Estudio Anti-Spoofing en la región

Autores: Paula Venosa, Nicolás Macia, Guillermo Pereyra
Coordinación/revisión: Graciela Martínez
Edición: Área de Comunicaciones
Área: CSIRT de LACNIC

Diciembre 2022

lacnic
www.lacnic.net

Resumen ejecutivo

Este estudio logró identificar fuentes de información para visibilizar la correcta implementación de buenas prácticas para mitigar el IP *spoofing* en redes de la región de América Latina y el Caribe. A partir del análisis de los datos obtenidos, se recomiendan acciones para que las organizaciones puedan detectar y mitigar este problema. De esta forma no solo se da a conocer el estado de adhesión a las buenas prácticas, sino también se muestra que es posible mejorar el estado de la región para prevenir el spoofing de direcciones IP.

Estas estadísticas brindan al CSIRT de LACNIC información para trabajar directamente con las organizaciones asociadas y reducir los riesgos de que los recursos asignados a éstas sean utilizados con fines maliciosos.

Abstract

Este informe constituye una única fuente de información para entender de qué se trata el IP *spoofing* y las problemáticas asociadas al mismo. Constituye un punto de partida para generar conciencia en los asociados de LACNIC respecto a esta temática, así como en toda la comunidad.

A lo largo del informe se brindan los resultados del relevamiento realizado sobre la presencia de redes que permiten realizar IP *Spoofing* y que pueden ser utilizadas para realizar ataques en América Latina y el Caribe. Se describen los ataques más comunes, además de la metodología aplicada, las fuentes consultadas, los resultados recogidos en un relevamiento inicial, las principales recomendaciones para su corrección y las conclusiones obtenidas.

También se ponen a disposición de la comunidad datos globales sobre el cumplimiento de buenas prácticas para mitigar el IP *spoofing* en la región, según las fuentes consultadas *Spoofer de CAIDA* y *Closed Resolver Project*.

Objetivo

Contribuir a la mejora de la seguridad en la región de LACNIC, alentando a las organizaciones asociadas a implementar buenas prácticas de seguridad para colaborar con la estabilidad y resiliencia de Internet.

Motivación

Mejorar los niveles de ciberseguridad de los recursos IP bajo la administración de LACNIC.

Minimizar el posible uso de técnicas de IP spoofing para llevar adelante ataques tanto a la infraestructura que soporta Internet como a las redes de los asociados, a través de la concientización, visibilización y aporte de herramientas a la comunidad de LACNIC para abordar esta problemática.

IP Spoofing

¿Qué es?

El *IP spoofing* es la suplantación de direcciones IP. Se refiere a la creación de paquetes IP con una dirección IP origen falsa, con el objetivo de hacerse pasar por otro sistema informático. La suplantación de direcciones IP permite a actores maliciosos realizar distintos tipos de ataques y su detección es muy dificultosa.

Para llevar a cabo el *IP spoofing*, un actor malicioso se vale de herramientas que modifican la dirección IP de origen del encabezado IP de los paquetes, de modo que el sistema receptor interprete que dichos paquetes provienen de otro lugar.

El IP *spoofing* es posible debido a la falta de implementación de políticas de filtrado adecuadas para prevenirlo.[1] [2] Estas políticas de filtrado deben encargarse de verificar la dirección de origen de los paquetes. Para referirnos a estas políticas hablaremos de *filtros de tipo source address validation* o *filtros de tipo SAV*.

Una organización puede, sin saberlo, permitir el IP *spoofing*, tanto sobre el tráfico que ingresa (*inbound IP spoofing*) como sobre el tráfico que sale a internet (*outbound IP spoofing*).

Inbound IP spoofing

Se refiere a la posibilidad de que ingrese a la organización, desde sistemas autónomos externos, tráfico con direcciones IP de origen de la propia organización o de redes *bogons.*[3] Estas situaciones no deberían ocurrir y se deben filtrar, pues:

- Los bloques IP de la organización deberían ser el destino del tráfico entrante y no el origen. El *inbound IP spoofing*, cuando utiliza direcciones IP de la propia organización, da lugar a diferentes ataques que pueden afectar a sus servicios y a sus usuarios.
- Las redes *bogons* no deberían aparecer en las tablas de ruteo de internet.

El *inbound IP spoofing* es posible cuando una organización no implementa filtros de tipo SAV sobre el tráfico entrante o, como los llamaremos a partir de ahora, *filtros de tipo inbound SAV*.

[1] RFC 2827 - Network Ingress Filtering: Defeating Denial of Service Attacks which employ IP Source Address Spoofing ‹https://www.rfc-editor.org/rfc/rfc2827›

[2] RFC 3704 - Ingress Filtering for Multihomed Networks ‹https://www.rfc-editor.org/rfc/rfc3704.html›

[3] Redes Bogons - ‹https://team-cymru.com/community-services/bogon-reference/›

Outbound IP spoofing

Se refiere a la posibilidad de que salga de la organización, hacia sistemas autónomos externos, tráfico IP que tiene direcciones IP de origen que no pertenecen a la propia organización. Estas situaciones no deberían ocurrir y se deben filtrar dichos paquetes, pues permiten la realización de distintos ataques que pueden afectar la disponibilidad de servicios de terceros.

El *outbound IP spoofing* es posible cuando una organización no implementa *filtros de tipo outbound SAV*.

¿Qué protocolos se aprovechan para realizar IP spoofing?

El IP *spoofing* se realiza en la capa de red, a nivel IP. Sin embargo, dependiendo del ataque, podría cambiar el protocolo encapsulado por IP.

Por ejemplo, en ataques de DoS[4] por inundación o *flooding* se pueden utilizar diversos protocolos:

- HTTP para saturar el servicio dado.
- TCP con *flags* específicos, como syn, fin, rst, etc., para saturar el servicio dado.
- ICMP para dar más volumen al ataque y saturar la capacidad de la red o del servidor.
- UDP para dar más volumen al ataque y saturar la capacidad de la red o del servidor.

Para los que se conocen como *ataques de DDoS* y otros en los que se necesita interactuar con servicios utilizando tráfico *spoofeado*, se utilizan protocolos de aplicación que se transportan sobre UDP.

Las aplicaciones que utilizan estos protocolos brindan lo que se conoce como *servicios reflexivos*. Estos servicios, que utilizan UDP como protocolo de transporte, no necesitan de un *handshake* previo para el inicio de la conexión.

[4] DoS / DDoS - ‹https://www.osi.es/es/actualidad/blog/2018/08/21/que-son-los-ataques-dos-y-ddos›

¿Qué ataques se aprovechan del IP spoofing y cómo lo explotan?

Ataques de DoS por inundación

El objetivo de un ataque de DoS es comprometer la disponibilidad de un servicio en la organización víctima. En caso de que la organización donde está el *host* controlado por el actor malicioso no implemente filtros de tipo outbound SAV, este podría ocultar el origen del ataque realizando IP *spoofing*.

Ataques DDoS por amplificación

Para que los ataques de DDoS sean posibles, deben darse dos condiciones. Por un lado, la organización desde donde realiza el ataque el actor malicioso no tiene implementados filtros de tipo *outbound* SAV y, por otro, es necesaria la identificación de servicios reflexivos y amplificables vulnerables accesibles desde Internet.[5]

En el informe «Estudio de recursos de internet presentes en la región de LACNIC con protocolos UDP que posibilitan ataques de tipo DDoS» se trata la segunda problemática.[6] Allí se describe cómo es posible realizar ataques de denegación de servicio distribuido o DDoS.

De manera resumida, el actor malicioso debe:

1. Identificar servicios reflexivos y amplificables vulnerables, accesibles desde internet.
2. Enviar requerimientos *spoofeados* a los servicios vulnerables identificados previamente (presentes en internet por desconocimiento), utilizando como IP de origen la IP del objetivo del ataque.
3. Realizar de manera coordinada el paso anterior desde distintos *hosts*, en organizaciones que no aplican filtros de tipo *outbound* SAV, posibilitando así incrementar el volumen del ataque, de manera proporcional a la cantidad de *hosts* utilizados por el actor malicioso.

---

[5] ‹https://www.cisa.gov/uscert/ncas/alerts/TA14-017A›

[6] ‹https://imasd.lacnic.net/reportes/ciberseguridad/estudio-de-recursos-de-internet-presentes-en-la-region-de-lacnic-con-protocolos-udp-que-posibilitan-ataques-de-tipo-ddos.pdf ›

NSNXAttack

NSNXAttack es un ataque que, a diferencia de los anteriores, se aprovecha de que la organización víctima no implementa filtros de tipo *inbound* SAV. Estos ataques se valen del hecho de que un actor malicioso puede realizar consultas al DNS *resolver* como si estuviese utilizando un *host* de la organización. En particular, en este tipo de ataque se busca afectar la disponibilidad del servicio brindado por el DNS recursivo de la organización.[7]

DNS cache poisoning

El DNS *cache poisoning*[8] es un ataque para proveer un servidor de nombres de dominio (DNS) de datos que no se originan en fuentes autoritativas. Los problemas de DNS *cache poisoning* pueden darse independientemente de que la organización víctima del ataque implemente filtros de tipo *inbound* SAV. Sin embargo, la no implementación de dichos filtros puede facilitar el ataque.

El DNS *cache poisoning* afecta la integridad de los datos provistos por el servicio de DNS, lo que ofrece el eslabón inicial de un ataque más complejo. Por ejemplo, uno que se valga de esta situación para realizar *phishing* a usuarios, estafas bancarias[9] o el *defacement* de sitios web.[10]

Otros posibles ataques

Dado que el *inbound IP spoofing* permite a un actor malicioso interactuar con servicios internos desde direcciones locales, podrían existir otros protocolos de aplicación en el que

---

[7] ‹https://www.nxnsattack.com/shafir2020-nxnsattack-paper.pdf›

[8] ‹https://es.wikipedia.org/wiki/Envenenamiento_de_DNS›

[9] ‹https://www.zscaler.it/blogs/security-research/brazilian-bank-targeted-phishing-site-and-dns-poisoning›

[10] ‹https://www.theguardian.com/technology/2017/aug/31/wikileaks-hacked-ourmine-group-julian-assange-dns-attack›

distintos vectores de ataque se aprovechen de que no se aplique el paradigma de confianza cero, conocido como *zero trust.* [11] [12] [13] [14]

Beneficios de tomar acciones contra el IP spoofing

Los asociados de LACNIC se pueden beneficiar de tomar acciones para evitar el IP *spoofing* con la mitigación de distintos ataques que aprovechan el *inbound IP spoofing* contra su organización. En particular, se pueden mencionar ataques como *NSNXAttack* y DNS *cache poisoning*, así como otros que puedan surgir en el futuro, basados en el aprovechamiento de la confianza dada por comunicaciones internas en ambientes no alineados con el paradigma de *confianza cero*.

Asimismo, la comunidad de internet se puede beneficiar de que los distintos actores tomen acciones para evitar el IP *spoofing*, al limitar las redes desde las que es posible lanzar ataques de DDoS.

Fuentes de información disponibles

Para este estudio se utilizaron fuentes de datos activas que pueden brindar información pertinente sobre el estado del IP *spoofing* en la región.

Estas son:

- ‹https://www.caida.org/projects/spoofer/›
- ‹https://closedresolver.korlabs.io/›

[11] ‹https://storage.googleapis.com/pub-tools-public-publication-data/pdf/43231.pdf›

[12] ‹https://cloud.google.com/blog/topics/developers-practitioners/what-zero-trust-identity-security›

[13] ‹https://www.cloudflare.com/es-es/learning/security/glossary/what-is-zero-trust/›

[14] ‹https://www.akamai.com/es/our-thinking/zero-trust/zero-trust-security-model›

El proyecto Spoofer de CAIDA

El proyecto spoofer de CAIDA[15] es referencia obligada para obtener datos sobre problemáticas relacionadas al IP *spoofing*. Funciona desde 2006 y cuenta con un cliente a partir del cual se pueden ejecutar tests. Con estos testeos se evalúa si una red permite tanto *outbound IP spoofing* como *inbound IP spoofing*.

En la siguiente figura se puede observar la arquitectura del *spoofer* de CAIDA, sus componentes y la interacción:

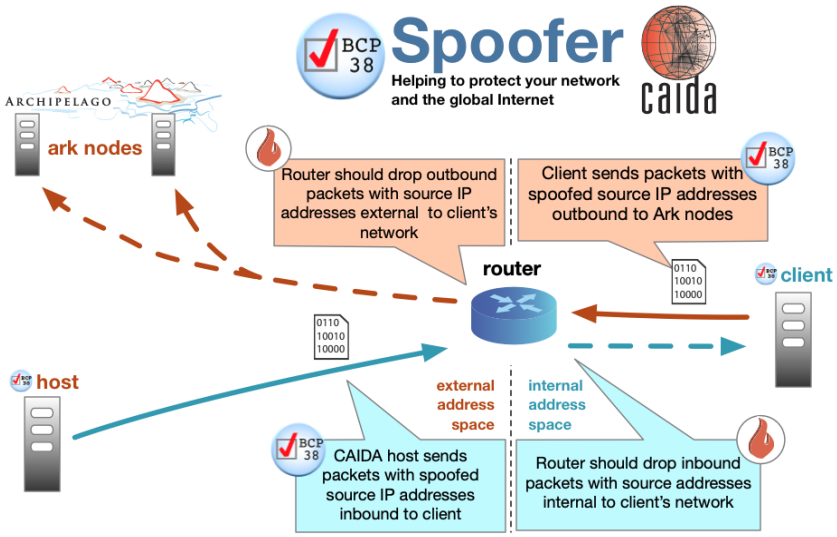


Fig. 1- Spoofer CAIDA ‹https://www.caida.org/projects/spoofer/images/spoofer-diagramer.png›

En el cliente está el *software* que ejecuta los tests para evaluar las políticas de filtrado de la organización. El cliente envía paquetes, con una dirección IP de origen *spoofeada*, a los nodos *ark* del archipiélago de CAIDA.

En caso de aplicar políticas de filtrado, el *router* de la organización descarta los paquetes salientes con dirección IP de origen externa a la red del cliente.

Los *nodos ark* funcionan como *honeypots*. Si reciben los paquetes generados en el cliente con IP *spoofeada*, se determina entonces que la organización no filtra el *outbound* IP

[15] ‹https://www.caida.org/projects/spoofer/›

spoofing. Si los *nodos ark* no reciben dichos paquetes, significa que la organización está filtrando el *outbound* IP *spoofing*.

El *host* de CAIDA es el dispositivo externo desde el cual se realiza la evaluación del *inbound IP spoofing*. Desde allí se envían paquetes con direcciones IP de origen *spoofeadas* hacia el cliente.

En caso de aplicar políticas de filtrado, el *router* de la organización descarta los paquetes entrantes con direcciones IP de origen que no deberían ser aceptadas como direcciones IP internas de la red del cliente.

El proyecto brinda para cada país información de la cantidad de bloques IP testeados, la cantidad de bloques que no filtran el *outbound IP spoofing* y la cantidad de bloques que sí lo hacen. También provee reportes que muestran datos respecto a la remediación en cada uno de estos países.

Entre las ventajas de este proyecto destacamos que:

- Brinda al usuario información sobre la existencia en la organización de filtros de tipo *inbound* y *outbound* SAV.
- Permite evaluar el IP *spoofing,* tanto en redes IPv4 como en redes IPv6.
- La aplicación cliente que se instala en el dispositivo testea automáticamente la existencia de filtros de tipo *inbound* y *outbound* SAV en las redes a las que dicho dispositivo se conecta.

Como desventajas podemos mencionar que:

- Los datos públicos muestran solamente el *outbound* IP *spoofing*.
- Requiere la instalación de una aplicación en el cliente.
- Tiene datos parciales de redes IPv4 e IPv6.

El Proyecto Closed Resolver

En el año 2019, The Closed Resolver Project[16] hizo un relevamiento de todo el espacio de direcciones IPv4 utilizando el DNS para evaluar la presencia de filtros adecuados contra el *inbound* IP *spoofing*. Se utilizó el DNS para realizar consultas que parecieran provenientes de *hosts* de la misma organización donde funciona el servidor de DNS. Cuando el destinatario de la evaluación era un servidor DNS recursivo, era posible la detección del *inbound* IP *spoofing* a partir de la secuencia observada en la siguiente figura (extraída del artículo que describe el objetivo del proyecto).[17]

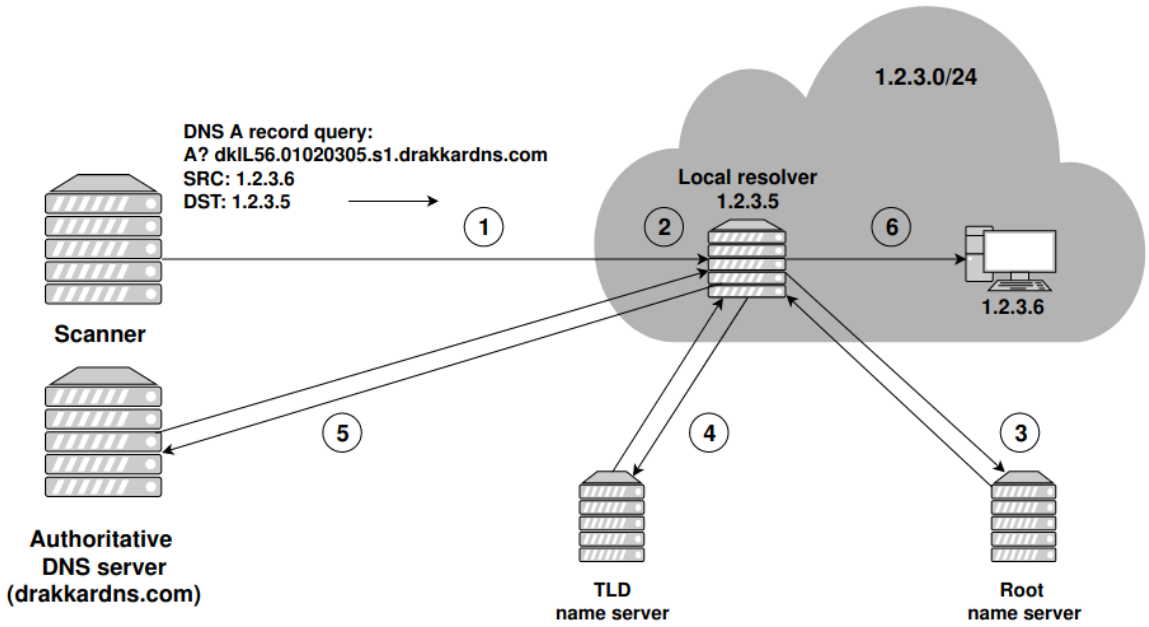


Fig. 2- Metodología de The Closed Resolver Project para la evaluación del *inbound* IP *spoofing* ‹https://arxiv.org/pdf/2002.00441.pdf›

La metodología observada en la figura muestra que:

1. Se realizan requerimientos de DNS a diferentes direcciones IPv4 (1.2.3.5 en el ejemplo) provenientes desde direcciones IP de la propia organización (1.2.3.6 en el ejemplo).

16 ‹https://closedresolver.korlabs.io/›

17 Imagen extraída de ‹https://arxiv.org/pdf/2002.00441.pdf›

2. En caso de que no implemente filtros de tipo *inbound* SAV será posible el *inbound* IP *spoofing* y el paquete llegará al destino (1.2.3.5 en el ejemplo).
3. Si en la dirección de destino del requerimiento de DNS funciona un *resolver* de DNS, este intentará resolver la consulta (en el ejemplo, se consulta el registro A de dkIL56.01020305.s1.drakkardns.com). Para ello, el resolver inicialmente contactará a un *root server* que brindará la información de los Name Servers (NS s) asociados al Top Level Domain (TLD).
4. Seguidamente, se contactará con uno de los NSs asociados al TLD correspondiente. La respuesta del TLD brindará información sobre los NSs asociados al dominio *drakkardns.com.*
5. Dado que el dominio *drakkardns.com* del ejemplo, es controlado por The Closed Resolver Project, al recibir la consulta se puede confirmar que proviene de una red que no implementa filtros de tipo *inbound* SAV.
6. Finalmente, el servidor de DNS brinda la información solicitada a quien cree que le hizo la consulta (1.2.3.6 en el ejemplo).

Las principales ventajas de este proyecto son:

- Tiene datos completos de redes IPv4.
- No requiere la instalación de ninguna aplicación.

Entre las desventajas distinguimos que:

- Solamente brinda al usuario información de buenas prácticas relacionadas con la aplicación de filtros de tipo *inbound* SAV.
- Para IPv6 solo se cuenta con datos parciales, pues, al no poder escanear todo el espacio de direcciones, se utiliza el servicio *hitlist*[18] para realizar los tests sobre direcciones IPv6 que responden.

18 ‹https://ipv6hitlist.github.io/›

¿Cómo determinar si su organización implementa filtros de tipo SAV?

Para saber rápidamente si una organización implementa adecuadamente filtros de tipo *inbound* y *outbound* SAV, se podría evaluar de forma manual, verificando si son posibles las siguientes situaciones:

- *Inbound* IP *spoofing*: si es posible enviar desde internet hacia la organización tráfico con direcciones IP de origen pertenecientes a bloques IP de la organización.
- *Outbound* IP *spoofing*: si es posible enviar desde la organización hacia internet tráfico con direcciones IP de origen pertenecientes a bloques IP que no sean de la organización.

Mecanismo manual de detección del IP spoofing

Evaluar manualmente si es posible realizar IP *spoofing* es una forma práctica de entender y corroborar el problema. Para detectar manualmente si una organización permite tanto el *inbound* IP *spoofing* como el *outbound* IP *spoofing*, necesitamos contar con:

- Una herramienta para generar tráfico manipulado. Para esto se utilizará *Hping3.*[19]
- Una herramienta para monitorear la llegada del tráfico manipulado. Para esto se utilizará Tcpdump.[20]
- Un dispositivo de red, conectado a la red de la organización que se va a evaluar, que tenga permitida la recepción sin filtros de tráfico a un servicio determinado (por ej.: TCP/443).
- Un dispositivo de red conectado a un sistema autónomo externo a la organización a evaluar.

19 Hping3 ‹http://www.hping.org/›

20 Tcpdump ‹https://www.tcpdump.org/›

Verificación manual de inbound *IP* spoofing

La falta de filtros de tipo *inbound* SAV permiten el *inbound* IP *spoofing*. Esto se puede verificar manualmente generando tráfico *spoofeado* en un dispositivo conectado a un sistema autónomo externo usando como direcciones IP origen a direcciones IP propias de la organización.

El tráfico debe estar dirigido a un dispositivo conectado a la red de la organización. En este último dispositivo es donde podremos concluir, en caso de observar la llegada del tráfico manipulado, que es posible el *inbound* IP *spoofing*.

Los pasos para realizar la verificación manual serían:

Dispositivo conectado a un sistema autónomo externo

```
# Asumiendo que
# la IP del dispositivo conectado a un sistema autónomo externo es y.y.y.y
# la IP del dispositivo en la organización tiene la IP x.x.x.1
# la IP que se usará como IP de origen en la prueba será la dirección x.x.x.2
# Test de verificación inicial: Para verificar que el tráfico llega
hping -S -p 443 -c 1 x.x.x.1
# Test para verificar el inbound IP spoofing
hping -S -p 443 -c 1 -a x.x.x.2 x.x.x.1
```

Dispositivo conectado a la organización a evaluar

```
# Asumiendo que la interfaz de red es la eth0 y usando filtros para observar
# solamente el tráfico de interés
tcpdump -n -i eth0 host y.y.y.y or host x.x.x.2
```

Dependiendo de la actividad mostrada por el comando **tcpdump** se podrá concluir si es posible el *inbound* IP spoofing. En base a las observaciones, los posibles escenarios son:

- Es posible el *inbound IP spoofing*: Para concluir con esto, con el comando *tcpdump* se debería poder observar el tráfico asociado al test de verificación inicial y el tráfico asociado al test del *inbound IP spoofing*.

```
IP y.y.y.y.7323 > x.x.x.1.443: Flags [S], seq 45611324, length 0
```

```
      IP x.x.x.1.443 > y.y.y.y.7323: Flags [R.], seq 0, ack 45611325, length
0

      IP x.x.x.2.1899 > x.x.x.1.443: Flags [S], seq 883275884, length 0

      IP x.x.x.1.443 > x.x.x.2.1899: Flags [R.], seq 0, ack 883275885, length
0
```

- No es posible el *inbound IP spoofing*: Para concluir con esto, con el comando *tcpdump* se debería poder observar solo el tráfico asociado al test de verificación inicial.

```
      IP y.y.y.y.7323 > x.x.x.1.443: Flags [S], seq 45611324, length 0

      IP x.x.x.1.443 > y.y.y.y.7323: Flags [R.], seq 0, ack 45611325, length
0
```

- No se puede determinar: En caso de no observar el tráfico asociado al test de verificación inicial, podemos concluir que existe un filtro de red que evita que podamos hacer la verificación en el dispositivo x.x.x.1, utilizando el puerto determinado (en el ejemplo, el puerto TCP/443).

***Verificación manual de* outbound *IP* spoofing**

De manera análoga, la falta de filtros de tipo *outbound* SAV permite el *outbound IP spoofing*. Esto se puede verificar manualmente generando tráfico *spoofeado* en un dispositivo conectado a la organización evaluada, usando como direcciones IP de origen a direcciones IP que no sean de la organización.

El tráfico debe estar dirigido a un dispositivo conectado a un sistema autónomo externo. En este último dispositivo es donde podremos concluir, en caso de observar la llegada del tráfico manipulado, que es posible el *outbound IP spoofing*.

Los pasos para la verificación manual serían:

Dispositivo conectado a la organización a evaluar

```
# Asumiendo que
# la IP del dispositivo en la organización tiene la IP x.x.x.1
# la IP del dispositivo conectado a un sistema autónomo externo es y.y.y.y
```

```
# la IP que se usará como IP de origen en la prueba será la dirección z.z.z.z

# Test de verificación inicial: Para verificar que el tráfico llega

hping -S -p 443 -c 1 y.y.y.y

# Test para verificar el outbound IP spoofing

hping -S -p 443 -c 1 -a z.z.z.z y.y.y.y
```

Dispositivo conectado a un sistema autónomo externo

```
# Asumiendo que la interfaz de red es la eth0 y usando filtros para observar
# solamente el tráfico de interés
tcpdump -n -i eth0 host x.x.x.1 or host z.z.z.z
```

Dependiendo de la actividad mostrada por el comando **tcpdump,** se podrá concluir si es posible el *outbound IP spoofing*. En base a las observaciones, los posibles escenarios que se podrán dar son:

- Es posible el *outbound IP spoofing*: Para concluir esto, con el comando *tcpdump* se debería poder observar el tráfico asociado al test de verificación inicial y el tráfico asociado al test del *outbound IP spoofing*.

```
IP x.x.x.1.7323 > y.y.y.y.443: Flags [S], seq 45611324, length 0
IP y.y.y.y.443 > x.x.x.1.7323: Flags [R.], seq 0, ack 45611325, length
0
IP z.z.z.z.1899 > y.y.y.y.443: Flags [S], seq 883275884, length 0
IP y.y.y.y.443 > z.z.z.z.1899: Flags [R.], seq 0, ack 883275885, length
0
```

- No es posible el *outbound* IP *spoofing*: Para concluir esto, con el comando *tcpdump* se debería poder observar solo el tráfico asociado al test de verificación inicial.

```
IP x.x.x.1.7323 > y.y.y.y.443: Flags [S], seq 45611324, length 0
IP y.y.y.y.443 > x.x.x.1.7323: Flags [R.], seq 0, ack 45611325, length
0
```

- No se puede determinar: En caso de no observar el tráfico asociado al test de verificación inicial, podemos concluir que existe un filtro de red que evita que podamos hacer la verificación.

Mecanismo automático de detección de IP spoofing

La evaluación manual del IP *spoofing* es una forma práctica para entender la problemática con el objetivo de mitigarla. Sin embargo, el uso de herramientas que automatizan su detección es la opción recomendada para tener una visión completa y

exhaustiva del problema. Hoy contamos con dos herramientas que ayudan a realizar la detección del IP *spoofing* de manera automática.

The Closed Resolved Project

El proyecto The Closed Resolved Project es un portal web que brinda información sobre la posibilidad de realizar *inbound IP spoofing*. Se utiliza la dirección IP del visitante para informar sobre los resultados.

Spoofer

Es una herramienta que brinda el proyecto Spoofer de CAIDA.[21] Este proyecto es la fuente principal de datos sobre la problemática del IP *spoofing*. La herramienta que brinda el proyecto es la que permite obtener los datos que se brindan a la comunidad.

Para poder hacer los tests, se debe instalar el *software* llamado *spoofer.*[22] Los tests tienen en cuenta una gran cantidad de situaciones particulares para evaluar tanto el *inbound IP spoofing* como el *outbound IP spoofing*.

Spoofer evalúa la posibilidad de realizar IP *spoofing* tanto con tráfico IPv4 como con tráfico IPv6. Además, evalúa si es posible utilizar como IP de origen direcciones IP que pertenecen a bloques IP no ruteables, como, por ejemplo, bloques IP privados[23] en IPv4 o ULA[24] en IPv6.

Si bien se dispone de una interfaz gráfica, la herramienta funciona como un servicio que hace tests de manera periódica, permitiendo evaluar distintas redes a las que el usuario se conecta.

---

[21] ‹https://www.caida.org/projects/spoofer/#download-client-software›

[22] ‹https://www.caida.org/projects/spoofer/#download-client-software›

[23] ‹http://www.ietf.org/rfc/rfc1918.txt›

[24] ‹https://www.ietf.org/rfc/rfc4193.txt

En la Figura 3 se puede ver cómo se presenta la herramienta al usuario. Se puede ver que el usuario dispone de la funcionalidad para iniciar nuevos tests. También puede ver un resumen de tests realizados y acceder al reporte.

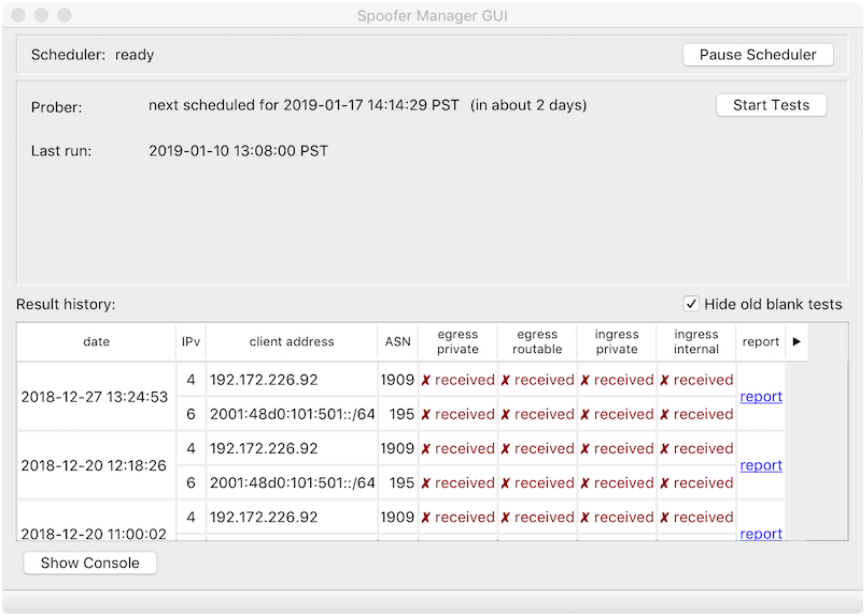


Fig. 3 - Interfaz gráfica del cliente Spoofer de CAIDA

Los reportes generados tienen detalles sobre el *inbound* y el *outbound* IP *spoofing*. En la Figura 4 se puede ver de manera resumida el resultado de tests para IPv4, IPv6, uso de direcciones no ruteables, uso de direcciones ruteables y si es posible tanto el *inbound* como el *outband* IP *spoofing*. En la Figura 5 se puede visualizar la cadena de sistemas autónomos por la que se rutean los paquetes *spoofeados* en las pruebas realizadas.

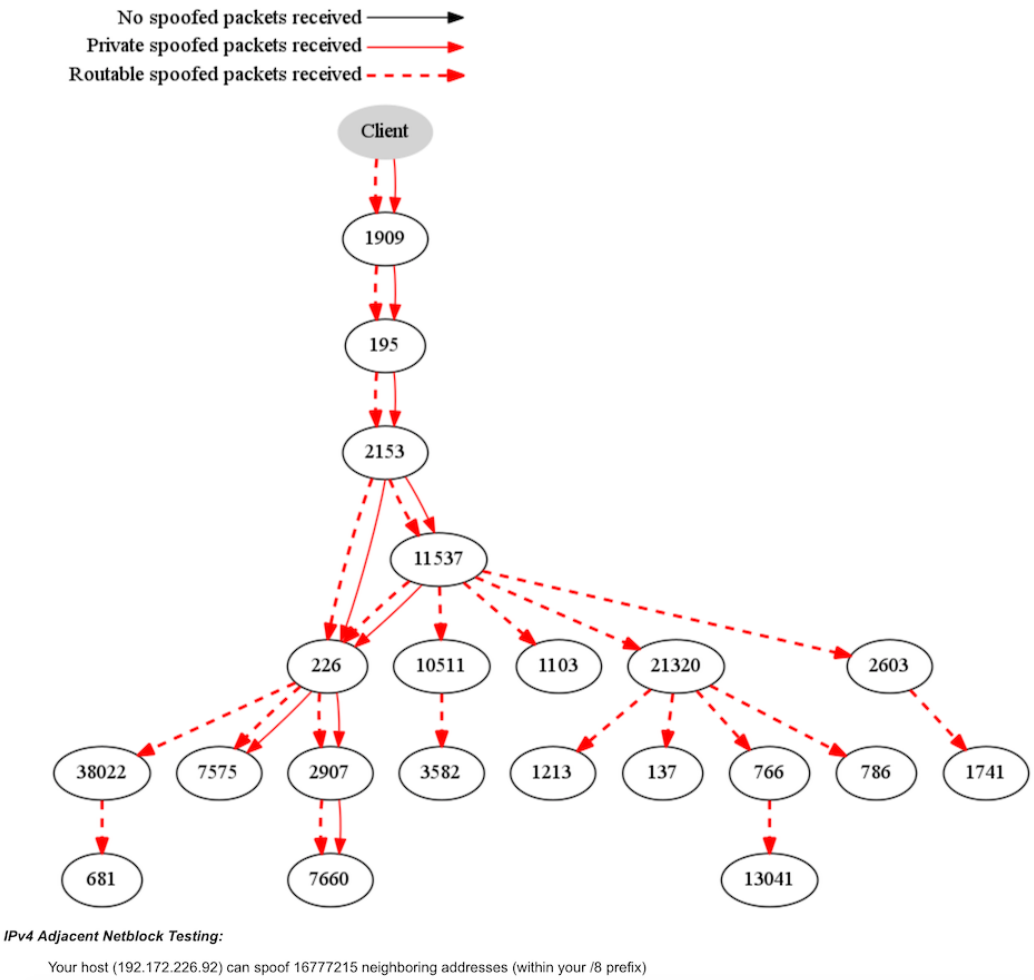
Test run at: 2018-12-27 21:24:53 GMT
Client operating system: OSX
Client IP (v4): [redacted]
Client AS (v4): 1909 (ALPHA-NAP-AS)
IPv4 Probes: 124
Client IP (v6): [redacted]
Client AS (v6): 195 (SDSC-AS)
IPv6 Probes: 82

Egress spoofing summary from the client to our server		
Source address type	**IPv4**	**IPv6**
Private - RFC1918 or ULA	received	received
Routable	received	received
Largest spoofable neighbor prefix	/8	/16
Ingress spoofing summary from our server to the client		
Source address type	**IPv4**	**IPv6**
Private - RFC1918 or ULA	received	received
Internal to the same IPv4 /31 or IPv6 /120 as the client	received	received

Meaning of result status in column:

received	Spoofed packet was received.
rewritten	Spoofed packet was received, but the source address was changed en route.
blocked	Spoofed packet was not received, but unspoofed packet was.
blocked	Spoofed packet was not received, but unspoofed packet was. Pattern of tests from this IP block indicates a switch from allowing spoofing to blocking it.
unknown	Neither spoofed nor unspoofed packet was received.

Fig. 4- Sección resumen del reporte del cliente Spoofer de CAIDA



IPv4 Adjacent Netblock Testing:

Your host (192.172.226.92) can spoof 16777215 neighboring addresses (within your /8 prefix)

Fig. 5- Sección IPv4 AS route del reporte del cliente Spoofer de CAIDA

Análisis de implementación de políticas *antispoofing* en la región LAC

En la presente sección se describe el estado de situación en la región respecto a la problemática de IP *spoofing* en redes que involucran recursos bajo la administración de LACNIC.

Utilizando las fuentes de información incluidas en la sección «Fuentes de información disponibles», se describen aquí los principales hallazgos.

Implementación de filtros de tipo outbound SAV

Los datos de esta sección fueron obtenidos de ‹https://spoofer.caida.org/country_stats.php›, el 17 de octubre de 2022

Observaciones sobre los testeos en la región

La cantidad de bloques evaluados en la región fue de 3082. Por cada AS puede haber varios bloques IP evaluados.

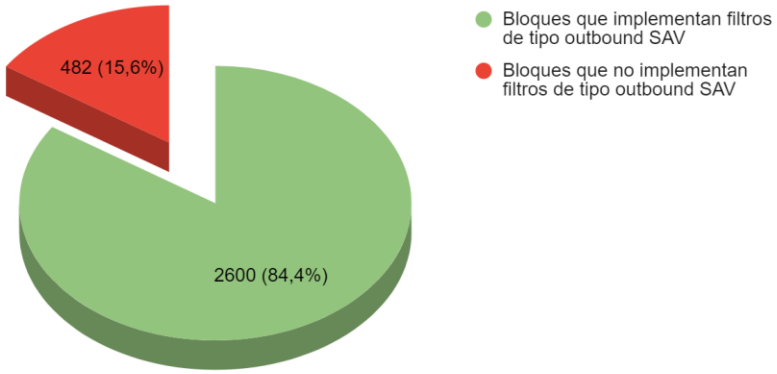


Figura 6

Figura 6 - Se observa que sobre el total de bloques evaluados, un 84,4% implementan filtros de tipo *outbound* SAV mientras que en el 15,6% restante no.

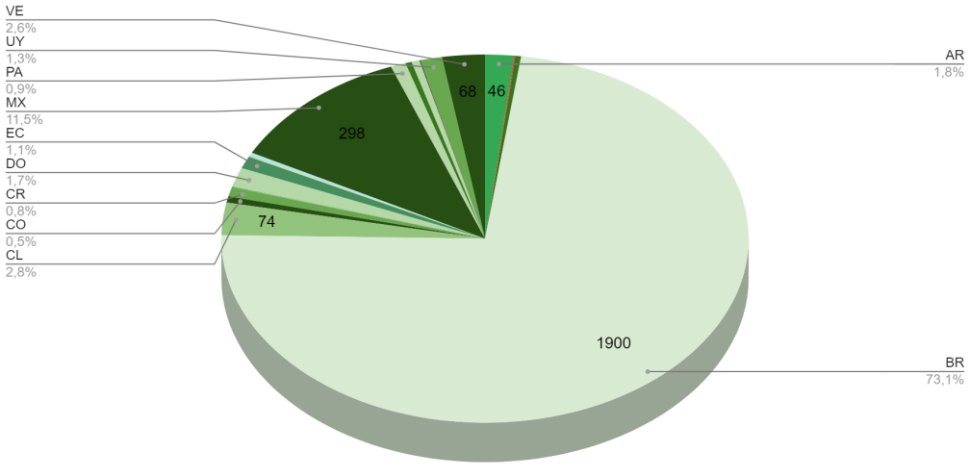


Figura 7

Figura 7 - Se observa el porcentaje que representan los bloques de cada país que implementan filtros de tipo *outbound* SAV sobre el total de bloques testeados que lo implementan (correspondiente al 84,4% de lo que muestra la Figura 6).

Los gráficos anteriores nos dan una primera visión sobre la implementación de filtrado de tipo *outbound* SAV en la región.

Observaciones sobre la distribución de los testeos en los países de la región

Para poder evaluar la situación por país, es importante tener en cuenta la distribución de los bloques analizados, es decir, la cantidad de testeos por país. En la Figura 8 se puede observar la cantidad de países de la región donde se realizaron testeos y la cantidad de países donde no hubo ninguno. En la Figura 9 se puede ver el detalle de cantidad de testeos por país.

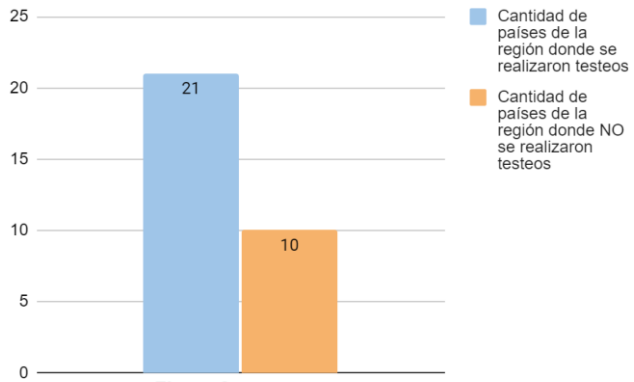


Figura 8

Como se observa en la Figura 8, sobre un total de 31 países, en 21 se realizaron testeos para verificar la aplicación de filtros de tipo *outbound* SAV, mientras que en 10 países no hubo testeos.

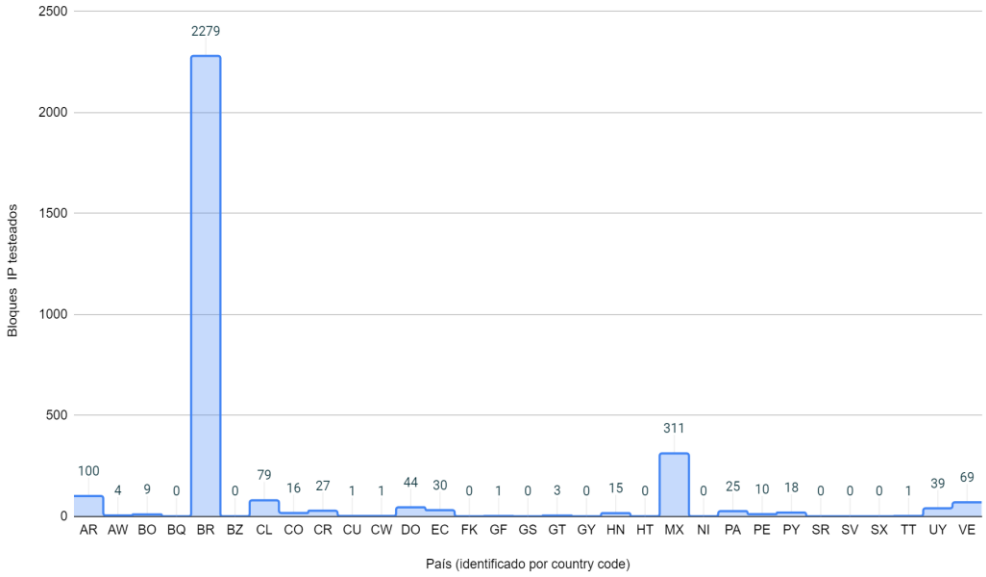


Figura 9

En la Figura 9 se puede observar que el país donde más testeos se realizaron fue Brasil (2279 bloques de red testeados), seguido de México (311 bloques testeados) y Argentina (100 bloques testeados). También se observan los 10 países en los que no se realizaron testeos: Bonaire, San Eustaquio y Saba, Belice, Islas Malvinas (Falkland Islands), Guyana, Islas Georgias del Sur y Sandwich del Sur, Haití, Nicaragua, Surinam, El Salvador y Saint Martin.

Observaciones sobre la implementación de filtros de tipo outbound SAV en los bloques testeados en cada país

En las figuras 10, 11,12 y 13 se puede observar la situación completa de cada país. Esas figuras no incluyen aquellos países donde no se realizaron testeos (Bonaire, San Eustaquio y Saba, Belice, Islas Malvinas (Falkland Islands), Guyana, Islas Georgias del Sur y Sandwich del Sur, Haití, Nicaragua, Surinam, El Salvador, Islas Georgias del Sur y Saint Martin).

En cada una de las figuras se muestra, sobre los bloques testeados por país, la cantidad de bloques de red que no implementan filtros de tipo *outbound* SAV y la cantidad de bloques de red que sí lo hacen:

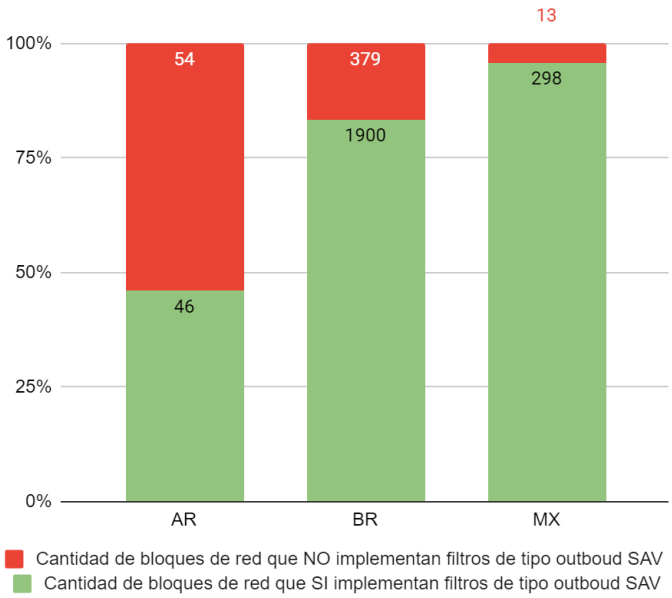


Fig. 10 Situación de Argentina, Brasil y México respecto al *outbound* IP *spoofing*

La Figura 10 incluye los tres países donde se realizaron más testeos (más de 100 testeos). Se puede observar que en Argentina se implementan filtros de tipo *outbound* SAV en un 46% de los bloques testeados. Mientras que, en Brasil se aplican filtros de tipo *outbound* SAV en un 83% del total de bloques testeados y en México se aplican filtros de tipo outbound SAV en un 96% del total de bloques testeados.

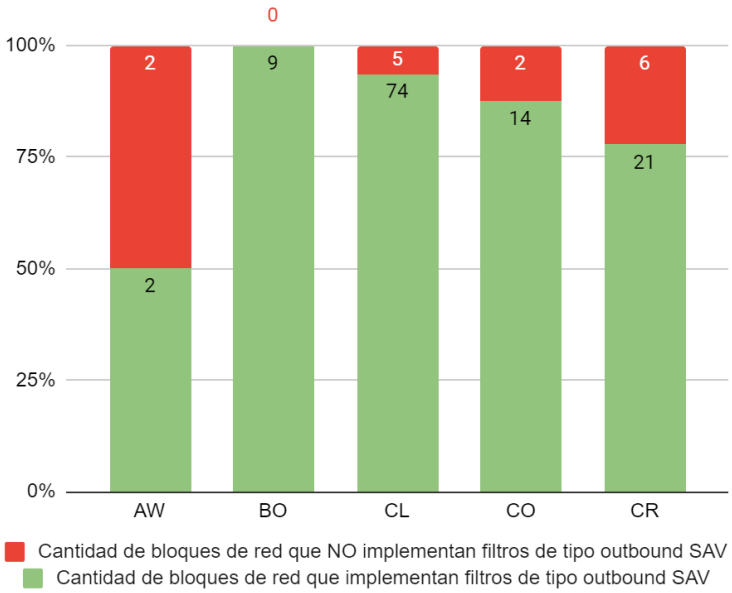


Fig. 11 Situación de Aruba, Bolivia, Chile, Colombia y Costa Rica respecto al *outbound* IP *spoofing*

En la Figura 11 se puede observar también que gran porcentaje (más del 75%) de los bloques testeados implementan filtros de tipo *outbound* SAV, tanto en Bolivia como en Chile, Colombia y Costa Rica. Solo en Aruba ocurre que el porcentaje de bloques de red que implementan filtros es más bajo, solo dos (50% de las redes testeadas) lo implementan

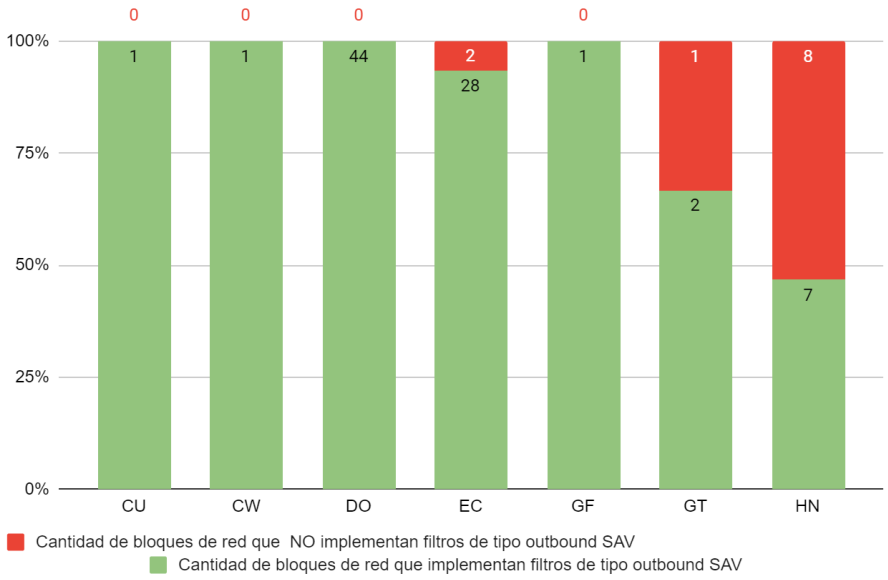


Fig. 12 Situación de Cuba, Curazao, Dominica, Ecuador, Guyana Francesa, Guatemala y Honduras respecto al *outbound IP spoofing*

En la Figura 12 se puede observar que en Cuba, Curazao, Dominica y Guyana Francesa todos los bloques de red testeados implementan filtros de tipo *outbound* SAV. En Ecuador dichos filtros se implementan en 28 bloques de red, lo cual representa más del 90% de las redes testeadas. Mientras que en Guatemala la implementación de filtros se da en dos de los tres bloques de red testeados (66%) y en Honduras el porcentaje de implementación es más bajo, casi del 50% (en siete de los bloques testeados).

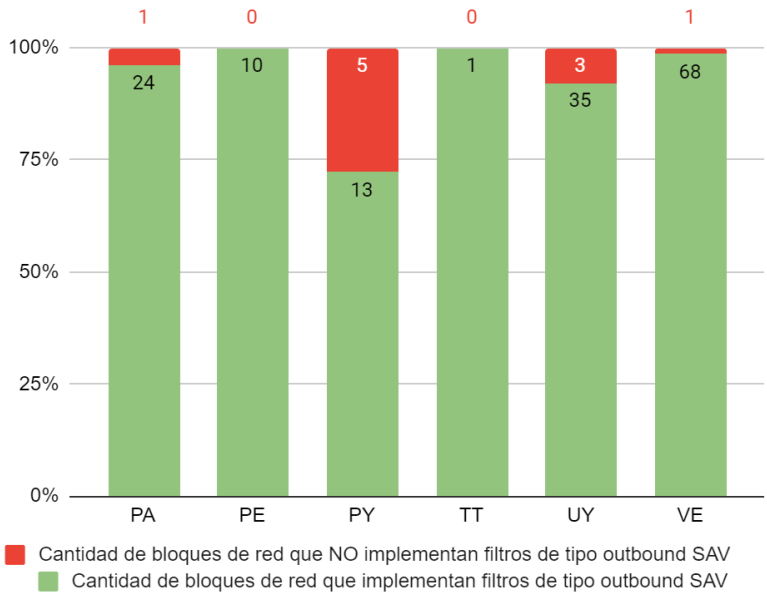


Fig. 13 Situación de Haití, Panamá, Perú, Paraguay, Trinidad y Tobago, Uruguay y Venezuela respecto al *outbound* IP *spoofing*

La Figura 13 muestra también que en todas las redes testeadas en Perú y Trinidad y Tobago se implementan filtros de tipo *outbound* SAV; mientras que en Panamá y Venezuela dichos filtros se implementan casi en la totalidad de las redes testeadas: 24 de 25 y 68 de 69 respectivamente. En Paraguay en 13 de las 18 redes testeadas implementan filtros de tipo *outbound* SAV y en Uruguay 35 de 38. La implementación se da en más del 70% en todos estos países, y en cuatro de ellos en casi el 100% de los bloques testeados.

Desde el punto de vista de la seguridad, el primer paso para mitigar un problema es conocer el estado de situación. Surge de la información relevada que una propuesta —a partir de este estudio— es fomentar la realización de tests que permitan visibilizar mejor la situación de la región respecto de la implementación de filtros de tipo *inbound* y *outbound* SAV.

En este sentido, es importante definir estrategias que permitan realizar testeos en aquellos países de la región donde no se cuenta con información y también incrementar los testeos en aquellos países donde se han testeado pocas redes.

Brasil, al momento de obtener el reporte, es el país que más testeos realizados registra a nivel mundial. En Brasil se analizaron 2279 bloques. Le sigue México con 311 bloques y Argentina con 100 bloques testeados. En el resto de los países se hicieron pruebas en menos de 100 bloques, incluso en algunos solo se testeó uno.

Implementación de filtros de tipo inbound SAV

Sobre la problemática relacionada con que las organizaciones permitan la recepción de tráfico *spoofeado* desde internet, en la Figura 14 se puede apreciar la proporción de redes IPv4 /24 por país que se confirmaron vulnerables por The Closed Resolver Project, dado que no implementan filtros de tipo *inbound* SAV.

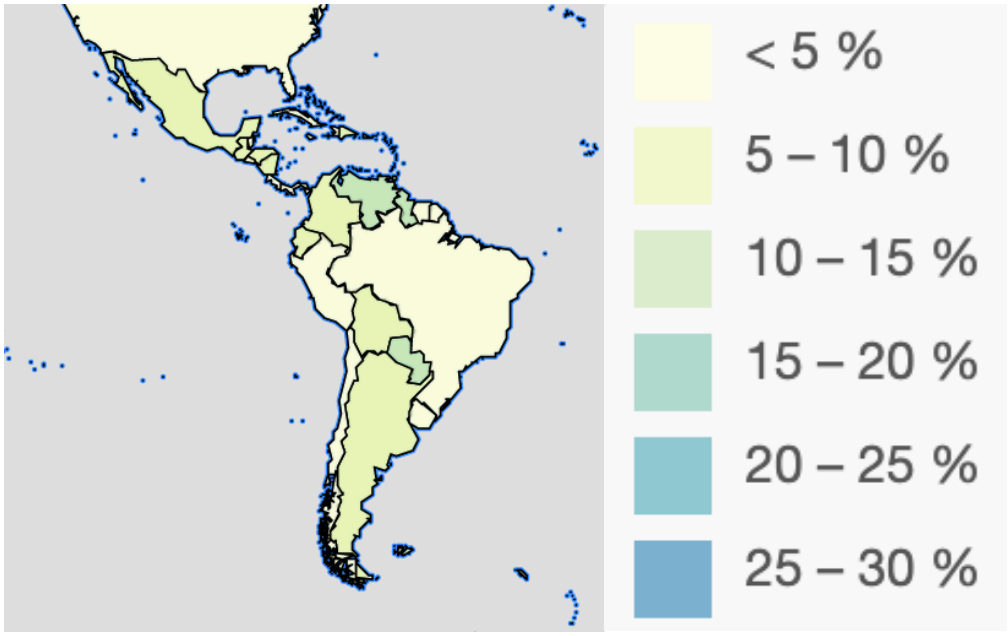


Fig. 14 - Proporción de redes IPv4 /24 por país vulnerables al *inbound* IP *spoofing*. ‹https://closedresolver.korlabs.io/#results›, 17 de octubre de 2022

De acuerdo a la información brindada por esta fuente, en todos los países de la región el porcentaje de redes vulnerables al inbound IP spoofing se encuentra por debajo del 20%.

En particular se puede observar que Brasil, Chile, Guyana Francesa, Perú, Uruguay y Surinam son los que presentan menor porcentaje de redes vulnerables al inbound IP spoofing; mientras que Guyana, Paraguay y Venezuela son los que presentan mayor porcentaje de redes vulnerables al inboud IP spoofing.

Mecanismos para mitigar el IP *spoofing*

La mitigación del IP *spoofing* es posible principalmente aplicando filtros de tipo *inbound* y *outbound* SAV. Como se mencionó, la falta de filtros de tipo SAV puede resultar en diferentes amenazas. Los filtros de tipo SAV son indicados inicialmente en la RFC 2827 para abordar la problemática de los ataques de DDoS basados en el uso de servicios reflexivos y amplificables.

Sin embargo, es importante entender que la implementación de este tipo de filtros también ayuda a proteger la red de otro tipo de ataques, como los referidos anteriormente.

Considerando cuestiones sobre el mantenimiento de los filtros de tipo SAV referidos en la RFC 2827 y a situaciones particulares de redes *multihomed*, surge como actualización la RFC 3704. En esta RFC se proponen diferentes modos de implementar los filtros de tipo SAV. Cada uno de ellos será el más adecuado de usar según las distintas situaciones. Tienen, además, diferentes grados de efectividad. Los métodos propuestos son:

- Filtros de tipo SAV basados en ACL.
- Distintas versiones de filtros de tipo SAV basadas en mecanismos uRPF (Unicast Reverse Path Forwarding). Los mecanismos uRPF verifican de manera dinámica si la dirección de origen de un paquete recibido existe en la tabla de enrutamiento. Si no está en la tabla de enrutamiento, el paquete se filtra.
    - *Strict*: verifica que el ruteo a la dirección de origen del paquete se realiza a través de la misma interfaz de red que la que recibió el paquete.
    - *Feasible*: verifica si hay alguna ruta a la dirección de origen del paquete (no necesariamente la mejor) que se realiza a través de la misma interfaz de red que la que recibió el paquete.
    - *Loose*: verifica que exista una ruta a la dirección de origen del paquete. La interfaz es irrelevante.

- *Loose* ignorando *default routes*: verifica que exista una ruta a la dirección de origen del paquete, pero sin considerar en la evaluación posibles *default routes* configuradas. La interfaz es irrelevante.

Recomendaciones generales

Las siguientes recomendaciones se realizan utilizando las que da la RFC 2827,[25] la RFC 3704,[26] la guía de implementación anti-*spoofing* de MANRS[27] y la documentación provista por el Anti-Spoofing Task Force de RIPE.[28]

Con el objetivo de evitar tanto el *inbound* como el *outbound* IP *spoofing*, se recomienda implementar políticas de filtrado *inbound* y *outbound* SAV, utilizando algunos de las siguientes técnicas:

- Filtros de tipo SAV basados en ACL

  Dependiendo de la cantidad de prefijos, puede resultar un problema el mantenimiento manual de las ACL que implementan tanto el *inbound* SAV como el *outbound* SAV.

- Filtros de tipo SAV basados en el mecanismo Strict uRPF

  Es una excelente alternativa para redes que tienen un ruteo simétrico en el que no es necesario realizar el mantenimiento manual de ACL.

- Filtros de tipo SAV basados en el mecanismo Feasible uRPF

  En la guía de implementación anti-*spoofing* de MANRS se indica:

[25] ‹https://www.rfc-editor.org/rfc/rfc2827›

[26] ‹https://www.rfc-editor.org/rfc/rfc3704›

[27] ‹https://www.manrs.org/netops/guide/antispoofing/›

[28] ‹https://www.ripe.net/publications/docs/ripe-431›

Los operadores de red dudan en usar uRPF en el núcleo de sus redes, debido al temor de perder accidentalmente tráfico válido que ha tomado una ruta inesperada a través de su red. El mecanismo Feasible uRPF debería resolver estos problemas.

Tips de configuración para Cisco

A continuación, se describe una implementación posible en Cisco de una ACL para evitar el tráfico *spoofeado*.

La Figura 15 representa el ejemplo dado en la guía de implementación anti-*spoofing* de MANRS.[29] En dicha guía se muestra la ACL que el ISP debe implementar en la interfaz marcada con un recuadro rojo para evitar que el cliente envíe tráfico IP *spoofeado* a internet.

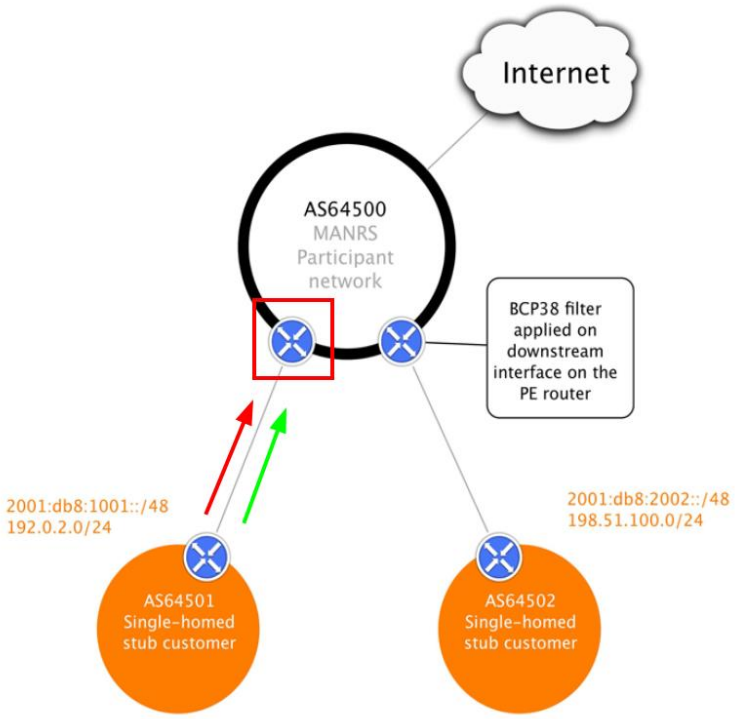


Fig. 15 - Topología de ejemplo para filtro de tipo inbound SAV basado en ACL

[29] ‹https://www.manrs.org/netops/guide/antispoofing/

Suponiendo que los bloques IP de que dispone el cliente son 192.0.2.0/24 y 2001:db8:1001::/48, la ACL que el ISP configuraría para evitar recibir tráfico IP *spoofeado* sería:

```
    ip access-list extended customer1-in-ipv4
 permit ip 192.0.2.0 0.0.0.255 any
!
ipv6 access-list customer1-in-ipv6
 permit ipv6 2001:db8:1001::/48 any
!
interface x
 ip access-group customer1-in-ipv4 in
     ipv6 traffic-filter customer1-in-ipv6 in
```

Para implementar filtros de tipo SAV basados en el mecanismo Strict uRPF, se pueden activar los filtros usando la siguiente configuración en las interfaces de red:

```
    ip cef
    ipv6 cef
    ip verify unicast source reachable-via rx
ipv6 verify unicast source reachable-via rx
```

Tips de configuración para Juniper

A continuación, se describe la implementación en Juniper de una ACL para evitar el tráfico *spoofeado*.

Utilizando el ejemplo dado en la guía de implementación anti-*spoofing* de MANRS[30], se muestra la ACL que el ISP puede implementar para evitar que el cliente envíe tráfico *spoofeado* a internet.

```
firewall {
family inet {
filter customer1-in-ipv4 {
term allowed-sources {
from {
source-address {
192.0.2.0/24;
}
}
then accept;
}
}
}
family inet6 {
filter customer1-in-ipv6 {
term allowed-sources {
```

[30] ‹https://www.manrs.org/netops/guide/antispoofing/›

```
from {
source-address {
2001:db8:1001::/48;
}
}
then accept;
}
}
}
}
interfaces x {
unit 0 {
family inet {
filter {
input customer1-in-ipv4;
}
}
family inet6 {
filter {
input customer1-in-ipv6;
}
}
}
}
```

Para implementar filtros de tipo SAV basados en el mecanismo Strict uRPF, se pueden activar en Juniper, usando la siguiente configuración en las interfaces de red:

```
family inet {
 rpf-check;
}
family inet6 {
 rpf-check;
}
```

Para implementar filtros de tipo SAV basados en el mecanismo uRPF modo Feasible, en Juniper, es necesario activarlo de manera global:[31]

```
[edit routing-options forwarding-table]
 unicast-reverse-path feasible-paths;
```

Y también activar en la configuración de las interfaces de red:

```
family inet {
 rpf-check; {
 mode loose;
 }
}
family inet6 {
 rpf-check; {
```

[31] ‹https://www.juniper.net/documentation/us/en/software/junos/interfaces-ethernet-switches/static-routing/topics/ref/statement/unicast-reverse-path-edit-routing-options.html›

```
    mode loose;
    }
  }
```

Tips de configuración para MikroTik

En MikroTik, la manera de configurar una ACL para evitar que salga a internet tráfico *spoofeado* es definiendo una lista de direcciones IPs permitidas para la organización. Esta lista debe considerar todas las IPs y/o bloques de direcciones que podrían ser el origen de los paquetes que salen a internet.

Crear una regla de *firewall*[32] que acepte el tráfico que cumpla las siguientes condiciones:

- *Chain: forward*
- *Src. Address List*: La lista de direcciones permitidas[33]
- *Out. Interface*: Interfaz de WAN que se conecta a internet

Crear una regla de firewall que deniegue lo que cumpla las siguientes condiciones:

- *Chain: forward*
- Out. Interface: Interfaz de WAN que se conecta a internet.

Para implementar filtros de tipo SAV basados en el mecanismo Strict uRPF, se pueden activar en MikroTik desde el menú IP → Settings → RP Filter = strict [34].

Nota: El mecanismo Feasible URPF no está soportado en la última versión estable al momento de realizar este informe: RouterOS 7.5.

[32] ‹https://help.mikrotik.com/docs/display/ROS/Filter›

[33] ‹https://help.mikrotik.com/docs/display/ROS/Address-lists

[34] ‹https://help.mikrotik.com/docs/display/ROS/IP+Settings›

Recomendaciones

En suma, para evitar la suplantación de direcciones IP se recomienda:

- Evaluar el estado de la aplicación de filtros de tipo *inbound* y *outbound* SAV en los recursos asignados con la herramienta *spoofer* del proyecto Spoofer de CAIDA.
- Implementar filtros de tipo *inbound* y *outbound* SAV.
- A partir del conocimiento dado por este estudio, fomentar los testeos y la aplicación de las buenas prácticas recomendadas para mitigar los problemas asociados al IP *spoofing*.
- Utilizar estadísticas provistas por el proyecto Spoofer de CAIDA para darle mayor visibilidad a la problemática en la comunidad y para generar conciencia.
- Analizar alternativas para incentivar la realización de testeos, de forma de motivar a los asociados a sumarse a la iniciativa.

Conclusiones

De acuerdo a los datos relevados en el presente estudio, si bien la mayoría de las redes testeadas tienen una correcta implementación de las buenas prácticas asociadas a la mitigación del IP spoofing en la región, aún resta un porcentaje del universo analizado sin actuar sobre dicha problemática.

En este informe se ha unificado terminología y se ha organizado información de diferentes fuentes. Por ello resulta una fuente autónoma y relevante para que los asociados puedan comprender la problemática del spoofing de direcciones IP, los mecanismos con los que cuentan para identificar en qué estado se encuentran sus redes con respecto a ella y qué acciones pueden realizar para mejorar la seguridad de las mismas.

Se espera que a partir de este estudio se incrementen de forma significativa las redes que aplican técnicas anti-spoofing y, de esta manera, colaborar con la seguridad, estabilidad y resiliencia de Internet.