INTERNET HOLES - ELIMINATING IP ADDRESS FORGERY

Series Introduction

The Internet is now the world's most popular network and it is full of potential vulnerabilities. In this series of articles, we explore the vulnerabilities of the Internet and what you can do to mitigate them.

An Introduction IP Address Forgery

The Internet Protocol (IP) (RFC791) provides for two and only two functions. It defines a datagram that can be routed through the Internet, and it provides a means for fragmenting datagrams into packets and reassembling packets into the original datagrams. To quote from RFC791:

> The internet protocol is specifically limited in scope to provide the functions necessary to deliver a package of bits (an internet datagram) from a source to a destination over an interconnected system of networks. There are no mechanisms to augment end-to-end data reliability, flow control, sequencing, or other services commonly found in host-to-host protocols. The internet protocol can capitalize on the services of its supporting networks to provide various types and qualities of service.

Here's a description of an IP datagram, also from RFC791:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|Version|  IHL  |Type of Service|          Total Length         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         Identification        |Flags|      Fragment Offset    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Time to Live |    Protocol   |         Header Checksum       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       Source Address                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                    Destination Address                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                    Options                    |    Padding    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                             data                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                             data                              |
\                                                               \
\                                                               \
|                             data                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|             data              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Description of an IP Datagram

Note that the 4th line of the description calls for the Source Address of the datagram. In the simplest form of IP address forgery, the forger only needs to create a packet containing a false Source Address and insert it into the Internet by writing it into the output device used to send information to the rest of the Internet. For the non-expert forger, there is a tool called iptest which is part of the free and publicly available ipfilter security package that automatically forges packets for the purpose of testing configurations or routers and other IP security setups.

The infrastructure of the Internet consists primarily of a set of gateway computers and packet routers. These systems have multiple hardware interfaces. They maintain routing tables to let them decide which output interface to send a packet out on based on the input interface that it came in on and the destination IP address specified in the packet. When a forged packet arrives at an infrastructure element, that element will faithfully route the packet toward the destination address, exactly as it would a legitimate packet.

How Can IP Address Forgery Be Used

At its root, IP address forgery is a method of deception, and thus it can be used in much the same way as other forms of deception. Dunnigan95 More specifically, and using Dunnigan and Nofi's classification scheme, here are some quick ideas about how IP address forgery might be used:

* Concealment: IP address forgery is commonly used to conceal the identity of an attacker, especially when denial of services is the goal of the attack.
* Camouflage: IP address forgery is used to make one site appear to be another as a way to convince the victim, for example, that an attack is from a University, when in fact it is from a competitor.
* False and Planted Information: IP address forgery can be used to create the impression that a particular site is acting maliciously in order to create friction or lead a defender to falsely accuse an innocent third party.
* Reuses: IP address forgery can be used to support another activity designed to gain the confidence of the defender. For example, a salesperson for information security products could create IP address forgeries in order to convince a client of the need for their services.
* Displays: IP address forgery has been used in order to lead defenders to believe that many sites are participating in an attack when in fact only a small number of individuals are responsible.
* Demonstrations: IP address forgery has been used to demonstrate a potential for untraceable attacks as a way to convince defenders not to try to catch attackers.
* Feints: IP address forgery can be used to try to fool an enemy into believing that an attack is coming from outside or from a particular direction, when the real attack is very different. This is a way to misdirect the enemy into spending limited resources in the wrong way.
* Lies: IP address forgery has been used to create a more convincing lie that somebody known to the defender is communicating with them about a particular matter.
* Insight: IP address forgery can be used to gain insight into how an opponent reacts and as a sort of probe to determine what sorts of responses are likely to arise.

Another way to view this issue is in terms of the net effect on information in information systems. Here is another way of viewing this issue with an example from each category.

* Corruption of Information: IP addresses are often used as the basis for Internet control decisions. For example, DNS updates are often designated as coming only from specific other servers. With IP address forgery, the entire DNS system could be corrupted, causing services to be rerouted through enemy servers.
* Denial of Services: The Internet is basically a fragile network that depends on the proper behavior and good will of the participants for its proper operation. Without wide-ranging changes to the way the Internet works, denial of services is almost impossible to prevent. For example, the same DNS attack could be used to cause widespread denial of services, or perhaps even to create loops in the packet delivery mechanisms of the Internet backbone.
* Leakage of Information: Forged IP addresses can be used to cause a host to take orders for the delivery of information to enemy sites by forging authorization as if it were from a legitimate authorizing site.
* Misplaced Liability: Forged IP addresses could be used, as described above under False and Planted Information, to cause defenders to assert claims against innocent bystanders and to lay blame at the wrong feet.

These are only some of the examples of what forged IP addresses can do. Without a lot of effort, many other examples can be created.

What Can We Do About It?

As individuals, there is little we can do to eliminate all IP address forgery, but as a community, we can be very effective. Here's how. Instead of having all infrastructure elements route all packets, each infrastructure element could, and should, enforce a simple rule. They should only route packets from sources that could legitimately come from the interface the packet arrives on.

This may sound complicated, but it really isn't. In fact, the technology to do this is already in place, and always has been. Virtually every router and gateway in existence today allows for the filtering of packets based on their input interface and IP source and destination address. This is a necessary component of their operation and is the basis for the way they route all packets.

The only change that has to be made is for these routers and gateways to enforce the network structure that is legitimately in place. Or in other words, the routers and gateways should refuse to route ridiculous packets. Here are some of the simpler examples of known bad packets:

* The IP address 127.0.0.1 is ONLY used for internal routing of packets from a host to itself. There is no legitimate IP datagram that should pass through a router or gateway with this as the source address. In fact, routing these packets is dangerous because they may be used to forge packets from the localhost which often has special privileges. A recent attack that causes denial of services involves sending a packet to a host's echo port with 127.0.0.1 as its source address and the echo port as it's source port. The echo port causes whatever packet it is sent to be

returned to its source. Since the source address is the same port on the same host, this packet creates an infinite loop which, in many cases, disables the computer.

* The IP address 0.0.0.0 is not legitimate - full stop. In fact, there's really no legitimate IP address that should traverse gateways containing a 0 for one of the address elements. Unfortunately, many routers use the '.0.' convention in their filtering tables to indicate any address from 0 to 255 (the whole range), so blocking these packets may be non-trivial in some infrastructure elements.
* The IP specification includes provisions for private subnetworks that are designated for internal use only. There is no legitimate reason to route packets from these addresses anywhere in the general Internet infrastructure. (RFC1597) These address ranges include 10.*.*.*, 172.16-32.*.*, and 192.168.*.* (where * indicates any value from 0 through 255). No packets should be routed through the Internet with these addresses as either their source or their destination.

The next step in eliminating IP address forgery is for the routers and gateways at each type of infrastructure element to enforce standards on each interface. Generally, the Internet is broken up into Backbone providers that provide wide area packet transport services, Private Networks which are owned and operated by companies, institutions, government agencies, and other parties for their own purposes, and Internet Service Providers (ISPs) that provide connections between the backbone elements and private networks (sometimes including other ISPs). These roles can be blurred at times, but they are adequate for our purposes.

* Private Networks: Each private network should;
    + 1) prevent all of the known-bad packets from crossing into or out of the organization,
    + 2) prevent packets with internal source addresses from passing inward,
    + 3) prevent packets with external source addresses from passing outward,
    + 4) prevent packets with external destination addresses from passing inward, and
    + 5) prevent packets with internal destination addresses from passing outward.
* ISPs: Each ISP should;
    + 1) prevent all of the known-bad packets from crossing into or out of their infrastructure,
    + 2) prevent any packet inbound from any of their clients with a source address not from that client's assigned address range from passing from the client network,
    + 3) prevent any packets with a destination address not in their client's address range from passing to the client network,
    + 4) prevent any packet not from this ISP's legitimate address range from entering the backbone, and
    + 5) prevent any packets originating from the backbone and not destined for one of their legitimate IP addresses from entering their network.

Two additional rules will assist the ISP's clients;

*
    + 6) prevent inbound traffic from the client with the client's address as a destination, and
    + 7) prevent outbound traffic to the client with the client's address claimed to be the source.
* Backbone Networks: Each backbone provider should;
    + 1) prevent all of the known-bad packets from crossing into or out of their infrastructure,

+ 2) prevent packets originating from any ISP with source addresses not in that ISP's range of legitimate source addresses from entering the backbone,
+ 3) prevent any packets not destined for an ISP's address range from entering that ISP,
+ 4) prevent any packets from any other backbone provider that could not be properly routed through that provider from entering their backbone, and
+ 5) prevent any packets from going to any other backbone provider unless they could legitimately be routed through that provider to reach their destination.

For backbones, this requires some effort, however the high volume of information they carry certainly justifies a little effort for protection.

Some Examples

As an aide to the less technically inclined, the following examples provide some real world implementation details.

This set of rules applies to a private network (in this case, the all.net class C network 204.7.229.*) and are written in the format of the Morningstar PPP (point to point protocol) Filter file:

```
#       Rule 1 for private networks
#       prevent known-bad address ranges from entering (or leaving)
!172.16-32.0.0                     # private network segment
!192.168.0.0                       # private network segment
!10.0.0.0                          # private network segment
!127.0.0.0                         # localhost network
#       Rule 2 for private networks
#       prevent internal source address packets from passing inward
!recv/src/204.7.229.0              # prevent inbound from our network
#       Rule 5 for private networks
#       prevent internal destination addresses from passing outward
#       Note that rule 5 is placed here because the filters are order dependent
!send/dst/204.7.229.0              # prevent our destinations from passing out
#       Rule 3 for private networks
#       prevent external source address packets from passing outward
send/src/204.7.229.0               # allow legitimate outbound sources
!send/src/0.0.0.0                  # prevent illegitimate outbound sources
#       Rule 4 for private networks
#       prevent external destinations from passing inward
recv/dst/204.7.229.0               # allow legitimate inbound destinations
!recv/dst/0.0.0.0                  # prevent illegitimate inbound destinations
```

The next set of rules applies to an ISP. In this case, we assume that the ISP has control over three class B networks that it uses to sell services to its clients. The class B networks used in this example have IP addresses of 123.7.*.*, 231.6.*.*, and 201.96.*.*. In this case, we have three different parts of the example:

This is the router connecting the ISP to the backbone, presented in the format of a Cisco router with interface 0 connected to the backbone and interface 1 connected to the ISP's internal network. It implements rules 1, 4, and 5 for the ISP.

```
#        Rule 1 for an ISP
#        prevent all of the known-bad address ranges
#        this should be done on all in and out connections
#        on all interfaces in all access control lists
All interfaces in and out
deny ip 172.16-32.0.0              # private network segment
deny ip 192.168.0.0                # private network segment
deny ip 10.0.0.0                   # private network segment
deny ip 127.0.0.0                  # localhost network

#        Rule 2 for an ISP
#        prevent inbound from client not in client's address range
#        DONE ELSEWHERE

#        Rule 3 for an ISP
#        prevent entry of packets not destined clients from passing their way
#        DONE ELSEWHERE

#        Rule 4 for an ISP
#        prevent exit of packets not from our class Bs
#        on interface 0 (backbone) out filter
Interface 0 out
permit ip 123.7.0.0
permit ip 231.6.0.0
permit ip 201.96.0.0
deny   ip 0.0.0.0

#        Rule 5 for an ISP
#        prevent entry of packets not destined for our class Bs.
#        on interface 0 (backbone) in filter
Interface 0 in
permit ip 123.7.0.0
permit ip 231.6.0.0
permit ip 201.96.0.0
deny   ip 0.0.0.0
```

Next, we implement rules 2 and 3 for each client by creating separate (in this example ppp) filters on the ISP's gateway computer. Again, using the Morningstar ppp Filter format and assuming that Class C IP network 201.96.1.* is assigned to this particular client:

```
#       Rule 1 for ISPs
#       prevent known-bad address ranges from entering (or leaving)
!172.16-32.0.0                      # private network segment
!192.168.0.0                        # private network segment
!10.0.0.0                           # private network segment
!127.0.0.0                          # localhost network
#        Rule 6 for an ISP
#        prevent inbound traffic from the client destined for the client
#        note that rule 6 is placed here because filters are order dependent
!recv/dest/201.96.1.0               # prevent inbound from client to self
#        Rule 7 for an ISP
#        prevent outbound traffic to the client claimed to be from the client
#        note that rule 7 is placed here because filters are order dependent
!send/src/201.96.1.0                # prevent outbound to client from client
#        Rule 2 for an ISP
#        prevent inbound from client not in client's address range
recv/src/201.96.1.0                # allow legitimate traffic
!recv/src/0.0.0.0                  # prevent all other traffic
#        Rule 3 for an ISP
#        prevent entry of packets not destined clients from passing their way
send/dest/201.96.1.0               # allow legitimate traffic
!send/dest/0.0.0.0                 # prevent all other traffic
```

Note that redundant protection is provided in several ways. The ISP protects the clients from backbone forgery both at the backbone router and at the client's ppp connection, and protects the backbone from IP forgery emanating from the ISP both by preventing forgery from clients and by preventing forgery from within the ISP. Similarly, the ISP provides redundant protection against improperly configured client hardware and software. The last two filter rules are to assure that even if the client is not properly configured to prevent forgery of internal addresses from the outside world or to prevent internal traffic from being sent out, this traffic is prevented.

This last example is a simplification of a wide area backbone network in which this particular router (no type specified) is at the junction between UK connections and non-UK connections. In this case, it is a reasonable assumption that all internal UK traffic should remain internal and that external traffic that gets to this node should be sent out of the UK never to return. This particular backbone node will be connected to non-UK traffic on interface 0, our previously described ISP on interface 1, and the rest of the internal UK backbone on interface 2.

```
#        Rule 1 for a backbone
#        prevent all of the known-bad packets from crossing
all-interfaces prevent in/out 172.16-32.0.0     # private network segment
all-interfaces prevent in/out 192.168.0.0       # private network segment
all-interfaces prevent in/out 10.0.0.0          # private network segment
all-interfaces prevent in/out 127.0.0.0         # localhost network

#        Rule 2 for a backbone
#        prevent packets originating from any ISP with non-ISP source address
interface-1 allow in from 123.7.0.0                  # ISP traffic
interface-1 allow in from 231.6.0.0                  # ISP traffic
interface-1 allow in from 201.96.0.0                 # ISP traffic
interface-1 prevent in from 0.0.0.0                  # no other inbound traffic

#        Rule 3 for a backbone
#        prevent packets not destined for an ISP from going there
interface-1 allow out to 123.7.0.0                 # ISP traffic
interface-1 allow out to 231.6.0.0                 # ISP traffic
interface-1 allow out to 201.96.0.0                # ISP traffic
interface-1 prevent out to 0.0.0.0                 # no other outbound traffic

#        Rule 4 for a backbone
#        prevent packets from other backbones that shouldn't come this way
interface-0 allow in to UK-1                       # UK traffic
interface-0 allow in to UK-2                       # UK traffic
...
interface-0 allow in to UK-n                       # UK traffic
interface-0 prevent in to 0.0.0.0                  # no other inbound traffic

#        Rule 5 for a backbone
#        prevent packets that should stay in our backbone from going out
interface-0 allow out from UK-1                    # UK traffic
interface-0 allow out from UK-2                    # UK traffic
...
interface-0 allow out from UK-n                    # UK traffic
interface-0 prevent out from 0.0.0.0               # no other outbound traffic
```

In this example, we have assumed that all UK traffic is on IP addresses identified as UK-1, ..., UKn.

What-ifs and Objections

WHAT IFS?

* What if a private network ignores the rules? It is to be expected than many private networks will ignore any such rules, either through ignorance, intent, or inattention. But even if all private networks ignored all of the rules, the rules for ISPs would prevent IP forgery from extending to the overall infrastructure.
* What if an ISP ignores the rules? If an ISP ignores the rules and allows IP forgery, the backbone can protect the rest of the Internet, at least to the point where forged packets within the ISP's domain remain within or are traceable to that domain. That means that the ISP's clients would be subject to IP forgeries from other clients of that ISP, but that the rest of the Internet would be able to trace all packets coming from that ISP to that ISP.
* What if the backbone ignores the rules? If all of the backbone providers ignore the rules, unless everyone else follows them, we will continue to have IP forgeries through the ISPs that don't follow the rules.
* What if combinations ignore the rules? Depending on the specific combinations, we will have more or fewer IP address forgeries. It turns out that a complete analysis of this issue is not simple enough to do in the space provided, but a simple conclusion can be drawn without a full analysis. As more Internet users and providers prevent IP address forgery, the job of the forger will become harder and harder. We don't all have to participate in order to have proper protection, but if we all fail to participate, the forgeries will continue.

OTHER OBJECTIONS

* Content (common carrier) objections: Many ISPs and backbone providers don't want or take responsibility for content in the Internet. Just like a telephone company, they don't want any role in examining or dictating the content of the messages - they only want to be a delivery service. It could be argued that examining the address information in an IP packet and preventing packets based on those addresses constitutes limitation of content. Of course the portion of the content involved here must be examined in order to route the information, and the routing used in the Internet already provides exclusion of packets based on IP address ranges. Furthermore, common carriers (in the U.S.) are allowed to listen to and filter traffic to the extent that this activity is done solely to assure the proper operation of the network. Thus this objection would seem to be moot.
* The cost is too high objection: In fact the cost is negligible. If the rules set forth herein were applied as a normal part of the installation and maintenance process, it would come to only a few minutes of effort during each installation. Even applying them to systems already in place requires only a few minutes of effort, again an insubstantial amount of effort well within the discretion of any systems administrator.
* The we don't want restrictions objections: There are a substantial number of people that want a total lack of restrictions on information flowing through the Internet. I generally agree with the principle of free information flow, except in cases where the freedom of one person infringes on the freedom of others. This impingement on other peoples' rights applies to certain types of information, such as routing information, that must be correct in order for the Internet to work properly. Since the restrictions described here only assure that the Internet works properly and don't restrict the content or flow of information, there is no restriction of the free flow of information here. Only increased assurance that those who want to use the media for legitimate purposes will continue to be able to do so.

Summary

This solution we presented:

* 1) is easy to implement,
* 2) makes good sense from a traffic standpoint,
* 3) allows all legitimate activity without any hinderence,
* 4) works even if all parties don't participate,
* 5) costs almost nothing to implement at each site,
* 6) does not require any changes in existing protocols of traffic patterns,
* 7) makes good sense for the security of each party that participates, and
* 8) can be done today.

All that remains is for the people in each of these organizations to implement these protections. Unlike so many of the problems in the Internet that are hard to solve and will require years of evolution, this problem can be solved now. We encourage you to implement these protections at your earliest convenience and to urge other to do so as well. Together, we can eliminate IP address forgery.