemes are defined: simple password and cryptographic authentication. Simple password authentication guards against routers inadvertently joining the routing domain (each router must first be configured with its attached networks' passwords before it can participate in routing), but is vulnerable to passive attacks. Anyone with physical access to the network can learn the password and compromise the security of the OSPF routing domain.

For cryptographic authentication, a shared secret key is configured on the routers attached to a common network/subnet. For each OSPF protocol packet, the key is used to generate/verify a message digest that is appended to the end of the OSPF packet. The algorithms used to generate and verify the message digest are specified implicitly by the secret key. Each OSPF packet is protected against replay attacks the same way RIP packets are (a non-decreasing sequence number is included in the packet).

In the event that the last key associated with an interface expires, since [16] states that it is unacceptable to revert to an unauthenticated condition, and not advisable to disrupt routing, a separate mechanism for smooth transition from one key to a new one is needed for OSPF too.

Since routing information (network topology) is not considered a sensitive information, OSPF cryptographic authentication option does not offer data confidentiality.

Shared Key Management for Routing Protocols

One of the critical issues for the previously described authentication schemes is the number of shared keys involved. In order to calculate the authentication data appended to the routing messages, the routers maintain one or more keys per interface. A critical issue may be the number of routers (interfaces) that share the same secret. There are mainly two approaches for this problem:

- a single globally-shared key is used to authenticate all routing messages exchanged in an AS,
- pairwise keys are configured for all possible pairs of routers.

The first method provides a very low level of security, while the second results in an unacceptable number of shared keys. The second solution can be still taken into consideration with RIP, where the protocol packets are exchanged between neighbours, but is considered awkward and unsuitable for link state routing [14]. For OSPF, which uses a mechanism for flooding routing information, this method provokes an unacceptable overhead due to recalculation of the authentication data at each router present in the flooding path. As a consequence, many authors consider the public key-based authentication approach more appropriate for use with link state routing protocols [17].

As with all the Internet authentication mechanisms, a capital issue is the key management procedure. It is obvious that having a strong cryptographic algorithm with a compromised key nullifies the protection offered by any authentication mechanism. [15] defines the key management requirements for RIP-2 authentication. The proposed management solutions are the classical two ones: manual and automated management of keys. An open issue remains as far as the latter method is concerned, since no IETF key management protocol has been generally accepted as a standard yet.

B. Public Key Authentication for OSPF

The experimental RFC-2154 [18] describes OSPF extensions in order to add digital signatures to Link State data, and to provide a certification mechanism for router data.

The keyed MD5 authentication method presented in the previous section is very useful for protection of protocol packets passed between neighbors, but, as we have outlined, does not address authentication of routing data that is flooded from source to eventual destination, through routers which may themselves be faulty or subverted. As a consequence, [18] proposes the following authentication scheme for OSPF:

- digital signatures are added to all OSPF LSA (Link State Advertisement) data. A LSA contains the state of a router adjacent links,
- a method is defined to distribute certified router information and keys,
- a neighbor-to-neighbor authentication algorithm (e.g., keyed MD5) is used to protect local protocol exchanges.

The LSAs that are flooded inside the Link State Update packets are individually protected by a digital signature. Each LSA is signed by the originator of that information and the signature stays with the data in its travel via OSPF flooding. This provides end-to-end integrity and authentication for LSA data. The digital signature attached to a LSA by the source router guarantees that the data comes from the advertising router. It also ensures that the data has not been modified by some other router in the course of flooding. If incorrect routing data is originated by a faulty router, the signature will identify the source of the problem (non repudiation).

For participating in the authentication scheme, each router has a pair of keys, a public and a private key. The private key is used to generate an unique signature of a block of data (the LSA), which is then appended to the LSA. The public key is used for signature verification. A distribution mechanism is mandatory for assuring that each router knows the public key of every other router. The key distribution is achieved by creating a new LSA, the Public Key LSA (PKLSA). This LSA is distributed via the standard OSPF flooding procedure. Flooding will ensure that a router public key is sent everywhere the router's signed LSAs are sent.

Even if this method scales well with OSPF, a problem still remains: any router can send out a public key and claim to be a given router, so the public key itself provides no assurance of the actual identity of the sender. This assurance must be provided by a trusted third party, the Trusted Entity (TE). This entity is a system that generates certificates for routers. In this case, a certificate is a packet of information about a router that identifies the router and supplies a public key. Certified router information includes the router identity, its role, the address ranges that the router may advertise, a timestamp and the router's public key. The certificate for a router is contained in a router PKLSA.

For verifying other routers' certificates, each router is configured with the TE's public key. A router receiving a PKLSA verifies the certificate using this key, and then verifies the whole signed LSA using the router's public key contained in the certificate. Successful verification provides assurance that the PKLSA was issued by the correct router, and that it has not been altered by any other router in the flood path.

The described authentication mechanism is not perfect. A compromised router can still distribute incorrect data in the information for which it itself is responsible. As a consequence, an AS employing digital signatures with this mechanism is not completely invulnerable to routing disruptions from a single router. For example, the area border routers and AS border routers will still be able to inject incorrect routing information (the outsider attacks). Also, any single internal router can be incorrect in the routing information it originates about its own links (insider attacks). This attacks cannot be addressed with cryptography alone, the only way to detect that something is wrong is to notice a disagreement between link state expressed by the two end-points of a link.

Apart from the vulnerabilities described above, the public key cryptography has the disadvantage that is quite expensive in terms of CPU time consumed for both the generation and verification of public key-based signatures. There are some work for providing more efficient, alternative techniques for these tasks. [14] describes such a method, based on public key digital signatures and one-way hash functions, which takes advantage of a technique for constructing hash chains similar to the one used in S/KEY one-time authentication.

V. Conclusions

Most of the attacks at the network infrastructure described in the present work begin with spoofing the IP source address of the victim. In fact, the more troublesome attacks for the Internet community are DoS attacks which employ forged source addresses [19]. As a consequence, a simple and effective defense consists in using ingress traffic filtering to prohibit DoS attacks to be propagated from “behind” an Internet Service Provider's (ISP) aggregation point. In a few words, all providers of Internet connectivity are urged to implement strict traffic filtering to prohibit attackers from using spoofed source addresses which do not reside within a range of legitimately advertised prefixes. An additional benefit of implementing this type of filtering is that it enables the originator of an attack to be easily traced to its true source, since the attacker would have to use a valid, and legitimately reachable source address.

It is an accepted fact that control and routing protocols need stronger security than the one that can be reached by simply using packet filtering. This paper outlines the importance of cryptographic authentication. For an authentication scheme to succeed a mechanism has to be standardized for dealing with the distribution of the cryptographic keys. Key management should be an intrinsic component of the basic security architecture in Internet. There currently exist a large variety of key management protocols, more of them are still in implementation and test phase. The work done in this area during the last few years is likely to converge towards IKE, a combination of ISAKMP and Oakley key exchange protocols.

The more efficient security mechanisms described in this article rely on public key cryptography. The Internet X.509 Public Key Infrastructure documents define public key certificates and certificate management protocols based on the X.509v3 standard. This may be a viable solution for all the security services which require strong authentication, data integrity and non-repudiation.

REFERENCES

[1] S. Kent, R. Atkinson, *Security Architecture for the Internet Protocol*, RFC 2401, Nov.1998

[2] J. Postel, *Internet Control Message Protocol*, RFC 792, 1981

[3] R. Braden, *Requirements for Internet Hosts-Communication Layers*, RFC 1122, 1989

[4] http://www.LOpht.com/advisories.html, LOpht Security Advisory, Aug. 1999

[5] S. Derring, *ICMP Router Discovery Messages*, RFC 1256, Sep. 1991

[6] http://www.cert.org/, *CA-96.26 Denial-of-Service Attack via ping*, CERT Coordination Center, 1996

[7] http://www.cert.org/, *CA-98.01 'smurf' IP Denial-of-Service Attacks*, CERT Coordination Center, 1998

[8] P. Kern, W. Simposon, *ICMP Security Failure Messages*, RFC 2521, Mar. 1999

[9] W. Simpson, *ICMP Domain Name Messages*, RFC 1788, Apr. 1995

[10] D. Harkins, D. Carrell, *The Internet Key Exchange(IKE)*, RFC 2409, Nov. 1998

[11] ipsec@lists.tislabs.com, IETF IPsec mailing list, Oct. 1999

[12] W. Fenner, *Internet Group Management Protocol, Version 2*, RFC 2236, Nov. 1997

[13] S. Bellovin, *Security Problems in the TCP/IP Protocol Suite*, ACM Computer Communications Review, Volume 19, Number 2, Apr. 1989

[14] R. Hauser, T. Przygienda, G. Tsudik, *Lowering security overhead in link state routing*, Computer Networks, vol. 31, Number 8, Apr. 1999

[15] F. Baker, R. Atkinson, *RIP-2 MD5 Authentication*, RFC 2082, Jan. 1997

[16] J. Moy, *OSPF Version 2*, RFC 2328, Apr. 1998

[17] R. Perlman, *Network Layer Protocols with Byzantine Robustness*, Ph.D. Thesis, Department of Electrical Engineering and Computer Science, MIT, Aug. 1988

[18] S. Murphy, B. Wellington, *OSPF with Digital Signatures*, RFC 2154, Jun. 1997

[19] P. Ferguson, D. Senie, *Network Ingress Filtering: Defeating Denial of Service Attacks which employ IP Source Address Spoofing*, RFC 2267, Jan. 1998