Journal of Computer and Communications, 2014, 2, 10-21
Published Online June 2014 in SciRes. http://www.scirp.org/journal/jcc
http://dx.doi.org/10.4236/jcc.2014.28002

Scientific Research

Practical Security Approaches against Border Gateway Protocol (BGP) Session Hijacking Attacks between Autonomous Systems

Stephen Brako Oti[1], James Ben Hayfron-Acquah[2]

[1]Information Technology Department, Methodist University College, Accra, Ghana
[2]Computer Science Department, Kwame Nkrumah University of Science & Tech., Kumasi, Ghana
Email: stephen.brako.oti@gmail.com, jbha@yahoo.com

Received 22 January 2014; revised 20 February 2014; accepted 28 February 2014

Abstract

The border gateway protocol (BGP) is the default inter domain routing protocol used on the internet for exchanging information between autonomous systems. Available literature suggests that BGP is vulnerable to session hijacking attacks. There are a number of proposals aimed at improving BGP security which have not been fully implemented. This paper examines a number of approaches for securing BGP through a comparative study and identifies the reasons why these proposals have not been implemented commercially. This paper analyses the architecture of internet routing and the design of BGP while focusing on the problem of BGP session hijacking attacks. Using Graphical Network Simulator 3 (GNS-3), a session hijack is demonstrated and a solution which involves the implementation of route filtering, policy-maps and route-maps on CISCO routers representing ASes is carried out. In the end, a workable industry standard framework for securing and protecting BGP sessions and border routers from exploitation with little or no modification to the existing routing infrastructure is demonstrated.

Keywords

Inter-Domain Routing, Session Hijacking, Bgp Security, Autonomous Systems

1. Introduction

The internet is a global decentralized network of networks comprised of end systems that originate or receive IP

How to cite this paper: Oti, S.B. and Hayfron-Acquah, J.B. (2014) Practical Security Approaches against Border Gateway Protocol (BGP) Session Hijacking Attacks between Autonomous Systems. *Journal of Computer and Communications*, **2**, 10-21. http://dx.doi.org/10.4236/jcc.2014.28002

packets usually identified by IP addresses.

The internet started off as a US Department of Defense (DoD) network to connect scientists and university professors around the world [1]. The internet has transformed the computer and the communications world like nothing before: providing opportunity for worldwide broadcasting, mechanisms for information dissemination and a medium for collaboration and interaction between individuals and their computers without regard for geographic location. The internet consists of thousands of autonomous systems (AS) each owned and operated by a single institution [2]. Hence the internet can be said to be a conglomeration of autonomous systems that define the administrative authority and routing policies of different organizations. Autonomous systems are made up of routers that run interior gateway protocols such as Routing Information Protocol (RIP), Enhanced Interior Gateway Routing Protocol (EIGRP), Open Shortest Path First (OSPF), Intermediate system-Intermediate System (IS-IS) within their boundaries and interconnect via an Exterior Gateway Protocol. The current internet de facto standard EGP is the Border Gateway Protocol Version 4 (BGP-4) defined in RFC 1771 on 4 March 1995 by Rekhter *et al.* [3] and revised in RFC 4271 on 4 January 2006 by Rekhter *et al*. in 2006 [4]. Exterior Gateway Protocols such as BGP logically binds the ASes that make up the internet together by providing a mechanism for BGP peers to exchange route information. BGP unfortunately possesses some fundamental vulnerability that could be exploited to carry out different forms of attack capable of destabilizing the Internet. This paper sheds light on a number of proposals set forth towards addressing numerous other BGP problems and also attempts to propose a solution to BGP session hijacking by simulating a BGP session hijack and a countermeasure using GNS-3 (Graphical Network Simulator) simulator.

Session hijacking

Session hijacking is when an attacker places himself in between the source device and the destination device. This is also known as the man in the middle attack.

BGP operates on trust. BGP speakers themselves inject bogus routing information either by masquerading as any other legitimate BGP speaker or by distributing unauthorized routing information as themselves. Hypothetically, we postulate that the problem of BGP session hijacking could be effectively mitigated through the strict enforcement of already known industry's best practices while utilizing the already deployed routing infrastructure. The solution we believe doesn't rest with overhead ridden protocol extensions such as Secure BGP (sBGP), Pretty Secure BGP (psBGP), Secure Origin BGP (soBGP) and a host of others, all of which rely on the use of additional layers of authentication and encryption. The overall effect is a protocol that is not commercially feasible due to the capacity of the existing deployed infrastructure which in most cases is not able to handle the sort of such overhead protocol extensions placed on it. In place of the several other protocols put forward for securing BGP sessions, we believe that the solution ensures that up streams (typically ISPs) of various ASes verifing their downlinks, *i.e.* the routers advertising routes through them, actually own the prefixes they are announcing. These uplinks must then set up filters to ensure that their downlinks are only allowed to advertise the routes that they own and nothing else. To buttress our view point, we design and implement simulations whose conclusions support our hypothesis. The simulations are implemented in Graphical Network Simulator (GNS-3) running standard industry deployed CISCO devices.

2. Related Work

2.1. Current Proposals for Securing BGP

We analyze some major proposals aimed at securing BGP through a comparative analysis of a number of tools and approaches available for securing BGP-pointing out their strengths and weaknesses; while bringing to light the relevance of the intended research.

2.2. Tools for Securing BGP

A number of mechanisms for securing BGP have been developed which begin at the session level and also includes the tools that are used to protect the TCP session at both the sending and receiving end. According to Gill *et al.* [5] the TTL security mechanism is one such proposal that could substantially limit the effective radius of potential attack on the session. There are two tools to protect the BGP TCP session from external disruption that rely on the use of a cryptographic function.

These are the use of IPSEC at the IP level proposed by Kent *et al*. [6] and the TCP MD5 signature option at

the TCP session level proposed by Rivest [7] and revised by Hefferman [8].

The MD5 signature option has some potential weaknesses when compared with IPSEC based on the assessment of Murphy [9] however the MD5 signature option is preferable to no form of TCP protection at all. The choice between IPSEC and MD5 is made by considering their key relative capabilities. No standard key rollover mechanism exists in MD5 as asserted by Behringer [10] alongside the cryptographic processing load it comes with; whereas the load of IPSEC processing is significantly higher than MD5 processing.

The cryptographic validation requirement of these two mechanisms provides room for a potential denial of service threat where a BGP speaker could be flooded with invalid messages each of which must be cryptographically processed before being detected as invalid and discarded [11]. In addressing the message integrity limitation, an approach is suggested by Schneier [12] which aims to provide transparent session level protection through the use of digital signatures. By this mechanism, a set of credentials is assigned that allows peers to verify the correctness of the information carried as the message payload in BGP.

The reason for the use of digital signatures instead of an integrity check which uses some form of shared secret key is due to the fact that the number and identities of all external recipients of the information is not known in advance [9].

Apart from being able to determine whether or not a message had been altered en route to the destination, a mechanism to actually verify the authenticity of the original information is necessary.

This meant that the digital signatures used had to be verified thus using some form of mechanism that authenticates the public key associated with an address prefix or an AS number [13].

2.3. Approaches for Securing BGP

A very significant contribution to this area is the secure BGP (SBGP) proposal by Kent *et al.* [14]. This happens to be one of the most complete contributions in this direction despite the fact that the assumptions relating to the processing capabilities of the routing equipment needed to run the protocol far exceeds what is available in real life.

2.3.1. Secure Border Gateway Protocol (sBGP)

The sBGP protocol places digital signatures over the address and AS path information contained in routing advertisements and defines an associated PKI for validation of these signatures. sBGP defines the correct operation of a BGP speaker in terms of constraint placed on individual protocol messages, including ensuring that all protocol UPDATE messages have not been allowed in transit between the BGP peers and that the UPDATE messages were sent by the peer is indicated. The basic security framework proposed in sBGP is that of digital signatures thus x.509 certificates and PKI's. This enables BGP speakers to identify and authorize other BGP speakers as well as AS administrators and address prefix owners. The verification framework for sBGP requires a PKI for address allocation, where every address assignment is reflected in an issued certificate [14]. In addition, sBGP proposes the use of IPSEC to secure the inter-router communication paths as well as the use of attestations. An address attestation is produced by an address holder, and authorizes a nominated AS to advertise itself as the origin AS for a particular address prefix.

There are a number of significant issues that have been identified with sBGP including the computation burden for signature generation and validation as well as the increased load in BGP session restart. There is also the issue of piecemeal deployment and the completeness of route attestations [15].

2.3.2. Secure Origin Border Gateway Protocol (SoBGP)

A refinement to the sBGP approach is secure origin BGP (SoBGP) proposed by White [16] in an effort to find a middle ground between the additional security processing overhead and the capabilities of deployed routing systems and security infrastructure. Here, the requirements for AS path verification are relaxed and the nature of the related Public Key Infrastructure is altered to remove the requirement for a strict hierarchical address PKI that precisely reflects the address distribution framework. The overall approach proposed in soBGP represents a different set of design trade-offs to sBGP, where the amount of validated material is a BGP UPDATE message is reduced. This can reduce the processing overhead for validation of UPDATE messages. In soBGP each local BGP speaker assembles a validated inter-AS topology map as it collects AS PolicyCerts, and each AS path in UPDATE messages is then checked to see if the AS sequence matches a feasible inter-AS path in this map. The

avoidance of a hierarchical PKI for the validation of AuthCerts and EntityCerts could be considered a weakness in this approach, as the derivation of authority to speak on addresses is very unclear in this model.

2.3.3. Pretty Secure BGP (PSBGP)

Another refinement of the SBGP model is pretty secure BGP (psBGP) proposed by Oorschot *et al.* [17]. This approach represents a similar effort aimed at achieving a compromise between security and deployed capability through the introduction of a trust rating for assertions based on assessment of confidence in corroborating material. psBGP puts forward the proposition that the proposals relating to the authentication of the use of an address in a routing context must either entirely rely on the use of signed attestation that need to be validated in the context of PKI, or rely on the authenticity of information contained in Internet Routing Registries. The weakness of routing registries is that the commonly used controls in the registry are insufficient to validate the accuracy or the current authenticity of the information that is represented as being contained in a route registry object. psBGP allows for partial path signature to exist, mapping the validation outcome to a confidence level rather than a more basic BGP model of accepting an AS path only if the AS path in the BGP UPDATE completely verifiable. The essential approach of psBGP is the use of a reputation scheme in place of a hierarchical address PKI, but the value of this contribution is based on accepting the underlying premise that a hierarchical PKI for addresses is feasible. psBGP appears to be needlessly complex and bears much of the characteristics of making a particular solution for the problem, rather than attempting to craft a solution within the bounds of the problem space.

2.3.4. Interdomain Route Validation (IRV)

Another technique, inter-domain route validation (IRV) proposed by Goodell *et al.* [18] attacks the problem from a different angle by extending the existing model of Internet Route Registries into per-AS route registries. It attempts to replace the configuration of the BGP protocol with security credentials, in a query based credential retrieval system. The approach assesses the security function as an incremental overlay on the existing routing infrastructure.

This approach is midway between the strict AS path test of sBGP that validates that the UPDATE message was passed along the AS sequence described in the AS Path and the soBGP AS Path feasibility that validates that there is a set of AS peer connections that correspond to the AS sequence. Here the validation test is that each AS in the sequence is currently advertising this prefix to the next AS in sequence.

This IRV architecture has a number of issues that are not completely specified, including IRV discovery, IRV query redirection, authentication of queries and responses, selective responses, transparent layer protection and imposed overheads. It is unclear how an IRV response is to be validated, and how the relying party can verify that the received response originated from the IRV server of the AS in question, that the response has not been altered in any way, and that the response represents the actual held state in the queries AS. A similar concern lies in the estimation of additional overhead associated with performing a query to each AS in the AS Path for every received BGP update. It is also unspecified whether the query and response is a pre-condition for acceptance of a route would appear to offer a route robust form of security; it is also the case that IRV would be unreachable until the route is accepted.

There is no clear cut solution to the problem of routing security that attains a balance between security and acceptable deployment overhead [19]. Current research on BGP security focuses on the integrity, authenticity, and verifiability of routing information [20]. A more stable routing system capable of providing stable routing state is also capable of verifying routing information updates.

We believe the solution to BGP session hijacking does not necessarily rest with protocol extensions or modifications but rather the implementation of policy based routing such as the use of filters, route-maps and policy maps as well as a number of already known best practices in the service provider industry. This approach will require no additional hardware but what exist already and should guarantee a level of BGP session security at service provider level if implementation is efficiently done. Unlike the earlier suggested approaches to securing BGP, this approach focuses on the optimization of BGP configuration using Defensive Routing Policies from Cisco IOS to create a more robust BGP session better equipped to forestall session hijacking attacks while taking into consideration the processing capabilities of the existing routing infrastructure or equipment in other to prevent undue overhead associated with the implementation of other proposals for securing BGP.

3. Session Hijacking Simulation

Session hijacking involves intrusion into an ongoing BGP session by masquerading as a legitimate peer in a BGP session [21]. The difference as compared to a TCP reset attack is that session hijacking attack may be designed to achieve more than simply bringing down a session between BGP peers. The objective may be to change routes used by a peer or black holing [22]. In EBGP, neighbor routers add all routes they receive from their downstream neighbors into their routing table and then advertise those routes to the next hop. BGP session hijacking could occur when ISPs do not filter advertisements. An attacker could then hijack such an ISP and use their routers to advertise any prefix they want leading to a diversion of traffic or a black hole as the simulations would demonstrate. Session hijacking attacks could result in serious outages culminating in a complete loss of connectivity. Specific instances include the 2008 incident where at least eighty US universities had their traffic diverted to block access to their site from inside the country but accidentally black holed the route in the global BGP table [23]. In studying the session hijacking problem, case studies were developed that feature a number of routers representing ISPs and other autonomous systems connected together through EBGP peering. To successfully carry out the attack, a rogue AS (autonomous System) was included in the case study to advertise legitimate routes to its downstream neighbors creating a masquerade effect that causes downstream neighbors to forward traffic towards the rogue AS with the fake identity.

Routes forwarded to this AS by its downstream neighbors are not forwarded to the next hop address (router) consequently creating a Black Hole.

The purpose of this experiment is to demonstrate a Border Gateway Protocol (BGP) session hijack. To facilitate this demonstration, Graphical Network Simulator 3 (GNS-3) running Cisco routers have been employed.

The experiment consists of three main scenarios.

1) Scenario one—this depicts a normal BGP operation.
2) Scenario two—in this scenario the BGP session hijacking is demonstrated.
3) Scenario three—the third and final scenario shows how this session hijacking can be prevented.

3.1. Scenario One—Normal Operation of BGP

Routers R1 to 7 all represent routers in different Autonomous Systems (ASs), each having multiple point-to-point uplinks through other transit ASs to the global Internet. Each of the afore-mentioned routers is running external BGP (EBGP) and is peering with each directly connected router. The routers that are of particular interest are routers R6 and R7. For the purposes of this lab, R6 will serve as the AS originating a public Internet Protocol (IP) address, 20.20.0.0/19, unto the global Internet. Inside AS 600, a server with a 20.20.20.20/24 address exists. All the routers in the various AS can access this server in AS 600. The BGP routes on each of the routers point to R6 to reach the 20.20.20.20/24 server. This is BGP in normal operation.

3.2. Scenario Two—Session Hijacking Attack

The session hijacking comes into effect when router R7, begins to advertise a 20.20.20.0/21. This is a subnet of the 20.20.0.0/19 being advertised by AS 600 (R6) albeit more specific prefix. The natural tendency of routing protocols to prefer a more specific route kicks in and the routers on the Internet (R1 through to R7) now use AS 700 (R7) as their path to reach the 20.20.20.20/24 IP. This initially creates a “black hole” on the Internet due to the fact that R7 does not actually have a node on its network with the 20.20.20.20/24 IP. To resolve this ‘black hole’ so R7 session hijacking can go unnoticed, R7 makes its advertisements/route for the 20.20.20.0/21 prefix undesirable to router R4 and R6 by prepending their Ass (400 and 600) in its advertisement to the aforementioned routers. This ensures that routers R4 and R6 never use R7 as their primary path to the 20.20.0.0/19 network, but rather R6s path. R7 then uses a static route to point any traffic coming through it and destined for the 20.20.20.20/24 server, to use R4 as its next hop router. This technique allows R7 to quietly receive all traffic meant for the 20.20.20.20/24 server, inspect and or alter it, and then forward it through R4 to its intended destination.

3.3. Scenario Three—Solution

To prevent this particular kind of BGP session hijacking, it is imperative that all the up streams (typically ISPs)

of the various ASes verify that their downlinks, *i.e.* the routers advertising routes through them, actually own the prefixes they are announcing. These uplinks must then setup filters to ensure that their downlinks are only allowed to advertise the routes that they own and nothing else.

In the case of this lab, R2, R8 and R4, the uplink providers of R7 check and implement filters to allow R7 to advertise only the routes that belong to that AS (**Figure 1**).

3.4. Implementation—Scenario One

- Normal BGP peering going on
- R6 in AS 600 has two uplinks to the global Internet, R4 and R5
- R6 owns and is advertising the 20.20.0.0/19 prefix

```
routerbgp 600
no synchronization
bgp log-neighbor-changes
network 20.20.0.0 mask 255.255.224.0
neighbor 21.202.0.1 remote-as 400
neighbor 21.202.0.1 soft-reconfiguration inbound
neighbor 21.202.0.1 route-map RBLOCKDEF out
neighbor 21.202.1.1 remote-as 500
neighbor 21.202.1.1 soft-reconfiguration inbound
neighbor 21.202.1.1 route-map RIN in
neighbor 21.202.1.1 route-map RBLOCKDEF out
no auto-summary
```

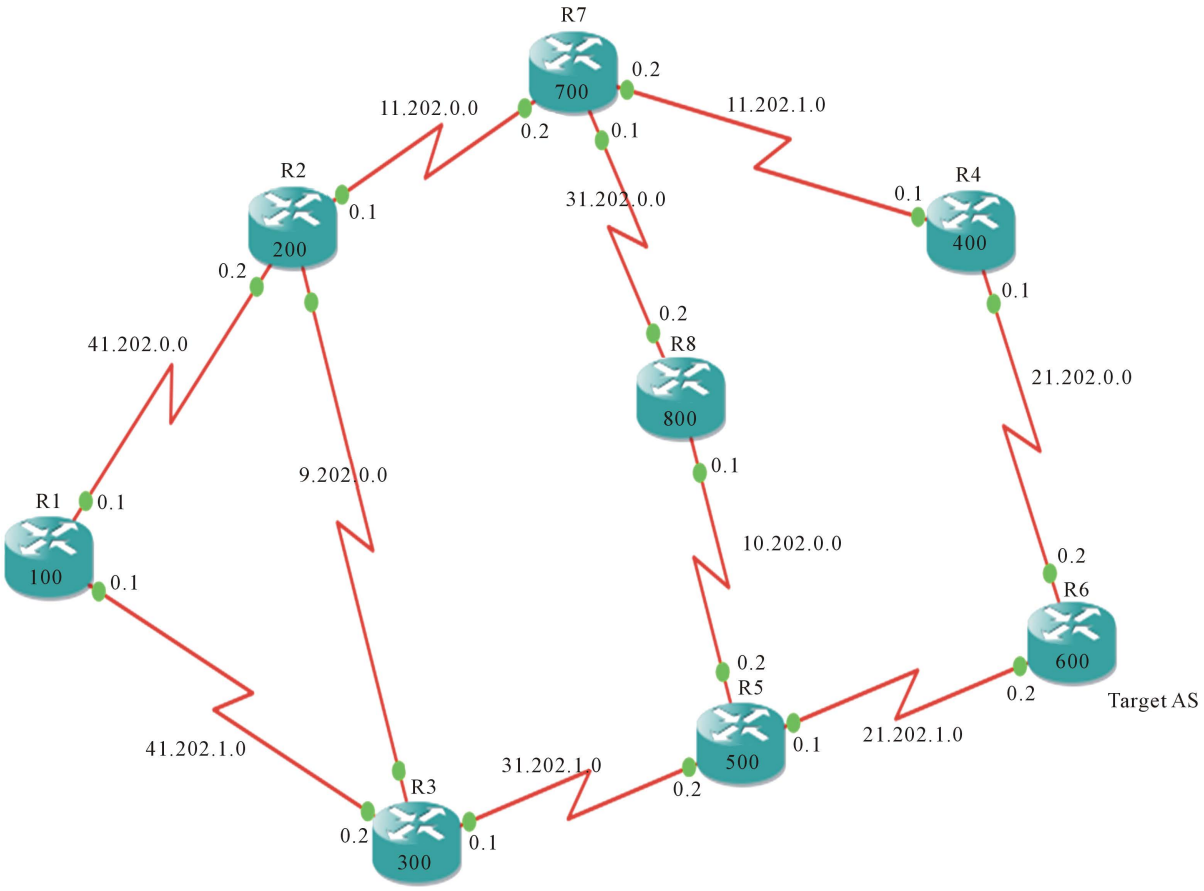

Figure 1. Shows the topology for BGP experiment.

- R6 has an IP/Server 20.20.20.20/24(Simulated with a loopback) that everybody on the Internet can reach, through R6

*Ping test from r*1 *to* 20.20.20.20/24
R1#ping 20.20.20.20
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 20.20.20.20, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 28/50/72 ms
Traceroute to 20.20.20.20/24 *from r*1
R1#traceroute 20.20.20.20
Type escape sequence to abort.
Tracing the route to 20.20.20.20
1 41.202.1.2 20 msec 4 msec 40 msec
2 31.202.1.2 [AS 300] 32 msec 40 msec 16 msec
21.202.1.2 [AS 500] 48 msec* 40msec
Pings to 20.20.20.20 *from r*7
R7#ping 20.20.20.20
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 20.20.20.20, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 32/43/52 ms
*Traceroute from r*7 *to* 20.20.20.20
R7#traceroute 20.20.20.20
Type escape sequence to abort.
Tracing the route to 20.20.20.20
1 11.202.1.1 4 msec 8 msec 28 msec
2 21.202.0.2 [AS 400] 16 msec* 36 msec

- R1 and R7 are learning about the routes from BGP via both R6's uplinks thus R4 and R5 as shown in the traceroute. R1 uses R5 (AS 500) to get to the 20.20.20.20/24Server

Traceroute to 20.20.20.20/24 *from r*1 *ends at r*6 *with* 21.202.1.2 *ip address*
R1#traceroute 20.20.20.20
Type escape sequence to abort.
Tracing the route to 20.20.20.20
1 41.202.1.2 20 msec 4 msec 40 msec
2 31.202.1.2 [AS 300] 32 msec 40 msec 16 msec
3 21.202.1.2[AS 500] 48 msec* 40msec
R7 uses R5 (AS 400) to get to the 20.20.20.20/24Server
*Traceroute from r*7 *to* 20.20.20.20 *ends at r*6 *with* 21.202.0.2 *ip address*
R7#traceroute 20.20.20.20
Type escape sequence to abort.
Tracing the route to 20.20.20.20
1 11.202.1.1 4 msec 8 msec 28 msec
2 21.202.0.2[AS 400] 16 msec* 36 msec

Scenario Two

A Rogue Router, R7, in as 700 begins to advertise a more specific route (20.20.20.0/22) than the 20.20.0.0/19 being advertised by R6. Due to the nature of routing protocols preferring longer/more specific prefixes, all the BGP routers now point to AS 700 to reach the 20.20.20.20/24 server. R7 originally does not have a server with the specified IP so traffic meant for that server coming from the BGP routers all end at R7's AS 700 and get dropped, essentially creating a "blackhole" for the 20.20.20.20/24 IP on the Internet. The "blackhole" is rectified by using an AS prepend to make the R7s path to the 20.20.0.0/19 network and ultimately the 20.20.20.20/24 server undesirable (**Figure 2**).

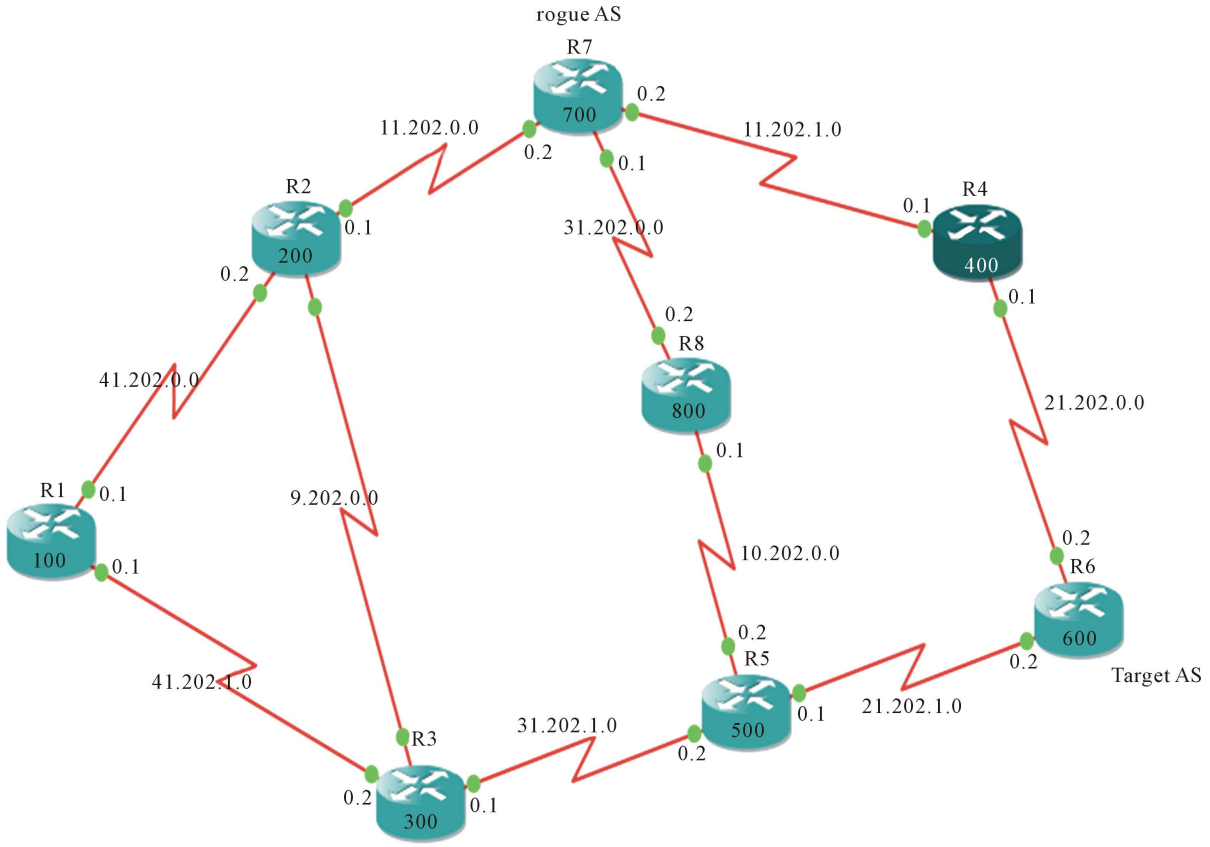

Figure 2. R7 begins to originate a more specific route of 20.20.20.0/22.

R7#sh run|sec router bgp
routerbgp 700
no synchronization
bgp log-neighbor-changes
network 20.20.20.0 mask 255.255.252.0
neighbor 11.202.0.1 remote-as 200
neighbor 11.202.0.1 soft-reconfiguration inbound
neighbor 11.202.0.1 route-map RDEF out
neighbor 11.202.1.1 remote-as 400
neighbor 11.202.1.1 soft-reconfiguration inbound
neighbor 11.202.1.1 route-map RPREPEND out
neighbor 31.202.0.2 remote-as 800
neighbor 31.202.0.2 soft-reconfiguration inbound
no auto-summary

- R1 now tries to get to the 20.20.20.20/24 IP/Server through R7 but is unable to reach the server because R7 does not actually have that IP/Server on its network. A traceroute from R1 below shows R1 changes its route to the R7's uplink R2, AS 200, to get to the 20.20.20.20/24 IP.

Traceroute from r1 to 20.20.20.20/24 ends at r7 11.202.0.2 ip
R1#traceroute 20.20.20.20
Type escape sequence to abort.
Tracing the route to 20.20.20.20
1 41.202.0.2 12 msec 8 msec 36 msec
2 11.202.0.2 [AS 200] 92 msec 20 msec 24 msec
3 11.202.0.2 [AS 200] !H * !H

- Ping test from R1 to the 20.20.20.20/24 ip reports the destination to be unreachable

R1#ping 20.20.20.20
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 20.20.20.20, timeout is 2 seconds:
U.U.U
Success rate is 0 percent (0/5)

- R7 resolves the blackhole by prepending the as 400 and 600 to its 20.20.20.0/22 advertisement to r4 and r6. This makes r7s path to the 20.20.0.0/19 network and ultimately the 20.20.20.20/24 server undesirable to r4 and r6.
- Using the *route-map* and *access control listprepend* and *accdef respectively*, r7 prepends the 400 and 600 ASes to its 20.20.20.0/22 advertisement to r4 and r6.

routerbgp 700
no synchronization
bgp log-neighbor-changes
network 20.20.20.0 mask 255.255.252.0
neighbor 11.202.1.1 remote-as 400
neighbor 11.202.1.1 soft-reconfiguration inbound
neighbor 11.202.1.1 route-map RPREPEND out
no auto-summary
!
ip access-list standard ACCDEF
permit 20.20.20.0 0.0.3.255
!
!
route-map RPREPEND permit 10
matchip address ACCDEF
set as-path prepend 400 600
!
route-map RPREPEND permit 15

- R4 and R6s routing table now show R6s (21.202.0.2) as the best path to the 20.20.0.0/19 network

R4#sh ipbgp 20.20.0.0
BGP routing table entry for 20.20.0.0/19, version 3
Paths: (1 available, best #1, table Default-IP-Routing-Table)
Advertised to update-groups:
1
600, (received & used)
21.202.0.2 from 21.202.0.2 (20.20.20.20)
Origin IGP, metric 0, localpref 100, valid, external, best

R6#sh ipbgp 20.20.0.0
BGP routing table entry for 20.20.0.0/19, version 2
Paths: (1 available, best #1, table Default-IP-Routing-Table)
Advertised to update-groups:
1
Local
0.0.0.0 from 0.0.0.0 (20.20.20.20)
Origin IGP, metric 0, localpref 100, weight 32768, valid, sourced, local, best

- R7 now quietly redirects all traffic passing through it to the 20.20.20.20/24 server by employing a static route to point to r4 (11.202.0.2/30) as the next-hop to the 20.20.20.20/24 server.

Static route on r7 pointing to r4
ip route 20.20.20.0 255.255.252.0 11.202.1.1

- R1 is now able to reach the 20.20.20.20/24 server through R7

Ping test from R1 to the 20.20.20.20/24 server

R1#ping 20.20.20.20
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 20.20.20.20, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 40/62/108 ms
A traceroute from R1 to the 20.20.20.20/24 server
R1#trace 20.20.20.20
Type escape sequence to abort.
Tracing the route to 20.20.20.20
1 41.202.0.2 24 msec 64 msec 8 msec
2 11.202.0.2 [AS 200] 36 msec 28 msec 52 msec
3 11.202.1.1 [AS 400] 48 msec 48 msec 40 msec
4 21.202.0.2 [AS 400] 44 msec* 52 msec

Scenario Three

- To prevent R7 from advertising a route/prefix that does not belong to that AS, the administrators of R7s uplink providers *i.e.* R4, R8 and R2 must verify from the internet registry which prefixes the AS700 can originate or announce. As a proactive measure, R4, R8 and R2 must also implement filtering mechanisms to ensure that R7 only advertises the routes that belong to it.
- In this lab, R7 owns and advertises the 70.70.70.0/24 prefix. After ensuring that R7 can and is allowed to advertise that prefix, R4, R8 and R2 implement the route-map below to filter R7s routes so it can only advertise the 70.70.70.0/24 route. However note that R7 can also choose to advertise a subset of the 70.70.70.0/24 prefix and its upstream providers can alter their filters to accommodate those prefixes.

A filter on R4, R8 and R2 to allow R7 to advertise only the 70.70.0.0/19 prefix
ip access-list standard PERMITVALIDIP
permit 70.70.70.0 0.0.0.255
!
!
route-map PERMITVALIDROUTE permit 10
matchip address PERMITVALIDIP
!
route-map PERMITVALIDROUTE deny 15

- R4, R8 and R2 now only receive the 70.70.70.0/24 advertisement from R7 preventing it from propagating the “illegal” 20.20.20.0/22 route. This effectively stops the BGP session highjack. The only valid route R4 receives from R7 is the 70.70.0.0/19 prefix

R4(config)#do shipbgpnei 11.202.1.2 received-route
BGP table version is 52, local router ID is 31.202.0.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

	Network	Next Hop	Metric	LocPrf	Weight	Path
*>	70.70.70.0/24	11.202.1.2	0		0	700 i

- R1 goes back to its original path to reach the 20.20.20.20/24 ip.

R1#trace 20.20.20.20
Type escape sequence to abort.
Tracing the route to 20.20.20.20
1 41.202.1.2 28 msec 96 msec 28 msec
2 31.202.1.2 [AS 300] 8 msec 56 msec 28 msec
3 21.202.1.2 [AS 500] 84 msec* 96 msec
*compare with traceroute of scenario one.

4. Conclusions

The objective of the above BGP session hijacking simulation is to bring to light the inherent vulnerability of

External BGP (EBGP) sessions and the measures taken to mitigate this vulnerability. A typical EBGP peering between different ASes on the Internet is described. A setting in which a number of ASes serve as uplinks/upstreams, passing on routes from the ASes originating these routes from their local systems, unto the global Internet.

In each of the scenarios, the flexibility of BGP is demonstrated. Naturally, BGPs comprehensive stock of attributes allows routes to be altered, forwarded and dropped with relatively minimal fuss. The widely accepted notion that BGP can be tweaked to behave in many ways and choose any of several paths as it traverses the Internet making it difficult to pinpoint a particular behavior as malicious.

For example, in the above scenarios, R2 could modify the routes it receives from R1, its downstream peer to make them seems like that R2 was originating those routes. This might not necessarily constitute malicious behavior since R2 could do this to make some routes seem preferable to other routes that it receives from other peers.

Again Router, R3 could for instance drop routes originating from R1's AS, but forward every other route. BGPs nature therefore makes policing the protocol to identify inconsistent behavior not entirely straight forward. Concerns have been raised that perhaps it is about a different Inter-domain routing protocol with fewer "quirks" and a more well defined stable set of rules (in a security sense) developed to replace the already ubiquitous BGP. This argument might hold some merits considering the fact that BGPs flexibility is what allows for our attacker (R7) to surreptitiously manipulate routes in such a way as to suite its malicious intentions.

Verifying and filtering the routes advertised by immediate peers can go some way to make BGP a more secure Inter-domain routing protocol. However, this might be an exercise in futility unless all peers thereof participate in this process of verification and filtering of routes. The task of getting every provider to filter routes is daunting if it is not a seemingly improbable quest.

In conclusion, it is imperative to point out that although BGP might be inherently flawed mainly because the propagation of routes by peering ASes is based fundamentally on trust, and this undoubtedly raises several levels of security concerns. However, the tradeoff between security and BGPs presently unmatch ability to adapt and traverse myriad paths to re-establish lost connectivity that cannot be ignored. Hence, we believe the current protocol in use can be effectively tweaked to provide an adequate assurance of security without necessitating modifications to the existing routing infrastructure as demonstrated in our simulations.

References

[1] Peter, I. (2012) Origins of the Internet. http://www.nethistory.info/History%20of%20the%20Internet/origins.html

[2] Sachdeva, M., Singh, G. and Kumar, K. (2011) Deployment of Distributed Defence against DDoS Attacks in ISP Domain. *International Journal of Computer Applications*, **15**, 29.

[3] Rekhter, Y. and Li, T. (1995) A Border Gateway Protocol 4 (BGP-4). RFC 1771, the Internet Engineering Task Force (IETF). http://www.ietf.org/rfc/rfc1771.txt

[4] Rekhter, Y. and Li, T. (2006) A Border Gateway Protocol 4 (BGP-4). RFC 4271, the Internet Engineering Task Force (IETF). http://www6.ietf.org/rfc/rfc4271

[5] Gill, V., Heasley, J. and Meyer, D. (2004) The Generalized TTL Security Mechanism (GTSM). RFC 3682 (Experimental), Internet Engineering Task Force, Obsoleted by RFC 5082. http://www.ietf.org/rfc/rfc3682.txt

[6] Kent, S. and Atkinson, R. (1998) Security Architecture for the Internet Protocol. RFC 2401 (Proposed Standard), Internet Engineering Task Force, Obsoleted by RFC 4301, Updated by RFC 3168. http://www.ietf.org/rfc/rfc2401.txt

[7] Rivest, R. (1992) The MD5 Message-Digest Algorithm. RFC 1321 (Informational), Internet Engineering Task Force. http://www.ietf.org/rfc/rfc1321.txt

[8] Heffernan, A. (1998) Protection of BGP Sessions via the TCP MD5 Signature Option. RFC 2385 (Proposed Standard), Internet Engineering Task Force. http://www.ietf.org/rfc/rfc2385.txt

[9] Murphy, S. (2001) BGP Security Analysis. http://tools.ietf.org/html/draft-murphy-bgp-secr-04

[10] Behringer, M. (2007) BGP Session Security Requirements. Internet-Draft (Informational). http://tools.ietf.org/html/draft-behringer-bgp-session-sec-req-02

[11] Christian, B. and Tauber, T. (2008) BGP Security Requirements. Internet-Draft (Informational). http://tools.ietf.org/html/draft-ietf-rpsec-bgpsecrec-10

[12] Schneier, B. (1995) Applied Cryptography: Protocols, Algorithms, and Source Code in C. John Wiley & Sons, Inc.,

New York.

[13] Cooper, D., Santesson, S., Farrell, S., Boeyen, S., Housley, R. and Polk, W. (2008) Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile. RFC 5280 (Proposed Standard), Internet Engineering Task Force. http://www.ietf.org/rfc/rfc5280.txt

[14] Kent, S., Lynn, C. and Seo, K. (2000) Secure Border Gateway Protocol (SBGP). *IEEE Journal on Selected Areas in Communications*, **18**, 582-592. http://dx.doi.org/10.1109/49.839934

[15] Zhao, M., Smith, S. and Nicol, D. (2005) The Performance Impact of BGP Security. *IEEE Journal on Network*, **19**, 42-48. http://dx.doi.org/10.1109/MNET.2005.1541720

[16] White, R. (2003) Securing BGP through Secure Origin BGP. *The Internet Protocol Journal*, **6**.

[17] van Oorschot, P.C., Wan, T. and Kranakis, E. (2007) On Interdomain Routing Security and Pretty Secure BGP (psBGP). *ACM Transactions on Information and System Security*, **10**, Article Number: 11. http://dx.doi.org/10.1145/1266977.1266980

[18] Goodell, G., Aiello, W., Griffin, T., Ioannidis, J., McDaniel, P. and Rubin, A. (2003) Working around BGP: An Incremental Approach to Improving Security and Accuracy of Interdomain Routing. *Proceedings of Internet Society Symposium on Network and Distributed System Security* (*NDSS* 03), San Diego, 6-7 February 2003.

[19] Chan, H.W., Dash, D., Perrig, A. and Zhang, H. (2006) Modeling Adoptability of Secure BGP Protocols. *Proceedings of the Joint International Conference on Measurement and Modeling of Computer Systems*, **36**, 279-290.

[20] Butler, K., Farley, T.R., McDaniel, P. and Rexford, J. (2010) A Survey of BGP Security Issues and Solutions. *Proceedings of the IEEE*, **98**, 100-122. http://dx.doi.org/10.1109/JPROC.2009.2034031

[21] Behringer, M. (2007) BGP Session Security Requirements. Internet-Draft (Informational). http://tools.ietf.org/html/draft-behringer-bgp-session-sec-req-02

[22] IETF, RFC 4272 (2006) BGP Security Vulnerabilities Analysis. http://www.ietf.org/rfc/rfc4272.txt

[23] Docstoc (2012) Border Gateway Protocol.