IETF
Internet-Draft
Intended status: Informational
Expires: 21 January 2026

L. Chen
Zhongguancun Laboratory
D. Li
Tsinghua University
L. Liu
L. Qin
Zhongguancun Laboratory

20 July 2025

Benchmarking Methodology for Source Address Validation
draft-chen-bmwg-savnet-sav-benchmarking-05

Abstract

This document defines methodologies for benchmarking the performance of intra-domain and inter-domain source address validation (SAV) mechanisms. SAV mechanisms are utilized to generate SAV rules to prevent source address spoofing, and have been implemented with many various designs in order to perform SAV in the corresponding scenarios. This document takes the approach of considering a SAV device to be a black box, defining the methodology in a manner that is agnostic to the mechanisms. This document provides a method for measuring the performance of existing and new SAV implementations.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on 21 January 2026.

Copyright Notice

Table of Contents

1. Introduction

Source address validation (SAV) is significantly important to prevent source address spoofing. Operators are suggested to deploy different SAV mechanisms [RFC3704] [RFC8704] based on their deployment network environments. In addition, existing intra-domain and inter-domain SAV mechanisms have problems in operational overhead and accuracy under various scenarios [intra-domain-ps] [inter-domain-ps]. Intra-domain and inter-domain SAVNET architectures [intra-domain-arch] [inter-domain-arch] are proposed to guide the design of new intra-domain and inter-domain SAV mechanisms to solve the problems. The benchmarking methodology defined in this document will help operators to get a more accurate idea of the SAV performance when their deployed devices enable SAV and will also help vendors to test the performance of SAV implementation for their devices.

This document provides generic methodologies for benchamarking SAV mechanism performance. To achieve the desired functionality, a SAV device may support many SAV mechanisms. This document considers a SAV device to be a black box, regardless of the design and implementation. The tests defined in this document can be used to benchmark a SAV device for SAV accuracy, convergence performance, and control plane and data plane forwarding performance. These tests can be performed on a hardware router, a software router, a virtual machine (VM) instance, or a container instance, which runs as a SAV device. This document outlines methodologies for assessing SAV device performance and comparing various SAV mechanisms.

1.1. Goal and Scope

The benchmarking methodology outlined in this draft focuses on two objectives:

* Assessing ''which SAV mechnisms performn best'' over a set of well-defined scenarios.

* Measuring the contribution of sub-systems to the overall SAV systems's performance (also known as ''micro-benchmark'').

This benchmark evaluates the SAV performance of individual devices (e.g., hardware/software routers) by comparing different SAV mechanisms under specific network scenarios. The results help determine the appropriate SAV deployment for real-world network scenarios.

1.2. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

2. Terminology

SAV Control Plane: The SAV control plane consists of processes including gathering and communicating SAV-related information.

SAV Data Plane: The SAV data plane stores the SAV rules within a specific data structure and validates each incoming packet to determine whether to permit or discard it.

Host-facing Router: An intra-domain router facing an intra-domain host network.

Customer-facing Router: An intra-domain router facing an intra-domain customer network which includes routers and runs the routing protocol.

AS Border Router: An intra-domain router facing an external AS.

3. Test Methodology

3.1. Test Setup

The test setup in general is compliant with [RFC2544]. The Device Under Test (DUT) is connected to a Tester and other network devices to construct the network topology introduced in Section 5. The Tester is a traffic generator to generate network traffic with various source and destination addresses in order to emulate the spoofing or legitimate traffic. It is OPTIONAL to choose various proportions of traffic and it is needed to generate the traffic with line speed to test the data plane forwarding performance.

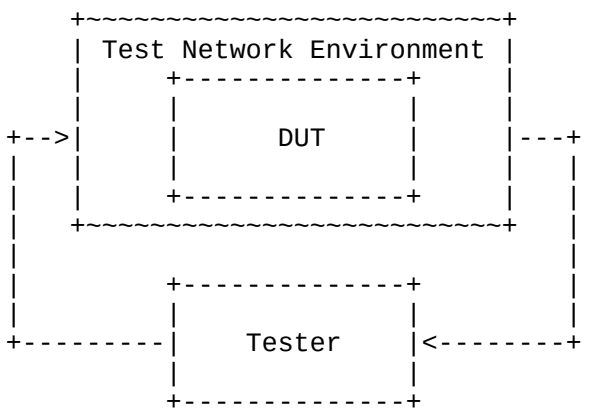


Figure 1: Test Setup.

Figure 1 illustrates the test configuration for the Device Under Test (DUT). Within the test network environment, the DUT can be interconnected with other devices to create a variety of test scenarios. The Tester may establish a direct connection with the DUT or link through intermediary devices. The nature of the connection between them is dictated by the benchmarking tests outlined in Section 5. Furthermore, the Tester has the capability to produce both spoofed and legitimate traffic to evaluate the SAV accuracy of the DUT in relevant scenarios, and it can also generate traffic at line rate to assess the data plane forwarding performance of the DUT. Additionally, the DUT is required to support logging functionalities to document all test outcomes.

3.2. Network Topology and Device Configuration

The placement of the DUT within the network topology significantly influences the SAV performance. Consequently, the benchmarking process MUST involve positioning the DUT at various locations throughout the network to thoroughly evaluate its performance.

The routing configurations of devices within the network topology can vary, and the SAV rules generated are contingent upon these configurations. It is imperative to delineate the specific device configurations employed during testing.

Moreover, it is essential to denote the role of each device, such as a host-facing router, customer-facing router, or AS border router within an intra-domain network, and to clarify the business relationships between ASes in an inter-domain network context.

When assessing the data plane forwarding performance, the network traffic produced by the Tester must be characterized by specified traffic rates, the ratio of spoofing to legitimate traffic, and the distribution of source addresses, as these factors can all impact the outcomes of the tests.

4. SAV Performance Indicators

This section lists key performance indicators (KPIs) of SAV for overall benchmarking tests. All KPIs MUST be measured in the bencharking scenarios described in Section 5. Also, the KPIs MUST be measured from the result output of the DUT.

4.1. False Positive Rate

The proportion of legitimate traffic which is determined to be spoofing traffic by the DUT across all the legitimate traffic, and this can reflect the SAV accuracy of the DUT.

4.2. False Negative Rate

The proportion of spoofing traffic which is determined to be legitimate traffic by the DUT across all the spoofing traffic, and this can reflect the SAV accuracy of the DUT.

4.3. Protocol Convergence Time

The control protocol convergence time represents the period during which the SAV control plane protocol converges to update the SAV rules when routing changes happen, and it is the time elapsed from the begining of routing change to the completion of SAV rule update. This KPI can indicate the convergence performance of the SAV protocol.

4.4. Protocol Message Processing Throughput

The protocol message processing throughput measures the throughput of processing the packets for communicating SAV-related information on the control plane, and it can indicate the SAV control plane performance of the DUT.

4.5. Data Plane SAV Table Refreshing Rate

The data plane SAV table refreshing rate refers to the rate at which a DUT updates its SAV table with new SAV rules, and it can reflect the SAV data plane performance of the DUT.

4.6. Data Plane Forwarding Rate

The data plane forwarding rate measures the SAV data plane forwarding throughput for processing the data plane traffic, and it can indicate the SAV data plane performance of the DUT.

4.7. Resource Utilization

The resource utilization measures the CPU and memory utilizations of the SAV process for intra-domain SAV and inter-domain SAV within the DUT.

5. Benchmarking Tests

5.1. Intra-domain SAV

5.1.1. False Positive and False Negative Rates

Objective: Evaluate the DUT’s false positive rate and false negative rate in handling legitimate and spoofing traffic across diverse intra-domain network scenarios, encompassing SAV implementations for customer or host networks, Internet-facing networks, and aggregation-router-facing networks.

In the following, this document introduces the classic scenarios of testing DUT for intra-domain SAV.

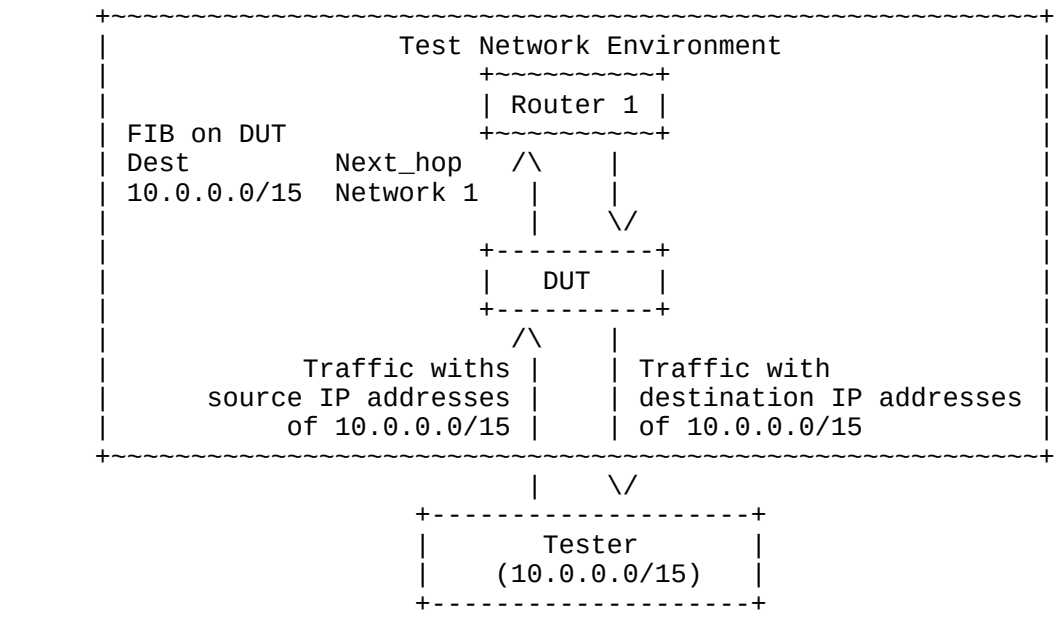


Figure 2: SAV for customer or host network in intra-domain symmetric routing scenario.

SAV for Customer or Host Network: Figure 2 shows the case of SAV for customer or host network in intra-domain symmetric routing scenario, and the DUT performs SAV as a customer/host-facing router and connects to Router 1 to access the Internet. Network 1 is a customer/host network within the AS, connects to the DUT, and its own prefix is 10.0.0.0/15. The Tester can emulate Network 1 to advertise its prefix in the control plane and generate spoofing and legitimate traffic in the data plane. In this case, the Tester configs to make the inbound traffic destined for 10.0.0.0/15 come from the DUT. The DUT learns the route to prefix 10.0.0.0/15 from the Tester, while the Tester can send outbound traffic with source addresses in prefix 10.0.0.0/15 to the DUT, which emulates the a symmetric routing scenario between the Tester and the DUT. The IP addrsses in this test case is optional and users can use other IP addresses, and this holds true for other test cases as well.

The *procedure* is listed below for testing SAV for customer or host network in intra-domain symmetric routing scenario:

1. First, in order to test whether the DUT can generate accurate SAV rules for SAV for customer or host network in intra-domain symmetric routing scenario, a testbed can be built as shown in Figure 2 to construct the test network environment. The Tester is connected to the DUT and performs the functions as Network 1.

2. Then, the devices including the DUT and Router 1 are configured to form the symmetric routing scenario.

3. Finally, the Tester generates traffic using 10.0.0.0/15 as source addresses (legitimate traffic) and traffic using 10.2.0.0/15 as source addresses (spoofing traffic) to the DUT, respectively. The ratio of spoofing traffic to legitimate traffic can vary, such as from 1:9 to 9:1.

The *expected results* are that the DUT can block the spoofing traffic and permit the legitimate traffic from Network 1 for this test case.

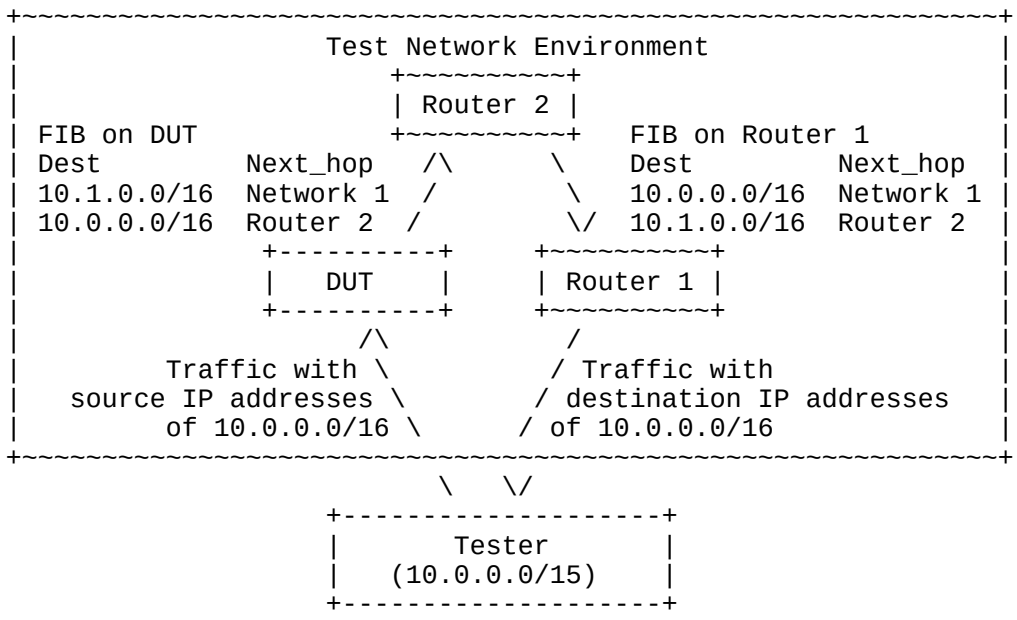


Figure 3: SAV for customer or host network in intra-domain asymmetric routing scenario.

Figure 3 shows the case of SAV for customer or host network in intra-domain asymmetric routing scenario, and the DUT performs SAV as a customer/host-facing router. Network 1 is a customer/host network within the AS, connects to the DUT and Router 1, respectively, and its own prefix is 10.0.0./15. The Tester can emulate Network 1 and performs its control plane and data plane functions. In this case, the Tester configs to make the inbound traffic destined for 10.1.0.0/16 come only from the DUT and the inbound traffic destined for 10.0.0.0/16 to come only from Router 1. The DUT only learns the route to prefix 10.1.0.0/16 from the Tester, while Router 1 only learns the route to the prefix 10.0.0.0/16 from Network 1. Then, the DUT and Router 1 avertise their learned prefixes to Router 2. Besides, the DUT learns the route to 10.0.0.0/16 from Router 2, and Router 1 learns the route to 10.1.0.0/16 from Router 2. The Tester can send outbound traffic with source addresses of prefix 10.0.0.0/16 to the DUT, which emulates the an asymmetric routing scenario between the Tester and the DUT.

The *procedure* is listed below for testing SAV for customer or host network in intra-domain asymmetric routing scenario:

1. First, in order to test whether the DUT can generate accurate SAV rules for SAV for customer or host network in intra-domain asymmetric routing scenario, a testbed can be built as shown in Figure 3 to construct the test network environment. The Tester is connected to the DUT and Router 1 and performs the functions as Network 1.

2. Then, the devices including the DUT, Router 1, and Router 2, are configured to form the asymmetric routing scenario.

3. Finally, the Tester generates traffic using 10.1.0.0/16 as source addresses (spoofing traffic) and traffic using 10.0.0.0/16 as source addresses (legitimate traffic) to the DUT, respectively. The ratio of spoofing traffic to legitimate traffic can vary, such as from 1:9 to 9:1.

The expected results are that the DUT can block the spoofing traffic and permit the legitimate traffic from Network 1 for this test case.

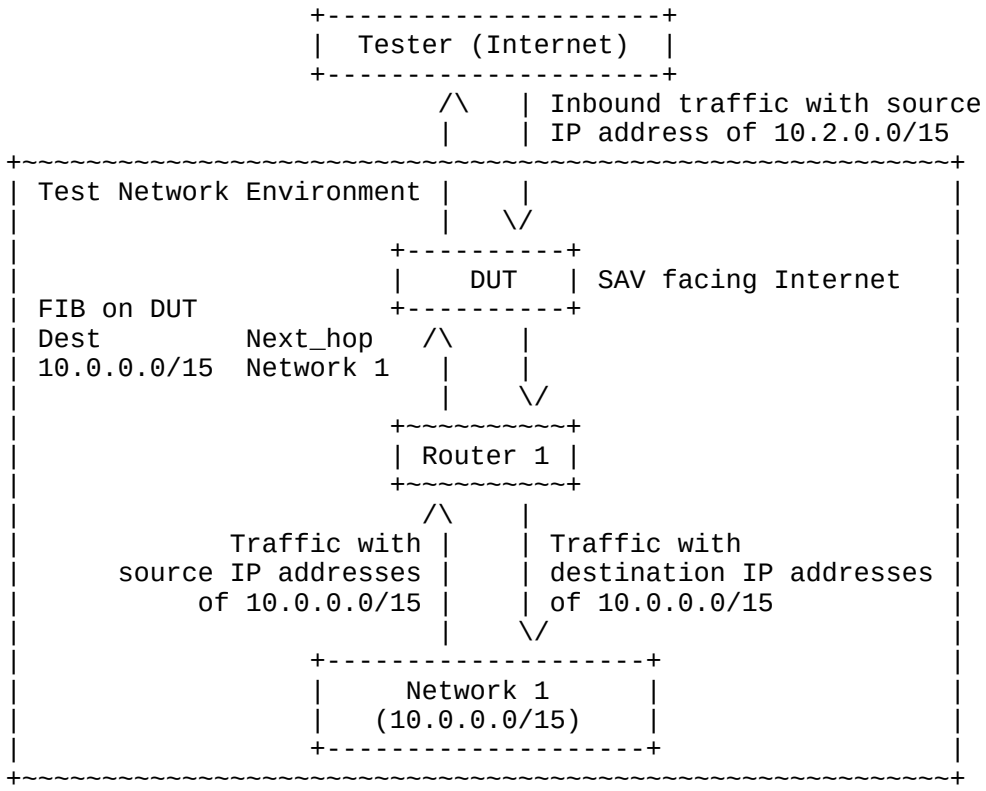


Figure 4: SAV for Internet-facing network in intra-domain symmetric routing scenario.

SAV for Internet-facing Network: Figure 4 illustrates the test scenario for SAV in an Internet-facing network within an intra-domain symmetric routing context. In this scenario, the network topology mirrors that of Figure 2, with the key distinction being the DUT's placement within the network. Here, the DUT is linked to Router 1 and the Internet, with the Tester simulating the Internet's role. The DUT executes Internet-facing SAV, as opposed to customer/host-network-facing SAV.

The *procedure* is listed below for testing SAV for Internet-facing network in intra-domain symmetric routing scenario**:

1. First, in order to test whether the DUT can generate accurate SAV rules for SAV for Internet-facing network in intra-domain symmetric routing scenario, a testbed can be built as shown in Figure 4 to construct the test network environment. The Tester is connected to the DUT and performs the functions as the Internet.

2. Then, the devices including the DUT and Router 1 are configured to form the symmetric routing scenario.

3. Finally, the Tester can send traffic using 10.0.0.0/15 as source addresses (spoofing traffic) and traffic using 10.2.0.0/15 as source addresses (legitimate traffic) to the DUT, respectively. The ratio of spoofing traffic to legitimate traffic can vary, such as from 1:9 to 9:1.

The *expected results* are that the DUT can block the spoofing traffic and permit the legitimate traffic from the Internet for this test case.

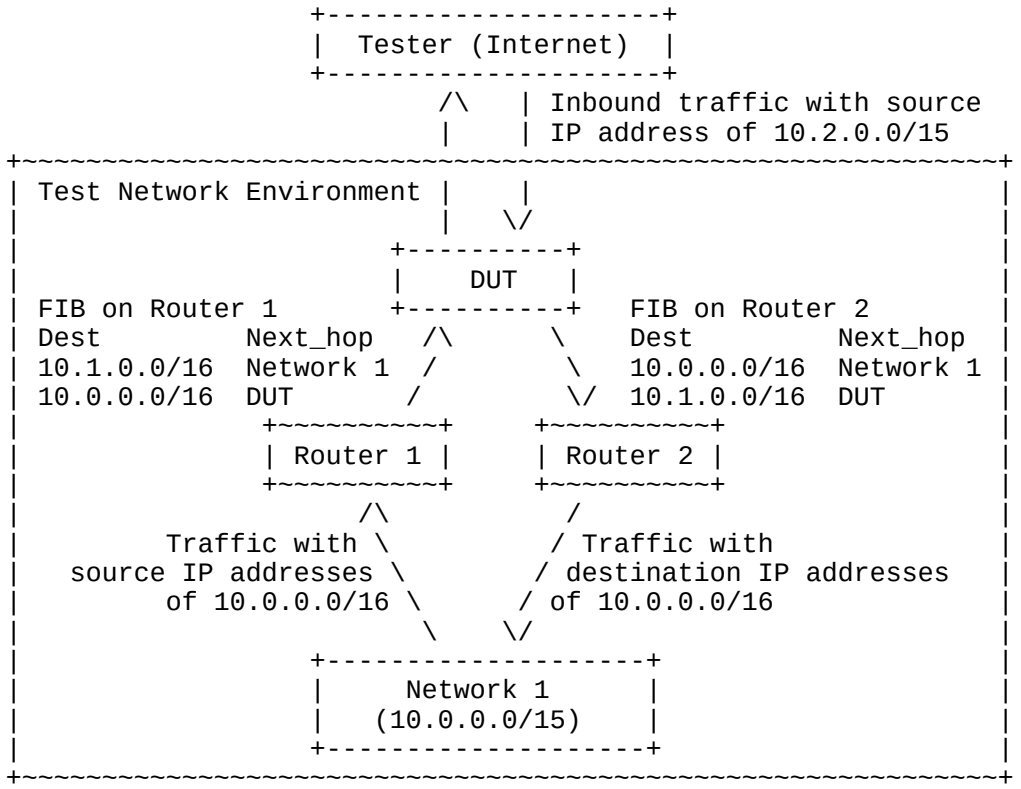


Figure 5: SAV for Internet-facing network in intra-domain asymmetric routing scenario.

Figure 5 shows the test case of SAV for Internet-facing network in intra-domain asymmetric routing scenario. In this test case, the network topology is the same with Figure 3, and the difference is the location of the DUT in the network topology, where the DUT is connected to Router 1 and Router 2 within the same AS, as well as the Internet. The Tester is used to emulate the Internet. The DUT performs Internet-facing SAV instead of customer/host-network-facing SAV.

The *procedure* is listed below for testing SAV for Internet-facing network in intra-domain asymmetric routing scenario**:

1. First, in order to test whether the DUT can generate accurate SAV rules for SAV for Internet-facing network in intra-domain asymmetric routing scenario, a testbed can be built as shown in Figure 5 to construct the test network environment. The Tester is connected to the DUT and performs the functions as the Internet.

2. Then, the devices including the DUT, Router 1, and Router 2 are configured to form the asymmetric routing scenario.

3. Finally, the Tester can send traffic using 10.0.0.0/15 as source addresses (spoofing traffic) and traffic using 10.2.0.0/15 as source addresses (legitimate traffic) to the DUT, respectively. The ratio of spoofing traffic to legitimate traffic can vary, such as from 1:9 to 9:1.

The *expected results* are that the DUT can block the spoofing traffic and permit the legitimate traffic from the Internet for this test case.

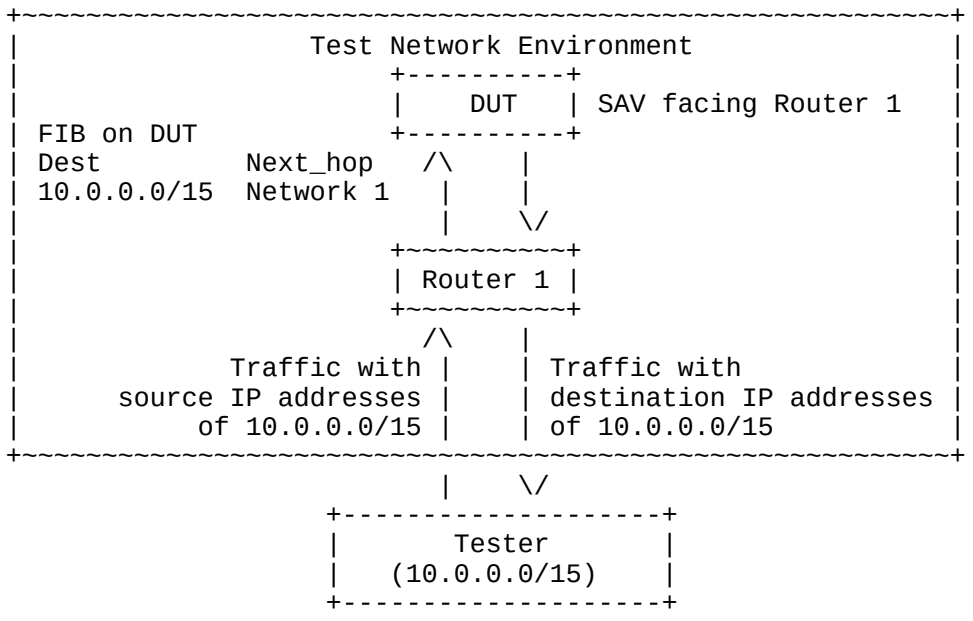


Figure 6: SAV for aggregation-router-facing network in intra-domain symmetric routing scenario.

SAV for Aggregation-router-facing Network: Figure 6 depicts the test scenario for SAV in an aggregation-router-facing network within an intra-domain symmetric routing environment. The test network

setup in Figure 6 is identical to that of Figure 4. The Tester is linked to Router 1 to simulate the operations of Network 1, thereby evaluating the false positive rate and false negative rate of the DUT as it faces the direction of Router 1.

The *procedure* is listed below for testing SAV for aggregation-router-facing network in intra-domain symmetric routing scenario:

1. First, in order to test whether the DUT can generate accurate SAV rules for SAV for Internet-facing network in intra-domain symmetric routing scenario, a testbed can be built as shown in Figure 6 to construct the test network environment. The Tester is connected to Router 1 and performs the functions as Network 1.

2. Then, the devices including the DUT and Router 1 are configured to form the symmetric routing scenario.

3. Finally, the Tester can send traffic using 10.1.0.0/15 as source addresses (legitimate traffic) and traffic using 10.2.0.0/15 as source addresses (spoofing traffic) to Router 1, respectively. The ratio of spoofing traffic to legitimate traffic can vary, such as from 1:9 to 9:1.

The expected results are that the DUT can block the spoofing traffic and permit the legitimate traffic from the direction of Router 1 for this test case.

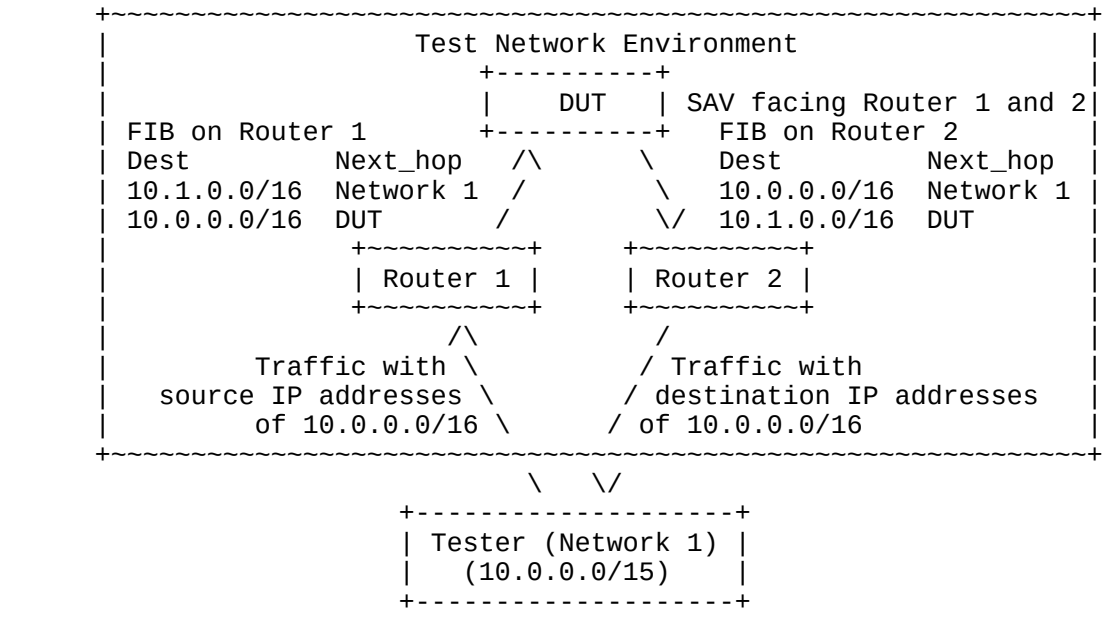


Figure 7: SAV for aggregation-router-facing network in intra-domain asymmetric routing scenario.

Figure 7 shows the test case of SAV for aggregation-router-facing network in intra-domain asymmetric routing scenario. The test network environment of Figure 7 is the same with Figure 5. The Tester is connected to Router 1 and Router 2 to emulate the functions of Network 1 to test the false positive rate and false negative rate of the DUT facing the direction of Router 1 and Router 2.

The *procedure* is listed below for testing SAV for aggregation-router-facing network in intra-domain asymmetric routing scenario:

1. First, in order to test whether the DUT can generate accurate SAV rules for SAV for aggregation-router-facing network in intra-domain asymmetric routing scenario, a testbed can be built as shown in Figure 7 to construct the test network environment. The Tester is connected to Router 1 and Router 2 and performs the functions as Network 1.

2. Then, the devices including the DUT, Router 1, and Router 2 are configured to form the asymmetric routing scenario.

3. Finally, the Tester generates traffic using 10.1.0.0/16 as source addresses (spoofing traffic) and traffic using 10.0.0.0/16 as source addresses (legitimate traffic) to Router 1, respectively. The ratio of spoofing traffic to legitimate traffic can vary, such as from 1:9 to 9:1.

The expected results are that the DUT can block the spoofing traffic and permit the legitimate traffic from the direction of Router 1 and Router 2 for this test case.

5.1.2. Control Plane Performance

Objective: Measure the control plane performance of the DUT, encompassing both protocol convergence performance and protocol message processing performance in response to route changes triggered by network failures or operator configurations. The protocol convergence performance is quantified by the protocol convergence time, which is the duration from the initiation of a routing change to the completion of the SAV rule update. The protocol message processing performance is characterized by the protocol message processing throughput, defined as the total size of protocol messages processed per second.

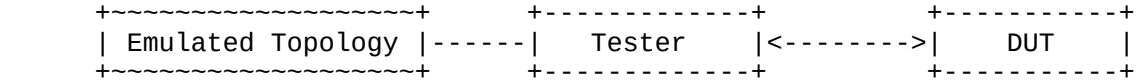


Figure 8: Test setup for protocol convergence performance measurement.

Protocol Convergence Performance: Figure 8 illustrates the test setup for measuring protocol convergence performance. The protocol convergence process of the DUT, which updates SAV rules, is initiated when route changes occur. These route changes, which necessitate the updating of SAV rules, can result from network failures or operator configurations. Consequently, in Figure 8, the Tester is directly connected to the DUT and simulates route changes to trigger the DUT's convergence process by adding or withdrawing prefixes.

The *procedure* is listed below for testing the protocol convergence performance:

1. First, in order to test the protocol convergence time of the DUT, a testbed can be built as shown in Figure 8 to construct the test network environment. The Tester is directly connected to the DUT.

2. Then, the Tester proactively withdraws the prefixes in a certern percentage of the overall prefixes supported by the DUT, such as 10%, 20%, ..., 100%.

3. Finally, the protocol convergence time is calculated according to the logs of the DUT about the beginning and completion of the protocol convergence.

Please note that withdrawing prefixes proportionally for IGP can be accomplished by proportionally shutting down interfaces. For instance, the Tester is connected to an emulated network topology where each interface links to an emulated device. Suppose the Tester connects to ten emulated devices through ten interfaces. Initially, these ten emulated devices advertise their prefixes to the DUT. To withdraw 10% of the prefixes, the Tester can randomly disable one interface connected to an emulated device. Similarly, to withdraw 20%, it can shut down two interfaces randomly, and this method applies to other proportions accordingly. This is merely a suggested approach, and alternative methods achieving the same objective are also acceptable.

The protocol convergence time, which is the duration required for the DUT to complete the protocol convergence process, should be measured from the moment the last hello message is received on the DUT from

the emulated device connected by the disabled interface until the SAV rule generation on the DUT is finalized. To accurately measure the protocol convergence time, the DUT's logs should record the timestamp of receiving the last hello message and the timestamp when the SAV rule update is completed. The protocol convergence time is then determined by calculating the difference between these two timestamps.

It is important to note that if the emulated device sends a "goodbye hello" message during the process of shutting down the Tester's interface, using the reception time of this goodbye hello message instead of the last hello message would yield a more precise measurement, as recommended by [RFC4061].

Protocol Message Processing Performance: The test of the protocol message processing performance uses the same test setup shown in Figure 8. The protocol message processing performance measures the protocol message processing throughput to process the protocol messages. Therefore, the Tester can vary the rate for sending protocol messages, such as from 10% to 100% of the overall link capacity between the Tester and the DUT. Then, the DUT records the size of the processed total protocol messages and processing time.

The *procedure* is listed below for testing the protocol message processing performance:

1. First, in order to test the protocol message processing throughput of the DUT, a testbed can be built as shown in Figure 8 to construct the test network environment. The Tester is directly connected to the DUT.

2. Then, the Tester proactively sends the protocol messages to the DUT in a certern percentage of the overall link capacity between the Tester and the DUT, such as 10%, 20%, ..., 100%.

3. Finally, the protocol message processing throughput is calculated according to the logs of the DUT about the overall size of the protocol messages and the overall processing time.

To measure the protocol message processing throughput, the logs of the DUT records the overall size of the protocol messages and the overall processing time, and the protocol message processing throughput is calculated by dividing the overall size of the protocol messages by the overall processing time.

5.1.3. Data Plane Performance

Objective: Evaluate the data plane performance of the DUT, encompassing both the data plane SAV table refreshing performance and the data plane forwarding performance. The data plane SAV table refreshing performance is quantified by the data plane SAV table refreshing rate, which indicates the speed at which the DUT updates its SAV table with newly implemented SAV rules. Concurrently, the data plane forwarding performance is measured by the data plane forwarding rate, which represents the total size of packets forwarded by the DUT per second.

Data Plane SAV Table Refreshing Performance: The assessment of the data plane SAV table refreshing performance utilizes the identical test configuration depicted in Figure 8. This performance metric gauges the velocity at which a DUT refreshes its SAV table with new SAV rules. To this end, the Tester can modulate the transmission rate of protocol messages, ranging from 10% to 100% of the total link capacity between the Tester and the DUT. This variation influences the proportion of updated SAV rules and, consequently, the proportion of entries in the SAV table. Subsequently, the DUT logs the total count of updated SAV table entries and the duration of the refreshing process.

The *procedure* is listed below for testing the data plane SAV table refreshing performance:

1. First, in order to test the data plane SAV table refreshing rate of the DUT, a testbed can be built as shown in Figure 8 to construct the test network environment. The Tester is directly connected to the DUT.

2. Then, the Tester proactively sends the protocol messages to the DUT in a certern percentage of the overall link capacity between the Tester and the DUT, such as 10%, 20%, ..., 100%.

3. Finally, the data plane SAV table refreshing rate is calculated according to the logs of the DUT about the overall number of updated SAV table entries and the overall refreshing time.

To measure the data plane SAV table refreshing rate, the logs of the DUT records the overall number of updated SAV table entries and the overall refreshing time, and the data plane SAV table refreshing rate is calculated by dividing the overall number of updated SAV table entries by the overall refreshing time.

Data Plane Forwarding Performance: The evaluation of the data plane forwarding performance employs the same test setup illustrated in Figure 8. The Tester is required to transmit a blend of spoofing and legitimate traffic at a rate equivalent to the total link capacity between the Tester and the DUT, while the DUT constructs a SAV table that utilizes the entire allocated storage space. The proportion of spoofing traffic to legitimate traffic can be adjusted across a range, for example, from 1:9 to 9:1. The DUT then records the aggregate size of the packets forwarded and the total duration of the forwarding activity.

The *procedure* is listed below for testing the data plane forwarding performance:

1. First, in order to test the data plane forwarding rate of the DUT, a testbed can be built as shown in Figure 8 to construct the test network environment. The Tester is directly connected to the DUT.

2. Then, the Tester proactively sends the data plane traffic including spoofing and legitimate traffic to the DUT at the rate of the overall link capacity between the Tester and the DUT. The ratio of spoofing traffic to legitimate traffic can vary, such as from 1:9 to 9:1.

3. Finally, the data plane forwarding rate is calculated according to the logs of the DUT about the overall size of the forwarded traffic and the overall forwarding time.

To measure the data plane forwarding rate, the logs of the DUT records the overall size of the forwarded traffic and the overall forwarding time, and the data plane forwarding rate is calculated by dividing the overall size of the forwarded traffic by the overall forwarding time.

5.2. Inter-domain SAV

5.2.1. False Positive and False Negative Rates

Objective: Measure the false positive rate and false negative rate of the DUT to process legitimate traffic and spoofing traffic across various inter-domain network scenarios including SAV for customer-facing ASes and SAV for provider/peer-facing ASes.

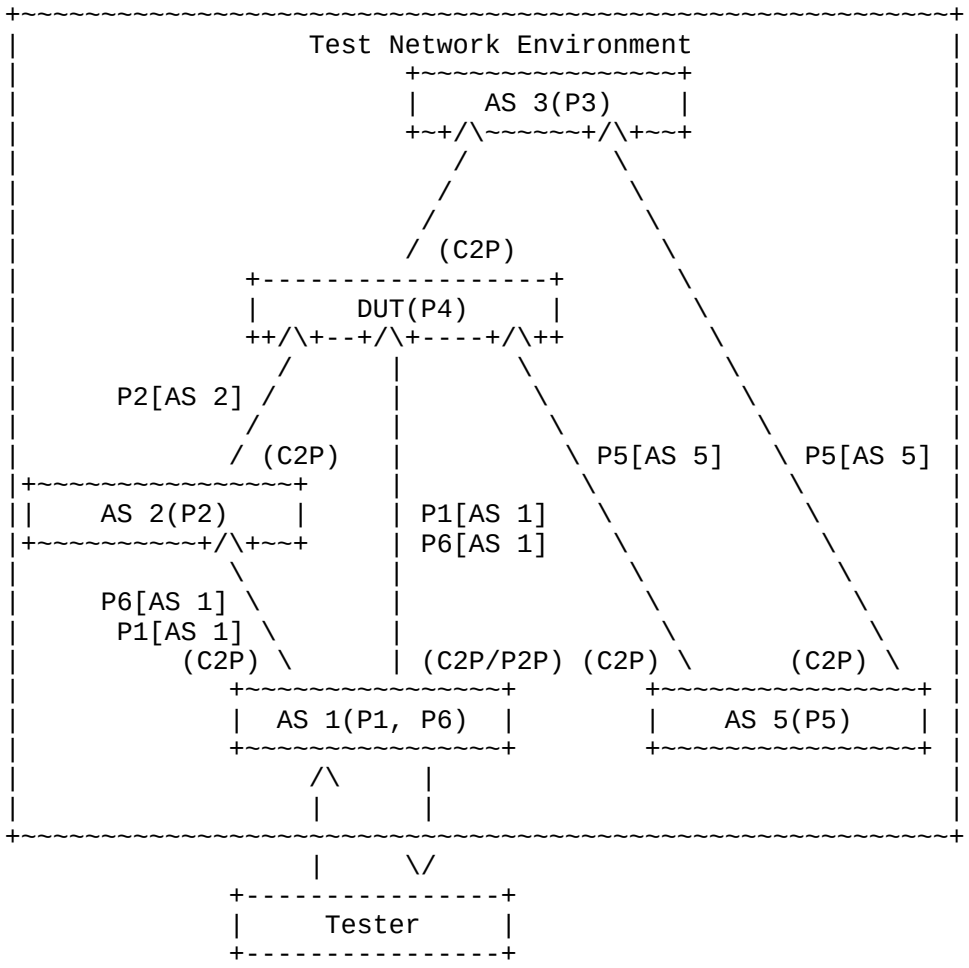


Figure 9: SAV for customer-facing ASes in inter-domain symmetric routing scenario.

SAV for Customer-facing ASes: Figure 9 presents a test case of SAV for customer-facing ASes in inter-domain symmetric routing scenario. In this test case, AS 1, AS 2, AS 3, the DUT, and AS 5 constructs the test network environment, and the DUT performs SAV as an AS. AS 1 is a customer of AS 2 and the DUT, AS 2 is a customer of the DUT, which is a customer of AS 3, and AS 5 is a customer of both AS 3 and the DUT. AS 1 advertises prefixes P1 and P6 to AS 2 and the DUT, respectively, and then AS 2 further propagates the route for prefix P1 and P6 to the DUT. Consequently, the DUT can learn the route for prefixes P1 and P6 from AS 1 and AS 2. In this test case, the legitimate path for the traffic with source addresses in P1 and

destination addresses in P4 is AS 1->AS 2->AS 4, and the Tester is connected to the AS 1 and the SAV for customer-facing ASes of the DUT is tested.

The *procedure* is listed below for testing SAV for customer-facing ASes in inter-domain symmetric routing scenario:

1. First, in order to test whether the DUT can generate accurate SAV rules for SAV for customer-facing ASes in inter-domain symmetric routing scenario, a testbed can be built as shown in Figure 9 to construct the test network environment. The Tester is connected to AS 1 and generates the test traffic to the DUT.

2. Then, the ASes including AS 1, AS 2, AS 3, the DUT, and AS 5, are configured to form the symmetric routing scenario.

3. Finally, the Tester sends the traffic using P1 as source addresses and P4 as destination addresses (legitimate traffic) to the DUT via AS 2 and traffic using P5 as source addresses and P4 as destination addresses (spoofing traffic) to the DUT via AS 2, respectively. The ratio of spoofing traffic to legitimate traffic can vary, such as from 1:9 to 9:1.

The *expected results* are that the DUT can block the spoofing traffic and permit the legitimate traffic from the direction of AS 2 for this test case.

Note that the locations of the DUT in Figure 9 can be set at AS 1 and AS 2 to evaluate its false positive rate and false negative rate according to the procedure outlined above. The expected results are that the DUT will effectively block spoofing traffic.

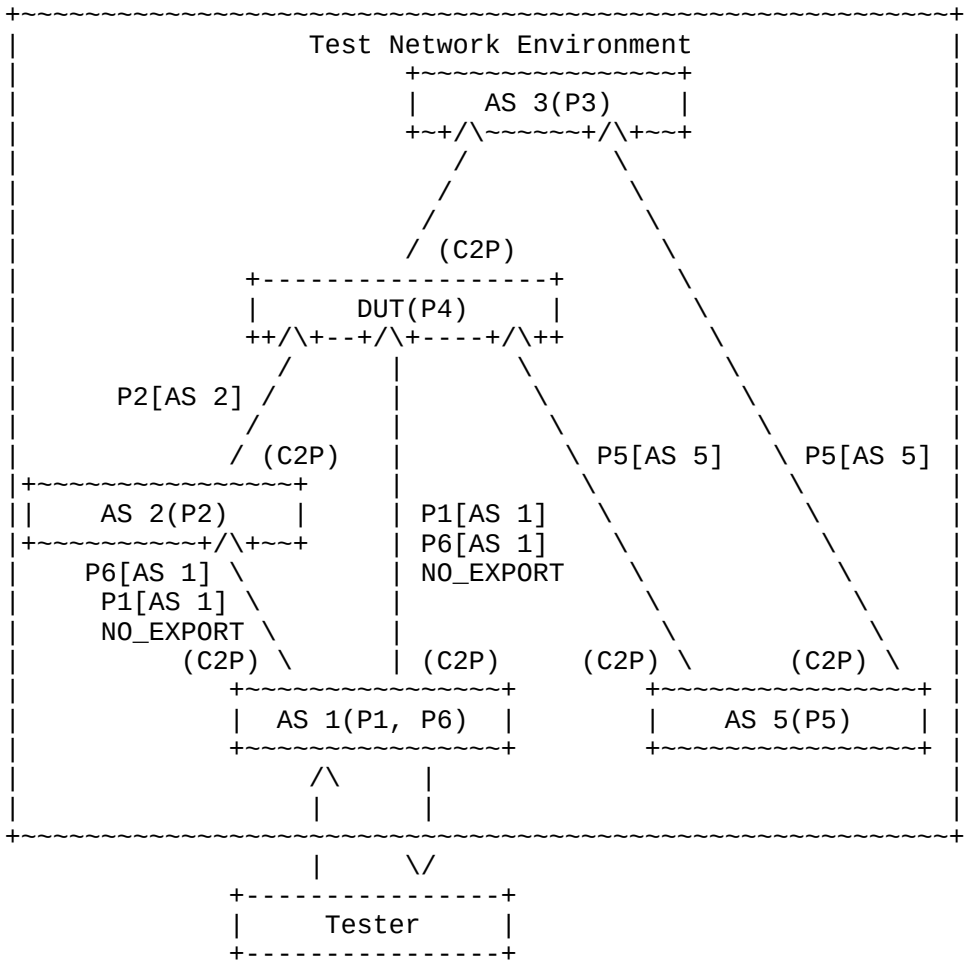


Figure 10: SAV for customer-facing ASes in inter-domain asymmetric routing scenario caused by NO_EXPORT.

Figure 10 presents a test case of SAV for customer-facing ASes in inter-domain asymmetric routing scenario caused by NO_EXPORT configuration. In this test case, AS 1, AS 2, AS 3, the DUT, and AS 5 constructs the test network environment, and the DUT performs SAV as an AS. AS 1 is a customer of AS 2 and the DUT, AS 2 is a customer of the DUT, which is a customer of AS 3, and AS 5 is a customer of both AS 3 and the DUT. AS 1 advertises prefixes P1 to AS 2 and adds the NO_EXPORT community attribute to the BGP advertisement sent to AS 2, preventing AS 2 from further propagating the route for prefix P1 to the DUT. Similarly, AS 1 adds the NO_EXPORT community attribute to the BGP advertisement sent to the DUT, resulting in the DUT not

propagating the route for prefix P6 to AS 3. Consequently, the DUT only learns the route for prefix P1 from AS 1 in this scenario. In this test case, the legitimate path for the traffic with source addresses in P1 and destination addresses in P4 is AS 1->AS 2->DUT, and the Tester is connected to the AS 1 and the SAV for customer-facing ASes of the DUT is tested.

The *procedure* is listed below for testing SAV for customer-facing ASes in inter-domain asymmetric routing scenario caused by NO_EXPORT:

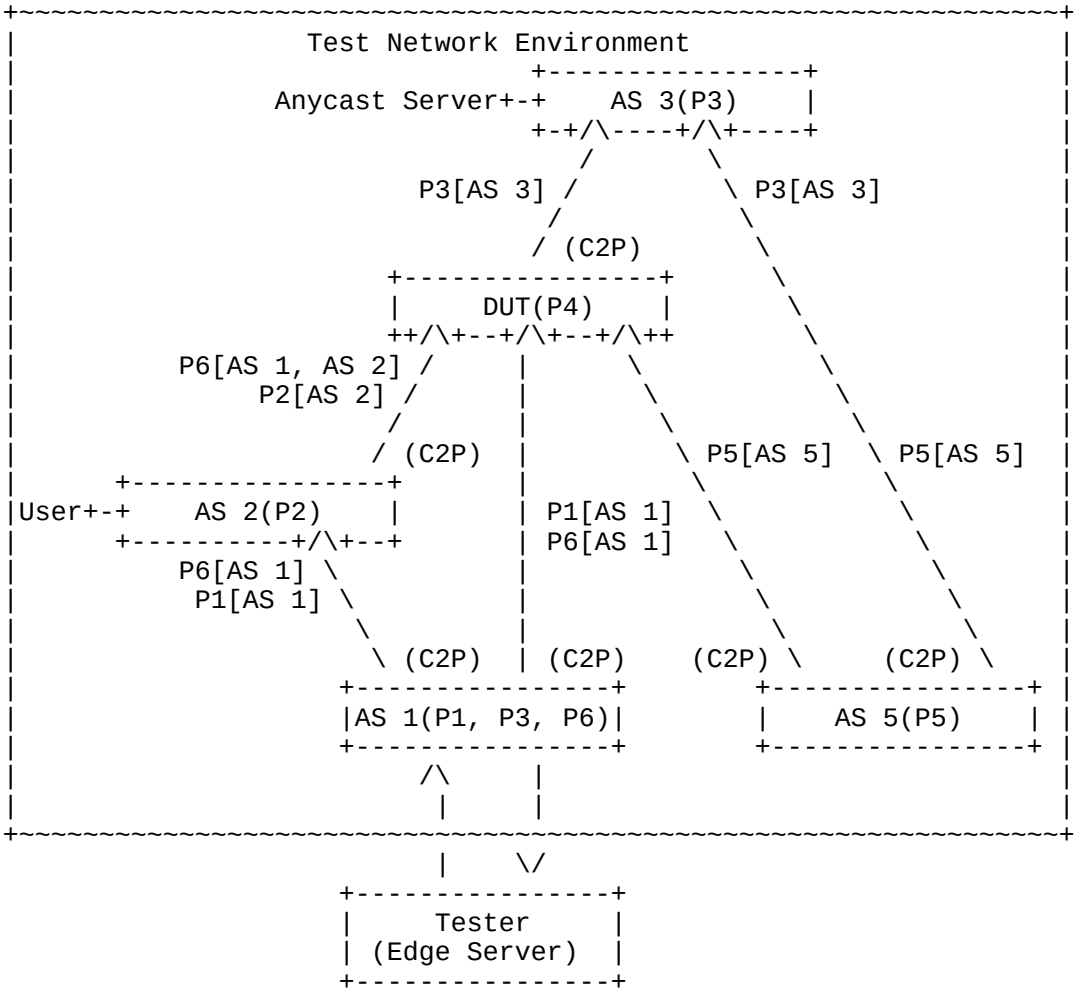
1. First, in order to test whether the DUT can generate accurate SAV rules for SAV for customer-facing ASes in inter-domain asymmetric routing scenario caused by NO_EXPORT, a testbed can be built as shown in Figure 10 to construct the test network environment. The Tester is connected to AS 1 and generates the test traffic to the DUT.

2. Then, the ASes including AS 1, AS 2, AS 3, the DUT, and AS 5, are configured to form the asymmetric routing scenario.

3. Finally, the Tester sends the traffic using P1 as source addresses and P4 as destination addresses (legitimate traffic) to the DUT via AS 2 and traffic using P5 as source addresses and P4 as destination addresses (spoofing traffic) to the DUT via AS 2, respectively. The ratio of spoofing traffic to legitimate traffic can vary, such as from 1:9 to 9:1.

The *expected results* are that the DUT can block the spoofing traffic and permit the legitimate traffic from the direction of AS 2 for this test case.

Note that the locations of the DUT in Figure 10 can be set at AS 1 and AS 2 to evaluate its its false positive rate and false negative rate according to the procedure outlined above. The expected results are that the DUT will effectively block spoofing traffic.



Within the test network environment, P3 is the anycast prefix and is only advertised by A
S 3 through BGP.

Figure 11: SAV for customer-facing ASes in the scenario of direct server return (DSR).

Figure 11 presents a test case of SAV for customer-facing ASes in the scenario of direct server return (DSR). In this test case, AS 1, AS 2, AS 3, the DUT, and AS 5 constructs the test network environment, and the DUT performs SAV as an AS. AS 1 is a customer of AS 2 and the DUT, AS 2 is a customer of the DUT, which is a customer of AS 3, and AS 5 is a customer of both AS 3 and the DUT. When users in AS 2 send requests to the anycast destination IP, the forwarding path is AS 2->DUT->AS 3. The anycast servers in AS 3 receive the requests

and tunnel them to the edge servers in AS 1. Finally, the edge servers send the content to the users with source addresses in prefix P3. The reverse forwarding path is AS 1->DUT->AS 2. The Tester sends the traffic with source addresses in P3 and destination addresses in P2 along the path AS 1->DUT->AS 2.

The *procedure* is listed below for testing SAV for customer-facing ASes in the scenario of direct server return (DSR):

1. First, in order to test whether the DUT can generate accurate SAV rules for SAV for customer-facing ASes in the scenario of DSR, a testbed can be built as shown in Figure 11 to construct the test network environment. The Tester is connected to AS 1 and generates the test traffic to the DUT.

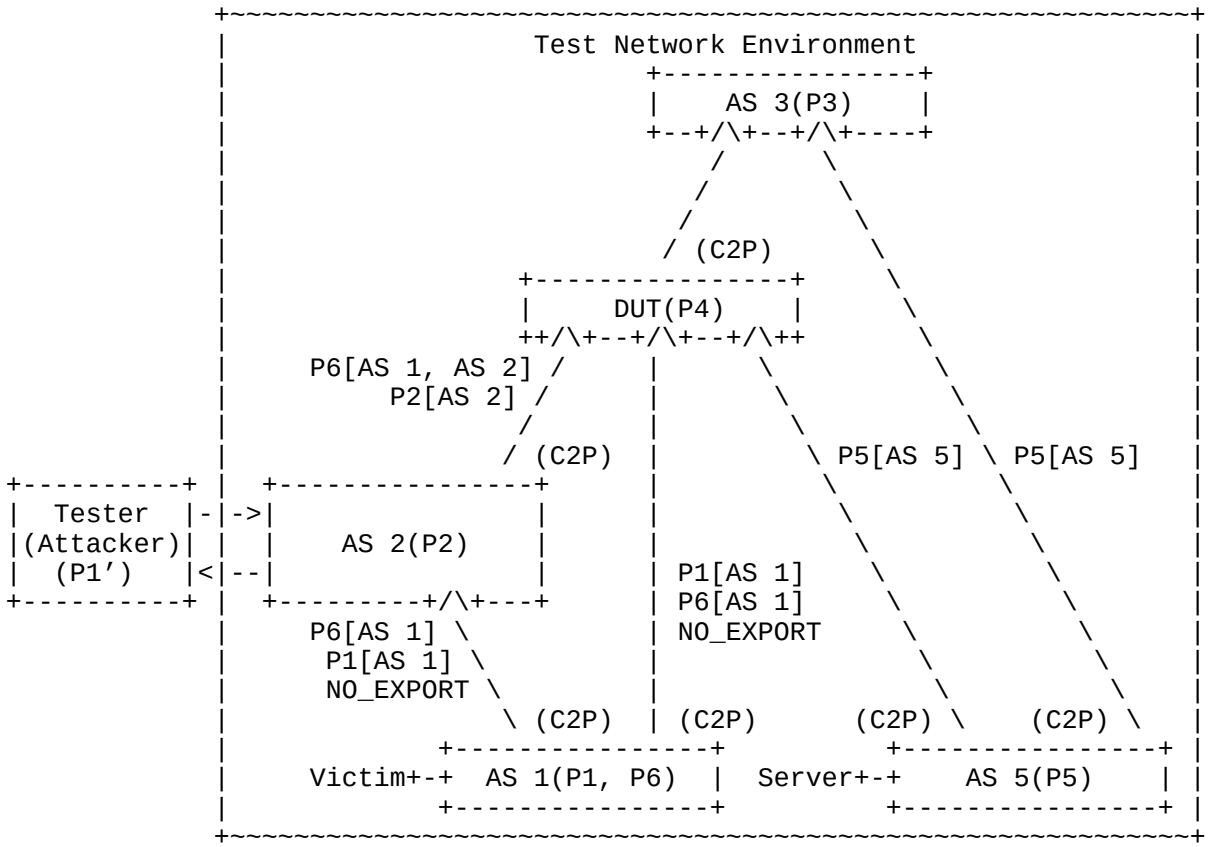
2. Then, the ASes including AS 1, AS 2, AS 3, the DUT, and AS 5, are configured to form the scenario of DSR.

3. Finally, the Tester sends the traffic using P3 as source addresses and P2 as destination addresses (legitimate traffic) to AS 2 via the DUT.

Note that in Figure 11, to direct the return traffic from the edge server to the user to the path AS 1->DUT->AS 2, the document recommends to config static route to direct the traffic with source addresses in P3 and destination addresses in P2 to the DUT.

The *expected results* are that the DUT can permit the legitimate traffic with source addresses in P3 from the direction of AS 1 for this test case.

Note that the locations of the DUT in Figure 11 can be set at AS 1 and AS 2 to evaluate its false positive rate and false negative rate according to the procedure outlined above. The expected results are that the DUT will effectively block spoofing traffic.



P1' is the spoofed source prefix P1 by the attacker which is inside of AS 2 or connected to AS 2 through other ASes.

Figure 12: SAV for customer-facing ASes in the scenario of reflection attacks.

Figure 12 depicts the test case of SAV for customer-facing ASes in the scenario of reflection attacks. In this test case, the reflection attack by source address spoofing takes place within DUT's customer cone, where the attacker spoofs the victim's IP address (P1) and sends requests to servers' IP address (P5) that are designed to respond to such requests. The Tester performs the source address spoofing function as an attacker. The arrows in Figure 12 illustrate the commercial relationships between ASes. AS 3 serves as the provider for the DUT and AS 5, while the DUT acts as the provider for AS 1, AS 2, and AS 5. Additionally, AS 2 is the provider for AS 1.

The *procedure* is listed below for testing SAV for customer-facing ASes in the scenario of reflection attacks:

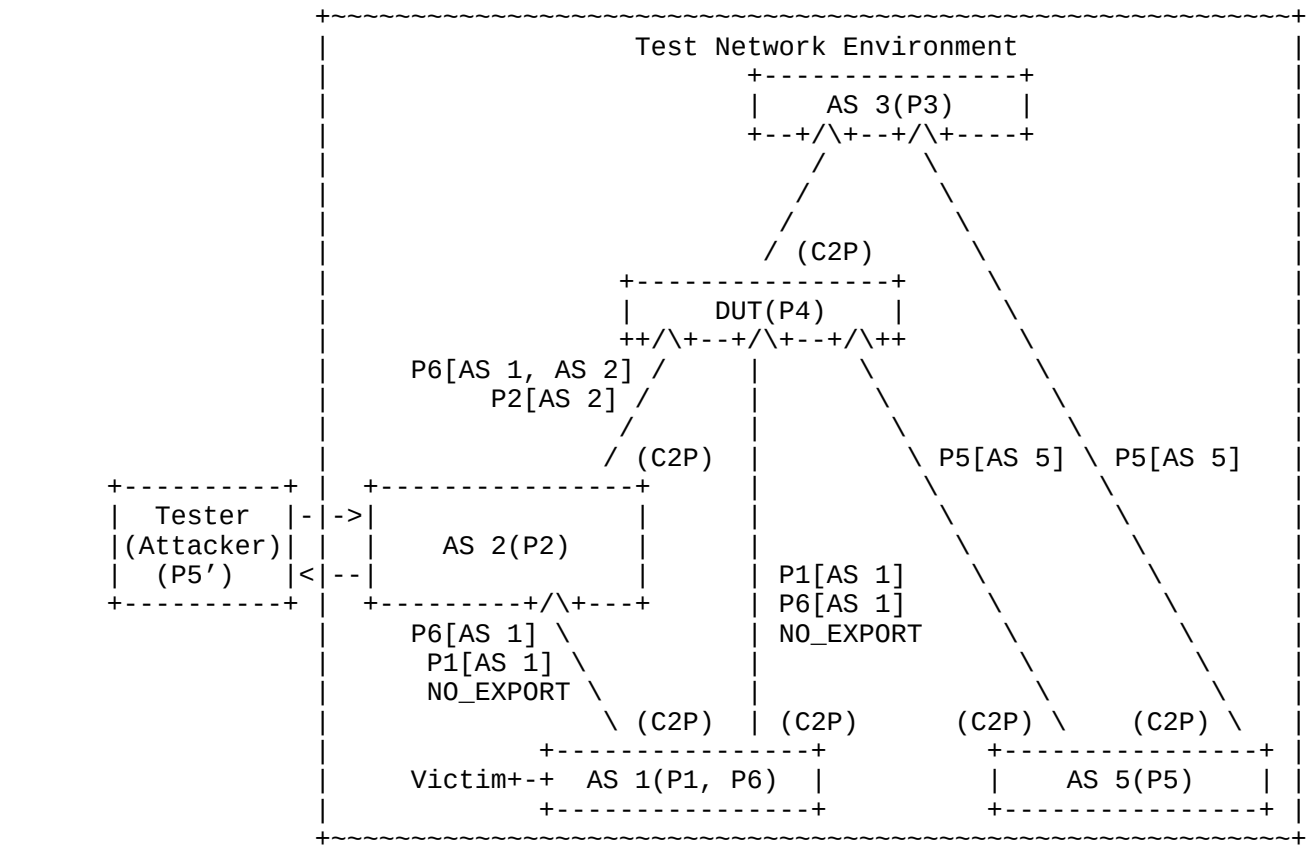
1. First, in order to test whether the DUT can generate accurate SAV rules for SAV for customer-facing ASes in the scenario of reflection attacks, a testbed can be built as shown in Figure 12 to construct the test network environment. The Tester is connected to AS 2 and generates the test traffic to the DUT.

2. Then, the ASes including AS 1, AS 2, AS 3, the DUT, and AS 5, are configured to form the scenario of reflection attacks.

3. Finally, the Tester sends the traffic using P1 as source addresses and P5 as destination addresses (spoofing traffic) to AS 5 via the DUT.

The *expected results* are that the DUT can block the spoofing traffic with source addresses in P1 from the direction of AS 2 for this test case.

Note that the locations of the DUT in Figure 12 can be set at AS 1 and AS 2 to evaluate its false positive rate and false negative rate according to the procedure outlined above. The expected results are that the DUT will effectively block spoofing traffic.



P5' is the spoofed source prefix P5 by the attacker which is inside of AS 2 or connected to AS 2 through other ASes.

Figure 13: SAV for customer-facing ASes in the scenario of direct attacks.

Figure 13 presents the test case of SAV for customer-facing ASes in the scenario of direct attacks. In this test case, the direct attack by source address spoofing takes place within the DUT's customer cone, where the attacker spoofs a source address (P5) and directly targets the victim's IP address (P1), overwhelming its network resources. The Tester performs the source address spoofing function as an attacker. The arrows in Figure 13 illustrate the commercial relationships between ASes. AS 3 serves as the provider for the DUT and AS 5, while the DUT acts as the provider for AS 1, AS 2, and AS 5. Additionally, AS 2 is the provider for AS 1.

The *procedure* is listed below for testing SAV for customer-facing ASes in the scenario of direct attacks**:

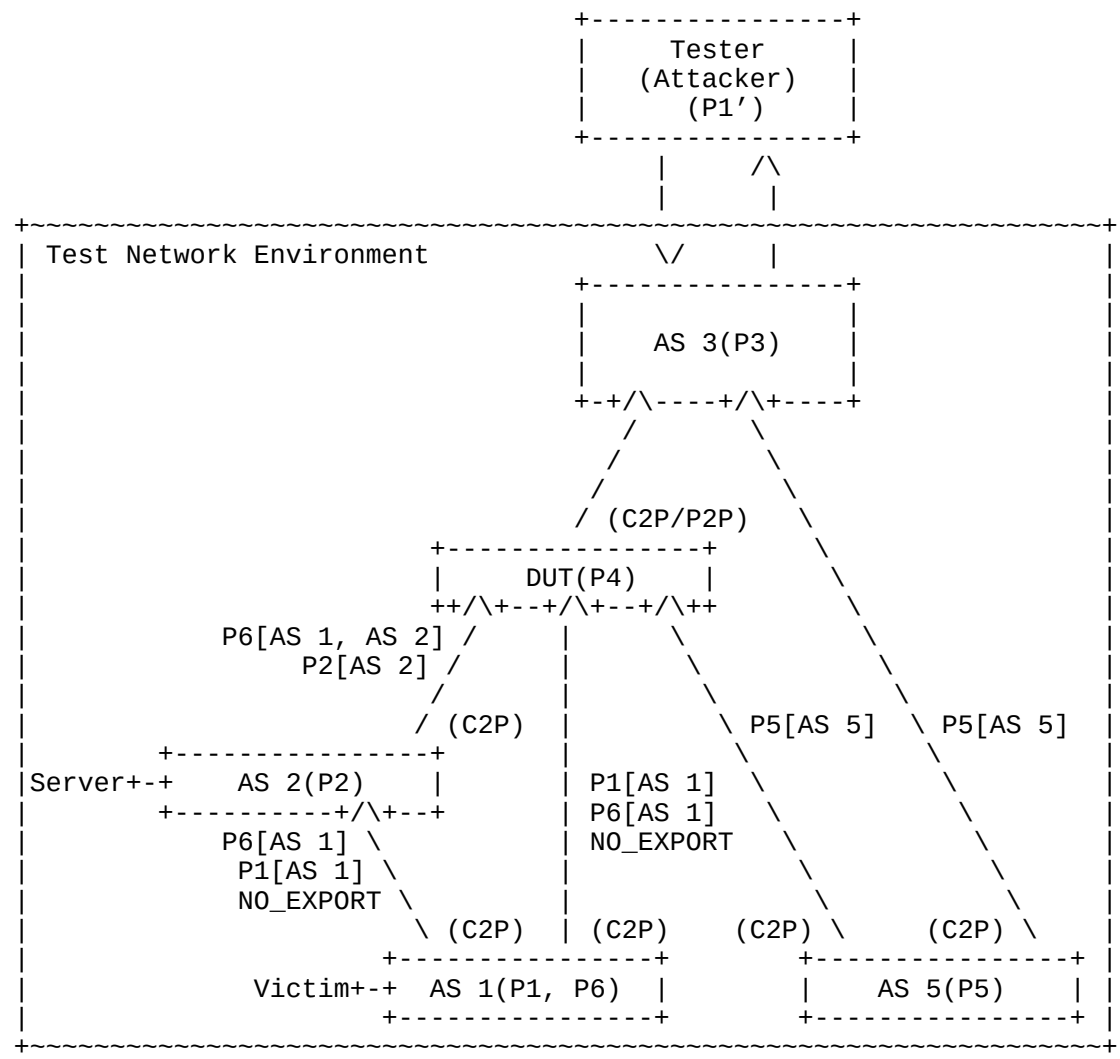
1. First, in order to test whether the DUT can generate accurate SAV rules for SAV for customer-facing ASes in the scenario of direct attacks, a testbed can be built as shown in Figure 13 to construct the test network environment. The Tester is connected to AS 2 and generates the test traffic to the DUT.

2. Then, the ASes including AS 1, AS 2, AS 3, the DUT, and AS 5, are configured to form the scenario of direct attacks.

3. Finally, the Tester sends the traffic using P5 as source addresses and P1 as destination addresses (spoofing traffic) to AS 1 via the DUT.

The *expected results* are that the DUT can block the spoofing traffic with source addresses in P5 from the direction of AS 2 for this test case.

Note that the locations of the DUT in Figure 13 can be set at AS 1 and AS 2 to evaluate its false positive rate and false negative rate according to the procedure outlined above. The expected results are that the DUT will effectively block spoofing traffic.



P1' is the spoofed source prefix P1 by the attacker which is inside of AS 3 or connected to AS 3 through other ASes.

Figure 14: SAV for provider-facing ASes in the scenario of reflection attacks.

SAV for Provider/Peer-facing ASes: Figure 14 depicts the test case of SAV for provider-facing ASes in the scenario of reflection attacks. In this test case, the attacker spoofs the victim's IP address (P1) and sends requests to servers' IP address (P2) that respond to such requests. The Tester performs the source address spoofing function as an attacker. The servers then send overwhelming

responses back to the victim, exhausting its network resources. The arrows in Figure 14 represent the commercial relationships between ASes. AS 3 acts as the provider or lateral peer of the DUT and the provider for AS 5, while the DUT serves as the provider for AS 1, AS 2, and AS 5. Additionally, AS 2 is the provider for AS 1.

The *procedure* is listed below for testing SAV for provider-facing ASes in the scenario of reflection attacks:

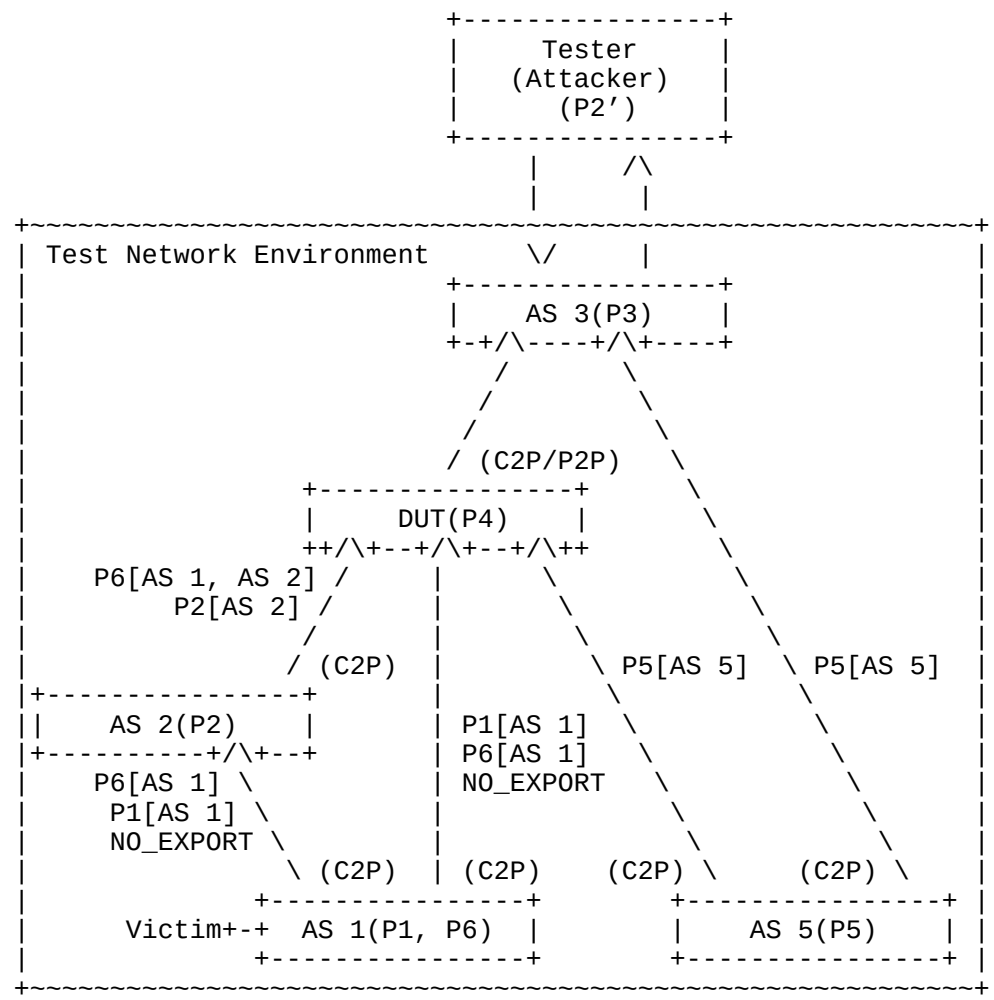
1. First, in order to test whether the DUT can generate accurate SAV rules for SAV for provider-facing ASes in the scenario of reflection attacks, a testbed can be built as shown in Figure 14 to construct the test network environment. The Tester is connected to AS 3 and generates the test traffic to the DUT.

2. Then, the ASes including AS 1, AS 2, AS 3, the DUT, and AS 5, are configured to form the scenario of reflection attacks.

3. Finally, the Tester sends the traffic using P1 as source addresses and P2 as destination addresses (spoofing traffic) to AS 2 via AS 3 and the DUT.

The expected results are that the DUT can block the spoofing traffic with source addresses in P1 from the direction of AS 3 for this test case.

Note that the locations of the DUT in Figure 14 can be set at AS 1 and AS 2 to evaluate its false positive rate and false negative rate according to the procedure outlined above. The expected results are that the DUT will effectively block spoofing traffic.



P2' is the spoofed source prefix P2 by the attacker which is inside of AS 3 or connected to AS 3 through other ASes.

Figure 15: SAV for provider-facing ASes in the scenario of direct attacks.

Figure 15 showcases a testcase of SAV for provider-facing ASes in the scenario of direct attacks. In this test case, the attacker spoofs another source address (P2) and directly targets the victim's IP address (P1), overwhelming its network resources. The arrows in Figure 15 represent the commercial relationships between ASes. AS 3 acts as the provider or lateral peer of the DUT and the provider for AS 5, while the DUT serves as the provider for AS 1, AS 2, and AS 5. Additionally, AS 2 is the provider for AS 1.

The *procedure* is listed below for testing SAV for provider-facing ASes in the scenario of direct attacks:

1. First, in order to test whether the DUT can generate accurate SAV rules for SAV for provider-facing ASes in the scenario of direct attacks, a testbed can be built as shown in Figure 15 to construct the test network environment. The Tester is connected to AS 3 and generates the test traffic to the DUT.

2. Then, the ASes including AS 1, AS 2, AS 3, the DUT, and AS 5, are configured to form the scenario of direct attacks.

3. Finally, the Tester sends the traffic using P2 as source addresses and P1 as destination addresses (spoofing traffic) to AS1 via AS 3 and the DUT.

The *expected results* are that the DUT can block the spoofing traffic with source addresses in P2 from the direction of AS 3 for this test case.

Note that the locations of the DUT in Figure 15 can be set at AS 1 and AS 2 to evaluate its false positive rate and false negative rate according to the procedure outlined above. The expected results are that the DUT will effectively block spoofing traffic.

5.2.2. Control Plane Performance

The test setup, procedure, and measures can refer to Section 5.1.2 for testing the protocl convergence performance and protocol message processing performance.

5.2.3. Data Plane Performance

The test setup, procedure, and measures can refer to Section 5.1.3 for testing the data plane SAV table refreshing performance and data plane forwarding performance.

5.3. Resource Utilization

When testing the DUT for both intra-domain (Section 5.1) and inter-domain SAV (Section 5.2) functionality, CPU utilization (both control plane and data plane) and memory utilization (both control plane and data plane) should be recorded. These measurements should be captured separately for each plane to enable detailed performance analysis.

6. Reporting Format

Each test has a reporting format that contains some global and identical reporting components, and some individual components that are specific to individual tests. The following parameters for test configuration and SAV mechanism settings MUST be reflected in the test report.

Test Configuration Parameters:

1. Test device hardware and software versions
2. Network topology
3. Test traffic attributes
4. System configuration (e.g., physical or virtual machine, CPU, memory, caches, operating system, interface capacity)
5. Device configuration (e.g., symmetric routing, NO_EXPORT)
6. SAV mechanism

7. IANA Considerations

This document has no IANA actions.

8. Security Considerations

The benchmarking tests outlined in this document are confined to evaluating the performance of SAV devices within a controlled laboratory environment, utilizing isolated networks.

The network topology employed for benchmarking must constitute an independent test setup. It is imperative that this setup remains disconnected from any devices that could potentially relay test traffic into an operational production network.

9. References

9.1. Normative References

[RFC3704] Baker, F. and P. Savola, "Ingress Filtering for Multihomed Networks", BCP 84, RFC 3704, DOI 10.17487/RFC3704, March 2004, <https://www.rfc-editor.org/rfc/rfc3704>.

[RFC8704] Sriram, K., Montgomery, D., and J. Haas, "Enhanced Feasible-Path Unicast Reverse Path Forwarding", BCP 84, RFC 8704, DOI 10.17487/RFC8704, February 2020, <https://www.rfc-editor.org/rfc/rfc8704>.

[RFC2544] Bradner, S. and J. McQuaid, "Benchmarking Methodology for Network Interconnect Devices", RFC 2544, DOI 10.17487/RFC2544, March 1999, <https://www.rfc-editor.org/rfc/rfc2544>.

[RFC4061] Manral, V., White, R., and A. Shaikh, "Benchmarking Basic OSPF Single Router Control Plane Convergence", RFC 4061, DOI 10.17487/RFC4061, April 2005, <https://www.rfc-editor.org/rfc/rfc4061>.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/rfc/rfc2119>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/rfc/rfc8174>.

9.2. Informative References

[intra-domain-ps]
"Source Address Validation in Intra-domain Networks Gap Analysis, Problem Statement, and Requirements", 2025, <https://datatracker.ietf.org/doc/draft-ietf-savnet-intra-domain-problem-statement/>.

[inter-domain-ps]
"Source Address Validation in Inter-domain Networks Gap Analysis, Problem Statement, and Requirements", 2025, <https://datatracker.ietf.org/doc/draft-ietf-savnet-inter-domain-problem-statement/>.

[intra-domain-arch]
"Intra-domain Source Address Validation (SAVNET) Architecture", 2025, <https://datatracker.ietf.org/doc/draft-ietf-savnet-intra-domain-architecture/>.

[inter-domain-arch]
"Inter-domain Source Address Validation (SAVNET) Architecture", 2025, <https://datatracker.ietf.org/doc/draft-wu-savnet-inter-domain-architecture/>.

Acknowledgements

Many thanks to Aijun Wang, Nan Geng, Susan Hares, Giuseppe Fioccola, Minh-Ngoc Tran, Shengnan Yue, Changwang Lin etc. for their valuable comments and reviews on this document.

Authors' Addresses

Li Chen
Zhongguancun Laboratory
Beijing
China
Email: lichen@zgclab.edu.cn

Dan Li
Tsinghua University
Beijing
China
Email: tolidan@tsinghua.edu.cn

Libin Liu
Zhongguancun Laboratory
Beijing
China
Email: liulb@zgclab.edu.cn

Lancheng Qin
Zhongguancun Laboratory
Beijing
China
Email: qinlc@zgclab.edu.cn