Addressing the challenge of IP spoofing

SEPTEMBER 2015

Nowhere in the basic architecture of the Internet is there a more hideous flaw than in the lack of enforcement of simple SAV (source-address validation) by most gateways.

Paul Vixie, "Rate-limiting State. The edge of the Internet is an unruly place" [1]

Internet Society™

Introduction

Spoofed Internet traffic is a persistent threat, and often the root cause of reflection Distributed Denial of Service (DDoS) attacks. While technical solutions for blocking spoofed traffic exist they are only effective and applicable close to the edge - computers and other end-devices connected to the net. This requires deployment of anti-spoofing measures by a vast majority of networks on a global scale – something that is not easy to achieve.

Unfortunately, right now there are few incentives, further aggravated by real costs and risks for implementing anti-spoofing measures. There is also an imbalance between the ease and low cost of launching a DDoS attack and the heavy economic and social impact that these attacks have.

The complexity, high visibility, and negative impacts of spoofing call for a comprehensive approach, including technology measures, better information and awareness, and social and regulatory tactics. Unfortunately little visible progress has been made in solving the problem. Demonstrating visible improvements (or at least the way forward) could produce a significant impact in both technical and policy planes due to the high profile of DDoS attacks.

> In February 2015, the Internet Society convened a roundtable bringing together network operators, vendors, leading security experts, and researchers in this area to discuss the problem of source IP address spoofing with a goal to better understand the challenges of addressing it, various factors that aggravate or help solve the problem, and to identify paths to improve the situation going forward.
>
> The objective was to identify elements of a comprehensive strategy and a roadmap for tackling this issue. This paper represents the main takeaways from this discussion and articulates possible elements of such a strategy.

What is Source IP spoofing?

IP address spoofing, or IP spoofing, is the forging of a source IP address field in IP packets with the purpose of concealing the identity of the sender or impersonating another computing system.

Fundamentally, source IP spoofing is possible because Internet global routing is based on the destination IP address. Or, more precisely, an Internet router with a default configuration (i.e. no special policy applied, like reverse path filtering) forwards packets from one interface to another looking up only the destination IP address.

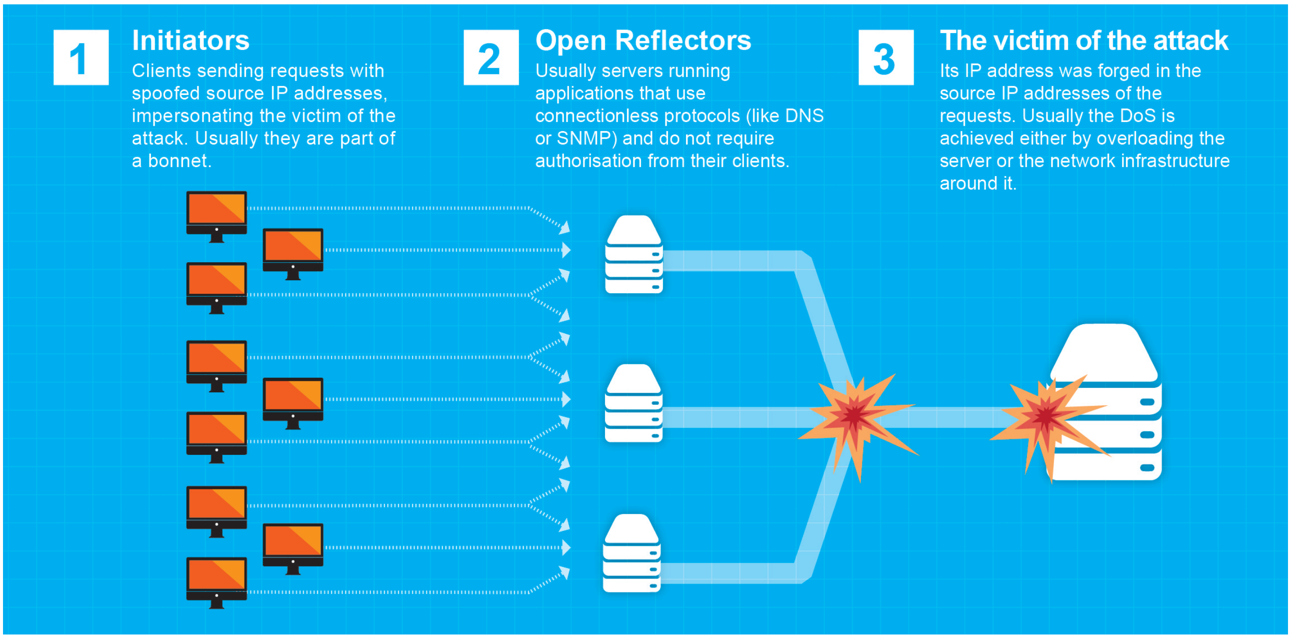
An application with sufficient privileges can modify the source IP address field of an IP packet to any syntactically correct value, and in most cases the packet will be sent through the network interface and in many cases will reach the destination.

Of course, an incorrect source IP address may hinder normal operation of communications: responses from the destination application or intermediary nodes (e.g. ICMP responses) will not reach the sender. But attacks mounted using the spoofing technique do not rely on properly set up communication flows. On the contrary, they abuse this feature, directing traffic flow of responses to the target identified by the forged source IP address.

Reflection and amplification DDoS Attack

A typical reflection and amplification DDoS attack exploits a common scenario: a compromised host emits packets with source IP addresses set to the IP address of the target of the attack, directed at a so-called reflector – a remote application that will respond to these packets/requests directing traffic to the victim. In many cases the size of the response is several times larger than the request itself, thus not only reflecting, but also amplifying the traffic toward the victim. Usually such attacks have a distributed nature – packets are sent from multiple sources to multiple reflectors, all configured with the same target. The volume of such attack can reach several hundred Gbps[2]. This is schematically shown on figure 1.

Figure 1. The anatomy of a reflection and amplification DDoS attack



1 **Initiators**
Clients sending requests with spoofed source IP addresses, impersonating the victim of the attack. Usually they are part of a bonnet.

2 **Open Reflectors**
Usually servers running applications that use connectionless protocols (like DNS or SNMP) and do not require authorisation from their clients.

3 **The victim of the attack**
Its IP address was forged in the source IP addresses of the requests. Usually the DoS is achieved either by overloading the server or the network infrastructure around it.

Such an attack is based on three ingredients:

Use of source IP spoofing with a datagram protocol, like UDP or ICMP. These protocols do not require setting up a connection (e.g. by a handshake between the applications) and therefore the first response can already carry a significant amount of data. It is important to note that some TCP implementations can also send significant amounts of data as part of the first response, which widens the spectrum of possible reflectors[3].

Reflectors and Amplifiers – remote applications that will respond to requests coming from the compromised hosts. These responses will be directed at the target specified by the spoofed source IP address in the requests. A “good” reflector is also an amplifier – its responses are several times larger than the requests. For example, for DNS applications, amplification can exceed 50 times. Also, DNS resolvers are ubiquitous – that is why they are often used as reflectors.

Botnets. In order to achieve significant traffic volumes but not attract attention to the real source of the attack, spoofed requests must be generated from many geographically distributed hosts. Botnets are perfect candidates for that. With the average botnet counting tens of thousands of compromised computers and more than 28 million open resolvers[4], mounting a multi-Gbps attack does not seem like a very difficult task.

Current mitigation strategies

The challenge of reflection and amplification DDoS attacks can be addressed by tackling initiators of the attack – hosts that send requests with spoofed IP packets, and reflectors – hosts that respond to these requests.

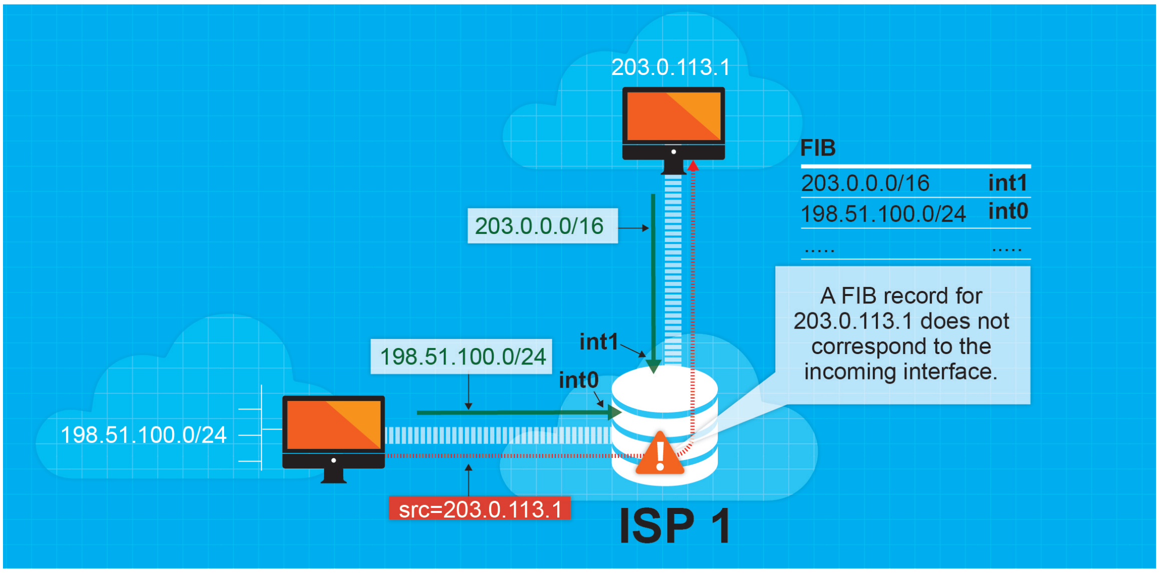
One of the measures for dealing with the reflector/amplifier side of the attack is limiting the scope of clients that are authorized to send requests. Usually these are clients coming from the same network where the reflector resides. For instance, simple access lists in the configuration of a DNS resolver can “close” an otherwise open resolver. Another measure is Response Rate Limiting, or RRL[5], which lowers the rate at which authoritative servers respond to high volumes of malicious queries.

To prevent an initiator from sending packets with forged source IP addresses, the following anti-spoofing measure have been developed:

> Ingress filtering described in BCP38 (http://tools.ietf.org/html/bcp38).

> Unicast Reverse Path Forwarding, or uRPF. To automate the implementation of BCP38 a technique was developed based on the router’s knowledge about connected networks. In its “strict” mode for a given network interface, a router will not accept packets originating in networks to which the router has no route through that particular network interface (reverse path), as shown in figure 2.

Figure 2. Unicast Reverse Path Forwarding, uRPF



FIB

203.0.0.0/16	int1
198.51.100.0/24	int0
.....	.....

A FIB record for 203.0.113.1 does not correspond to the incoming interface.

Unfortunately in more complex topologies of connected networks (e.g. multihomed customers) with traffic asymmetry, strict uRPF may not work, resulting in dropping legitimate packets. To overcome this, less strict modes of uRPF operation were developed.

In its “feasible” mode, the router (its forwarding table, FIB) maintains alternate routes to a given IP address. If the incoming interface matches with any of the routes associated with the IP address, then the packet is forwarded. Otherwise the packet is dropped.

Finally, in “loose” mode, each incoming packet's source IP address is tested against the FIB. The packet is dropped only if the source IP address is not reachable via any interface on that router. This is the safest, but also the least effective tool, especially in the IPv4 world, since soon almost every IP range will be in use due to the depletion of free address pools.

In any case, the uRPF becomes more fragile and less effective the more you move away from the source of the forged packet.

Preventing spoofing locally, as close to the host as possible, is more robust and limits the scope of a possible attack. The IETF Working Group on Source Address Validation Improvements (SAVI)[6] is working on solutions in this direction[7]. Implementing anti-spoofing in a local network segment makes possible application of more reliable and fine-grained filters, or bindings, like (IP address)-(MAC address)-(switch port), as opposed to (address range)-(router interface) in the case of traditional ingress filtering.

SAVI takes a similar approach to the DOCSIS source address verification mechanism that we will discuss further and extends it on all kinds of network topologies and technologies, not only broadband cable networks. The proposed mechanisms are purely network based and don't depend on supporting functionality in connected hosts.

Current challenges and the way forward

Measurements

While there is plenty of information about reflection DDoS attacks[8], including reports and statistical data, little is known about their actual source – the networks that allow source IP address spoofing. DDoS attacks, and specifically reflection/amplification attacks, are on the rise – reaching volumes as high as 300Gbps, but the analysis doesn’t usually go beyond the immediate “attackers” – reflectors that respond to spoofed requests, often amplifying them. But the reflectors are almost ubiquitous – from more than a few dozen million open resolvers[9] to zillions of simple TCP speakers[10]. So addressing the root cause is very important.

To address the problem and build a sensible strategy, we need more information about address spoofing: where the spoofing happens, how widely anti-spoofing measures are deployed, and what the general trend is. Taking into account its probably enormous scale, discussing address spoofing without credible data is like shooting in the dark.

Having credible data about the ability to spoof IP addresses by networks allows correlation of this vulnerability with type of network, its topology and geography. It allows the technical community to create more focused efforts, for instance addressing specific network environments.

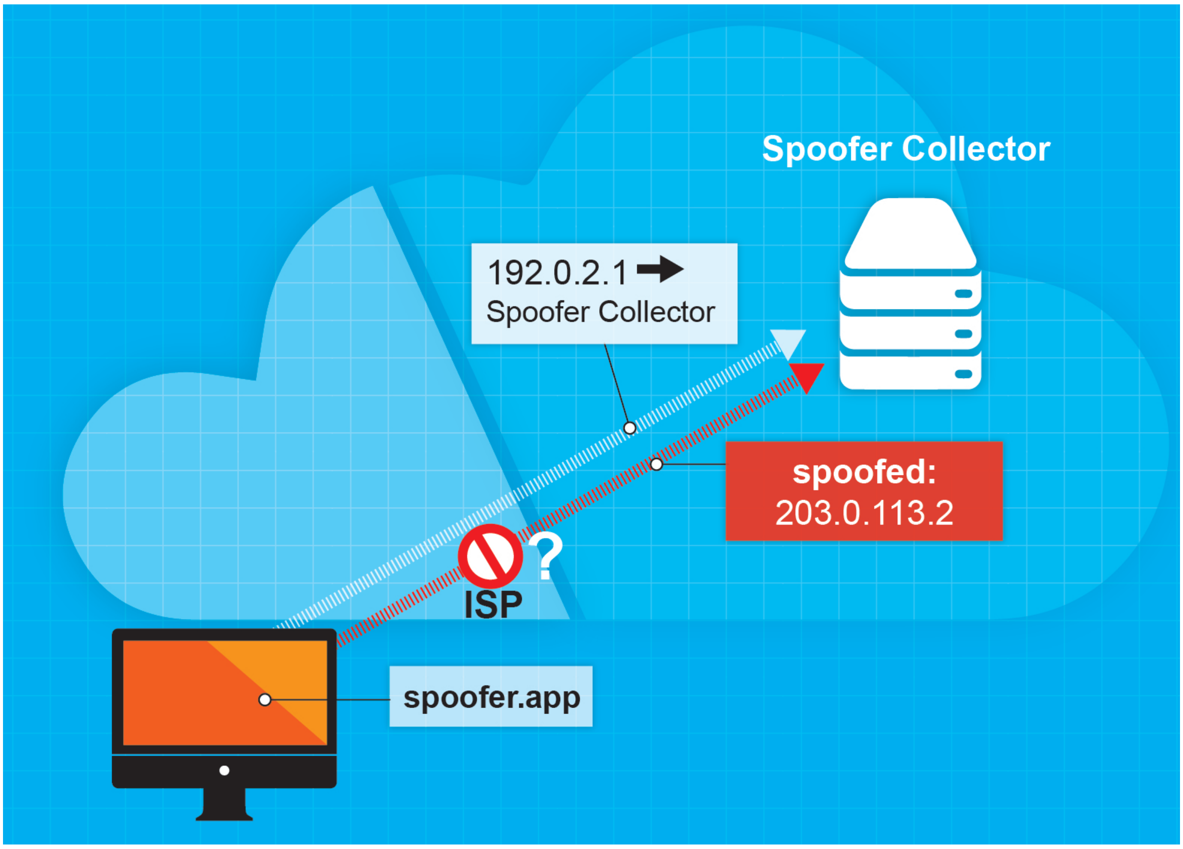
Another important factor is that accurate numbers allow us to credibly demonstrate the problem and to monitor the trend line to demonstrate progress.

Measurement methods and current measurement activities

Several techniques exist that allow one to infer if a network does source address validation. One of them relies on an insider doing the testing while two others have a limited capability to test ability to spoof from the outside. All methods have limitations and may produce biased results that are difficult to extrapolate.

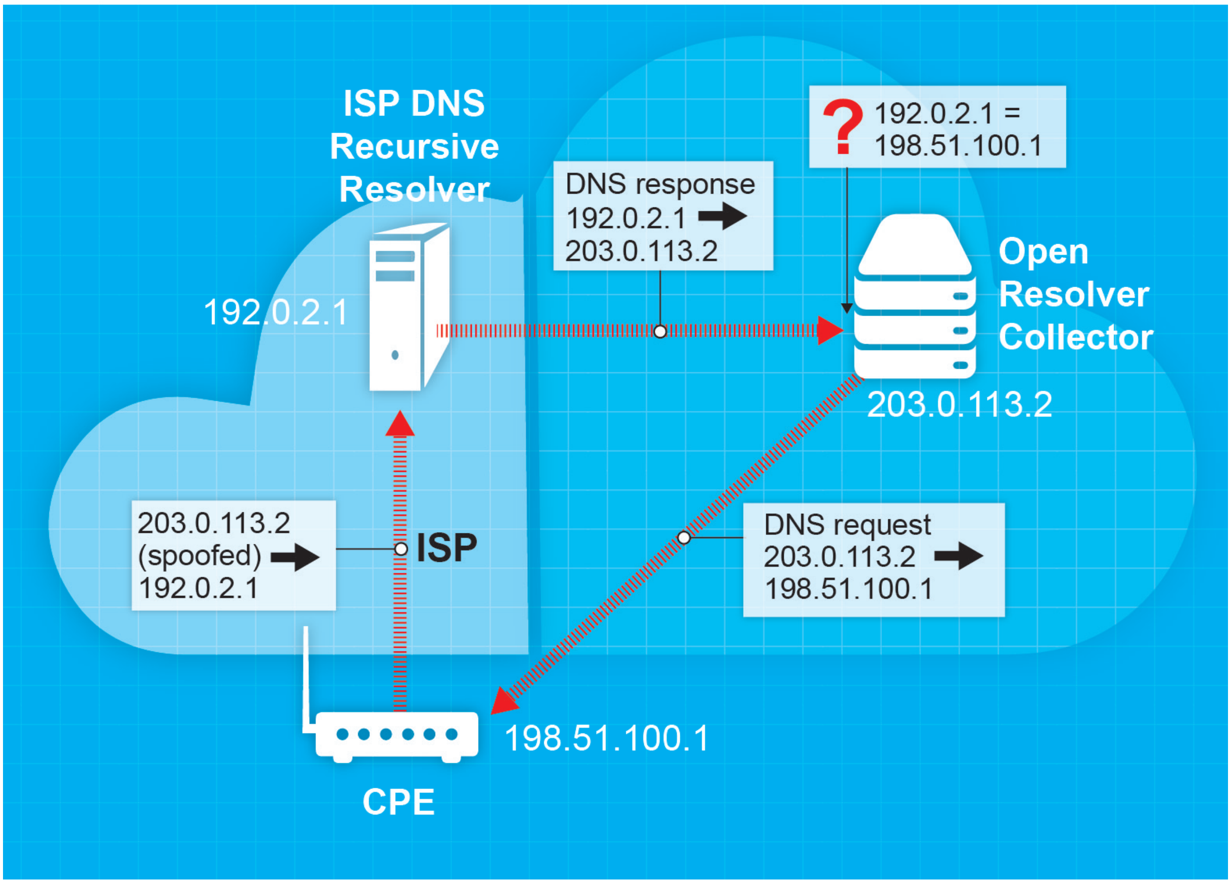
The first method is to run a specialized client, such as that provided by the Spoofer project (http://spoofer.caida.org/), which sends packets with spoofed addresses to a centralized server. The server archives meta-data surrounding the measurement, such as the IP address and origin AS of the client, and information about the types of source addresses that could be forged. This method is shown in Figure 3. This method requires an insider, someone with sufficient permissions[11] on a computer to be able to run the Spoofer test, which limits its application.

Figure 3. An internal probe method (Spoofer project)



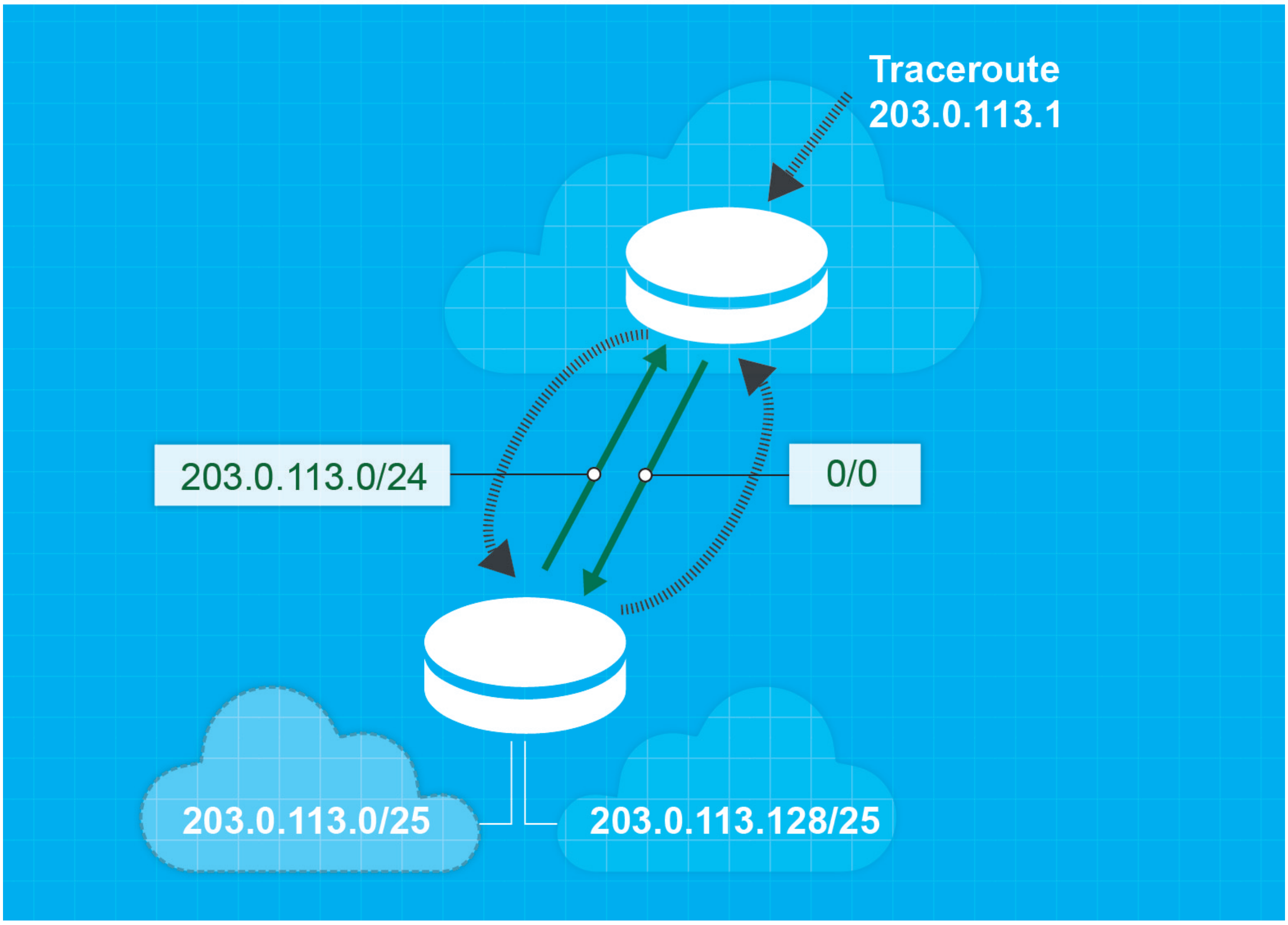
The second source of data is provided by the Open Resolver project (http://openresolverproject.org/), which makes use of a particular DNS implementation quirk of some CPE devices. Such DNS implementations forward the request with the source IP address of the request packet intact to an upstream resolver, when the request comes from a WAN interface. The upstream resolver then relays the DNS answer back to the original requestor – an open resolver prober. Because the source IP address belongs to the open resolver project's prober, and not the CPE's network, it implies that the CPE's network has inadequate source address validation. This method is presented in Figure 4.

Figure 4. A method making use of some CPE DNS implementations (OpenResolver project)



Traceroute data can sometimes reveal the absence of anti-spoofing measures, if bouncing between the final hops is observed. This method relies on a very specific scenario: where the customer advertises address space that it does not use internally, and where the customer has a default route pointed at its provider. Figure 5 schematically describes this method.

Figure 5. Using traceroute to detect spoofable networks



Possible improvements to the measurements

The main problem is that with all methods listed above, it is difficult to get statistically representative data, partly because of the limited dataset, and partly because the features they use are not uniformly distributed, producing potentially biased data. For example, Spoofer relies on volunteers running the tests inside a network, which may imply already a certain level of awareness about the issue and technical qualification. This may mean that tests are more frequently run in “cleaner” networks, rather than in networks where people do not care about spoofing.

Increasing awareness about this project and providing additional incentives to the users of a network to check if it allows spoofing may increase the data set. Better knowledge of a tester profile may help in identifying the bias and possibly fixing this. Another approach could be to try to employ tools like the Mechanical Turk[12] that allows individuals to perform defined tasks, known as HITs (Human Intelligence Tasks).

The method used by the OpenResolver project relies on a specific CPE implementation, and it is unclear how widespread this is. It also tests particular networking environments – home networks and small enterprises, and their providers.

Applying a big data approach might be helpful. There are other data sets that may be correlated. For example, big service providers are frequent targets of DDoS attacks, and might be able to share some appropriately anonymized data, and provide estimates of the prevalence of IP address spoofing in these attacks.

Traceability

Traceability of the spoofing initiator is a challenge even inside a single network, requiring telemetry and resources. There is a lack of monitoring tools, and especially forensics tools. Inter-provider traceability is even more complex, requiring a high degree of coordination and automation as well as trusted relationships between the networks. It looks like an unsolvable problem at the moment.

The problem is aggravated by the fact that, especially in the context of a DDoS attack, operators mainly focus on the “attack” leg – i.e. traffic from the victim to reflectors, while the spoofed traffic flows along the “initiator” legs, thus motivating less attention and resources to look there. Also, for tracing back spoofed traffic, cooperation is required from networks hosting the reflectors (e.g. open resolvers). In many cases these are unresponsive networks that do not really care about the negative effects they produce.

However, given all the difficulties in implementing this, especially across multiple operators, this doesn’t look like a feasible tool in the near future[13].

Network types and common operational practices

Mobile networks

Incentives

The main incentive for mobile networks is preserving scarce resources:

- Spectrum
- Backhaul capacity
- Device battery
- CGN infrastructure

Common setup and anti-spoofing measures

For an IPv4-provisioned network, uRPF[14] capability is enabled on the Packet Data Network Gateway (or similar function in a non-LTE network). CGNs are commonly used in mobile networks; in this case most spoofed traffic is dropped at the CGN.

For the inbound traffic BCP38 ACLs are deployed on AS border routers. The objective here is to drop spoofed traffic (external traffic that is claiming to come from either obviously wrong prefix blocks, RFC1918, or the provider's own space) so that it doesn't have to carry it any further in the network before it gets dropped, and to prevent reflection attacks (an attacker sends packets with the source of something in the carrier's network and a destination of an open reflection source that is also on-net). It is also protecting against things like UDP floods that attack a destination on the network (customer or infrastructure) directly but spoof the source address.

Broadband access providers

Incentives

For broadband providers the incentives fall into two broad categories:

- To protect against customers trying to squat on/hijack a specific address
- To protect against customers trying to participate in reflection attacks, either toward the provider's infrastructure or toward other customers, or even off-net destinations

Common setup and anti-spoofing measures

For effective protection against IP spoofing, proper control at the CMTS or DSLAM is important, especially if the broadband provider allows customers to deploy CPEs of their own choice. For instance in cable networks, DOCSIS 3.0 cable source verify is usually enabled on CMTS, which works similar to the uRPF strict mode, in that it rejects traffic from any address but the one associated with a given cable modem.

Commonly, most of the connected customers/households are NATed – their home LANs are sitting behind a CPE that also acts as a NAT. Although it seems like good enough protection, there are several concerns. One group of these concerns is related to the fact that there is no standard way of dealing with spoofed traffic received from the LAN: it depends on the reference architecture implementation as well as how much vendors customize that implementation. Many consumer CPEs are based on one of a couple of chipsets and software provided as a reference implementation by a few vendors (e.g. Broadcom, Intel, etc.) and various open-source software components, but then retail vendors (OEMs) take that architecture and augment it with features and additional code as they deem necessary, which results in significant diversity in implementation and lack of a baseline standard.

Possible treatment:

- Drop obvious junk before sending it to NAT
- RPF check, TCP sequence number window check, state comparison

- NAT everything that is recognizable as a routable IPv4 packet (not anti-spoofing)
- Forward/Bridge everything that doesn't have a source in the NATed inside block (assuming upstream box will drop). This is the specific implementation of the CPE logic used by the OpenResolver measurements of spoofability, discussed in the "Measurements" section.

Another important aspect is that IPv6 is a game changer, since a common setup does not assume NAT. Compliance with RFC6092 (https://tools.ietf.org/html/rfc6092) prevents spoofing by requiring that "outbound packets MUST NOT be forwarded if the source address in their outer IPv6 header does not have a unicast prefix configured for use by globally reachable nodes on the interior network", but it is unclear how widely it is implemented[15]. Devices that passed the IPv6 Ready Logo CE Router Interoperability Test by the InterOperability Laboratory UNH-IOL have ingress filtering that prevents spoofing from the interior network enabled by default[16].

Enterprises

Incentives

There are fewer incentives from their service providers to control spoofing. Since enterprise networks tend to have tight security policies, it may make sense to integrate measures, like egress filtering, in such policies.

It is not uncommon for small enterprise networks to use a NATed setup. In such case the issue is mostly non-existent, until IPv6 is deployed (assuming without a NAT). For this category of networks integrating anti-spoofing as a default configuration in the small business router setup could improve protection at the edge[17].

Datacenters and hosting providers

Incentives

Incentives to deploy anti-spoofing measures for this category of network operators may vary significantly depending on the network setup and business model. In general the main incentive would come from the requirement to isolate customers as well as to prevent accounting issues (e.g. a service hijack similar to broadband providers).

Common setup and anti-spoofing measures

It is common for hosting providers to provide public IP addresses for their customers themselves, and these IP addresses are properly registered to the provider so that complaints about usage can be tied back to the tenant responsible. Traffic from a tenant's virtual machines (VMs) to Internet destinations are NAT'ed to these public IP addresses. Filtering rules enforce that all packets sent must bear a source address associated with the sending tenant. No tenant is allowed to advertise arbitrary prefixes into BGP. Many providers employ special fraud teams to address abusive behavior, which can be detected via data analysis or reported by outside parties.

Of course there are less diligent providers that pay much less attention to address spoofing as long as it does not result in attacks on their own infrastructure or their other customers. In many respects the setup is similar to enterprises, perhaps even allowing more flexibility and less policy control.

Based on these characteristics, it seems that the datacenter and hosting environments are likely the main sources of spoofed traffic. There are known cases of hosting providers who fully deploy anti-spoofing measures, but it is unclear to what extent this is a typical case.

Deployment considerations and challenges

Deployment of anti-spoofing measures in general does not provide a direct and immediate return on investment. Yet, their deployment involves costs and additional risks. One of the ways to improve the chances these measures are deployed is to significantly decrease costs and risks associated with them. Network equipment capability, readily available configurations, and clear operational instructions/guidance play a crucial role here.

An inventory of device capabilities is needed.

Challenges related to equipment capabilities can be split into two main categories:

- Equipment not having necessary capabilities. For example not all vendors, and not all equipment types, support uRPF, or some of the more useful forms, like feasible uRPF. The problem is exacerbated by the fact that efficacy of the measures is proportional to the proximity of the point in the network where it is applied to the edge, usually suggesting lower-end devices with a limited set of features.
- Equipment with necessary capabilities, but unknown/untested performance implications of switching them on in specific environments. This creates FUD, fueled by claims that switching uRPF can have a 30% performance hit.

A comprehensive testing of most common equipment typically used in network parts where anti-spoofing is effective [see next point on guidance] could help build confidence for operators willing to use these features.

Anti-spoofing by default

Effectiveness of anti-spoofing measures is proportional to the proximity of application point to the place where address spoofing might happen – at the edge This is related to the certainty of what source addresses can originate traffic, which in turn results in reducing the potential risk of affecting legitimate traffic by such measures.

For instance, the first recommendation that tried to address the problem of source IP address spoofing, BCP38, assumed that measures are deployed one tier up from the edge – at an upstream of a stub network. More complex topology, for instance if a stub network has multiple upstreams, required more nuanced solutions, recommended in BCP84. Still, because the data plane is not always congruent with the control/routing plane, even these solutions may not work. For example a network may implement load balancing of traffic by announcing part of its address space to one of its upstream providers and another part to another upstream, while sending all traffic through only one upstream. Without explicit out-of-band communication between the stub and upstream networks, for the upstream network legitimate traffic is undistinguishable from the spoofed traffic.

Addressing the challenge at the edge (or close to the edge) seems only possible with automation and if anti-spoofing measures are switched on by default.

The main problem is that for most devices the topology of connected networks is still too complex to make assumptions about what addresses to expect on a given network interface. This gets into questions about how much an upstream network that wants to implement filters can assume they know about the topology of the downstream networks based on what the upstream knows about how to route to the downstream. In many cases it is impossible to determine whether a network is single-homed or multihomed, and what rules it might be using to decide which packets use which exit(s).

In many cases, an individual router simply does not have enough information about the surrounding topology. A close approximation of the cases where anti-spoofing (uRPF-like) could be switched on by default could be described as follows:

- For networks directly connected to an interface (i.e. networks, not learned via routing protocols, or by installed static routes)[18];
- For routes learned via SPF protocols (OSPF, IS-IS), unless they are affected by installing static routes, importing routes from other protocols, or using asymmetric metrics.

The first case looks like a very simple and isolated setup, but in many cases it is the building block for more complex networks. By increasing anti-spoofing protection in these cases we may expect hardening of the overall networks over time against spoofing.

Diffusion of anti-spoofing capabilities can also be facilitated by embedding them into other "building blocks" - key open source software components used by network equipment vendors.

Tailored operational guidance

There is some material out there to help network administrators implement anti-spoofing measures, but it is not well consolidated, sometimes too generic, and not well maintained. We need easy instructions tailored to typical environments and use cases. A BCOP (Best Current Operational Practice) document might solve part of the problem.

Incentives, communication and awareness

Deployment of anti-spoofing measures is to a great extent hindered by economic factors, like lack of practical deployment information and low return on investment.

In general terms and setting all the above-mentioned incentives aside, from a network operator perspective, spoofed traffic constitutes a negligible fraction and is usually unnoticeable (that is the whole idea of a reflection attack). It also contributes only to a specific type of attack – a reflection-amplification DDoS.

This view might also be shared by an external observer, of course, when such an observer is not under attack. Are the scale and resources needed to contain source IP address spoofing proportional to the scale of the problem they solve?

Given how disproportionally low the costs of mounting a reflection attack are compared to the damage, and collateral damage, it produces[19] – the answer is “yes”, especially in the long run. If nothing is done to keep spoofing to a reasonable and controlled level of containment the DDoS phenomena may get out of hand and the utility of the Internet may decrease significantly[20].

This brings up a question of better articulation of incentives, especially in a non-coordinated environment, where each network operator is acting independently.

Again, apart from a specific network environment, there are few direct incentives, like security protection of a network’s own assets. An important conclusion during the discussion was that raising general awareness among the actors at all levels up to management could facilitate an environment where other measures discussed here can meet a more favorable reception and therefore have a higher chance of being adopted or followed.

There is also a possibility to leverage a more coordinated environment when it comes to containing the problem. We can consider two main approaches: a voluntary coordinated adoption and a recommended or mandatory regulation.

For the voluntary adoption we can leverage the common understanding that a world without spoofing is beneficial to all network operators and their customers. The main obstacle remains that even those willing to spend some money are not getting the benefits of their actions back, since the risk of a potential DDoS attack depends on how many other networks apply anti-spoofing measures. A possible approach here is making a social contract between the parties making their contribution to the good of the commons and expecting the others to do the same. One example of such an approach is the Mutually Agreed Norms for Routing Security, or MANRS[21].

There are also examples of a regulatory approach, when certain actions are recommended or mandated by a national telecommunication authority. For example in Finland, network operators are required to:

“[…] prevent in their IP interconnection interfaces such IP traffic to the operator's network where the source address of a received IP packets 1) belongs to an IP address space that the telecommunications company itself administers or advertises, 2) belongs to an IP address space that is reserved for non-public use, or 3) do not belong to routes advertised by a telecommunications company that conveys traffic to other telecommunications companies.”[22].

It would be interesting and useful to survey Finnish ISPs’ experiences with implementing measures to comply with this regulation and also have statistically representative data on its effect on the spoofability of Finnish networks.

Conclusions

Measurements

1. Improving knowledge about the origin and other parameters of spoofed traffic is important for the development of an effective strategy.
2. Unbiased statistical data is needed to credibly demonstrate the problem and to be able to track the trend line to demonstrate progress.
3. Two main techniques - running an insider test (e.g.Spoofer) or a DNS reflection have limitations and may produce biased results that are difficult to extrapolate.
4. Applying a big data approach, correlating various data sets (e.g. traces of DDoS attacks, data about botnets) might be helpful.

Traceability

1. Traceability is a challenge inside a network, requiring telemetry and resources. Inter-provider traceability looks like an unsolvable problem at the moment. There is a lack of monitoring tools, and especially forensics tools.
2. The need for traceability comes not only as a mitigation tool or name-and-shame (albeit this will help accountability), but also from the requirement to be able to trace back to criminals and prosecute them, thus substantively changing the equation.

Deployment considerations

1. An inventory of device capabilities is needed.
2. There is a lot of material out there to build capacity, but it is not well consolidated and sometimes too generic. We need easy instructions tailored to typical environments and use cases. A BCOP doc is underway, and might solve part of the problem.
3. Anti-spoofing must be on by default. The only place you can really do anti-spoofing is the edge (or the closest possible). Addressing the challenge at the edge (or close to the edge) seems only possible with automation and if anti-spoofing measures are switched on by default.
4. Implementation of anti-spoofing mechanisms and solutions in key open source projects is equally important.

Incentives, communication and awareness

1. We need to be able to better articulate incentives: business continuity, reputation, and/or compliance (if a policy action is a possibility). Bring the message to decision makers.
2. Requesting anti-spoofing capabilities as a standard feature. RFPs referencing an RFC could be a helpful tool for that.
3. No one solution to the problem exists, but combined efforts in the areas discussed in this paper may make an impact.

Acknowledgements

This paper is largely based on discussions among the participants of the roundtable organized by the Internet Society in February 2015. Specifically: Fred Baker, Jared Mauch, Douglas Montgomery, Matthew Luckie, David Freedman, Jeff Howe, Wes George, Roland Dobbins, Chris Roosenraad, Paul Ferguson, Merike Kaeo, Jeff Haas, Paul Vixie, Eliot Lear, Olaf Kolkman, Andrei Robachevsky.

Endnotes

[1] "Rate-limiting State. The edge of the Internet is an unruly place", ACMqueue, volume 12, issue 2, https://queue.acm.org/detail.cfm?id=2578510

[2] See "Chronology of a DDoS: SpamHaus", http://blogs.cisco.com/security/chronology-of-a-ddos-spamhaus

[3] "Hell of a Handshake: Abusing TCP for Reflective Amplification DDoS Attacks", http://syssec.rub.de/research/publications/hell-of-a-handshake/

[4] See project "Open Resolver", http://openresolverproject.org/

[5] http://www.redbarn.org/dns/ratelimits

[6] http://datatracker.ietf.org/wg/savi/

[7] Source Address Validation Improvement (SAVI) Framework, RFC7039, http://datatracker.ietf.org/doc/rfc7039

[8] See "Denial-of-service attack", http://en.wikipedia.org/wiki/Denial-of-service_attack#Reflected_.2F_spoofed_attack

[9] According to the OpenResolver project, http://openresolverproject.org/

[10] Hell of a Handshake: Abusing TCP for Reflective Amplification DDoS Attacks, http://syssec.rub.de/research/publications/hell-of-a-handshake/

[11] It cannot be run as a browser plugin or a JavaScript, since it requires fabricating a packet with a spoofed source IP address.

[12] https://www.mturk.com/mturk/welcome

[13] There is some work underway (see DDoS Open Threat Signaling (DOTS) BoF at IETF92, http://trac.tools.ietf.org/bof/trac/wiki/BofIETF92) for a standards based approach for on-premise DDoS mitigation devices to communicate threat and telemetry data to parties involved (e.g. service alleviate the challenges mentioned for the traceability.

[14] Unicast RPF, http://en.wikipedia.org/wiki/Reverse_path_forwarding

[15] The "IPv4 and IPv6 eRouter Specification" (CM-SP-eRouter-I10-130808) by CableLabs lists this RFC6092 recommendation as "critical".

[16] The list of certified CE routers is available here: https://www.iol.unh.edu/registry/ipv6-cerouter

[17] See previous comments regarding compliance with RFC6092.

[18] But see some discussion about this case in sec.3 of “Experiences from Using Unicast RPF”, https://www.ietf.org/archive/id/draft-savola-bcp84-urpf-experiences-03.txt

[19] Spamhaus attack

[20] See, for instance, an example of a mitigation strategy, when the utility of the global Internet is sacrificed in order to be able to sustain a DDoS, if only for a local community: https://tn-init.nl.

[21] Aka Routing Resilience Manifesto, http://www.routingmanifesto.org/

[22] https://www.viestintavirasto.fi/attachments/maaraykset/M67A_2015.pdf

Internet Society
Galerie Jean-Malbuisson, 15
CH-1204 Geneva
Switzerland
Tel: +41 22 807 1444
Fax: +41 22 807 1445
www.internetsociety.org

1775 Wiehle Ave.
Suite 201
Reston, VA 20190
USA
Tel: +1 703 439 2120
Fax: +1 703 326 9881
Email: info@isoc.org