SECURITY AND COMMUNICATION NETWORKS
Security Comm. Networks (2013)
Published online in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/sec.920

RESEARCH ARTICLE

A combined solution for the Internet access denial caused by malicious Internet service providers

Marwan Abu-Amara*

Department of Computer Engineering, King Fahd University of Petroleum & Minerals, Dhahran, Saudi Arabia

ABSTRACT

The Internet is becoming a vital communication tool for individuals, businesses, and governments. Thus, the Internet access reliability is crucial especially against malicious behaviors. When a malicious higher-tier Internet service provider filters transit traffic for the purpose of dropping a specific network's packets, then an *Internet access denial* occurs. This paper presents a solution for the denial of the Internet access problem that combines a network address translation based solution with a tunnel-based solution. The network address translation based solution is efficient in terms of network performance but suffers from a server reachability problem; a problem that is solved by using a tunnel-based solution. Moreover, the paper evaluates the combined solution performance with respect to the end-to-end delay and the throughput metrics. The combined solution has insignificant effect on these two metrics when traffic originates from the denied network and is forwarded outside the denied network. In contrast, and dependent on the tunneling protocol used, the combined solution increases the end-to-end delay of the network by at least 6% and decreases the throughput of the traffic by at least 1.65% when the traffic is originated outside the denied network and is intended for servers inside the denied network.

KEYWORDS

malicious ISP; Internet access denial; Internet availability; tunnel protocols

***Correspondence**

Marwan Abu-Amara, Department of Computer Engineering, King Fahd University of Petroleum & Minerals, Dhahran, Saudi Arabia.
E-mail: marwan@kfupm.edu.sa

1. INTRODUCTION

The Internet is increasingly becoming an essential element of our daily lives. The Internet is made up of several interconnected autonomous systems (ASes) with each AS being composed of different hosts that are controlled by a single entity. For the most part, Internet service providers (ISPs) operate a group of ASes. According to their size and interconnections, ISPs are categorized into three tiers with tier-1 ISPs constituting the core of the Internet, while tier-3 ISPs provide end-users with Internet services. Thus, packets that are sent from one host to another are likely to be carried over multiple different tier ISPs. Furthermore, the interaction between different ISPs and ASes is facilitated through the use of the standard interdomain routing protocol known as Border Gateway Protocol (BGP). Moreover, to properly route the packets sent from or to a host through the Internet, each host in an AS is assigned a unique Internet protocol (IP) address.

With such a structure for the Internet, the reliability of the Internet access is greatly dependent on the Internet security. Accordingly, it is an established fact that the Internet's BGP has several security weaknesses [1,2]. For example, BGP is incapable of controlling the way traffic is forwarded through ASes. This is because the reachability paths included in the BGP advertisements are only considered to be "promises." Hence, BGP cannot guarantee that traffic will be routed along such paths [3]. Specifically, BGP helps a certain AS in controlling which neighbor AS will receive the packet. On the other hand, BGP does not control how the neighbor AS will subsequently route the packet to the destination. Hence, it is possible that the traffic will be forwarded through paths that the originator of the traffic is not aware of.

Such a behavior by BGP can result in many security concerns. For example, if a malicious ISP is part of any route to the destination, then the transmitted packets may possibly be forwarded through the malicious ISP. Hence, the malicious ISP can refuse to route traffic belonging to a specific network, blocking it from reaching a large number of destinations and causing an *Internet access denial* [4].

As noted in [4] and [5], the concept of an ISP causing an intentional denial of Internet access may seem implausible at first. Nevertheless, an ISP can become malicious and

deny a certain network Internet access for many reasons. For example, governments can be politically motivated in forcing ISPs to impose a denial of Internet access against a specific network so as to establish an Internet blockade against that network. Several examples of such activities are provided in [4] and [5].

Accordingly, this paper proposes a combined solution for the Internet access denial problem. Section 2 presents background material associated with the problem and the proposed solutions. Section 3 provides a further explanation of how the Internet access denial problem can take place. In Section 4, the paper provides a detailed description of the proposed combined solution. The validity of the proposed combined solution is provided in Section 5. Section 6 presents an evaluation of the combined solution impact on the network performance. A comparison between the proposed combined solution and work related to the BGP blackholing problem is presented in Section 7. Finally, the paper conclusions are provided in Section 8.

2. BACKGROUND

The Internet access denial problem along with countermeasures was presented by Mahmoud *et al*. [5] and Abu-Amara *et al*. [4]. Mahmoud *et al*. [5] provided a description of some of the security issues associated with the interdomain routing protocol BGP that can lead to an Internet access denial. Subsequently, Mahmoud *et al*. [5] proposed three solutions to countermeasure the Internet access denial problem: BGP tuning, virtual peering, and virtual transit. The proposed countermeasures depend on the presence of at least one additional non-malicious ISP that can be utilized to force outgoing and incoming traffic from and to the denied region to bypass the malicious ISP. Furthermore, Mahmoud *et al*. [5] provided a qualitative comparison between the three solutions with respect to traffic filtering, communication overhead, setup overhead, scalability, and difficulty to offset the solution. They concluded that the BGP tuning solution, which is based on the use of BGP traffic engineering techniques suggested by Quoitin [6–8] is the simplest approach among the three solutions reviewed in the paper and can be easily implemented by configuring the proper routers. On the other hand, the virtual peering solution, which depends on using standard tunneling protocols such as IP security (IPsec) [9], generic routing encapsulation (GRE) [10], and IP-in-IP [11,12], provides the most deterministic way of controlling the incoming traffic. In contrast, the virtual transit solution combines the scalability of the BGP tuning solution with the deterministic control of the inbound traffic provided by the virtual peering solution. Although Mahmoud *et al*. [5] provided convergence figures for the BGP tuning solution for different types of applications and traffic loads, no performance evaluation of the effect of the use of tunneling protocols in the other solutions was presented. Such a performance evaluation of the tunneling protocols is presented in this paper.

On the other hand, Abu-Amara *et al*. [4] identified that two conditions must be met for the denial of Internet access problem to occur. The two conditions are (i) packets pass through a malicious ISP and (ii) the packets are filtered out by the malicious ISP. Thus, the denial of Internet access problem can be solved by removing at least one of the two conditions. Accordingly, Abu-Amara *et al*. [4] provided a classification of the different solutions for the denial of the Internet access problem. They classified the solutions into two different categories: solutions that manipulate the traffic path to avoid passing through the network of the malicious ISP and solutions that prevent the malicious ISP from filtering out the traffic by hiding its identity. Moreover, Abu-Amara *et al*. [4] proposed a scalable identity concealing solution to the denial of the Internet access problem that is based on the concept of network address translation (NAT) [13,14]. Furthermore, Abu-Amara *et al*. [4] supplied an evaluation of the NAT-based solution's effect on the network performance. They concluded that introducing NAT as a solution has insignificant degradation on the network performance. In addition, Abu-Amara *et al*. [4] addressed the server reachability problem that is associated with NAT routers by introducing a novel approach and demonstrated that the approach has a significantly small impact on the network performance. However, the solution to the server reachability problem requires the development of a new network component that was referred to as a web switch. Although the NAT-based solution can be easily and efficiently deployed to handle traffic that is originated in the denied network and destined for the Internet, the requirement of the web switch makes the NAT-based solution not readily available when traffic is originated in the Internet and destined to the denied region. Moreover, although the concept of the web switch may be extended to some network applications such as simple mail transfer protocol but not necessarily to all network applications. Such a deployment problem is addressed in this paper by employing tunneling protocols. Furthermore, this paper provides a performance evaluation comparison between the web switch option of the NAT-based solution and the tunneling protocols-based solution.

3. PROBLEM DESCRIPTION

Higher-tier ISPs can manipulate the BGP routing protocol to maliciously deny a specific network from accessing the Internet. Furthermore, these higher-tier ISPs can claim to have reachability to destinations within the specific network while filtering out the traffic intended to that network. Therefore, a malicious higher-tier ISP can be thought of as an apparently authentic provider that advertises to the Internet a route to a specific network and advertises to the specific network routes to Internet destinations with the ill intention of blocking the specific network from accessing the Internet. This Internet access denial is achieved by intercepting and filtering out traffic sent from or destined to the specific network.

DOI: 10.1002/sec

The aforementioned scenario is depicted in Figure 1. As shown, the malicious ISP advertises to the remote network (C) that it can deliver traffic to both network (A) and network (B). It also advertises to the blocked network (A) that it can deliver traffic to both network (B) and network (C). However, when traffic is sent by the blocked network (A) to either network (B) or network (C), the malicious ISP intercepts that traffic and drops it. Similarly, when network (C) transmits traffic to network (A), the malicious ISP blocks that traffic and filters it out. On the other hand, when network (C) transmits traffic to network (B), the malicious ISP forwards that normally and the traffic successfully reaches network (B). Accordingly, the malicious ISP performs an Internet access denial against network (A).

As noted by Abu-Amara *et al.* [4], the malicious ISP size, location, and connectivity have a direct impact on the extent of the Internet access denial. More specifically, an Internet access denial can be caused by a lower-tier ISP only if it is present in the traffic's path. In contrast, a larger impact on the Internet access can be caused by higher-tier ISPs.

Because tier-3 ISPs do not act as transit for other networks, they only carry traffic that belongs to their networks. Therefore, a malicious tier-3 ISP can only block access to its own network. Hence, the impact of this type of ISP is limited to only a small set of hosts and services. On the other hand, malicious higher-tier ISPs can have more impact as they can block not only traffic that belongs to their networks but also all other traffic that passes through them in transit. For example, a malicious tier-2 ISP can block access to its own network and to all its customer ISPs' networks. Furthermore, Internet access denial by tier-1 ISPs presents a more severe problem. A malicious tier-1 ISP can isolate the victim network and block it from accessing a considerable segment of the Internet. Because of the major impact that a malicious higher-tier ISP can cause, solutions to the Internet access denial problem should be studied and deployed. Figure 2 shows a simplified network of ISPs of different tiers and how Internet access denial that is caused by higher-tier ISPs results in a larger inaccessibility to other parts of the network.

Consequently, this paper proposes and evaluates solutions to resolve the denial of Internet access problem by concealing the identity of the traffic belonging to the denied network. Once the traffic identity is concealed, the traffic can pass through the malicious higher-tier ISP without being intentionally dropped.

4. COMBINED SOLUTION FOR THE INTERNET ACCESS DENIAL PROBLEM

As stated in Section 2, Abu-Amara *et al.* [4] proposed a scalable NAT-based solution to resolve the denial of the Internet access problem. The solution requires the denied region to use NAT routers as gateway connections to neighboring networks and to use a group of IP addresses that are not blocked with the NAT routers. The non-blocked IP addresses are not part of the IP ranges registered to the denied network; they are obtained privately from a neighboring network and they are kept publicly registered to that neighboring network. Subsequently, the solution requires setting the NAT enabled gateway routers to use NAT to translate all outgoing traffic into the non-blocked public IP addresses. Once NAT is configured properly and enabled, the NAT routers will translate all

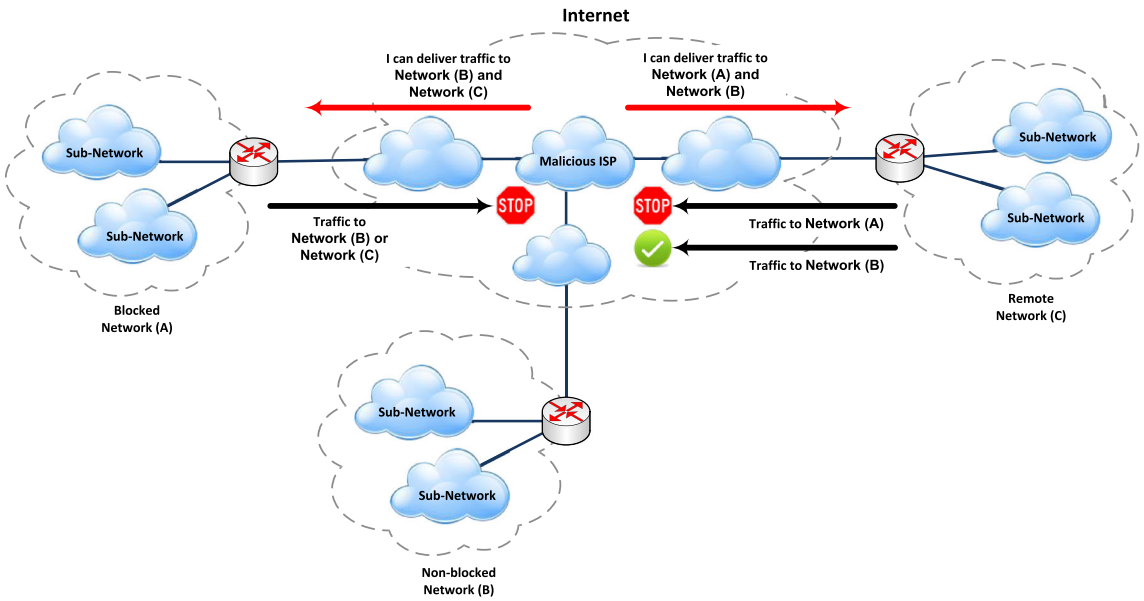


Figure 1. Malicious higher-tier Internet service provider blocking.

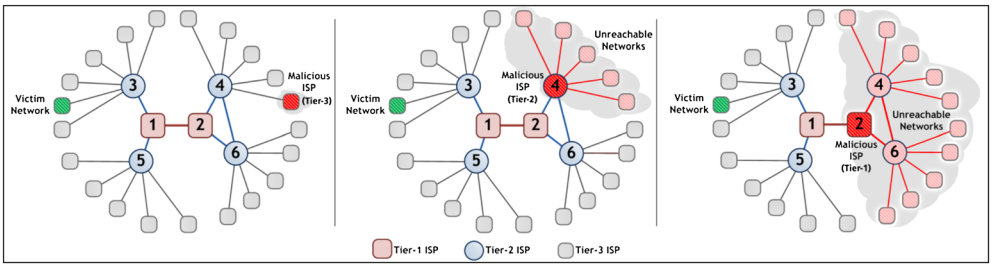


Figure 2. Impact of different tiers of malicious Internet service providers on Internet access denial.

traffic into the non-blocked public IP addresses allowing the clients within the denied region to send requests and receive responses. As a result of the IP address translation made by the NAT routers, each packet sent from the denied region will have a non-blocked IP address as the source address. As such, the outgoing packet does not contain any reference to the original source address belonging to the denied region. Hence, from the malicious ISP point of view, each of the packets sent by the denied region network appears to be originated from the non-blocked neighboring network from which the non-blocked IP addresses were obtained. Thus, even if traffic is forwarded through the malicious ISP, it will be impossible for the malicious ISP to recognize that it belongs to the denied region, and the malicious ISP will route it normally through its network. Hence, the solution hides the identity of the traffic belonging to the denied region's network by virtue of using the NAT feature on the gateway router.

Further, Abu-Amara *et al*. [4] identified the need for the NAT-based solution to be scalable due to the NAT router memory and the transport layer port number exhaustions. This is a direct result of the translation table management performed by the NAT router to keep track of all the replaced source IP addresses and port numbers of the outgoing packets. As such, the scalability of the solution is achieved by using more than one NAT router as gateway routers. Each NAT router is responsible for handling a subset of the denied region's network and is assigned a unique pool of non-blocked IP addresses, as depicted in Figure 3. The partitioning of the denied region's network can be performed on the basis of the physical topology. The denied region's network is partitioned into a number of subnetworks, and each subnetwork uses its own NAT router to translate traffic.

The proposed NAT-based solution is readily deployable for traffic originating from the network of the denied region and makes the solution very suitable for such traffic. However, because of the server reachability problem associated with NAT routers [4], the solution is not suitable when the traffic is originated outside the denied region's network and is destined to the denied region. Although Abu-Amara *et al*. [4] proposed a solution to the server reachability problem by using a web switch inside the denied region's network, such a solution is not readily deployable as the web switch is a new network component that needs further development. Moreover, the proposed web switch approach may not necessarily be extended to all network applications. A possible work around is for the denied region network to obtain several uncritical IP addresses borrowed from a cooperating network to assign

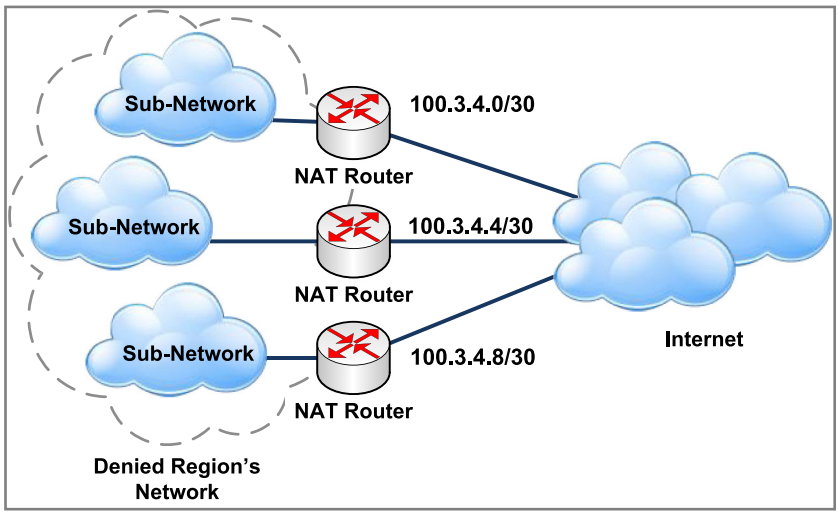


Figure 3. Scalable network address translation based solution design.

DOI: 10.1002/sec

to the servers it has. Although the routing will be more difficult, a more critical concern is that the servers have to publicly register their newly borrowed IP addresses with the DNS so that the servers can be properly accessed. Subsequently, the malicious ISP can easily discover the denied region's newly borrowed IP addresses and block them making such IP addresses useless.

Accordingly, a solution that depends on using tunneling protocols is proposed to resolve the server reachability problem making such a solution readily deployable. The tunnel-based solution establishes a tunnel along the path between the denied region's network and a remote network using any of the standard tunneling protocols.

Figure 4 shows the tunnel-based solution network setup. For the solution to work properly, the presence of at least two non-blocked cooperating networks outside the denied region is needed. The two cooperating networks are labeled in Figure 4 as *cooperating network (A)* and *cooperating network (B)*. As shown in Figure 4, the *cooperating network (A)* must exist in between the denied region's network and the malicious ISP network. Moreover, the *cooperating network (A)* must be a neighbor of the denied region network so as to simplify the tunnel establishment. Likewise, and as shown in Figure 4, the *cooperating network (B)* must exist in between the malicious ISP network and the remote network. Note that if the remote network is directly connected to the malicious ISP network, then the remote network must act as the *cooperating network (B)* for the solution to work properly.

Each of the cooperating networks will dedicate an IP address to be used for the creation and the operation of the tunnel. The dedicated IP address provided by the *cooperating network (A)* will be privately assigned to the gateway router R1's interface that connects it with router R2 of the *cooperating network (A)*. It should be pointed out that the denied region network can consult the list of AS paths stored in the routing information base table [15] of the gateway router R1 as well as *route views* [16] to select candidate ASes to become the cooperating networks. As it will be explained in Section 5, the selection of the *cooperating network (B)* plays an important role in attracting the traffic originating from the remote network and directing it toward the denied region's network. Once two candidate ASes are selected by the denied region then a service level agreement can be established with them to facilitate the creation of the tunnel and to maintain the secrecy of the actual intention behind the creation of the tunnel. Because the two IP addresses used for the creation of the tunnel belong to non-blocked networks, then the use of a standard tunnel setup protocol to activate the tunnel will be considered by the malicious ISP as legitimate. Therefore, it is unnecessary to keep the creation of the tunnel invisible from the malicious ISP.

With the network setup of Figure 4, a tunnel is then established between the gateway router R1 of the denied region's network and the gateway router R4 of the *cooperating network (B)*. Hence, routers R1 and R4 become the two end points of the tunnel. As part of the tunnel establishment, the tunnel setup protocol adds an entry to the routing table of R4 so that if a packet is received with a destination IP address owned by the denied region then it should be encapsulated and sent over the tunneled link. In the meantime, the intra-AS routing protocol propagates to the other routers of the *cooperating network (B)* that if a packet is received with a destination IP address owned by the denied region then it should be forwarded to router R4. On the other hand, if any router of the *cooperating network (B)* receives a packet that has a destination IP address that is not owned by the denied region then the remaining rules of the routing table of that router should direct the router to forward the packet to its ultimate destination without encapsulation.

Subsequently, traffic is transmitted from the remote network to the denied region's network, through the established tunnel, by first transmitting from the remote network to the *cooperating network (B)*. Once traffic is

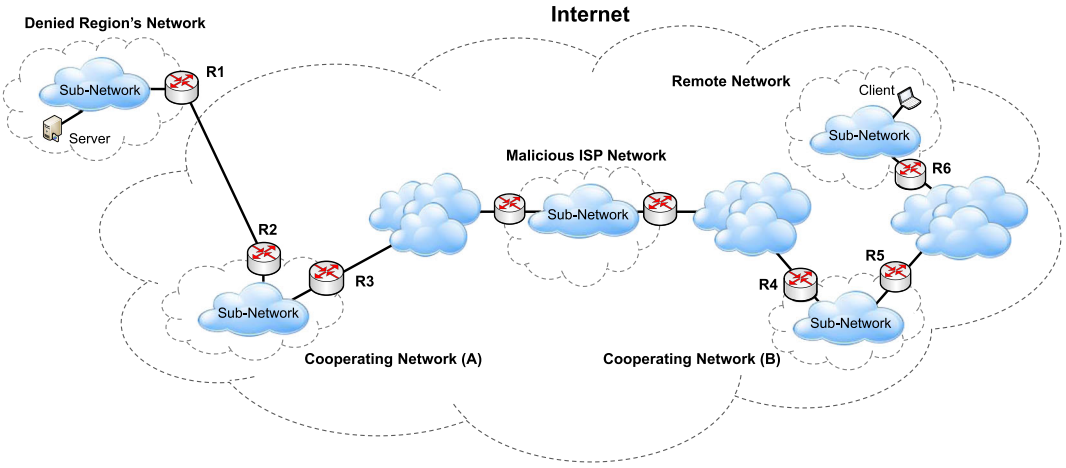


Figure 4. Tunnel-based solution network setup.

received at the *cooperating network (B)*, then the routers in that network will forward it to router R4. On the basis of R4's routing table, router R4 determines then to encapsulate the traffic and forwards it over the established tunnel, which passes through the malicious ISP to the *cooperating network (A)*. Router R2 of the *cooperating network (A)* continues to forward the traffic over the established tunnel to router R1 of the denied region's network. Once traffic is received at router R1, the end of tunnel, it is decapsulated by router R1, and it is forwarded by router R1 to the intended server of the denied region's network. The reverse traffic follows the reverse direction of the same path used earlier.

Note that although one endpoint of the tunnel is router R1 of the denied region's network, the traffic exchanged through the tunnel is encapsulated and will have an IP address that is not owned by the denied region's network. Hence, the identity of the traffic exchanged through the tunnel is concealed, and to the malicious ISP network, the traffic appears to be intended for the *cooperating network (A)*. Thus, the malicious ISP network will be misled into forwarding the traffic belonging to the denied network without being able to drop it. Moreover, a packet sent from or destined to the denied region network can be made more secure, if needed, by using an encrypted tunnel. Accordingly, it becomes impossible for the malicious ISP to discover that the packet is sent from or destined to the denied region network even if the malicious ISP attempts to perform deep packet inspection.

Unlike the NAT-based solution, the tunnel-based solution does not need to perform any IP address translations nor does it require carrying out any transport layer port number translations. Accordingly, the tunnel-based solution does not need to manage a separate translation table as in the case of the NAT-based solution. Instead, the tunneling protocol adds a single entry to the routing table of the two end points of the tunnel when the tunnel is created. Once the tunnel is created then every packet that reaches the entry point of the tunnel gets encapsulated by appending a new header to the packet. The newly appended header contains the tunnel entry point and the tunnel exit point IP addresses as the source and the destination IP addresses, respectively. Hence, the tunnel-based solution does not cause router memory exhaustion nor does it cause transport layer port numbers exhaustion. As such, the entire traffic belonging to the denied region network can be carried over a single tunnel. Thus, scalability is not an issue for the tunnel-based solution.

5. VALIDITY OF THE PROPOSED COMBINED SOLUTION

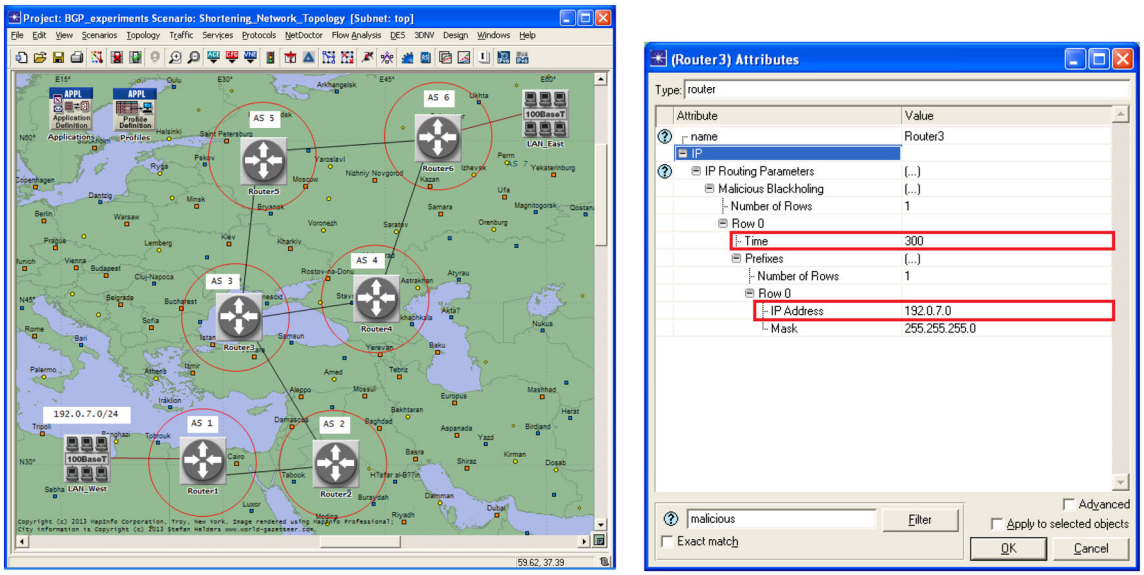
To validate the proposed combined solution, it is necessary to verify two aspects. The first aspect is concerned with showing that the identity of the traffic originating from or destined to the denied region is hidden. The second aspect is concerned with verifying that the traffic destined to the denied region will be attracted toward the *cooperating network (B)* of Figure 4. The purpose of verifying the first aspect is to show that the traffic will not be intentionally dropped by the malicious ISP based on discovering the identity of the traffic. On the other hand, the purpose of verifying the second aspect is to show that the malicious ISP network will not be able to prevent the *cooperating network (B)* from utilizing the established tunnel to protect the traffic sent by the remote network to the denied region against intentional dropping.

The verification of the first aspect was outlined in Section 4 and follows directly from the fact that the use of NAT routers and tunneling in the Internet to hide the traffic identity is well established. On the other hand, to verify the second aspect, it is important to consider how the *cooperating network (B)* in Figure 4 is selected. The selection of the *cooperating network (B)* is based on one of the following two cases:

Case (1): It is preferable to select the *cooperating network (B)* to be on the path taken by the BGP UPDATE message sent by the malicious ISP network and received by the remote network. Subsequently, the AS number of the *cooperating network (B)* will be included in the AS-Path of the UPDATE message received by the remote network. Hence, the *cooperating network (B)* will receive the traffic sent by the remote network and will simply forward it through the established tunnel to the denied region.

Case (2): If the selection based on Case 1 is not feasible, then the *cooperating network (B)* must be selected such that the number of ASes in between the *Cooperating Network (B)* and the remote network is less than or equal to the number of ASes included in the UPDATE message received by the remote network from the malicious ISP network. In this case, the *cooperating network (B)* can benefit from the AS-Path shortening approach proposed in [5] to advertise a shorter AS-PATH that can be used by the remote network to deliver traffic to the denied region. The concept behind the AS-Path shortening approach is that when a router selects between two BGP routes, it selects the route that has the shortest AS-Path. Hence, because the *cooperating network (B)* has an established tunnel to the denied region, then it can send an UPDATE message claiming the ownership of the denied region prefixes. Recalling how the *cooperating network (B)* was selected in this case, then when the UPDATE message is received at the remote network, it will have an AS-Path that is shorter than the AS-Path included in the UPDATE message received from the malicious ISP network. Thus, the remote network selects the route with the shorter AS-Path and, accordingly, forwards the traffic toward the *cooperating network (B)*.

To verify Case (2), the small network shown in Figure 5 (a) is simulated using OPNET Modeler, Riverbed Technology, San Francisco, California, USA. [17]. As illustrated in



(a) (b)

Figure 5. Validation simulation network setup.

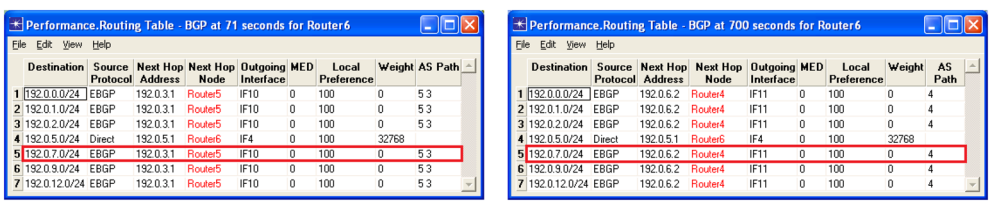
Figure 5(a), the denied region is represented by AS1 with an assigned prefix of 192.0.7.0/24, the *cooperating network (A)* is represented by AS2, the malicious ISP network is represented by AS3, the *cooperating network (B)* is represented by AS4, and the remote network is represented by AS6. The simulation is setup to invoke the malicious activity by Router3 of AS3 at time 300 s as shown in Figure 5(b). Subsequently, a tunnel is setup between Router4 of AS4 and Router1 of AS1, and an UPDATE message is sent from Router4 of AS4 to Router6 of AS6.

Examining the BGP tables of Router6 of AS6 before and after invoking the malicious activity by Router3 of AS3 verifies which route is selected by Router6 of the remote network AS6 to reach Router1 of the denied region AS1. Specifically, Figure 6(a) shows that the AS-PATH selected by Router6 to reach the denied region before invoking the malicious activity is "AS5 AS3," while Figure 6(b) illustrates that the AS-PATH selected by Router6 to reach the denied region after invoking the malicious activity is "AS4."

Moreover, the first and the second graphs in Figure 7 demonstrate that prior to invoking the malicious activity at time 300 s the traffic sent by the remote network is exchanged between Router3 of the malicious AS3 and Router1 of the denied region AS1. On the other hand, the third and the fourth graphs in Figure 7 show that after completing the establishment of the tunnel and the transmission of the UPDATE message from Router4 of AS4 to Router6 of AS6 at time 350 s, the traffic sent by the remote network is exchanged between Router4 of the *cooperating network (B)* and Router1 of the denied region. Furthermore, it is observed from the fifth graph in Figure 7 that the traffic is being intentionally dropped by Router3 of the malicious AS3 between 300 and 350 s and subsides thereafter marking the inability of Router3 to sustain the intentional dropping of traffic against the denied region.

6. PERFORMANCE EVALUATION

The use of a tunneling protocol in any communication introduces an extra overhead to the packets carried by the tunnel. Table I summarizes the maximum size of extra overhead



(a) (b)

Figure 6. Border gateway protocol routing table of Router6 of the remote network.

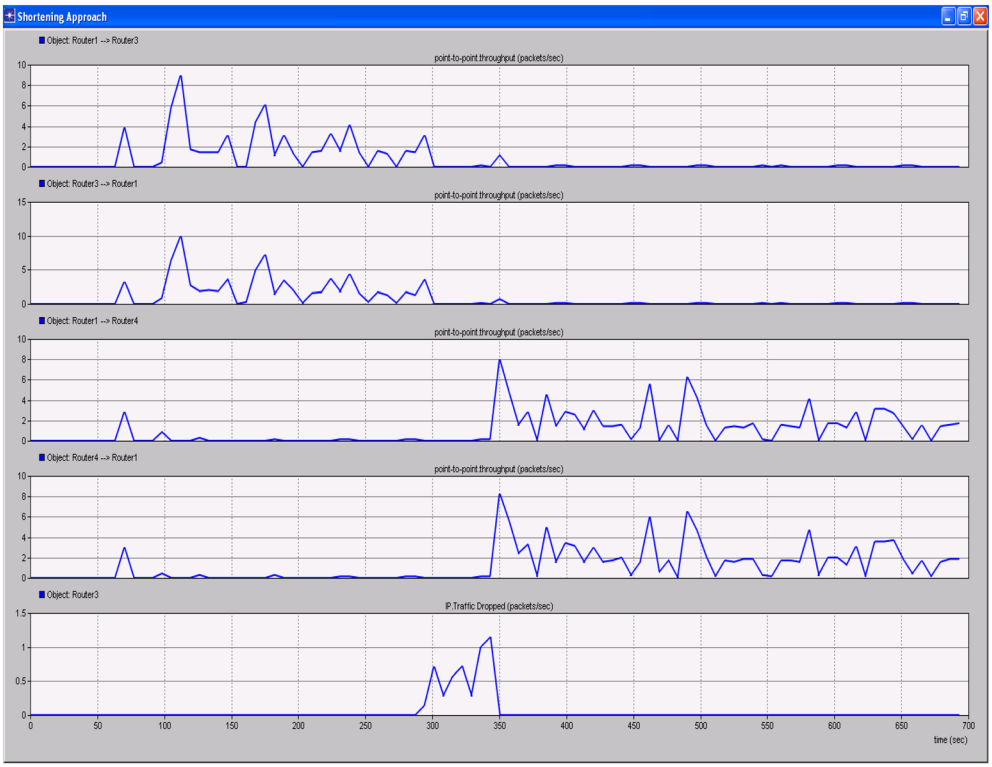


Figure 7. Throughput observed by denied region and drop rate at malicious Internet service provider network.

Table I. Tunneling protocol maximum size of overhead.

Tunneling protocol	Encrypted tunnel	Maximum size of extra overhead
IP-in-IP	No	20 bytes
GRE	No	24 bytes
GRE with checksum	No	28 bytes
IPsec with DES	Yes	77 bytes
IPsec with AES	Yes	85 bytes

IP, Internet protocol; GRE, generic routing encapsulation; IPsec, Internet protocol security; AES, advanced encryption standard; DES, data encryption standard.

introduced by IP-in-IP, GRE, GRE with checksum, IPsec used in conjunction with data encryption standard (DES), and IPsec used in conjunction with advanced encryption standard (AES). Note that AES provides a stronger encryption scheme than DES [18–20].

Accordingly, Section 6.2 compares the performance of the tunnel-based solution under malicious ISP activity against the normal operation with no malicious ISP activity and with no tunnels. The performance evaluation examines the network end-to-end delay and throughput when using different network applications and different loads. Moreover, Section 6.3 compares the tunnel-based solution results against the results of the web switch option of the NAT-based solution reported by Abu-Amara *et al.* [4]. The performance evaluation is accomplished through the use of OPNET Modeler simulations [17]. The details of the simulation setup are discussed in Section 6.1.

6.1. Tunnel-based solution simulation setup

The simulated network is depicted in Figure 4. The remote network has 10 connected hosts that act as clients, whereas the denied region's network has 10 connected hosts that act as servers for the considered applications. Each host is interconnected to its respective network by a 100 Mbps Fast Ethernet link. Moreover, the generic router model provided by OPNET was used for the simulations. The generic router model supports many protocols, including IP-in-IP, GRE, GRE with checksum, and IPsec tunneling protocols; the tunneling protocols used for the simulations. Routers of different networks are interconnected using 1.544 Mbps DS-1 links.

Two network applications are simulated: hypertext transfer protocol (HTTP) and video conferencing. HTTP is selected to represent applications that run over transmission control protocol (TCP), while video conferencing is selected to represent applications that run over user datagram protocol (UDP). Each application is simulated using three traffic scenarios: low, medium, and high traffic. The low traffic scenario uses 25% of the available link's bandwidth, which is about 380 kbps. The medium traffic uses 50% of the bandwidth (about 770 kbps). The high

traffic utilizes 75% of the bandwidth (about 1200 kbps). These scenarios are selected to evaluate the performance of each tunnel protocol under different traffic loads. Each simulation is executed five times, and the results of the five runs are averaged out. The performance evaluation considered the end-to-end delay metric as well as the traffic throughput metric.

It should be pointed out that the approach taken in evaluating the tunnel-based solution is similar to that taken by Abu-Amara *et al*. [4] when evaluating the network performance impact of the NAT-based solution.

6.2. Performance evaluation results

The end-to-end delay is measured by each simulation run, and it is computed by measuring the time a client needs to forward a packet to a server. This takes into account the transmission times, the propagation times, the queuing delays, and the added encapsulation and decapsulation delays associated with tunneling. Results of the end-to-end delay simulations are presented and discussed in Section 6.2.1. In addition, each simulation measures the impact of tunneling on the traffic throughput by computing the total traffic a host sends and receives per second. Results of the throughput simulations are presented and discussed in Section 6.2.2. The results for the two metrics are generated for each traffic type, for different traffic loads, and for each tunneling protocol.

6.2.1. End-to-end delay simulation results

The results of the HTTP simulation are provided in Figure 8. The absolute end-to-end delay is depicted in Figure 8(a), while the end-to-end delay percentage increase which is computed as $(Delay_{WithTunnel} - Delay_{WithNoTunnel})/(Delay_{WithNoTunnel})$ is illustrated in Figure 8(b). The largest percentage increase is about 39% in the case of 25% loading with the IPsec with AES tunneling protocol. The end-to-end delay percentage increase is mainly attributed to the increase in the processing time needed for encapsulating and decapsulating and encrypting and decrypting the tunnel traffic. Another reason for the end-to-end delay percentage increase is associated with the fact that the tunnel encapsulation and encryption may result in fragmentation [21]. Thus, more packets will need to be processed.

Another observation that is clear from Figure 8(b) is that as the traffic load increases the percentage increase of the end-to-end delay decreases. This is mainly because higher traffic results in higher queuing delay, which eventually becomes more significant than the time needed for processing the tunnel traffic. Moreover, it is apparent from Figure 8(b) that the IP-in-IP tunneling protocol experiences the smallest end-to-end delay percentage increase. This observation is attributed to the fact that the amount of overhead bytes added by the IP-in-IP tunnel is the smallest when compared with the other simulated protocols. Subsequently, IP-in-IP produces fewer fragments than the other tunneling protocols.

The other application considered in the simulation is video conferencing, which utilizes UDP as a transport layer. The absolute end-to-end delay is illustrated in Figure 9(a), while Figure 9(b) presents the end-to-end delay percentage increase. It can be deduced from Figure 9(b) that the end-to-end delay percentage increase results for video conferencing are comparable with the results found for HTTP, although significantly lower. The lower end-to-end delay percentage increase for video conferencing is mainly due to two reasons. The first reason is associated with the fact that the video conferencing packet size is relatively smaller than the HTTP packet size. The second reason is that video conferencing uses the UDP protocol as a transport layer, and therefore, it does not require acknowledgements. Hence, video conferencing requires less time to transmit the packets than HTTP, which results in reduction in the overall end-to-end delay.

Figure 9(b) shows further that, similar to HTTP, the IP-in-IP tunneling protocol experiences the smallest end-to-end delay percentage increase. Moreover, Figure 9(b) shows that as the traffic load increases, the percentage increase of the end-to-end delay decreases. This is again because higher traffic results in higher queuing delay, which eventually becomes more significant than the time needed for processing the tunnel traffic. Accordingly, it

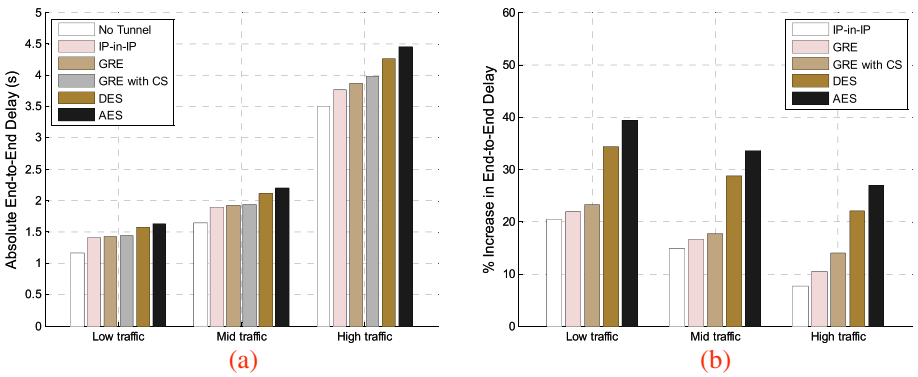


Figure 8. End-to-end delay for hypertext transfer protocol traffic.

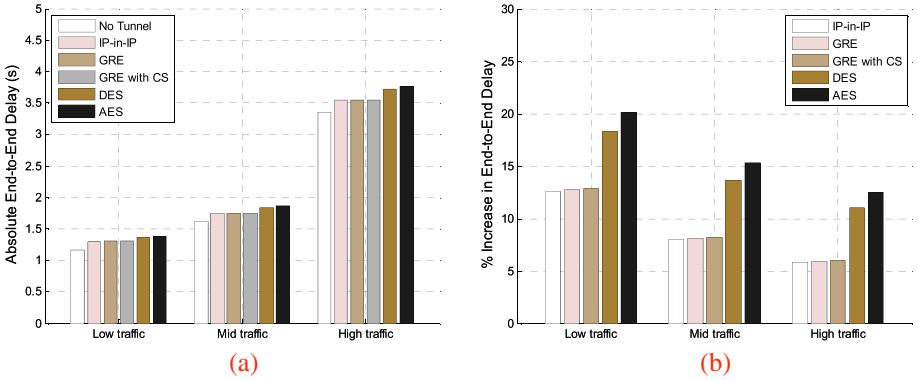


Figure 9. End-to-end delay for video conferencing traffic.

can be concluded that for massive networks such as tier-1 and tier-2 ISPs, where traffic volumes are much higher, the relative impact of tunneling on the end-to-end delay would be even smaller. In addition, the modern routers improve their processing performance by employing multi-core processors in their design so as to process the incoming packets in parallel at Gigabytes per second rates [22]. Hence, the tunnel processing delay becomes less significant to the overall end-to-end delay.

6.2.2. Throughput simulation results

The results for the percentage of throughput decrease for the HTTP application computed as $(Throughput_{WithNoTunnel} - Throughput_{WithTunnel})/Throughput_{WithNoTunnel}$ are illustrated in Figure 10. It can be concluded from Figure 10 that the traffic load has no impact on the percentage decrease in the throughput. This is because the overhead bytes introduced by tunneling are constant and are added to each packet irrespective of the traffic load. Hence, the percentage decrease in the throughput remains constant for all considered tunneling protocols. Also, Figure 10 shows that when IP-in-IP is used, the throughput percentage decrease is the least because it has the smallest added overhead when compared with the other tunneling protocols.

The other application considered in this simulation is video conferencing, which runs over UDP. As Figure 11 illustrates, the percentage decrease in the throughput results are comparable with the results found for HTTP. It is clear from Figure 11 that the traffic load has no impact on the throughput percentage decrease for the same reason provided in the case of HTTP. Hence, the percentage decrease in the throughput remains constant for all considered tunneling protocols. Moreover, Figure 11 shows that the percentage decrease in the throughput when using IP-in-IP is the smallest because it adds the least amount of overhead bytes when compared with the other tunneling protocols.

When comparing Figure 11 to Figure 10, it is noted that the percentage decrease in the throughput of video conferencing is lower than that of HTTP. This is mostly because the size of the overhead of a video conferencing packet, which uses UDP as a transport layer, is smaller than the size of the overhead of an HTTP packet, which uses TCP as a transport layer.

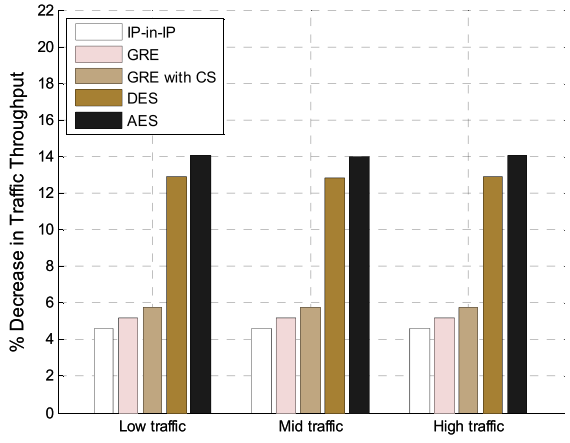


Figure 10. Percentage throughput decrease for hypertext transfer protocol.

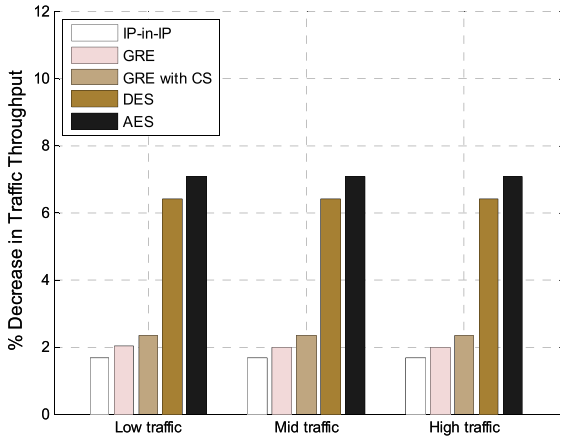


Figure 11. Percentage throughput decrease for video conferencing.

DOI: 10.1002/sec

6.3. Tunnel-based solution results versus network address translation based solution results

The tunnel-based solution results obtained in Section 6.2 are compared against the results of the web switch option of the NAT-based solution, referred to here as the web switch solution, that were provided by Abu-Amara *et al.* [4]. Specifically, the comparison looks at the HTTP end-to-end delay and the HTTP throughput for both low traffic loads and high traffic loads of the web switch solution and the IP-in-IP tunnel-based solution. The IP-in-IP tunneling protocol was selected for comparison because it demonstrated the best performance metrics among the protocols simulated in this paper.

The simulated web switch solution is shown in Figure 12 and is similar to that of Figure 4 that was used for the tunnel-based solution. In Figure 12, router R1 of the denied region's network acts as both a NAT gateway router and as a web switch. The web switch feature of router R1 accepts the TCP connection from a client, receives the HTTP request, and then decides which server should handle this request based on the *host* part of the HTTP request header [23]. The request then is handed over to the selected server.

Similar to the tunnel-based solution, the *cooperating network (A)* is needed for the web switch solution and must exist in between the denied region's network and the malicious ISP network. Note that if the denied region's network is directly connected to the malicious ISP network, then the *cooperating network (A)* must also be a neighbor of the denied region's network for the solution to work properly. The purpose of the *cooperating network (A)* is to provide the NAT gateway router R1 with an IP address that is not owned by the denied region's network. In contrast, the presence of the *cooperating network (B)* is not necessary for the web switch solution. However, for a fair comparison with the tunnel-based solution, the *cooperating network (B)* is kept in the simulated network setup of the web switch solution.

Similar to the simulation setup for the tunnel-based solution, the remote network in the web switch solution has 10 connected hosts that act as clients, whereas the denied region's network has 10 connected hosts that act as servers for the considered applications. Each host is interconnected to its respective network by a 100 Mbps Fast Ethernet link. Moreover, the generic router model provided by OPNET was used for the simulations. Also, each simulation is executed five times, and the results of the five runs are averaged out.

The absolute end-to-end delay for both the web switch solution and the IP-in-IP tunnel-based solution is shown in Figure 13(a). On the other hand, Figure 13(b) depicts the end-to-end delay percentage increase for both the web switch solution and the IP-in-IP tunnel-based solution. Figure 13(b) shows that the IP-in-IP tunnel-based solution has considerably more end-to-end delay than the web switch solution. This is because the IP-in-IP tunnel-based solution requires more processing time, which is associated with the encapsulation and the decapsulation of the tunnel traffic. Moreover, Figure 13(b) shows that for both solutions, the end-to-end delay percentage increase for the high traffic load scenario is smaller than the low traffic load scenario. The reason is that the processing delay for high traffic becomes less significant than the queuing delay. Hence, the end-to-end delay percentage increase caused by the web switch and the tunneling delays is smaller.

The other measure of performance considered in the comparison is throughput. Figure 14 illustrates the throughput percentage decrease caused by the introduction of a web switch and tunneling. Figure 14 shows that the IP-in-IP tunnel-based solution has more percentage of throughput decrease than the web switch solution. This is because the IP-in-IP tunnel-based solution introduces extra overhead to each packet that enters the tunnel as compared with the web switch solution packets.

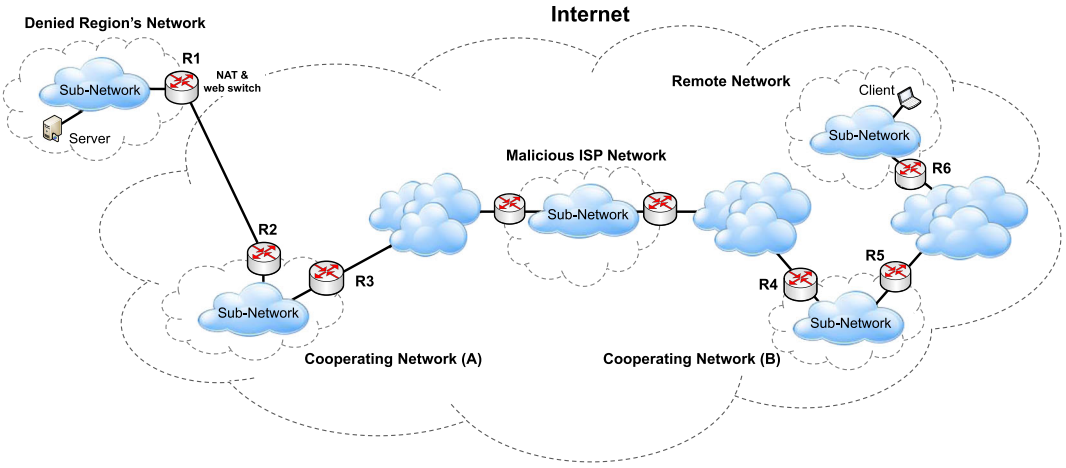


Figure 12. Network setup of the web switch solution.

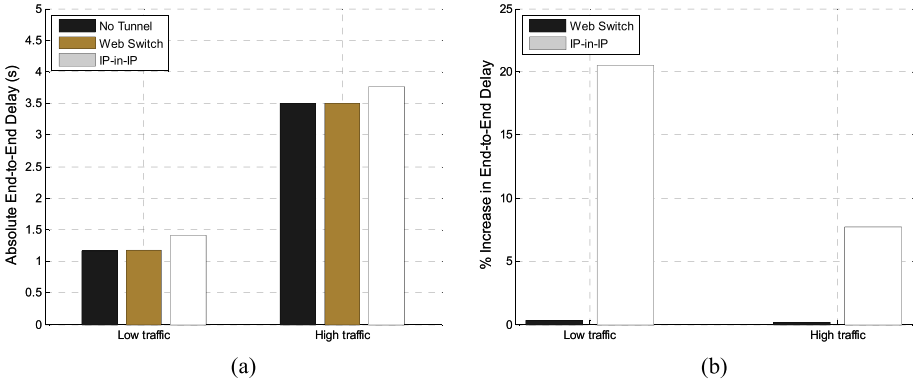


Figure 13. End-to-end delay for hypertext transfer protocol traffic.

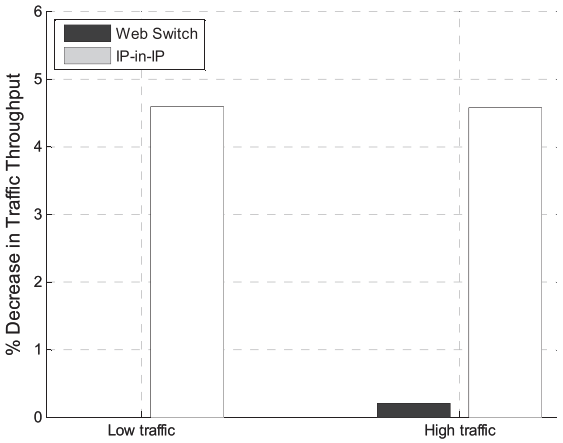


Figure 14. Percentage decrease of throughput for hypertext transfer protocol traffic.

Although the simulation results show that the web switch solution has insignificant impact on the end-to-end delay and on the traffic throughput, the web switch solution is not readily deployable. Furthermore, the web switch solution may not be easily adapted for applications other than HTTP. In contrast, because the relative impact of the IP-in-IP tunnel on the network end-to-end delay would be even smaller for massive networks with high traffic volumes, the IP-in-IP tunnel-based solution provides for an acceptable alternative solution to overcome the server reachability problem.

Accordingly, a combined solution of the NAT-based solution and the tunnel-based solution is recommended. When traffic is originated from within the denied region's network and destined outside the denied region's network, then the NAT-based solution should be used. This is because the NAT-based solution has a very insignificant impact on the network performance as compared with the tunnel-based solution. On the other hand, the tunnel-based solution should be used for traffic originating outside the denied region's network and is destined for servers within the denied region's network. Although the performance of the web switch solution is better than the performance of the tunnel-based solution, the tunnel-based solution is preferred in this case because it is readily deployable, and it is transparent to all network applications. The recommended combined solution network setup is shown in Figure 15. Note that scalability can be also extended to the combined solution in a similar fashion to what was carried out in Figure 3.

7. COMPARISON WITH BLACKHOLING RELATED WORK

Malicious blackholing occurs when traffic destined to a specific prefix is directed to a particular router through the use of incorrect BGP UPDATE messages that advertise false routes, and then, the router intentionally drops that traffic [24]. Accordingly, the Internet access denial problem can be thought of as a malicious blackholing whose intention is to drop traffic destined to or originated from a specific prefix. Consequently, it is important to compare the proposed combined approach presented in this paper against the proposed blackholing countermeasures found in the literature.

In general, the blackholing countermeasures can be divided into two main categories: countermeasures that aim to secure the communication channel's routing protocol and countermeasures that target hiding the identity of the transmitted data. The following subsections present a number of the blackholing countermeasures for each of the two categories while comparing such countermeasures against the proposed combined approach.

7.1. Countermeasures to secure the routing protocol

Most of the blackholing countermeasures that focus on securing the communication channel's routing protocol target the removal of invalid routes that could lead to blackholing. For example, route filtering [25] and Secure BGP (S-BGP) [26] are considered to be among the major proposed countermeasures that fall under this category.

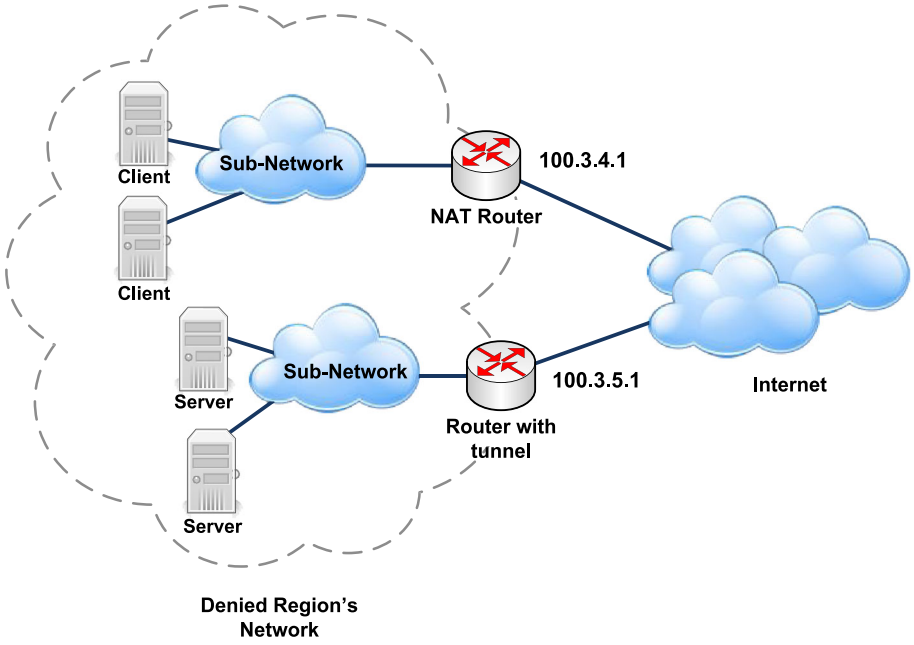


Figure 15. Recommended combined solution network setup.

The route filtering approach is achieved by creating access control lists (ACLs) of prefixes or ASes. The ACLs are then used by the router when sending or receiving BGP UPDATE messages. Outgoing UPDATE messages will be filtered according to the ACLs so as to control which routes are announced to peers. Similarly, the ACLs are used to filter incoming UPDATE messages by verifying that the origin AS of a route truly owns the corresponding prefix so as to prevent blackholing. To perform the filtering, ISPs must know the owner of each address block, which can be obtained from the Internet routing registries. Unfortunately, the Internet routing registries databases are often not updated [27], and ISPs do not query them frequently enough. In addition, a change in either the AS topology or the ASes routing policies that is not reflected accordingly in the ACLs can cause either correct routes to be dropped or fake routes to be allowed.

On the other hand, S-BGP is designed to protect routers from erroneous or malicious UPDATE messages by incorporating strong authorization and authentication capabilities to BGP based on public key cryptography. More specifically, S-BGP introduces a public key infrastructure in the interdomain routing infrastructure to authorize prefix ownership and validate routes. In addition, S-BGP introduces a new attribute to UPDATE messages whose goal is to ensure the authorization of routing UPDATE messages and to prevent route modifications by intermediate S-BGP routers. Finally, S-BGP can use IPsec for all routing messages if routing confidentiality is needed.

Secure BGP can play a significant role in securing the routing infrastructure against blackholing but only if widely deployed. However, wide deployment of S-BGP is hampered by the fact that many ISPs are reluctant to adopt S-BGP because it requires the presence of a hierarchical public key infrastructure and distribution system. Another issue preventing the wide deployment of S-BGP is the performance overhead resulting from the requirement to verify and sign each UPDATE message by each S-BGP router it goes through. Accordingly, if an initialization or a reboot of a BGP peering session takes place, then a large number of routes will be received by the affected S-BGP routers in a short time interval. This will result in a significant amount of performance overhead that will be observed by the affected S-BGP routers. An additional obstacle facing the wide deployment of S-BGP is the large memory space needed by each S-BGP router to store the public keys needed for route verifications. The space requirement can be significant for an S-BGP router with large number of peers [28]. Moreover, simulations of S-BGP show that path convergence times would increase by as much as double through adoption of S-BGP [29]. As such, S-BGP has not been widely deployed so far [30].

Variants of S-BGP have been proposed to resolve the aforementioned issues. These variants include secure origin BGP [31] and interdomain route validation [32]. However, ISPs perceive both S-BGP and its variants as adding additional complexity, infrastructure, and cost to the network and could potentially affect convergence [30]. Hence, so far, the only solutions deployed in the Internet in wide use are the use of route filtering [2].

In comparison, the proposed combined approach presented in this paper is able to deliver traffic to the destination even with the presence of erroneous or malicious UPDATE messages as long as the sender has a working path to the destination. Hence, the proposed combined approach is more resilient against blackholing than the

route filtering approach and more readily available for deployment when compared against S-BGP and its variants.

7.2. Countermeasures to hide the identity of transmitted data

The blackholing countermeasures under this category can be subdivided further into two subcategories: countermeasures that depend on the presence of a peer-to-peer (P2P) overlay network and countermeasures that utilize the existing Internet infrastructure.

Examples of the countermeasures that depend on the presence of a P2P network include the onion routing [33,34] where each peer acts as an onion router. Subsequently, each source host encrypts the message it wants to transmit multiple times with different encryption public keys obtained from the onion routers that are used to route the message. Encrypting the message with different keys preserves the identity of the source and the destination of the message as well as the secrecy of the path through which the message was routed. Moreover, intermediate onion routers are not aware of the content of the message, its original source or its final destination. A similar approach is followed by Tarzan [35], MorphMix [36], Cashmere [37], Crowds [38], and Hordes [39] where messages are repeatedly encrypted initially and then decrypted layer-by-layer at the intermediate routers.

The performance degradation of implementing such a solution is very high [40,41]. The solution would require a number of routers on the Internet that support the deployed anonymous routing protocol. In addition, a large cryptographic overhead is added to each router. Moreover, the number of hops that the message would traverse increases the end-to-end delay. In contrast, the proposed combined approach presented in this paper utilizes the existing Internet infrastructure and standard protocols to circumvent blackholing. Consequently, the performance degradation of the combined approach is lower than that of the countermeasures that depend on the presence of a P2P network.

On the other hand, availability centric routing (ACR) [42] is an example of the countermeasures that utilize the existing Internet infrastructure. The central philosophy behind ACR is that the delivery of traffic can take place as long as the sender is able to find a working path to the valid destination given that such a path exists. Hence, it is not necessary to secure the communication channel's routing protocol to guarantee the delivery of traffic between the sender and the destination. ACR utilizes transit ASes to keep track of multiple routes to each destination and to monitor the availability of such routes. When a route to a destination becomes unavailable due to blackholing, for example, it is removed from the list of the possible paths to that destination. The available routes are provided to the sender on a request basis so the sender can select a path for delivering the traffic to the destination. Once a path is selected, the sender encrypts the traffic before transmitting it over the selected path. Encrypting the traffic protects the identity of the source and the destination of the traffic.

The proposed combined approach presented in this paper shares a number of similarities with the ACR approach. Specifically, both approaches attempt to deliver the traffic over a working path while hiding the identity of the source and the destination of the traffic. Nevertheless, one key difference between the two approaches is related to the way routes to a destination are selected. The ACR approach collects available routes to a destination that goes around a blackholing AS while excluding all routes that pass through the blackholing AS. On the other hand, the proposed combined approach uses any available route to a destination even if the route passes through a blackholing AS. Because there can be many destinations that are served by only one higher-tier AS, the proposed combined approach has a major advantage over ACR in cases where the higher-tier AS is the source of the blackholing.

8. CONCLUSIONS

The Internet access denial problem is a malicious activity that is caused by a higher-tier ISP to force an Internet blockade against the network of a specific region. This paper presented a combined solution to countermeasure the denial of the Internet access problem by considering two traffic identity hiding techniques: NAT and tunneling. The paper further discussed the performance of the tunnel-based technique of the combined solution by considering the network end-to-end delay and the traffic throughput and compared it against the performance of the web switch option of the NAT-based technique. It was shown that the web switch option of the NAT-based technique has better performance than the tunnel-based technique. However, the web switch option of the NAT-based technique requires further development and, therefore, makes it not readily available. Accordingly, it is recommended that when traffic is originated from within the denied region's network and destined outside the denied region's network, then the NAT-based technique should be used. In contrast, the tunnel-based technique should be used for traffic originating outside the denied region's network and is destined for servers within the denied region's network.

ACKNOWLEDGEMENT

The author acknowledges the support provided by King Abdulaziz City for Science and Technology (KACST) through the Science & Technology Unit at King Fahd University of Petroleum & Minerals (KFUPM) for funding this work through project number 08-INF97-4 as part of the National Science, Technology and Innovation Plan.

REFERENCES

1. Kranakis E, van Oorschot PC, Wan T. On inter-domain routing security and pretty secure BGP (psBGP). Carleton University, School of Computer Science, Technical Report TR-05-08, September 2005.
2. Butler K, Farley T, McDaniel P, Rexford J. A survey of BGP security issues and solutions. *IEEE/ACM Transactions on Networking* January 2010; **98**: 100–122.
3. Mao Z, Rexford J, Wang J, Katz R. Towards an accurate AS-level traceroute tool. *Proceedings of the 2003 conference on Applications, technologies, architectures, and protocols for computer communications*, Germany, August 2003; 365–378.
4. Abu-Amara M, Al-Baiz A, Mahmoud A, Sqalli M, Azzedin F. A scalable NAT-based solution to Internet access denial by higher-tier ISPs. *Journal of Security and Communication Networks* 2012, John Wiley, **6**(2):194–209, February 2013.
5. Mahmoud A, Alrefai A, Abu-Amara M, Sqalli M, Azzedin F. Qualitative analysis of methods for circumventing malicious ISP blocking. *Arabian Journal for Science and Engineering* 2012; **37**(7):1911–1928.
6. Quoitin B. BGP-based interdomain traffic engineering. *Ph.D. Dissertation*, Universite catholique de Louvain, Louvain-la-Neuve, Belgium, August 2006.
7. Quoitin B, Bonaventure O. A cooperative approach to interdomain traffic. *Proceedings of the 1st Conference on Next Generation Internet Networks Traffic Engineering*, Rome, Italy, 2005.
8. Quoitin B, Pelsser C, Swinnen L, Bonaventure O, Uhlig S. Interdomain traffic engineering with BGP. *IEEE Communications Magazine* May 2003; **41**(5): 122–128.
9. Atkinson R. Security architecture for the Internet protocol. RFC 1825, Internet Engineering Task Force, August 1995.
10. Farinacci D, Li T, Hanks S, Meyer D, Traina P. Generic Routing Encapsulation (GRE). RFC 2784, Internet Engineering Task Force, March 2000.
11. Simpson W, Daydreamer. IP in IP tunneling. RFC 1853, Internet Engineering Task Force, October 1995.
12. Perkins C. IP encapsulation within IP. RFC 2003, Internet Engineering Task Force, October 1996.
13. Egevang K, Francis P. The IP network address translator (NAT). RFC 1631, Internet Engineering Task Force, May 1994.
14. Rekhter Y, Moskowitz B, Karrenberg D, Groot G, Lear E. Address allocation for private Internets. RFC 1918, Internet Engineering Task Force, February 1996.
15. Rekhter Y, Li T, Hares S. A border gateway protocol 4 (BGP-4). RFC 4271, Internet Engineering Task Force, January 2006.
16. Route Views. http://www.routeviews.org/. Accessed 25 April, 2013
17. OPNET Modeler. http://www.opnet.com/. Riverbed Technology, San Francisco, California, USA. Accessed 1 August, 2012.
18. Danielyan E. Goodbye DES, welcome AES. *The Internet Protocol Journal* June 2001; **4**(2): 15–21, Cisco.
19. Frankel S, Glenn R, Kelly S. The AES-CBC cipher algorithm and its use with IPsec. RFC 3602, Internet Engineering Task Force, September 2003.
20. National Institute of Standards and Technology (NIST). Advanced encryption standard (AES). Federal Information Processing Standard (FIPS) publication 197, November 2001.
21. Perkins C. Minimal encapsulation within IP. RFC 2004, Internet Engineering Task Force, October 1996.
22. Cisco IPsec and SSL VPN Solutions Portfolio. Cisco, 2008.
23. Fielding R, Gettys J, Mogul J, *et al*. Hypertext transfer protocol - HTTP/1.1. RFC 2616, Internet Engineering Task Force, June 1999.
24. Nordström O, Dovrolis C. Beware of BGP attacks. *ACM SIGCOMM Computer Communication Review* April 2004; **34**(2): 1–8.
25. Caesar M, Rexford J. BGP routing policies in ISP networks. *IEEE Network* November–December 2005; **19**(6): 5–11.
26. Kent S, Lynn C, Seo K. Secure border gateway protocol (S-BGP). *IEEE Journal on Selected Areas in Communications* April 2000; **18**(4): 582–592.
27. Mahajan R, Wetherall D, Anderson T. Understanding BGP misconfiguration. *ACM SIGCOMM Proceedings of the 2002 conference on Applications, technologies, architectures, and protocols for computer communications*, August 2002; 3–16.
28. Kent S, Lynn C, Mikkelson J, Seo K. Secure border gateway protocol (S-BGP) real world performance and deployment issues. *Proceedings of the Symposium on Network and Distributed System Security*, San Diego, CA, February 2000; 1–14.
29. Nicol D, Smith S, Zhao M. Evaluation of efficient security for BGP route announcements using parallel simulation. *Simulation Modelling Practice and Theory* July 2004; **12**(3–4): 187–216, Elsevier.
30. Meyer D, Partan A. BGP security, availability, and operator needs. Presentation at NANOG-28 meeting, June 2003.
31. White R. Securing BGP through secure origin BGP. *The Internet Protocol Journal* September 2003; **6**(3): 15–22, Cisco.
32. Goodell G, Aiello W, Griffin T, Ioannidis J, McDaniel P, Rubin A. Working around BGP: an incremental approach to improving security and accuracy of

interdomain routing. *Proceedings of The 10th Annual Network and Distributed System Security Symposium*, San Diego, CA, February 2003; 75–85.
33. Goldschlag D, Reed M, Syverson P. Onion routing. *Communications of the ACM* February 1999; **42**(2): 39–41.
34. Han J, Liu Y, Xiao L, Ni L. A mutual anonymous peer-to-peer protocol design. *Proceedings of the 19th IEEE International Symposium on Parallel and Distributed Processing*, April 2005; 68–77.
35. Freedman M, Sit E, Cates J, Morris R. Introducing tarzan, a peer-to-peer anonymizing network layer. *Peer-to-Peer Systems, Lecture Notes in Computer Science* 2002; **2429**:121–129.
36. Rennhard M, Plattner B. Introducing MorphMix: peer-to-peer based anonymous Internet usage with collusion detection. *Proceedings of the 2002 ACM workshop on Privacy in the Electronic Society*, 2002; 91–102.
37. Zhuang L, Zhou F, Zhao BY, Rowstron A. Cashmere: resilient anonymous routing. *Proceedings of the 2nd conference on Symposium on Networked Systems Design & Implementation* 2005; **2**:301–314.
38. Reiter M, Rubin A. Anonymous Web transactions with crowds. *Communications of the ACM* 1999; **42**(2): 32–48.
39. Shields C, Levine BN. A protocol for anonymous communication over the Internet. *Proceedings of the 7th ACM conference on Computer and communications security*, Athens, Greece, 2000; 33–42.
40. Syverson P, Reed M, Goldschlag D. Anonymous connections and onion routing. *IEEE Journal on Selected Areas in Communications* May 1998; **16**(4): 482–494.
41. Liu J, Kong J, Hong X, Gerla M. Performance evaluation of anonymous routing protocols in MANETs. *Proceedings of the 2006 IEEE Wireless Communications and Networking Conference*, Vol. **2**, April 2006; 646–651.
42. Wendlandt D, Avramopoulos I, Andersen D, Rexford J. Don't secure routing protocols, secure data delivery. *Proceedings of the 2006 ACM SIGCOMM Workshop on Hot Topics in Networking*, November 2006; 1–6.