CHALMERS

BGP Threats and Practical Security

Master of Science Thesis in Networks and Distributed Systems

Akhtar Zeb

Muhammad Farooq

Chalmers University of Technology
University of Gothenburg
Department of Computer Science and Engineering
Gothenburg, Sweden, March 2011

BGP Threats and Practical Security

Examiner:Ali Salehson

Chalmers University of Technology
Department of Computer Science and Engineering
SE-412 96 Gothenburg
Sweden
Telephone + 46 (0)31-772 1000

Abstract

Border Gateway Protocol (BGP) is the routing protocol being used for exchanging path information among routers in the Internet. A smooth service of the Internet depends on BGP but there is much vulnerability in BGP that can be exploited to disrupt the Internet services. BGP is vulnerable to many attacks due to the lack of inherent security measures in its design. Although many protocols are proposed to provide security in BGP, but up-to-date none of them has been implemented in practical world due to deployment issues.

In this thesis, we studied the BGP protocol, possible attacks on BGP and their countermeasures proposed in literature and research. We have designed and implemented case studies defining different attacks and their mitigation in Chalmers Networking Laboratory at Lindholmen, Chalmers campus. BGP is complex protocol; we have studied BGP and its vulnerabilities in detail. We did comparative and analytical study of the security protocols built for BGP in order to explore the reasons for their non-deployment in real world.

We considered frequently used and best practices proposed by large Internet Service Providers (ISPs) to avoid many attacks and problems by employing services like route filtering, route dampening and prefix limiting. A sample BGP network has been built using Cisco equipment available in the lab along with all possible security threats to test the protection provided by practical security measures.

List of Figures

Table of Contents

Chapter 1 Introduction

Internet is the network of computer networks in the world communicating using TCP/IP standard protocols. World Wide Web (WWW) is the mostly used part of the Internet. The Internet has become basic necessity of life in contemporary world and its applications include every aspect of life, whether it is business, education, office, social networking, gaming, dating, banking, booking travel or hotel and searching information about anything one can think.

Originally the Internet was designed with very less care about security, but presently Internet is being used for many critical applications for business such as banking and stock exchange, transport control system and telemedicine that all demand safety, scalability and reliability of the Internet.

1.1 Internet Architecture

Internet is across the whole world. An organization or home user wants to access the Internet will connect to a local Internet Service Provider (ISP). An ISP is someone who is providing global Internet connectivity. In reality the Local ISP is getting Internet services from Regional ISP which in turn is connected to major ISP called National ISP as illustrated in Figure 1-1 below.

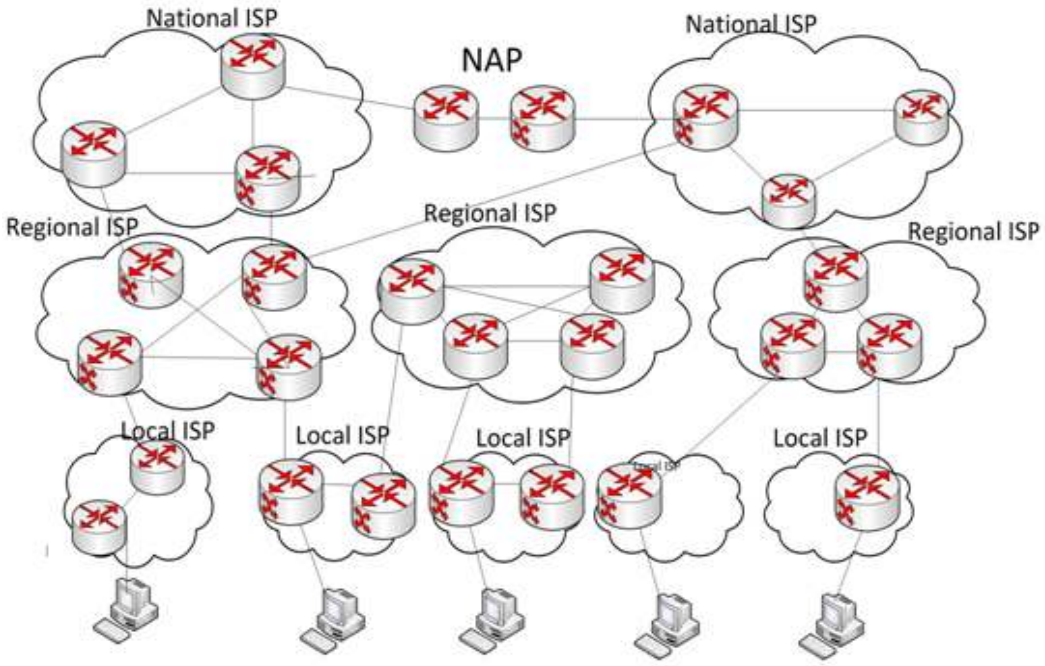

Figure 1-1 Internet architecture

These National ISPs connect to each other through Network Access Points (NAP). Moreover, many ISPs are nowadays directly connected to each other in what is known as ISP peering in order to achieve redundancy and efficiency [1].

One part of the Internet that has common policies and is under single administrative control can be considered as one entity to make the global picture of the Internet simple. This entity comprises a group of routers and computers belonging to one organization and is called Autonomous System (AS) and will be identified by assigning an AS unique number.

Routing protocol is required to exchange routing information dynamically among routers. The Interior Gateway Protocol (IGP) like OSPF, EIGRP or RIP is used to exchange routing information within AS. Exterior Routing Protocol (EGP) like BGP is used for sharing routing information between ASs. The Information, exchanged using routing protocols, is about all reachable networks. Every router in the Internet must have the information about all networks in its routing table to achieve global connectivity.

1.2 Internet Addressing

Every host connected to the Internet must have some name to be identified like every human being and place in the world. The Internet Protocol (IP) defines addressing scheme to be used for Internet. In early years the IP address range was divided in address classes for simplicity and one network from one class is allocated to organization on demand. IP Addressing is managed by the Internet Corporation for Assigned Names and Numbers (ICANN) and the Internet Assigning Number Authority (IANA). Below is the flow of the Internet address assignment

- IANA assigns prefix and AS number range to National ISPs.
- Regional ISPs get their part of the prefix range from National ISPs.
- Regional ISP assigns smaller prefix to Local ISP from the available address range.
- Organization and home user get their IP address from Local ISP.

IP address has two parts, network part and host part. Network part is used to identify different networks based on their classes and used for routing information by routers. In the Classful scheme, the IP address range is divided into different classes known as A, B and C. Each class identifies the fixed number of bits belonging to network and host part in the 32-bit IP address. With the global expansion of Internet it becomes difficult for routers to maintain entries for two millions class C networks due to memory and processing power limitations.

1.3 Classless Interdomain Routing (CIDR)

CIDR, a new enhanced, classless addressing and routing scheme is therefore introduced in 1993 to replace the original IP addressing scheme. With CIDR for example many contiguous Classful networks may be combined in one larger classless network represented by a single prefix. The prefix will be associated with prefix length corresponding to the

number of bits in subnet mask. The prefix of a supernet will have less number of bits in the subnet mask than the original combined Classful networks. In this way CIDR scheme reduces entries in routing tables of the national ISP routers as shown in figure 1-2.

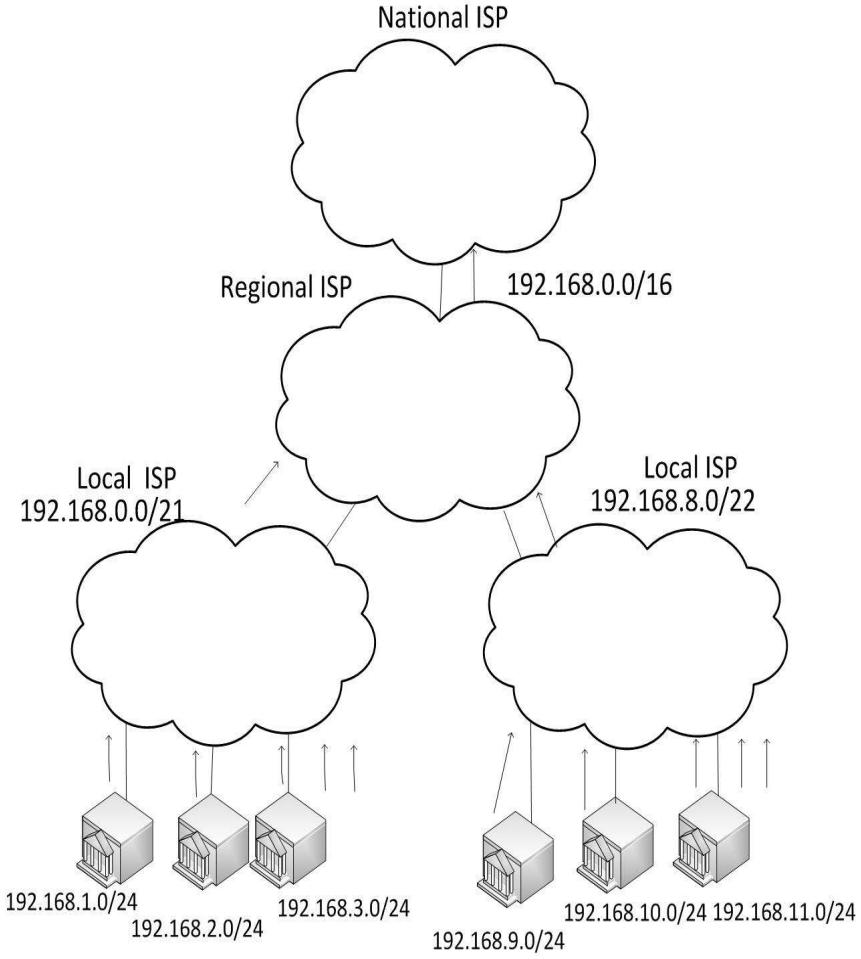

Figure 1-2 Classless IP addressing scheme

Local ISP (also called Tier3) routers have information about networks locally connected to it but advertise shorter prefix to upward regional ISP(also called Tier2) routers and regional ISP routers summarize these aggregated routes further to national ISPs(also called Tier1). So that national ISP routers have less number of entries in their routing tables and they only determine the direction of destination towards regional ISP routers and the local ISPs will finally forward the traffic to the intended destination [2]. Since BGP is used to exchange routing information among different autonomous systems and the contemporary version (BGP version 4) has the capability of carrying subnet mask in advertisement – thus supporting CIDR and route aggregation.

In order to make this CIDR scheme really effective, IANA assigns the contiguous address block in different regions and the routes can be summarized and larger prefix is advertised upwards.

CIDR Allocation to Regions

- Europe 194.0.0.0–195.255.255.255
- North America 198.0.0.0–199.255.255.255
- Central/South America 200.0.0.0–201.255.255.255
- Others 204.0.0.0–205.255.255.255
- Others 206.0.0.0–207.255.255.255

Figure 1-2 above shows an example of IP prefix assignment from national ISPs down to local ISPs and users.

Chapter 2 Background

BGP is the routing protocol used over the Internet to exchange routing information among different autonomous systems [3]. Having qualities of high scalability and ability to apply policy-based routing, BGP differs from interior gateway protocols like OSPF and EIGRP which have low convergence time but cannot be scaled to run at the Internet level. BGP speaker establishes unicast TCP connection (port 179) with a BGP peer and this TCP connection is used for exchanging information between the peers. TCP takes care of message reliability and retransmission making it sure that message sent is received indeed.

Once the TCP connection is established, BGP peers can establish peer relationship with each other before exchanging route information. When such peer relationship is established between two routers, they will exchange their routing information about the reachable destinations and their paths and therefore BGP is considered as path vector routing protocol. In order to form and terminate peer relationship as well as to exchange routing information, BGP peers exchange several messages such as Open, Update, Keepalive and Notification. During the operation of establishing peer relationship and exchanging routes, BGP routing process moves through different states such as Idle, Connect, Active, OpenSent, OpenConfirmed and Established.

2.1 BGP Messages

BGP Routing protocol exchange different messages for coordinating information between participating routers in BGP domain.

2.1.1 Open Message

Open message contains BGP version number, AS number and holddown time that need to be negotiated with neighbour before establishing peer relationship. The AS number will decide the type of connection as external BGP (eBGP) or internal BGP (iBGP). eBGP runs between two routers in different autonomous systems to exchange information with each other while iBGP runs between two routers in the same AS to spread the routing information within organization.

2.1.2 Keepalive Message

Once peer relationship is established, the peer routers will keep on sending Keepalive messages to each other. Keepalive messages serve the purpose of saying hello and informing peers about each other status.

2.1.3 Update Message

Update message contains the real stuff: new prefixes to be advertised and prefixes being withdrawn (that are previously advertised).

Update message will have a list of withdrawn prefixes or/and new prefixes with their path attributes.

2.1.4 Notification Message

Notification message is sent whenever some error or mismatch is detected, Notification message is sent to peer and connection is closed. The very important point here to note is that whenever a peer relationship is broken between two peers, all routes learned through said peer are purged out from the routing database.

2.2 BGP States

BGP process running on router moves through different states during its execution.

Idle State

When the router is not configured for BGP, BGP process on this router is in Idle state waiting for start event which is the manual peer configuration of BGP. Once BGP process is started on router, it initializes TCP resources, starts listening and moves to Connect state.

Connect State

In connect state BGP process waits for establishment of TCP connection after which Open message will be sent for peer negotiation and BGP process moves to OpenSent state. If TCP connection could not be established router moves to Active state and waits for connect retry timer to expire before attempting TCP connection again.

Active State

In Active state, the router is actively trying to establish TCP session with peer after which Open message is sent and router transits to OpenSent state.

OpenSent State

OpenSent state means router has sent an Open message containing its BGP parameters and waiting for the corresponding Open message from peer to negotiate and establish BGP peer relationship. Here the connection type is determined whether its eBGP or iBGP, holddown time negotiated and Keepalive messages started to be exchanged and state transits to OpenConfirmed. If the corresponding Open message received has some mis-match or if the Notification message (reporting mis-match) is received, state transits to Idle.

OpenConfirm State

OpenConfirm state means BGP peering relationship has been established between the peers and they have started exchanging Keepalive messages. In this state if Keepalive is received BGP process on that router moves to Established state and holddown timer is started. If Notification message or TCP reset is received, BGP router moves to Idle State.

Established State

Established state is the final state where routing information is exchanged among peers. If notification message, TCP Reset is received or keepalive is not received (in due time) and

holddown timer expired, connection is dropped and router moves to Idle state. BGP state transition diagram is shown in Figure 2-1 below.

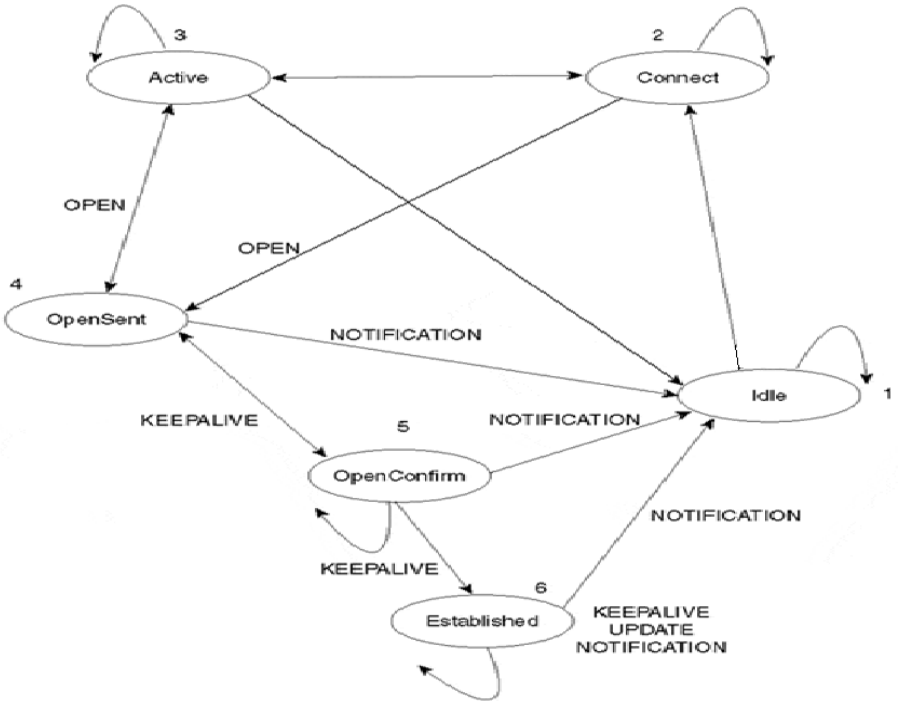

Figure 2-1BGP state machine

2.3 Path Attributes

Path attributes are characteristics associated with BGP routes being advertised. These path attributes make BGP metric, BGP doesn't use simple metric like cost in OSPF. However BGP path selection involves complex algorithm by taking decision on the basis of these path attributes.

There are different categories of path attributes as illustrated in Figure 2-2 below.

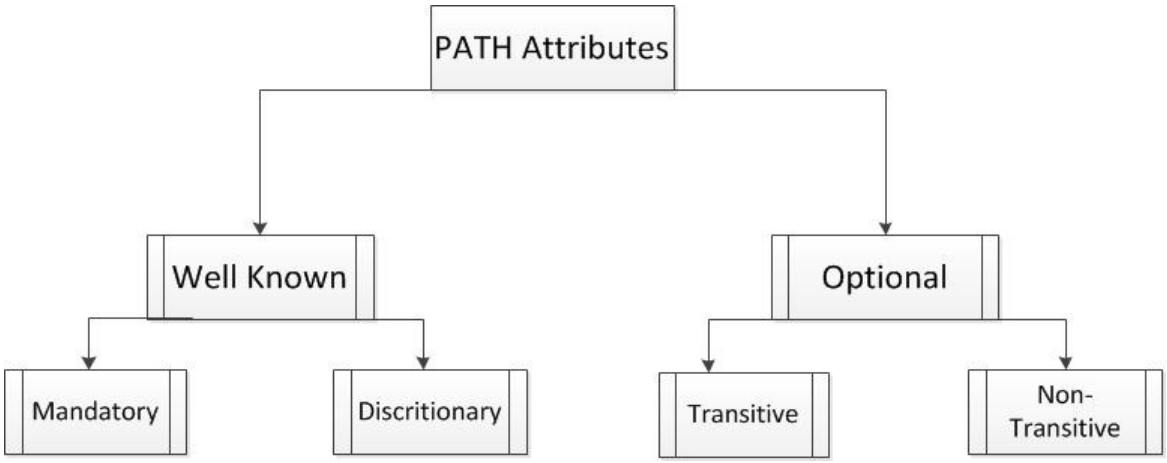

Figure 2-2 BGP path attributes hierarchy

2.3.1 Well-Known Attributes

Well-known attributes must be recognized and supported by every BGP implementation. This category has further two divisions.

WELL-KNOWN MANDATORY

Well-known attributes are supported by every BGP router and well-known mandatory attributes are included in every BGP route advertisement. Examples are ORIGIN, AS_Path, NEXT-HOP.

Well-Known Discretionary

Well-known discretionary attributes must be supported and understood by every BGP speaking router but it is not necessary to include them in every advertisement. Example is LOCAL_PREF.

2.3.2 Optional Attributes

These are the attributes which need not to be supported by every BGP implementation.

Optional Transitive

Optional Transitive attributes needs to be forwarded along route advertisement even if that router doesn't recognize it. Examples are AGGREGATOR and COMMUNITY.

Optional Non transitive

These are optional attributes that can be deleted by router, if router doesn't recognize them. Example is Multi Exit Discriminator (MED).

2.3.3 The AS_Path Attribute

AS_Path is a well-known mandatory attribute associated with every BGP route and contains the list of autonomous systems that must be passed through to reach the prefix advertised by this route advertisement with originating AS is listed at last. When the Network Layer Reachability Information (NLRI) is advertised to eBGP peer the router prepends it's AS number in the beginning of list (AS_Path). In Figure 2-3, AS number 20 advertises prefix 10.0.0.0 towards AS number 30 and writes its own AS numbers (20) in AS_Path attribute. Similarly AS number 30 prepends 30 to AS_Path attribute before advertising the prefix to eBGP peer in AS number 40. However AS number 20 has direct connection to AS number 40 and advertises the prefix 10.0.0.0 with AS_Path 20 to AS 40 as well. Now AS number 40 have two routes to prefix 10.0.0.0 but with different attributes.

BGP is called path vector routing protocol, as by default the route having shortest path is selected, if no other setting is manipulated. For example as shown in figure 2-3, routers in AS number 40 will chose the direct path towards AS number 20 to reach the prefix 10.0.0.0 instead of the other path having longer AS_Path.

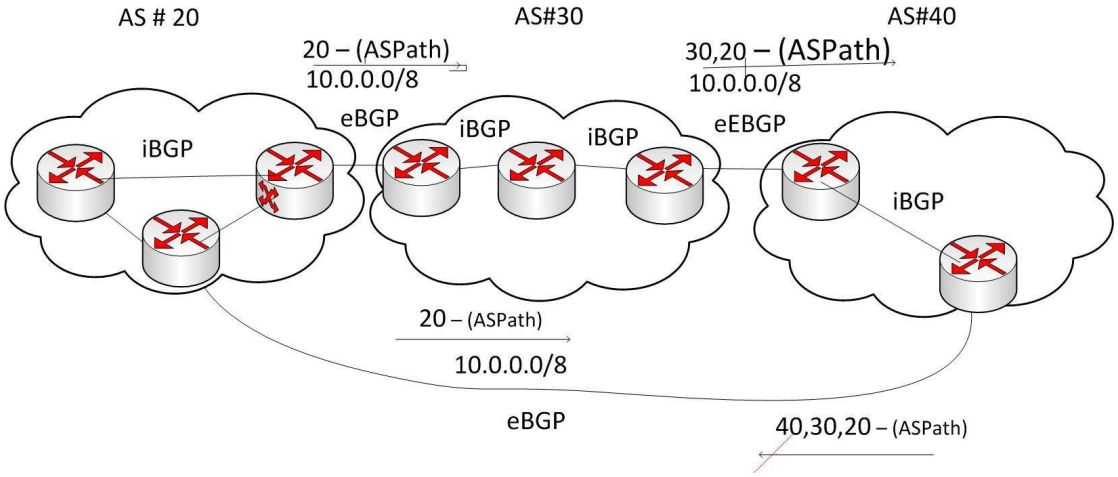

Figure 2-3 AS_Path attribute

AS_Path attribute is also used as loop prevention mechanism in BGP. When router receives NLRI containing its own AS number in the AS_Path list, it will simply ignore this advertisement considering this path as possible loop. For example if AS number 40 advertises the information for prefix 10.0.0.0 back to AS number 20, it will simply ignore this path considering it as possible loop since AS number 20 is already listed in AS_Path attribute.

2.3.4 The Origin Attribute

ORIGIN attribute defines how the associated route is originated at first place. ORIGIN attributes may have different values of origin code:

IGP: NLRI learned through configuration of network command will have origin code of IGP.

EGP— The NLRI learned through eBGP are marked with origin of EGP.

Incomplete— Route learned through redistribution from some other routing protocol is assigned with Origin attribute value of Incomplete.

2.3.5 The Next_Hop Attribute

NEXT_HOP attribute associated with NLRI refers to next-hop IP address where the traffic should be directed to reach destination advertised by this NLRI. But unlike eBGP, in iBGP the next-hop is not always peer router's exit interface IP Address.

When advertising to different autonomous systems, the next-hop is the IP address of routers outgoing interface. For example in figure 2-4 , when router A in AS number 200 is advertising about prefix 6.6.0.0/16 to router B in AS 100, next-hop attribute value will be outgoing interface IP address i.e 7.7.7.1.

When advertising to same autonomous system (iBGP peer) and prefix is internal, next-hop is outgoing interface IP address of router that originated this advertisement. For example in figure 2-4 , router B will advertise about prefix 5.5.0.0/16 with next-hop of 1.1.1.1 to router C and router C will advertise this prefix to router D, still with next-hop of 1.1.1.1 (outgoing IP address of router that originated this information). Recursive lookup is required in this case, when router D has to reach prefix 5.5.0.0/16 it has to recursively look up for 1.1.1.1 in its routing table and send the traffic destined to 5.5.0.0/16 towards 1.1.1.1.

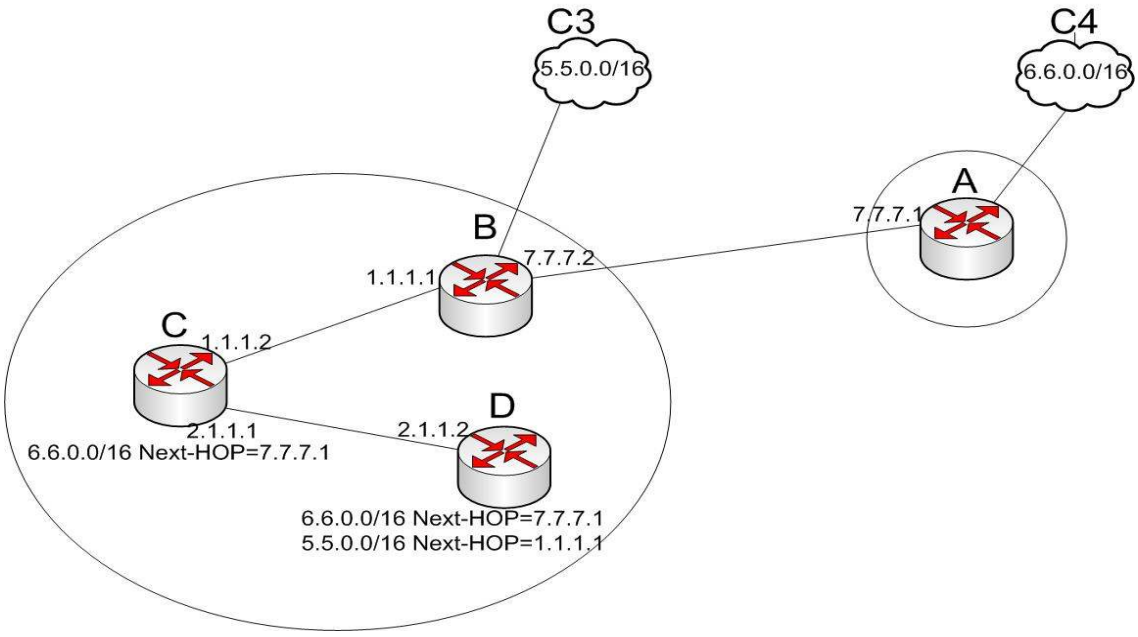

Figure 2-4 NEXT_HOP attribute

When advertising a prefix that is learned from another AS to an internal peer, the next-hop is IP address of the outgoing interface of the external AS peer from which NLRI was learned first. For example router B learned NLRI about prefix 6.6.0.0/16 through router A

which is external peer, next-hop in this case is 7.7.7.1. Router B will advertise this prefix to router D through router C with next-hop unchanged (i.e 7.7.7.1). Now router D has to do recursive lookup for 7.7.7.1 to reach 6.6.0.0/16, but 7.7.7.0/30 is external network and router D doesn't know about that network.

Solution is either to advertise this 7.7.7.0/30 network to IGP which can distribute it to all internal routers or change the next-hop at router A by instructing to use the next-hop-self Cisco IOS command.

2.3.6 The Local_Pref Attribute

LOCAL_PREFERENCE, a well known discretionary attribute is used inside autonomous system to set and communicate the local preference for certain route. If a router has two routes to the same destination, a route having higher local preference is preferred over the one that have lower local preference value. The boundary router of autonomous system assigns the local preference value and then it is communicated inside this autonomous system, so that all routers within autonomous system can take the preferred path.

For example in Figure 2-5 below, router R0 receives two routes to network 192.168.1.0/30, one from R1 and one from R2. Routes learned through R1 have been assigned local preference of 150 while routes learned through R2 have been assigned with local preference of 250. So when R0 has to send traffic to 192.168.1.0/30, it will prefer the link B as it has higher local preference.

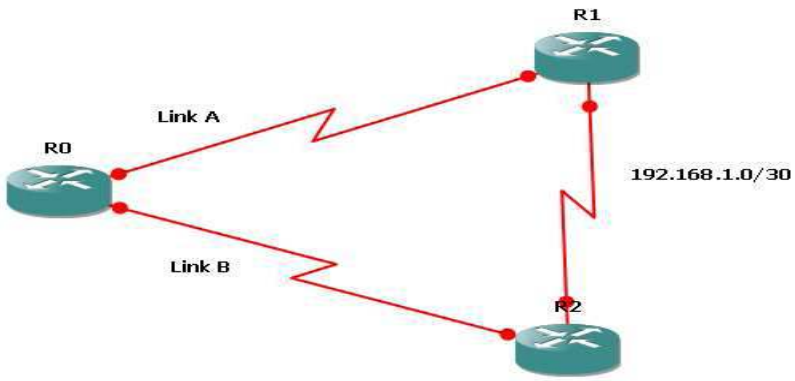

Figure 2-5 Local preference attribute

2.3.7 The Multi_Exit_Disc Attribute

BGP have total control on traffic leaving network and can enforce any traffic policy using path attributes but have very little control over inbound traffic. MED is the attribute through which ISP can suggest a possibly better way towards it from outside world. It is not bounding on receiving external peer to obey suggested MED. Lower value of MED is preferred.

For example in Figure 2-6 below an organization running BGP under AS number 100 and have internal network 192.168.1.0. Border router A will advertise this route to ISP with MED value 100 on link A and border router B will advertise this same network to ISP with MED value 200 on link B. Now the ISP has two paths towards internal network and it may choose the one with lower MED i.e. through link A.

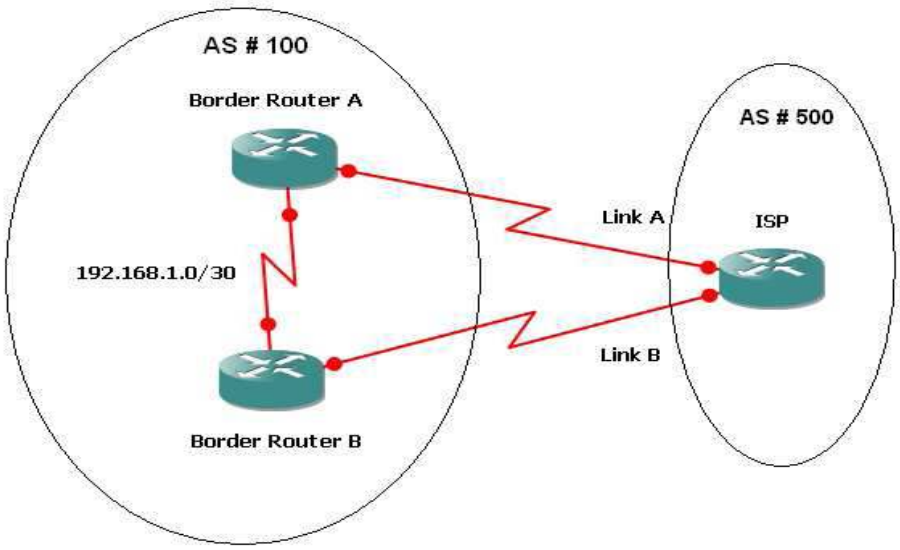

Figure 2-6 Multi_Exit_DISC attribute

2.3.8 Administrative Weight

Cisco proprietary attribute is valid on single route only and not advertised to any peer internal or external. Higher value of weight is preferred if there are two routes to same destination.

2.4 BGP Route Selecton Algorithm

First step is to check the availability of next-hop IP address, if next-hop is not reachable route is not considered [4].

1. The route having highest weight is preferred.

2. If two routes have equal value of weight attribute, Local_Pref attribute is checked and one having higher local_pref value is preferred.

3. If local_pref is equal, check if one of the route is learned through IGP; it's preferred.

4. Route having shortest AS_Path is preferred, if decision is not made in previous steps.

5. Check origin of route, IGP is preferred over EGP and INCOMPLETE is least preferred.

6. If multiple routes tie upto that point, choose the route with the lowest MED value.

7. Select route learned through eBGP over iBGP.

8. Prefer the route having lowest IGP metric to next-hop address.

9. If no decision is made upto that point, use both links, if maximum path command is configured.

10. Otherwise if maximum path command is not configured, use the route with lowest BGP Router ID.

Chapter 3 Security Issues in BGP

3.1 Routing Protocol Security

Routing protocol is used to convey path information among participating routers, any kind of error or attack on the routing protocol results in absence or incorrect routing information which will have devastating effect on the network traffic. Accurate and timely learning of this information is necessary for smooth running of the network [5].

There are different vulnerabilities and threats related to routing protocols. Vulnerabilities in routing protocols may come from **three** aspects; firstly from vulnerabilities in the underlying protocols being used, secondly from misconfiguration and human errors and lastly from some hi-jacked trusted partner. When an attacker, for example, outbreaks and takes control over some trusted router then the attacker uses the mutual trust relationship between the compromised router and its peer to spread erroneous information. Therefore we can say that the routing protocol is as secure as weakest entity in the routing domain.

3.2 BGP Security

BGP inherits the general security problems of routing protocols but such problems in BGP have very large effect due to its huge installed base and big domain. BGP being used in the Internet involves thousands of participating routers with peers always coming in and going away resulting in frequent changes of the BGP trust model. BGP V4 was designed to provide scalable and controllable routing protocol with almost no security concerns in mind [3]. BGP itself doesn't provide any security functionality [6].

However it is also possible to find weak router in BGP domain and use it to launch attack against the Internet routing. Moreover, BGP is the most complex routing protocol to configure and maintain and even minor error in its configuration can have disturbing effect. Thirdly BGP security problems become more serious due to large scale propagation of errors and consequently the magnitude at which the services got interrupted [8].

BGP routing protocol lacks many safety checks to provide security guarantees, some of these missing checks are listed below.

- There is no peer authentication in BGP by default.
- BGP has no means to assure that route updates received in update message are un-modified.
- No mechanism to check that prefixes advertised in update message are owned by the advertiser or authorized by owner to advertise these prefixes.
- No mechanism to check correctness of path attributes associated with updates.

- In case of withdrawn routes, it must be ensured that the same peer has advertised these routes previously.

BGP doesn't perform these checks and hence vulnerable to deliberate and accidental wrong advertisements.

There were many attacks and misconfiguration in BGP that resulted in major outage of the Internet in history.

3.2.1 AS7007 Internet Blockage Incident

AS7007 received routing updates from its downstream ISP and advertised them to upstream peers believing these updates were correct. But downstream ISP starts forwarding prefixes with manipulated length and attributes for almost all Internet. AS7007 had not implemented any security mechanism like route filtering to identify and prevent such errors or attacks resulting in forwarding of malicious information to upstream ISPs. Routers receiving this malicious information began forwarding data packets towards AS7007 which became a blackhole for the traffic [9].

On January 2006, "Con-Ed Steals the 'Net" is another incident in which false information was believed and advertised further leading to spoilage of the routing tables and wrong routing decisions in the global Internet [10].

3.2.2 YouTube Blockage by PTA

Similarly, on 24 February 2008, Pakistan Telecom Authority (PTA) advertised specific prefix of YouTube. PTA intended to block YouTube access in Pakistan and advertised the specific prefix 208.65.153.0/24 pointing to Null0 interface (sort of bit bucket), this prefix was part of the prefix used by YouTube i.e 208.65.153.0/22. The intention was that the traffic destined to YouTube would be forwarded to Null0 interface and hence YouTube would get blocked inside Pakistan. But due to some mistake the same route was advertised to upstream ISP (PCCW AS number 3491). PCCW, believing this information were correct, advertised this information to other peers as well, just because there was no authentication system present in BGP. All routers in the Internet who learned this information preferred this more specific route for YouTube traffic and started forwarding packets destined to YouTube towards this more specific route. In this way PTA trashed these packets by forwarding towards the null interface and hence blocked the Internet for almost two hours [11].

3.3 Attacks on TCP

BGP being based on TCP, inherits the security weakness of TCP as well, TCP problems can be exploited to disrupt BGP routing protocol. There are many well-known problems in TCP [12], such as:

- SYN Flooding
- TCP RST

3.3.1 SYN Flooding

SYN Flooding is a possible denial of service attack against TCP protocol. In TCP Three way handshake is performed before connection establishment [13] as shown in Figure 3-1.

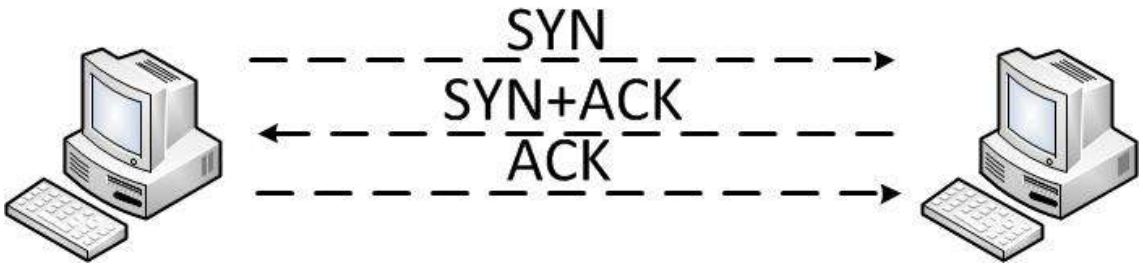

Figure 3-1 TCP handshake

Originating peer sends SYN packet to establish TCP connection, recipient peer on receipt of SYN, send SYN + ACK and allocates data structure Transmission Control Block (TCB) in kernel memory to hold information about this uncompleted connection until the corresponding ACK is received. Attacker sends many SYN packets from spoofed address of dead host which is unable to send corresponding ACK and complete or destroy this connection. Ultimately kernel memory is full and recipient peer cannot establish TCP connection with authorized peers which will cause denial of service for them.

If BGP router is put under SYN flooding attack, it might fail to establish TCP connections with legitimate peers and therefore will not establish peer relationship resulting in lack of information or denial of service.

Solutions

There are some mature TCP implementations that give some protection against this attack. Moreover newer transport layer protocols proposed have dealt with this issue by replacing 3-way handshake with 4-way handshake. No data structure is allocated until the originator proves its interest to establish the connection by replying through ACK.

It's more effective to prevent TCP SYN attack from happening than hardening victim, for which proper filtering of spoofed address by ISPs is much effective. If ISP filters the spoofed outgoing packets, it will become difficult for attacker to send large number of SYN packets.

3.3.2 TCP RST

TCP connection is established and all BGP peer negotiations and route updates pass through this TCP connection. Therefore if peer or some malicious attacker breaks this TCP connection and BGP session will be torn down. When router receives TCP RST packet, the corresponding TCP connection is closed. This will need that some malicious user has access to peer or can guess sequence number in recipient packets, in order to send TCP packet with RST bit set which can bring the TCP connection down.

When TCP connection is broken, BGP router deletes the corresponding peer relationship with peer and all routes learned through this peer are purged out from routing database. Router also withdraws related routes that it has advertised to other peers and the affect propagates. TCP connection may come up again and BGP peer relationship is established, now the routes are advertised again, this may result in route flapping and put much load on communicating links and router processor.

3.4 Attacks against BGP Protocol

There are many attacks and problems that are possible due to vulnerabilities and lack of security checks in BGP protocol itself. These problems are mainly due to missing security checks mentioned in ***section 3.2***. Other problems are due to weakness in the way the protocol is working; like when any BGP peer drop the BGP connection when notification message is received which results in processing load on router and route flapping [14].

3.4.1 Source of Attacks in BGP

First of all we need to identify the sources of attacks and the launching pad for BGP security problems; they can be classified into following list.

A. Injection of false information from attacker.

B. False information from trusted partner

   I. Partner is hi-jacked

   II. Originate false information due to some configuration error.

   III. Originating false information intentionally to accomplish some goals.

C. Vulnerabilities in Router's Operating System.

3.5 Security problems in BGP

There are many possible methods for influencing path selection in BGP protocol and there are many reasons to do so. Path selection depends upon the advertised routes and their path attributes which attackers can also manipulate to influence path selection [16].

3.5.1 Incorrect Routing Updates

A BGP peer can originate false routing updates i.e a route advertisement claiming to reach the prefix which it doesn't own and cannot reach or advertising more specific prefixes; where such incidents had happened in history like YouTube blockage.

Receiving router has no mechanism to validate and sanitize the received updates. Therefore recipient has to believe this information and spread it further which results in incorrect routing tables. Acceptance of this wrong information depends on its origin; false information originated from Tier 1 ISP is accepted by many routers and has devastating effect

whereas false information generated at Tier 3 router has limited effect. Origination of false updates is possible due to sources mentioned in section 3.4.1.

3.5.2 De-Aggregation

De-aggregation is the process of advertising more specific prefix instead of summary routes. A router receiving this more specific prefix will prefer this routing advertisement above any other path selection methods because more specific prefixes are always preferred while making routing path selection decision. As shown in figure 3-2, a legitimate router C is forwarding reachability information about prefix 192.168.0.0/16 and attacking router is forwarding reachability information about 192.168.1.0 /24 towards router A. Router A will install both of these advertisements in its routing table and always prefer path towards attacking router for traffic destined towards 192.168.1.0 due to longest prefix match rule.

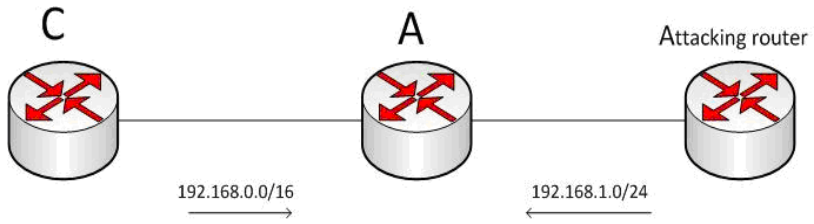

Figure 3-2 De-aggregation example

Advertisement of more specific prefixes due to misconfiguration is major cause of most interruptions happened in history of BGP.

3.5.3 Manipulation of Path Attributes

BGP routing protocol make routing decision on the basis of path attributes associated with routes. Peers can manipulate these path attributes to attract other peers to use path towards them or to cause some impact on routing decisions.

Receiving router can reset most of these path attributes when routing updates are received. But some attributes like AS_Path cannot be manipulated. Sending router can append its own AS number to AS_Path attribute to influence routing decisions.

Another attribute Multi_Exit_Disc (MED) may be also manipulated such that the values are changed to advise other ASs about incoming path to reach own organization. Route with Lowest MED value is preferred.

Every route has path attribute NEXT_HOP, which contains the value of next-hop IP address. An advertising peer can modify the path attribute next-hop to direct the traffic towards desired points.

Malicious user or attacker can exploit these path manipulation methods to achieve benefits for its own organization. This can create problems for other competitive organizations, as explained below.

3.5.4 Blackhole

Advertising prefix reachability to peers is actually a promise to deliver packets towards this prefix. Therefore a router will send the traffic destined to this prefix to the peer who has advertised reachability information about this prefix. But if the originating router does not have information about this prefix, it cannot deliver the traffic towards its intended destination and may drop the traffic resulting in a blackhole.

3.5.5 Eavesdrop

A router can eavesdrop the protected traffic by propagating false routing information and attracting traffic towards some un-protected or compromised routers. In the Internet there are many possible paths between any pair of receiver and sender. One can originate or block some route advertisements to force certain router to use some insecure path. Attacker can also attract traffic towards it by originating false information and then can forward it to its intended destination i.e. acting as man in the middle.

3.5.6 Congestion, delay and Loops

Routers may advertise the routing information to deliver large number of traffic through one path or some less optimal path which may result in congestion and delay of traffic. This path manipulation is done to cause denial of service problems.

Incorrect and incomplete routing information in routing tables of complete domain may result in incorrect decisions which may also cause routing loops.

Chapter 4 Protection measures available for securing BGP

4.1 Peer Authentication

TCP MD5-based authentication is supported in BGP, where in this scheme a key is pre-shared between participating routers. The sending router generates hash of every message along with the secret key and sends this hash concatenated with the message. The receiving router also calculates the hash on message plus the secret key and match this hash value with received hash, if the values are matched; message is accepted otherwise message is dropped [16].

In this way TCP MD5 authentication can be used to authenticate peer i.e. to verify that the peer who is advertising information is legitimate peer. TCP MD5 has many weaknesses discovered, therefore new protocol TCP-AO [17] is being proposed in the literature but its support is not incorporated in BGP standard yet.

Peer authentication can protect against attacks generated from outsiders, but misconfiguration and compromised routers can still be used to advertise false information from legitimate peers who have passed through MD5 authentication test.

False information due to misconfiguration or router hack is still possible and there are no checks in BGP, therefore many security enhancements for BGP are proposed. There are two complete proposed enhancements in BGP that can provide secure exchange of routing information.

4.2 Secure BGP (S-BGP)

S-BGP [18] is new secure form of BGP, which verifies that:

- Update message is received from legitimate peer and is unmodified.
- Advertiser owns or is authorized by owners to advertise the prefix whose information is contained in update.
- If the removal of route is requested in update, same route should have been advertised by the sending router previously.

S-BGP structure has four main elements that are used to provide above reliability:-

- A *Public Key Infrastructure* (PKI) to represent prefix and AS number ownership.
- Address Attestation issued to AS by owner of prefix; authorizing specific AS.
- Every router along the AS_Path issues route attestation to peer incrementally about previous paths.
- IPSec is proposed to be used in S-BGP for session security.

4.2.1 S-BGP Public Key Infrastructure

S-BGP requires PKI for authorization of prefixes and AS number allocation to different organizations. X.509 (V3) digital certificate in parallel to existing IP and AS allocation is required. In this structure, each AS assigned with public prefix and AS number will also be assigned with public certificate which binds the prefix and AS number to organization's public key.

There are two attestations, route attestation and address attestation attached with every update advertisement. Address attestation is issued by the owner of the IP prefix to AS who will be authorized to advertise this prefix. Route attestation is achieved by every router before advertising the update to its peer. This attestation verifies that last AS in AS_Path (sending AS) is authorized by previous AS to advertise these prefixes. New optional path attribute is defined in S-BGP to communicate this attestation information between BGP peers.

Receiving routers use these attestations and digital certificates to assure the reliability of the received updates. S-BGP ensures that owner of prefixes (contained in update) has authorized the advertiser to advertise these prefixes using address attestation attached to the update and the public key of owner. Similarly, it is also verified that all ASs along the path are also authorized to advertise prefixes to their peers by using Route attestations.

But checking all these attestations and storing the required certificates demand very high processing power and memory resources respectively.

While the size of routing table at core routers in the Internet is growing as shown in Figure 4-1 below; core routers are hardly managing their current functionality with this number of routing entries in their routing tables and they cannot bear extra load of route attestation and address attestation imposed by S-BGP.

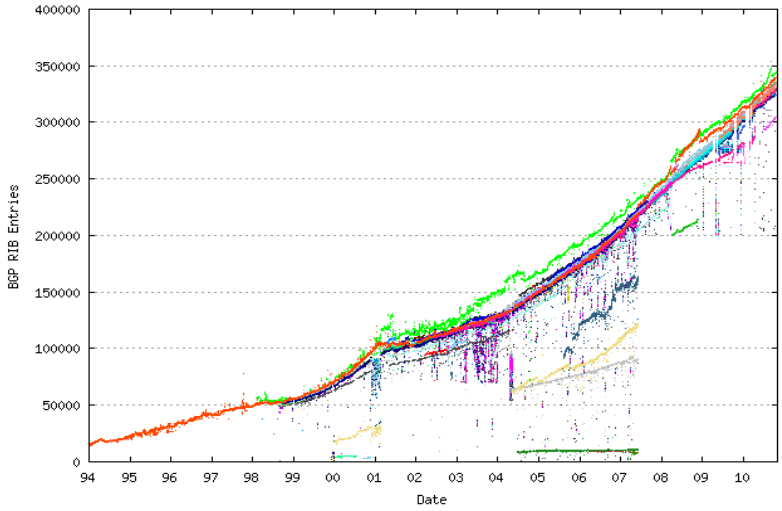

Figure 4-1 Routing table growth at core routers [20]

S-BGP can solve many BGP security problems but it is not being deployed due to three main reasons [19].

- Need of high processing power.
- Establishment of PKI infrastructure that is pre-requisite for S-BGP.
- Not implemented by main vendors and ISPs. Everybody is waiting for each other to take first step.

4.3 Secure Origin BGP (SoBGP)

Secure origin BGP (SoBGP) is another modified form of BGP to achieve security goals in BGP routing. SoBGP provides the following defence promises

- Verifying that AS originating advertisement for prefix 'A' is authorized to advertise this prefix.
- Verifying that AS originating route to prefix 'A' can actually reach this prefix.
- Verifying that the AS_Path attribute is valid and acceptable.

SoBGP defines and uses three certificates to achieve the above mentioned security guarantees

1) EntityCert
2) AuthCert
3) PolicyCert

AS will generate these certificates and other ASs will use these received certificates to verify authenticity of update messages. These certificates are signed using AS private key and EntityCert is used to distribute public key corresponding to this private key. EntityCert is signed by key of some well known verifier like Verisign and its key can be used to authenticate EntityCert.

AuthCert is used to prove ownership and authorization to use certain prefixes. AS will generate this AuthCert and sign it using the private key (corresponding to public key in EntityCert). Receiving AS will verify that advertiser is authorized to advertise these prefixes by using this AuthCert.

PolicyCert is used to distribute topology information to other AS. All ASs will build their own topology map using received PolicyCert and verify the updates according to this topology.

New BGP message 'Security message "is defined to carry these certificates between autonomous systems. While SoBGP is being deployable it will be able to solve many security problems, but in reality it doesn't meet these security guarantees [21] and it will be little too heavy for already heavy burdened Internet core routers.

4.4 Practical Implementable Solutions

There are many solutions proposed to secure BGP like S-BGP, SoBGP, pretty secure BGP (psBGP), but they either do not solve all known problems associated with BGP or they are not implemented due to cost factors coupled with them. But router vendors and large ISPs have developed some methods and practices to provide security in BGP. These good practices can greatly increase the security of BGP and avoid many problems associated with BGP without putting any severe load on BGP speaking routers. These practices include:-

- Router hardening
- Generalized TTL security mechanism
- Route dampening
- Maximum prefix limiting
- Limiting AS_Path length
- Prefix filtering

4.4.1 Router Hardening

Router is the main device that generates and advertises BGP advertisements to peers. If attacker can in somehow manage to get privileged access to trusted BGP router, this compromised router can be used to distribute malicious information to its peers. Therefore to achieve BGP security our first step should be hardening participating routers, so that no malicious user can get access to them [22]. This hardening includes:-

- Physical security
- Limited access to router
- Deny spoofed packets
- Use of secure and encrypted password in order to protect access
- Disable all unused services

Control of physical access to router by authorized people only, and limiting remote access can be achieved by configuring access list to permit some IP addresses and block (deny) all other explicitly.

Cisco has introduced the new feature “Auto Secure” that can be used to improve the security of Cisco router. This tool can be used to generate configuration script that will disable unused services and create and apply strong passwords to connections towards router.

4.4.2 Generalized TTL Security Mechanism

Peer authentication can protect a router from peering with un-authorized BGP speaker. But if many routers are sending request to TCP 179 port to become BGP peers even though they don’t know passwords, this may cause denial of service. They send too many fake requests that make router busy in evaluating those requests and saying sorry to them.

BGP TTL hack protection by generalized TTL Security mechanism can be used to prevent such scenarios or attacks [22]. By default eBGP peers are mostly directly connected to each other and BGP TTL hack protection exploit this fact to prevent denial of service attacks from

routers that are more than one hop away from eBGP speaking router. The method is that eBGP router generates BGP message with TTL value of 1.

eBGP router accepts any BGP message with TTL value greater than equal to zero, so routers that are many hops away can send BGP messages by setting TTL value to higher value in such a way that when BGP message arrive at receiver side, TTL value is greater than or equal to zero. To prevent this attack, we can configure BGP router to accept packets having TTL value greater than equal to 254. Maximum allowed TTL value is 255. Therefore directly connected neighbour will generate the BGP message with TTL value set to 255 and these messages arrived at recipient side with TTL value 254 and hence got accepted. Remember this TTL value is checked after decrementing [30]. Any message from router not directly connected to BGP router will have TTL less than 254 and hence got rejected. For example in figure 4-2 message send by router B to A will be accepted but message send from C to A will be discarded due to its TTL value as, router A is configured to receive only BGP message with TTL value greater than or equal to 254.

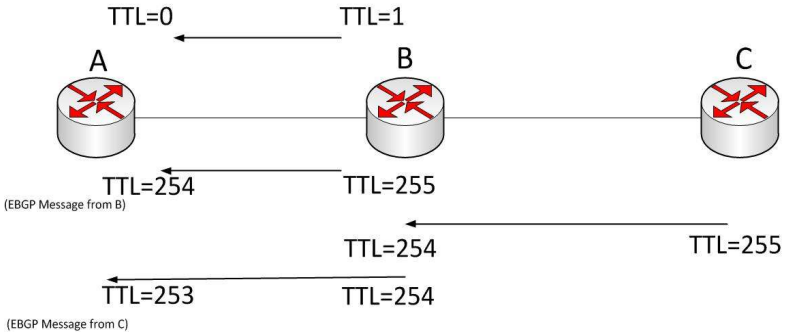

Figure 4-2 BGP TTL hack

4.4.3 Route Dampening

Route flapping is a sequence of advertisement, withdrawal and re-advertisement of any route. Route flapping may happen due to configuration error, OS error, attack or intentional flapping to launch an attack against BGP routing protocol. Route flapping has cascading effect as shown in figure 4-3 below, router A withdraws previously advertised route 100.10.0.0/16 to B. B has to re-evaluate routing table and forward the new advertisements to upstream routers C, D and E which may have further upstream connections [23].

BGP routing table re-evaluation is very processor intensive task and can result in very high CPU utilization or denial of service, because BGP routers are very heavily loaded with more than 3 hundred thousand entries in routing table. Many flapping routes may make the router too busy to perform its normal function of routing.

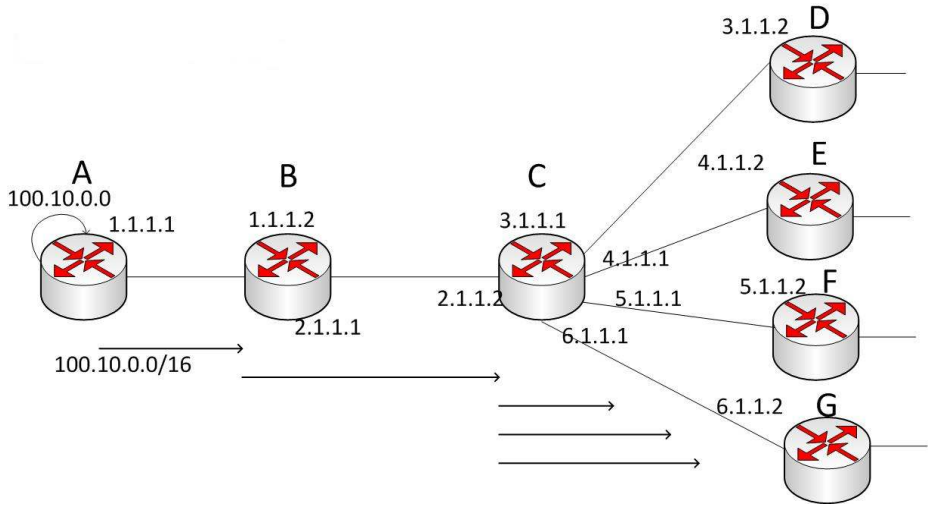

Figure 4-3 Route flapping

Route dampening is the feature supported by Cisco routers to control flapping route advertisement. When previously advertised route is withdrawn and advertised again, route is said to be flapped. Route dampening feature increment penalty assigned with route each time it flaps and starts suppressing route when its penalty reach some threshold. At this stage the route is neither advertised nor accepted from peers.

Five variables are used to control the advertisement of route and their default value is:

Penalty = 1000 for each flap

Suppress Limit = 2000

Half-life = 15 minutes

Re-use Limit = 750

Each time a route flaps, its penalty is incremented by 1000 and when penalty value reaches suppress limit i.e 2000. Route will be suppressed and it will not be advertised to peers. If route keeps on flapping penalty value associated with this route keeps on incrementing with each flap and remains equal or higher than suppress limit and route will continue to be suppressed. But if route does not flap for half-life i.e 15 minutes, its penalty value will be decreased to half and it keeps on decrementing by half after every 5 seconds afterwards.

The route will be active again and advertised to peers when penalty value associated with this route becomes less than or equal to re-use limit i.e 750 by default.

In this way we can block advertisements of flapping routes and avoid extra load of re-evaluating the routing table again and again due to some flapping routes.

4.4.4 Limiting Maximum Prefix Received

BGP core routers may have more than 350,000 entries in its routing table and they share this information with each other to provide end-to-end Internet connectivity. Routers having large number of routing entries have potential of overflowing the memory of peering routers especially in case of misconfiguration, hack or exploitation of software OS vulnerability.

BGP allows setting the limit on the number of prefixes that can be received from any peer. This maximum-prefix command allows configuring the limit and the action to be taken when this limit is reached.

When configuring maximum-prefix limit on peer, we need to calculate

- The possible number of prefixes expected to receive under normal circumstances.
- Limit should be configured more than the amount calculated in first step.
- Threshold limit (when to start generating warning messages)

Command syntax: neighbour *neighbour-id maximum-prefix threshold* [warning –only]

Warning-only is optional parameter, and if included only warning messages are generated when the maximum limit is reached. Default behaviour is to start generating warning messages when percentage (threshold) is reached and to tear the peer connection when maximum-prefix limit is reached.

Example command: neighbour 1.1.1.1 100 80

If above command is configured on router A. Router A will start generating warning messages when number of prefixes received from the neighbour router B reached 80 and peer connection is torn down when the total reaches 100.

But expected number of prefixes is changing with dynamic peer relationship and these changes need to be updated, conveyed and configured promptly, otherwise problems may rise.

4.4.5 Limiting AS_Path Length

AS_Path attribute attached to route advertisement contains the list of autonomous systems that need to traverse to reach NLRI advertised in update message. BGP route algorithm looks at AS_Path length and the route having shorter AS_Path length is preferred over longer one. BGP allows making AS_Path longer by repeating the same AS number many times in AS_Path attribute. This technique is used by some ISPs to make some paths preferable over others i.e. to achieve traffic engineering goals. An organization can repeat its own AS number in AS_Path attribute to make it longer, in this way traffic engineering goals can be achieved without affecting loop prevention functionality of AS_Path attribute.

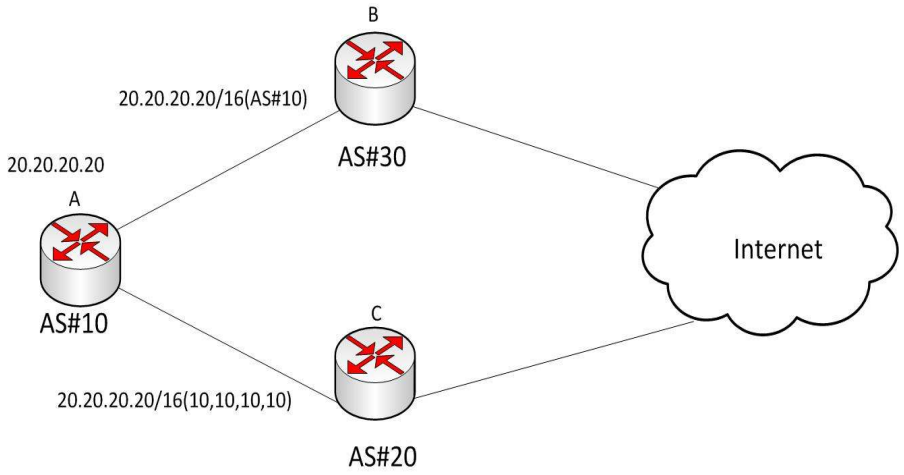

Figure 4-4 AS_Path length

For example as shown in figure 4-4 above, router A in AS #10 send advertisement to B and C about prefix 20.20.20.20/16. But advertisement send to B has AS number 10 listed in AS_Path attribute while advertisement sent to C has AS number 10 listed four times in its AS_Path attribute. Now rest of the internet receiving route to 20.20.20.20/16 from both B and C will always prefer the route going through B due to its shorter AS_Path length [25].

But care is required while appending AS_Path as there were some problems resulted due to manipulation of attribute in the past. Different router vendors have different methods of configuring this prepending and different tolerance to maximum AS-Limit. For example in one attack, one vendor has vulnerable way of configuring number of repetitions in AS_Path attribute that resulted in very AS_Path. This very long attribute when advertised to other vendors, exploits many vulnerabilities in other vendor IOS as their operating system didn't handled such AS_Path.

Therefore care must be given while configuring these AS_Path prepending**s** and **a** limit should be imposed so that longer AS_Path attribute are rejected by default.

4.4.6 Prefix Filtering

BGP is vulnerable to false route advertisement due to hack, error or misconfiguration. We have seen this happened in history. Any BGP router either misconfigured or hacked by some malicious user can be used to originate bad routing information and this information will propagate in the Internet poisoning all routers. There are many security solutions proposed as discussed above, but they are either non-deployable (due to lack of physical resources like memory and CPU) or they provide only partial security guarantees [26] [27].

Prefix Filtering is the most widely used method by ISPs to prevent attacks on BGP and it has been very effective too. Administrators are able to decide which prefixes should be advertised and received from peers and then filters are configured respectively to block advertisements and receipt of unwanted prefixes from any peer.

Administrator needs to do some preparation before implementing prefix filtering, one must clearly decide which prefixes should be advertised and which ones are expected to be accepted.

Decision of filtering incoming and outgoing packets largely depends upon kind of peer relationship. There can be three types of peer relationship**.**

Peer with downstream customer

ISP assigns prefixes to its downstream customer and should accept only these prefixes in advertisement from customer.

Peer with another ISP

Routes exchange with peers depends on agreement that has been negotiated. Each peer has informed other **peers** about the prefixes supposed to be advertised and received and prefix filters according to this agreement must be implemented at both ends.

Peer with upstream ISP

Upstream **or** Transit provider provides connectivity to the rest of the Internet; routing advertisements received from upstream providers vary with situation.

- In case of single provider, only default route is received.
- In case of multi-homing, default plus some upstream customer routes are advertised.
- Complete routing table is received from upstream routers in some cases when there are multiple exit points and routers are intended to take optimal decision for outward traffic.

But a filter must be configured for not to accept its own route from upstream providers.

4.4.7 Filtering Bogons

In addition to these filters which are depending upon peer relationship, there must be some general filtering implemented on every BGP-speaking router to prevent anomalies. There are some IP addresses that cannot be used as valid unicast IP addresses across the Internet and therefore routing advertisement originating from these addresses must be filtered at all routers. Filters must be implemented on every router participating in BGP routing protocol to filter routes of these addresses that are called Bogon Addresses [28].

Proper filtering of theses Bogon addresses prevents network from origination of denial of service attack, because most of the time a spoofed address from the Bogon addresses list is used as source IP address in packets that are aimed to launch such attacks.

Bogon addresses include the private IP addresses defined in RFC 1918 [29], the automatically configured address range and addresses that are not assigned by IANA currently. Bogon list is dynamically changing and administrator needs to update the filters promptly. Bogon list is shown in figure 4-5 below [28].

185.0.0.0/8	5.0.0.0/8
10.0.0.0/8	23.0.0.0/8
37.0.0.0/8	39.0.0.0/8
100.0.0.0/8	102.0.0.0/8
105.0.0.0/8	106.0.0.0/8
127.0.0.0/8	169.254.0.0/16
172.16.0.0/12	179.0.0.0/8
192.0.0.0/24	192.0.2.0/24
198.18.0.0/15	192.168.0.0/16
203.0.113.0/24	198.51.100.0/24
224.0.0.0/4	240.0.0.0/4

Figure 4-5 Bogons List of IP addresses

4.4.8 Re-assurance of Peer Filtering

Peer filters are implemented in a way to re-enforce each other, so that if something goes wrong at one end it will be handled by partner at other end.

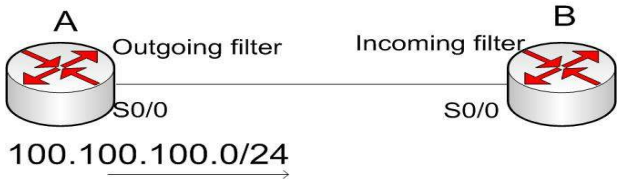

Figure 4-6 Re-assurance of filtering

For example in figure 4-6, router A needs to filter 100.100.100.0/24 route from **the** advertisement sent to router B. Outgoing filter is configured at Serial 0/0 **on** router A to prevent such outgoing advertisement and similarly on serial 0/0 interface of router B, filter is configured to reject incoming advertisement containing 100.100.100.0/24 route.

In this way, if due to some error, misconfiguration or attack router A may advertise false information but it will not be accepted by other peer **i.e.** filters are re-enforcing each other.

Chapter 5 Case Studies

We have designed and implemented several case studies on GNS module (emulating the Cisco IOS) illustrating these attacks, possible protection measures and their effectiveness.

Case Study 1

5.1 BGP Path Attributes and Policy Routing

In this first case study we will show how BGP routing protocol is used to achieve policy-based routing by manipulating BGP path attributes. Later case studies will go through possible security problems that can disrupt this policy-based-routing behaviour of BGP routing protocol. Figure 5-1 below shows the topology diagram for this case study.

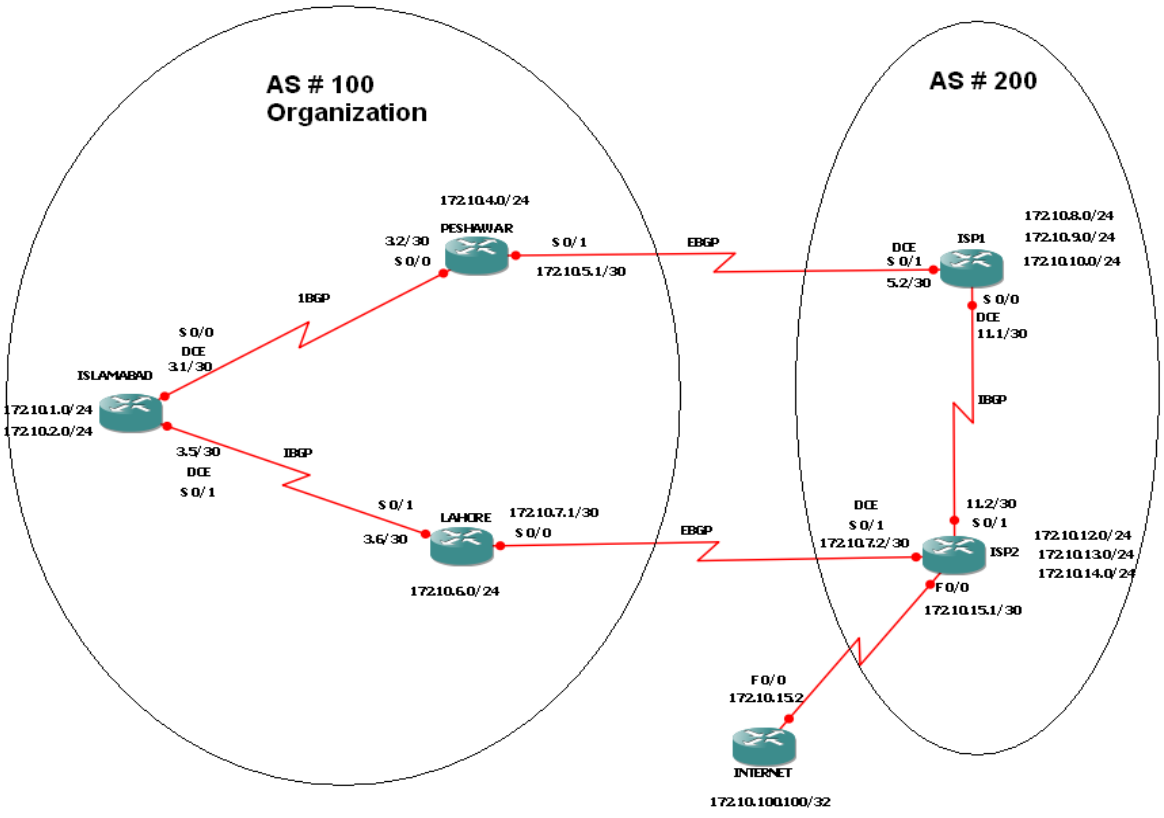

Figure 5-1 Case study 1 topology diagram

In our first case study an organization ABC has three branches in three different cities under autonomous system number 100. Islamabad (headquarter) is connected to Lahore branch and Peshawar branches. So Islamabad has internal BGP (iBGP) relationship with both Lahore and Peshawar branch. Peshawar branch has external BGP (eBGP) relationship with ISP-1 and Lahore branch has eBGP relationship with ISP-2. These ISPs are in same autonomous system (AS number 200) and have iBGP relationship between them.

ISP-2 has a connection to ISP-3 (emulating the Internet) and ISP-2 router has a static route to ISP-3. Islamabad (Headquarter) router receives multiple paths to different destinations from ISP-1 and ISP-2 through BGP updates. It might be better to go through ISP-1 for some routes and vice versa for some other routes. Moreover an administrator at Islamabad may want to implement some policy regarding use of paths going through different ISPs. We have implemented this routing policy using Local Preference attribute of BGP routes.

Many networks are running in Islamabad and different servers are hosted in both of them. These servers can be reached by users connected to the Internet. Internet has two paths to this network; one is through ISP-1 and the other is through ISP-2, so we implement load balancing here using MED attribute of BGP. Servers in network 172.10.1.0 are reached through ISP-1 and servers in network 2 are reached through ISP-2. ISP-2 has connection to ISP-3 and has static route to it. ISP-2 has redistributed this static route into BGP routing updates and this route will be advertised to eBGP peers Peshawar and Lahore and they in turn advertise it to Islamabad. So Islamabad has connectivity to ISP-3 network in its routing table and makes routing decision according to policy.

5.2 Policy Routing using BGP attributes

5.2.1 Policy Routing for Outgoing Traffic

Peshawar router is receiving information from ISP-1 about 172.10.8.0 - 172.10.15.0/21. Route map is configured on this router to change the local preference of routes received on this router.

Access list 1 is configured to identify networks that are nearer through Peshawar, i.e routes 172.10.8.0 to 172.10.11.0/22.

access-list 1 permit 172.10.8.0 0.0.3.255

Access list 2 is configured to identify networks that are not best reachable through Peshawar router i.e. 172.10.12.0 to 172.10.15.0/22.

access-list 2 permit 172.10.12.0 0.0.3.255

The 1st group of networks is assigned with local preference of 200 using route map. The 2nd group of networks is assigned with local preference of 50 at boundary router i.e Peshawar. This route map is applied to all routes coming in from the neighbour ISP-1. Now the Peshawar router will advertise these routes with changed local preference attribute value to Islamabad.

Similarly Lahore router is also receiving information from ISP-2 about 172.10.8.0 to 172.10.15.0/21. Access lists are configured on Lahore router like Peshawar router and the 1st group is assigned with local preference of 50 and second group is assigned with local preference of 200; just opposite to Peshawar router. As Lahore router has better path to 2nd group of networks i.e. 172.10.12.0 to 172.10.15.0/22. Lahore router has also static route to Internet router i.e. ISP-3 and this static route is redistributed into BGP routing updates. Now Lahore router will advertise this redistributed route as well as other routes with changed attributes information to Islamabad router. Now Islamabad router has two paths to both groups of networks, but route to 1st group is coming with higher preference from Peshawar router while route to 2nd group is coming with higher preference from Lahore router. So Islamabad Router will route traffic destined to group 1 through Peshawar router and traffic destined to group 2 through Lahore router.

5.2.2 Policy routing for Incoming Traffic

We have two networks in headquarter (Islamabad) and these networks are advertised to peers. Peshawar router is receiving two network routes from Islamabad router i.e. 172.10.1.0/24 and 172.10.2.0/24. Peshawar router is advertising these networks to ISP-1. Our policy dictates that traffic coming in from Internet to the internal network 172.10.1.0 should pass through Peshawar router. So we configure Peshawar router to advertise 172.10.1.0 route with higher MED and 172.10.2.0 route with lower MED.

Similarly Lahore router is also receiving information about these two networks from Islamabad and advertises this information to ISP-2. But our policy dictates that traffic coming in from Internet to network 172.10.2.0 should pass through Lahore. So our Lahore router will advertise this network to ISP-2 with higher MED and the other network with lower MED value assigned. ISP-1 and ISP-2 will exchange this routing information with each other but both routers have updated MED value. So ISP-1 and ISP-2 will route traffic to network 172.10.1.0 through Peshawar router and traffic to 172.10.2.0 through Lahore router.

5.2.3 Conclusion and Results

Now Islamabad router has routes to all networks but it will use the path through Peshawar router for destinations 172.10.8.0 to 172.10.11.0/21. Islamabad router will forward the traffic through Lahore router for destinations 172.10.12.0 to 172.10.15.0/21 and 172.10.100.100/32.

Now the shortest path is chosen and both links are utilized so that no link bandwidth is wasted and load balancing is been done. Fault tolerance is also present as one path fails traffic will be routed through other path. This is the main feature of BGP routing protocol; to support policy based routing. Similarly ISPs have route to both internal networks 172.10.1.0 and 172.10.2.0, and ISPs will route traffic to these internal networks according to our policy and doing load balancing between two available paths. Traffic to network 172.10.1.0 is coming through Peshawar router and traffic to network 172.10.2.0 is coming through Lahore router.

Case Study 2

5.3 iBGP: Black Hole Routing and Rule of Synchronization

iBGP is also needed in case of transit ISP as shown in Figure 5-2 below i.e. when ISP is learning some BGP routes at one end and transferring them to other ISP on the other end. In this case BGP routes along with their path attributes must be conveyed to other end of AS and iBGP is used inside AS for this purpose. If BGP routes are redistributed in IGP to propagate towards the other end they will lose their path attributes and other ISP at receiving end cannot implement policy control.

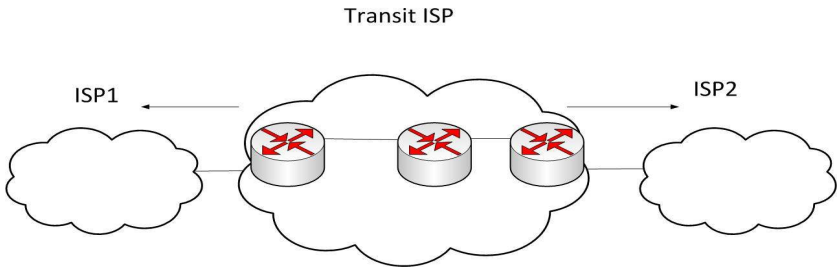

Figure 5-2 Transit ISP requires iBGP

Interior Border Gateway Protocol (iBGP) is needed to convey routing information within organization when there are multiple exit points towards the Internet and routers within organization should prefer one exit point for certain routes and vice versa. But iBGP is vulnerable to black hole problem and routing loops due to some iBGP specific rules. For instance, edge router in ISP or organization has BGP peer relationship with external AS and exchange BGP routes with this peer. There is shared connection link between edge router and external peer, so edge router has connected route to reach the external peer.

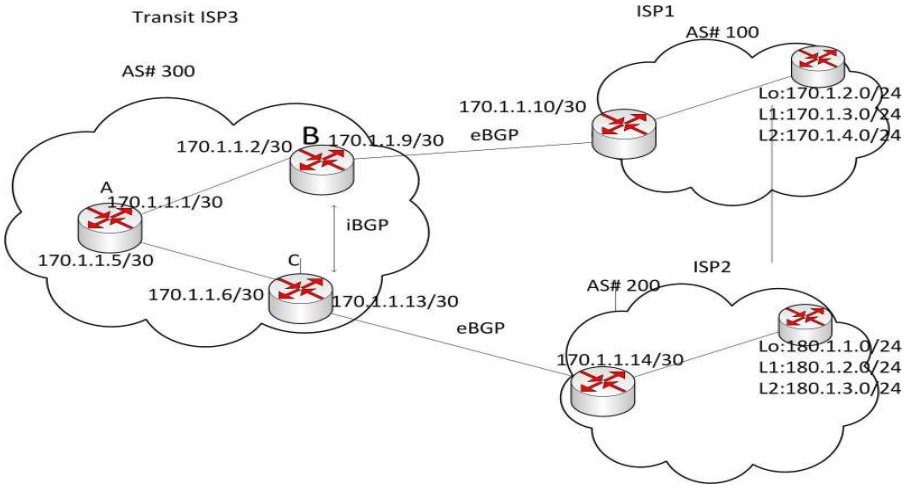

Figure 5-3 CASE STUDY2 topology diagram

When this edge router forwards BGP routes to internal peers, it doesn't change the next-hop by default and if internal routers don't have reachability information about next-hop address, it cannot use this routing information.

For example in this case study AS 300 is learning some routes from ISP-1 and some from ISP-2. Internal routers in this organization must be supplied with this information to choose the right shortest exit point i.e. Router A should forward packets destined to 170.1.2.0 towards ISP-1 and packets destined to network 180.1.1.0 towards ISP-2. In this case study we will study the possible configuration difficulties with iBGP. In figure 5-3 is topology diagram for this case study.

```
A#sh ip route

Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is 170.1.1.65 to network 0.0.0.0

    170.1.0.0/16 is variably subnetted, 10 subnets, 3 masks
C       170.1.1.33/32 is directly connected, Loopback0
D       170.1.1.8/30 [90/2681856] via 170.1.1.2, 00:02:20, Serial0/0
D       170.1.1.12/30 [90/2681856] via 170.1.1.6, 00:01:59, Serial0/1
B       170.1.3.0/24 [200/0] via 170.1.2.1, 00:13:47
D       170.1.2.0/24 [90/2681856] via 170.1.1.2, 00:02:20, Serial0/0
C       170.1.1.0/30 is directly connected, Serial0/0
C       170.1.1.4/30 is directly connected, Serial0/1
B       170.1.4.0/24 [200/0] via 170.1.2.1, 00:13:47
S       170.1.1.97/32 is directly connected, Serial0/1
S       170.1.1.65/32 is directly connected, Serial0/0
B*   0.0.0.0/0 [200/0] via 170.1.1.65, 00:13:48
A#
```

Figure 5-4 Routing table of router A

Here Router A is receiving routing updates from B and C about 170.1.2.0/24 – 170.1.4.0/24 and 180.1.1.0/24-180.1.3.0/24, but it is not considering some of these routes as valid BGP routes and these routes are not installed in routing table (shown in Figure 5-4 above and 5-5 below). Routes 180.1.1.0, 180.1.2.0, and 180.1.3.0 are not present in this routing table, as router A doesn't have routing information towards next-hop mentioned in these routes that is 180.1.1.1.

```
A#sh ip bgp
BGP table version is 18, local router ID is 170.1.1.33
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
* i0.0.0.0          180.1.1.1          0     100          0   200 i
```

```
*>i                 170.1.2.1        0          100             0      100 i
r>i170.1.2.0/24     170.1.2.1        0          100             0      100 i
*>i170.1.3.0/24     170.1.2.1        0          100             0      100 i
*>i170.1.4.0/24     170.1.2.1        0          100             0      100 i
*i180.1.1.0/24      180.1.1.1        0          100             0      200 i
*i180.1.2.0/24      180.1.1.1        0          100             0      200 i
*i180.1.3.0/24      180.1.1.1        0          100             0      200 i
A#
```

Figure 5-5 Routing table of router A (showing invalid routes)

Solution:

1) Change next-hop to edge routers (router B or C) interface address when advertising routing information to internal peers (router C) by using command:

   Neighbor Neighbor_iD next-hop self

2) Create static route to external peer address at edge routers and redistribute this information to internal routers, so that they have information about how to reach external next-hop.

5.3.1 iBGP Split Horizon

iBGP peer doesn't forward routing information learned from one iBGP peer to other iBGP peer known as iBGP split horizon. iBGP has this rule to prevent routing loops as loop prevention mechanism provided by AS_Path attribute doesn't work in iBGP . iBGP routes are being exchanged among routers within single autonomous system ,therefore AS number is not prepended to AS_Path attribute. But this may result in inaccurate information, partial information at different routers which may result in black holing or routing loop. For example suppose following partial peering:

- Router A and B are iBGP peers,
- Router A and C are iBGP peers,
- But routers B & C are not iBGP peers with each other.

When router B has learned external routes from ISP-1 then it will advertise these routes to router A. As router C is receiving external routes from ISP-2 it will advertise these routes to router A. But router A doesn't forward information learned from B to C and from C to B due to the rule of split horizon. Therefore router C doesn't have information about prefixes 170.1.2.0/24- 170.1.4.0/24 and router B has not learned information about prefixes 180.1.1.0/24- 180.1.3.0/24. This is shown in the output of router B in Figure 5-6 below.

B#sh ip route

Gateway of last resort is 170.1.2.1 to network 0.0.0.0

170.1.0.0/16 is variably subnetted, 10 subnets, 3 masks
S 170.1.1.33/32 is directly connected, Serial0/0
C 170.1.1.8/30 is directly connected, Serial0/1
D 170.1.1.12/30 [90/3193856] via 170.1.1.1, 00:07:51, Serial0/0
B 170.1.3.0/24 [20/0] via 170.1.2.1, 00:25:13
S 170.1.2.0/24 is directly connected, Serial0/1
C 170.1.1.0/30 is directly connected, Serial0/0
D 170.1.1.4/30 [90/2681856] via 170.1.1.1, 00:08:12, Serial0/0
B 170.1.4.0/24 [20/0] via 170.1.2.1, 00:25:13
D 170.1.1.97/32 [90/2681856] via 170.1.1.1, 00:08:12, Serial0/0
C 170.1.1.65/32 is directly connected, Loopback0
B* 0.0.0.0/0 [20/0] via 170.1.2.1, 00:25:39

Figure 5-6 Routing table of router B (has all routes)

Similarly, router C also has learned partial information as shown in its routing table in Figure 5-7 below.

C#sh ip route

Gateway of last resort is 180.1.1.1 to network 0.0.0.0

170.1.0.0/16 is variably subnetted, 8 subnets, 3 masks
S 170.1.1.33/32 is directly connected, Serial0/1
D 170.1.1.8/30 [90/3193856] via 170.1.1.5, 00:09:44, Serial0/1
C 170.1.1.12/30 is directly connected, Serial0/0
D 170.1.2.0/24 [90/3193856] via 170.1.1.5, 00:09:44, Serial0/1
D 170.1.1.0/30 [90/2681856] via 170.1.1.5, 00:09:44, Serial0/1
C 170.1.1.4/30 is directly connected, Serial0/1
C 170.1.1.97/32 is directly connected, Loopback0
D 170.1.1.65/32 [90/2681856] via 170.1.1.5, 00:09:44, Serial0/1

180.1.0.0/16 is variably subnetted, 4 subnets, 2 masks
B 180.1.1.0/24 [20/0] via 180.1.1.1, 00:27:05
S 180.1.1.1/32 is directly connected, Serial0/0
B 180.1.3.0/24 [20/0] via 180.1.1.1, 00:27:05
B 180.1.2.0/24 [20/0] via 180.1.1.1, 00:27:06
B* 0.0.0.0/0 [20/0] via 180.1.1.1, 00:27:34

Figure 5-7 Routing table of router C (has learned routes)

Similarly, If only B & C are iBGP peers, B has learned routing information from ISP-1 about 170.1.2.0/24- 170.1.4.0/24 then it will forward this routing information to router C and

router C has learned routing information from ISP-2 about 180.1.1.0/24 - 180.1.4.0/24 then it will forward this routing information to router B.

Now suppose packets destined to this 170.1.2.0/24 network arrived at router C from ISP-2 or originated inside from router C. Router C will look its routing table to route the packets towards destination.

This table tells C to reach 170.1.3.0 traffic should be forwarded towards next-hop 170.1.1.10/24 (IP address of eBGP peer). Now router C looks in its routing table again to determine path towards 170.1.1.10/24 (Router B).

This look-up determines 170.1.1.5 (Router A) as next-hop which is directly connected and traffic is now forwarded towards Router A. Now Router A will look its routing table to route traffic towards 170.1.3.0/24. Routing table of router A as shown in Figure 5-8 below does not contain the reachability information to 170.1.3.0/24 as it has not been in iBGP peer relationship with router B.

```
A#sh ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

     170.1.0.0/16 is variably subnetted, 10 subnets, 3 masks
C       170.1.1.33/32 is directly connected, Loopback0
D       170.1.1.8/30 [90/2681856] via 170.1.1.2, 00:02:20, Serial0/0
D       170.1.1.12/30 [90/2681856] via 170.1.1.6, 00:01:59, Serial0/1
D       170.1.2.0/24 [90/2681856] via 170.1.1.2, 00:02:20, Serial0/0
C       170.1.1.0/30 is directly connected, Serial0/0
C       170.1.1.4/30 is directly connected, Serial0/1
S       170.1.1.97/32 is directly connected, Serial0/1
S       170.1.1.65/32 is directly connected, Serial0/0
```

Figure 5-8 Routing table of router A (with no iBGP)

But router A did not learn this information previously and becomes a black-hole point. Therefore traffic is discarded here resulting in black hole or forwarded towards some default route which may result in routing loops. Connectivity test output in Figure 5-9 proves the same.

```
C#ping 170.1.3.1
Type escape sequence to abort.
Sending 5, 100-byte ICMp Echos to 170.1.3.1, timeout is 2 seconds:
U.U.U
Success rate is 0 percent (0/5)

C#traceroute 170.1.3.1
Type escape sequence to abort.
Tracing the route to 170.1.3.1

    1 170.1.1.5 [AS 200] 8 msec 8 msec 8 msec
```

```
     2 170.1.1.5 [AS 200] !H  !H   *

C#
```

Figure 5-9 Connectivity test (ping and trace route) from router C

iBGP has rule of synchronization to prevent this problem, this rule states that never consider a route learned through iBGP peer unless a same route is learned through IGP protocol i.e. EIGRP or OSPF. Therefore if synchronization was not disabled on router C, router C would use the route for 170.1.2.0 at first place.

In other words this rule implies that all BGP routes are redistributed in IGP, so that all routers inside organization have consistent information and routing loops and black-holes can be avoided.

But redistribution of BGP routes into IGP is not feasible, an alternate solution to this problem is full mesh iBGP connection i.e. every internal router is peer with every other internal router, so independent iBGP connection is established among all internal routers and information is communicated to all and we can disable rule of synchronization in this case. But again full mesh iBGP is difficult to configure and maintain by routers.

5.4 Many solutions

5.4.1 Route Reflectors

Route reflectors are used to reduce the number of iBGP connections required. Without router reflector, AS having N number of routers requires n*(n-1)/2 number of iBGP connections to achieve full mesh connectivity as shown in Figure 5-10 below.

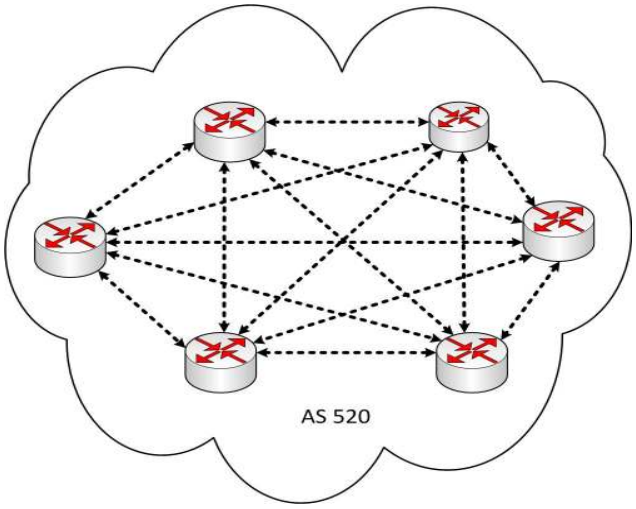

Figure 5-10 BGP full meshed

Route reflector concept is somewhat similar to designated router concept in OSPF routing protocol, here one router inside AS is configured as route reflector and all other routers work as client. Now every client needs to connect to router reflector to achieve full mesh connectivity and number of required iBGP connections are reduced to N-1.

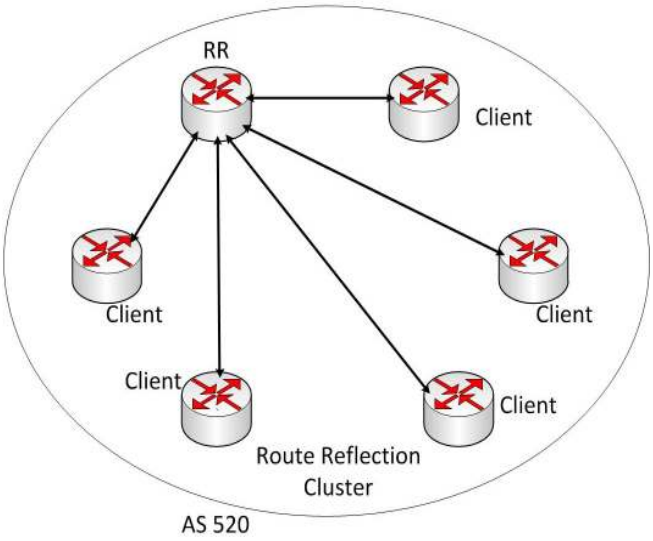

Figure 5-11 BGP route reflector

Route reflector learns BGP route information from one client and advertises this information to all other clients. In this way this information is propagated to whole AS without full mesh iBGP as shown in figure 5-11 above.

Case Study 3

5.5 BGP Attacks and Misconfiguration on sample network

In this case study we will show how BGP information can be manipulated maliciously. Figure 5-12 below shows the topology of case study 3.

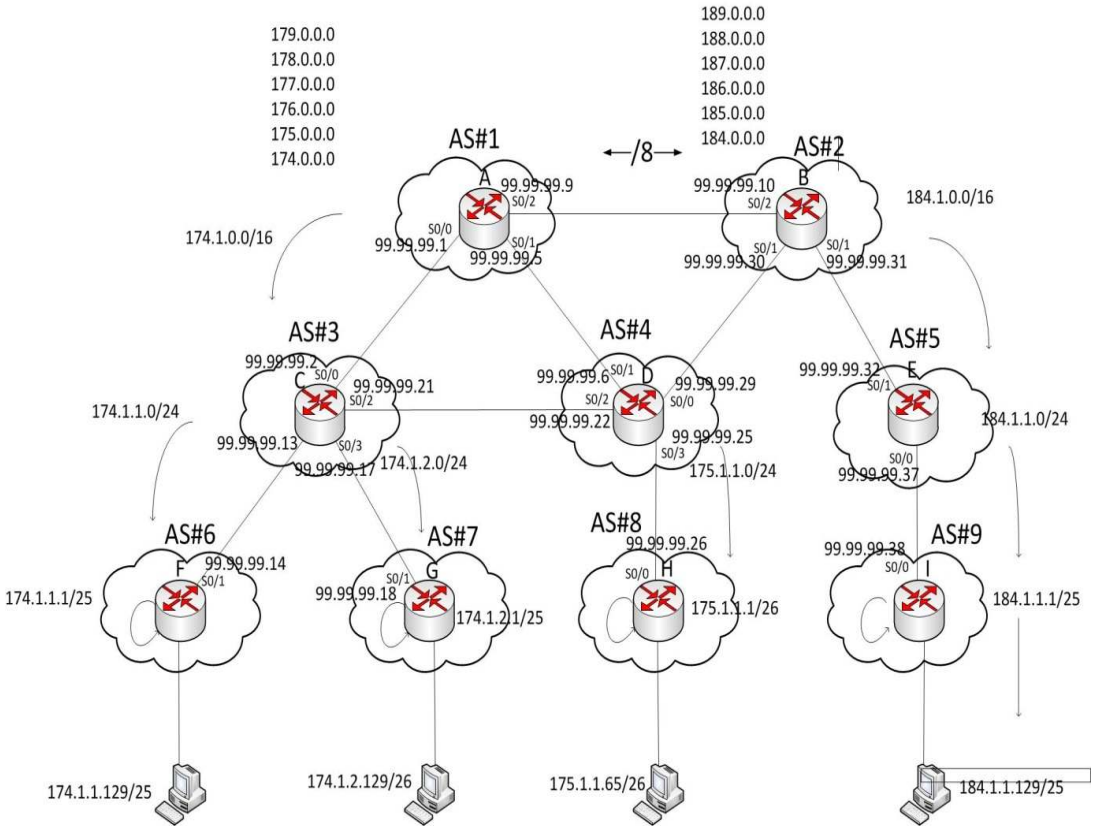

Figure 5-12 CASE STUDY 3 topology diagram

Each router represents single AS, in reality there are many routers inside any organization but most of the time there is one router communicating with outside world. Keeping this thing in mind and in order to make the topology simple we represent each AS with single router. Figure 5-12 shows the topology diagram for case study emulating BGP attacks and problems. Table 5-13 below contains router name to AS number mapping.

Router name and their corresponding AS number

Router Name	A	B	C	D	E	F	G	H	I
AS #	1	2	3	4	5	6	7	8	9

Figure 5-13 Router name and their corresponding AS number table

AS number 1 and 2 represent Tier 1 ISP, AS 3, 4 and 5 represent Tier 2 and AS 6, 7, 8 and 9 represent tier 3 ISP. Tier 1 ISP-A (ASN-1) owns prefixes 174.0.0.0/8 to 179.0.0.0/8

ISP-A assigns 174.1.0.0/16 to AS 3 and 175.1.0.0/16 to AS number-4. AS number 3 assigns sub-prefixes 174.1.1.0/24 and 174.1.2.0/24 to AS number 6 and 7 respectively. On the other hand ISP-B (AS number 2) holds prefixes 184.0.0.0/8 to 189.0.0.0/8, it assigns 184.1.0.0/16 to AS number 5, which assigns 184.1.1.0/24 to AS number 9. All autonomous systems exchange routing prefixes using BGP routing protocol with each other to represent the Internet at small scale. These autonomous systems successfully exchanged routing information with each other and have full end-to-end connectivity.

Routing table of router A (i.e. ISP-1) is shown in Figure 5-14 below.

```
A# sh ip bgp
BGP table version is 22, local router ID is 179.0.0.1
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  Weight     Path
*  174.0.0.0/8      0.0.0.0                32768      i
*>                  0.0.0.0             0  32768      i
s  174.1.0.0        99.99.99.42            0          2 4 3 i
s                   99.99.99.44            0          4 3 i
s>                  99.99.99.43         0  0          3 i
*  175.0.0.0/8      0.0.0.0               32768       i
*>                  0.0.0.0             0  32768      i
s  175.1.0.0        99.99.99.43            0          3 4 i
s                   99.99.99.42            0          2 4 i
s>                  99.99.99.44         0  0          4 i
*> 176.0.0.0/8      0.0.0.0             0  32768      i
*> 177.0.0.0/8      0.0.0.0             0  32768      i
*> 178.0.0.0/8      0.0.0.0             0  32768      i
*> 179.0.0.0/8      0.0.0.0             0  32768      i
*  184.0.0.0/8      99.99.99.44            0          4 2  i
*>                  99.99.99.42         0  0          2 i
*  185.0.0.0/8      99.99.99.44            0          4 2 i
  *>                99.99.99.42         0  0          2 i
*  186.0.0.0/8      99.99.99.44            0          4 2 i
*>                  99.99.99.42         0  0          2 i
*  187.0.0.0/8      99.99.99.44            0          4 2 i
*>                  99.99.99.42         0  0          2 i
*  188.0.0.0/8      99.99.99.44            0         42 i
*>                  99.99.99.42         0  0          2 i
*  189.0.0.0/8      99.99.99.44            0          4 2 i
*>                  99.99.99.42         0  0          2 i

Routing table of  AS number 1
```

Figure 5-14 Initial complete routing table of router A (ASN number1)

All organization routes are present in routing table of router at AS number 1 and it can successfully route information to any prefix as shown in tests conducted below in Figure 5-15

below.

```
A# ping 174.1.1.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 174.1.1.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 16/28/48 ms
A# ping 175.1.1.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 175.1.1.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 20/48/116 ms
A# ping 184.1.1.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 184.1.1.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 28/43/92 ms
A# ping 174.1.2.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 174.1.2.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 20/40/56 ms
A#
```

Figure 5-15 Successful ping test from A

Similarly routing table at AS number 2 has routes to all organizations and can successfully route data to all destinations, as shown in routing table in figure 5-16 below.

routing table of AS number 2

```
B# sh ip bgp
BGP table version is 19, local router ID is 189.0.0.1
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     Weight     Path
*  174.0.0.0/8      99.99.99.44              0                4 1 i
*>                   99.99.99.41              0          0     1 i
*> 174.1.0.0        99.99.99.44              0                4 3 i
*  175.0.0.0/8      99.99.99.44              0                4 1 i
*>                   99.99.99.41              0          0     1 i
*> 175.1.0.0        99.99.99.44              0          0     4 i
*  176.0.0.0/8      99.99.99.44                         0     4 1 i
*>                   99.99.99.41              0          0     1 i
*  177.0.0.0/8      99.99.99.44                         0     4 1 i
*>                   99.99.99.41              0          0     1 i
*  178.0.0.0/8      99.99.99.44                         0     4 1 i
*>                   99.99.99.41              0          0     1 i
*  179.0.0.0/8      99.99.99.44                         0     4 1 i
*>                   99.99.99.41              0          0     1 i
*  184.0.0.0/8      0.0.0.0                             32768 i
*>                  0.0.0.0                   0          32768 i
s> 184.1.0.0        99.99.99.45              0          0     5i
```

```
  s> 184.1.1.0/24    99.99.99.45              0          59 i
*> 185.0.0.0/8      0.0.0.0          0       32768      i
*> 186.0.0.0/8      0.0.0.0          0       32768      i
*> 187.0.0.0/8      0.0.0.0          0       32768      i
*> 188.0.0.0/8      0.0.0.0          0       32768      i
*> 189.0.0.0/8      0.0.0.0          0       32768      i
```

Figure 5-16 Intial complete routing table of B

All organization routes are present in routing table of router at AS number 2 and it can successfully route information to any prefix as shown in tests conducted below in Figure 5-17.

```
B#ping 174.1.1.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 174.1.1.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 20/40/68 ms
B#ping 184.1.1.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 184.1.1.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 16/60/96 ms
B#ping 175.1.1.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 175.1.1.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 28/42/76 ms
B#ping 174.1.2.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 174.1.2.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 20/49/112 ms
```

Figure 5-17 Successful connectivity tests from B

5.6 Possible problems

As stated in attacks section, any BGP router participating in BGP domain can generate incorrect/false information to peers. This may be intentional, due to mistake or the router has been hacked and being used to generate this false information. For example if any router in AS 6,7 or 8 originates routing updates 184.1.1.0/24 towards AS number 3 and 4. AS number 3 and 4 will believe this information to be true, as there is no route advertisement / attributes authentication mechanism in BGP. These AS consider this information as valid and install this route in its routing table and forward this information to upstream ISPs i.e. AS number 1.

5.6.1 False Information Origination

Suppose if AS number 6 install false information about prefix 184.1.1.0/24 in its routing table and forward it to upstream AS number 3. Routing table of AS number 6 (F) routers populated with this false information to spread as shown in Figure 5-18 below.

```
F#sh ip bgp
BGP table version is 20, local router ID is 174.1.1.1
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            Weight     Path
*> 174.0.0.0/8      99.99.99.43              0      3 1 i
*> 174.1.0.0        99.99.99.43              0      3 i
s> 174.1.1.0/25     0.0.0.0              32768      i
r> 174.1.1.0/24     0.0.0.0              32768      i
*> 175.0.0.0/8      99.99.99.43              0      3 1 i
*> 175.1.0.0        99.99.99.43              0      3 4 i
*> 176.0.0.0/8      99.99.99.43              0      3 1 i
*> 177.0.0.0/8      99.99.99.43              0      3 1 i
*> 178.0.0.0/8      99.99.99.43              0      3 1 i
*> 179.0.0.0/8      99.99.99.43              0      3 1 i
*> 184.0.0.0/8      99.99.99.43              0      3 1 2 i
*> 184.1.1.0/24     0.0.0.0              32768      i
*> 185.0.0.0/8      99.99.99.43              0      3 1 2 i
*> 186.0.0.0/8      99.99.99.43              0      3 1 2 i
*> 187.0.0.0/8      99.99.99.43              0      3 1 2 i
*> 188.0.0.0/8      99.99.99.43              0      3 1 2 i
*> 189.0.0.0/8      99.99.99.43              0      3 1 2 i
```

Figure 5-18 Router F originating false information

This BGP table is advertised to AS number 6 and it will also install this route in its routing table. Bad information is conveyed to and accepted by router C as shown in its routing table below in Figure 5-19.

```
C#sh ip bgp
BGP table version is 20, local router ID is 99.99.99.43
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 Weight     Path
*  174.0.0.0/8      99.99.99.44                0      4 1 i
*>                  99.99.99.41             0  0      1 i
*  174.1.0.0        0.0.0.0                    32768  i
*>                  0.0.0.0                 0  32768  i
s> 174.1.1.0/24     99.99.99.46             0  0 6    i
s> 174.1.2.0/24     99.99.99.47             0  0 7    i
*  175.0.0.0/8      99.99.99.44                0 4 1  i
*>                  99.99.99.41             0  0 1    i
*> 175.1.0.0        99.99.99.44             0  0 4    i
*  176.0.0.0/8      99.99.99.44                0 4 1  i
*>                  99.99.99.41             0  0 1    i
*  177.0.0.0/8      99.99.99.44                0 4 1  i
*>                  99.99.99.41             0  0 1    i
*  178.0.0.0/8      99.99.99.44                0 4 1   i
```

```
*>                99.99.99.41          0         0      1 i
*  179.0.0.0/8    99.99.99.44                    0      4 1 i
*>                99.99.99.41          0         0      1 i
*  184.0.0.0/8    99.99.99.44                    0      4 2 i
*>                99.99.99.41                    0      1 2 i
*> 184.1.1.0/24   99.99.99.46          0         0      6 i
*  185.0.0.0/8    99.99.99.44                    0      4 2 i
*>                99.99.99.41                    0      1 2 i
*  186.0.0.0/8    99.99.99.44                    0      4 2 i
*>                99.99.99.41                    0      1 2 i
*  187.0.0.0/8    99.99.99.44                    0      4 2 i
*>                99.99.99.41                    0      1 2 i
*  188.0.0.0/8    99.99.99.44                    0      4 2 i
*>                99.99.99.41                    0      1 2 i
*  189.0.0.0/8    99.99.99.44                    0      4 2 i
*>                99.99.99.41                    0      1 2 i
```

Figure 5-19 False information from F received at C

AS number 3, 4 and 1 receives this information and installs this route in routing table, though it is false. They will believe it as valid and start forwarding data packets towards it resulting in blackhole/denial of service as shown in connectivity test performed on this router in Figure 5-20 below.

```
C#ping 184.1.1.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 184.1.1.1, timeout is 2 seconds:
U.U.U
Success rate is 0 percent (0/5)
D#ping 184.1.1.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 184.1.1.1, timeout is 2 seconds:
U.U.U
Success rate is 0 percent (0/5)
A#ping 184.1.1.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 184.1.1.1, timeout is 2 seconds:
U.U.U
Success rate is 0 percent (0/5)
A#
```

Figure 5-20 Failed ping Test from router C

AS number 1 and 4 will also forward this route to AS number 2, but AS number 2 will not install this route in its routing table as it already has route with same mask with smaller AS_Path length. Therefore AS 2, 5 and 9 will ignore this false route advertisement and hence can forward traffic successfully. Figure 5-21 below show the success full connectivity test from router in AS number 2.

```
B#ping 184.1.1.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 184.1.1.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 4/36/84 ms
B#
```

Figure 5-21 Successful ping test from router B

5.6.2 De-Aggregation

Similarly if some malicious or misconfigured router advertises other AS prefixes with more specific prefixes instead of summary routes. This route advertisement is believed by almost all AS and traffic destined to specific prefix being advertised is black holed.

Suppose AS number 9 originates prefix 174.1.2.0/25 to AS 5. This information is propagated to all AS and all will believe this information and send traffic destined to 174.1.2.0/25 towards AS number 9 due to longest match rule. Routing table of router E with maliciously modified information is shown in Figure 5-22 below.

```
E#sh ip bgp
BGP table version is 22, local router ID is 99.99.99.45
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  Weight  Path
*> 174.0.0.0/8      99.99.99.42                      0  2 1 i
*> 174.1.0.0        99.99.99.42                      0  2 4 3 i
*> 174.1.2.0/25     99.99.99.49             0        0  9 i
*> 175.0.0.0/8      99.99.99.42                      0  2 1 i
*> 175.1.0.0        99.99.99.42                      0  2 4 i
*> 176.0.0.0/8      99.99.99.42                      0  2 1 i
*> 177.0.0.0/8      99.99.99.42                      0  2 1 i
*> 178.0.0.0/8      99.99.99.42                      0  2 1 i
*> 179.0.0.0/8      99.99.99.42                      0  2 1 i
*> 184.0.0.0/8      99.99.99.42             0        0  2 i
*> 184.1.0.0        0.0.0.0                 0    32768  i
*> 184.1.1.0/24     99.99.99.49             0        0  9 i
*> 185.0.0.0/8      99.99.99.42             0        0  2 i
*> 186.0.0.0/8      99.99.99.42             0        0  2 i
*> 187.0.0.0/8      99.99.99.42             0        0  2 i
*> 188.0.0.0/8      99.99.99.42             0        0  2 i
*> 189.0.0.0/8      99.99.99.42             0        0  2 i

E#ping 174.1.2.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 174.1.2.1, timeout is 2 seconds:
U.U.U
Success rate is 0 percent (0/5)
E#
```

Figure 5-22 Router with de-aggregated false information

But in de-aggregation case, more specific prefix is advertised maliciously and it got propagated through all networks. Below is Figure 5-23 showing corrupted routing table of router C.

```
C#sh ip bgp
BGP table version is 28, local router ID is 99.99.99.43
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 Weight     Path
* 174.0.0.0/8       99.99.99.44                    0      4 1 i
*>                  99.99.99.41              0     0      1 i
* 174.1.0.0         0.0.0.0                    32768      i
*>                  0.0.0.0                  0 32768      i
s> 174.1.1.0/24     99.99.99.46              0     0      6 i
s> 174.1.2.0/25     99.99.99.44                    0      4 2 5 9 i
s> 174.1.2.0/24     99.99.99.47              0     0      7 i
* 175.0.0.0/8       99.99.99.44                    0      4 1 i
*>                  99.99.99.41              0     0      1 i
*> 175.1.0.0        99.99.99.44              0     0      4 i
* 176.0.0.0/8       99.99.99.44                    0      4 1 i
*>                  99.99.99.41              0     0      1 i
* 177.0.0.0/8       99.99.99.44                    0      4 1 i
*>                  99.99.99.41              0     0      1 i
* 178.0.0.0/8       99.99.99.44                    0      4 1 i
*>                  99.99.99.41              0     0      1 i
* 179.0.0.0/8       99.99.99.44                    0      4 1 i
*>                  99.99.99.41              0     0      1 i
* 184.0.0.0/8       99.99.99.44                    0      4 2 i
*>                  99.99.99.41                    0      1 2 i
*> 184.1.1.0/24     99.99.99.46              0     0      6 i
* 185.0.0.0/8       99.99.99.44                    0      4 2 i
*>                  99.99.99.41                    0      1 2 i
* 186.0.0.0/8       99.99.99.44                    0      4 2 i
*>                  99.99.99.41                    0      1 2 i
* 187.0.0.0/8       99.99.99.44                    0      4 2 i
*>                  99.99.99.41                    0      1 2 i
* 188.0.0.0/8       99.99.99.44                    0      4 2 i
*>                  99.99.99.41                    0      1 2 i
* 189.0.0.0/8       99.99.99.44                    0      4 2 i
*>                  99.99.99.41                    0      1 2 i
C#  ping 174.1.2.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 174.1.2.1, timeout is 2 seconds:
U.U.U
Success rate is 0 percent (0/5)
C#
```

Figure 5-23 Routing table and ping results from router C

Case Study 4

5.7 Implementing practical security solutions for BGP Network

In this case study we have implemented possible practical security measures mentioned in chapter 4 to protect against BGP misconfiguration and attacks and have checked its effectiveness against most popular attacks in the history of BGP. Below in Figure 5-24 is topology diagram for this case study.

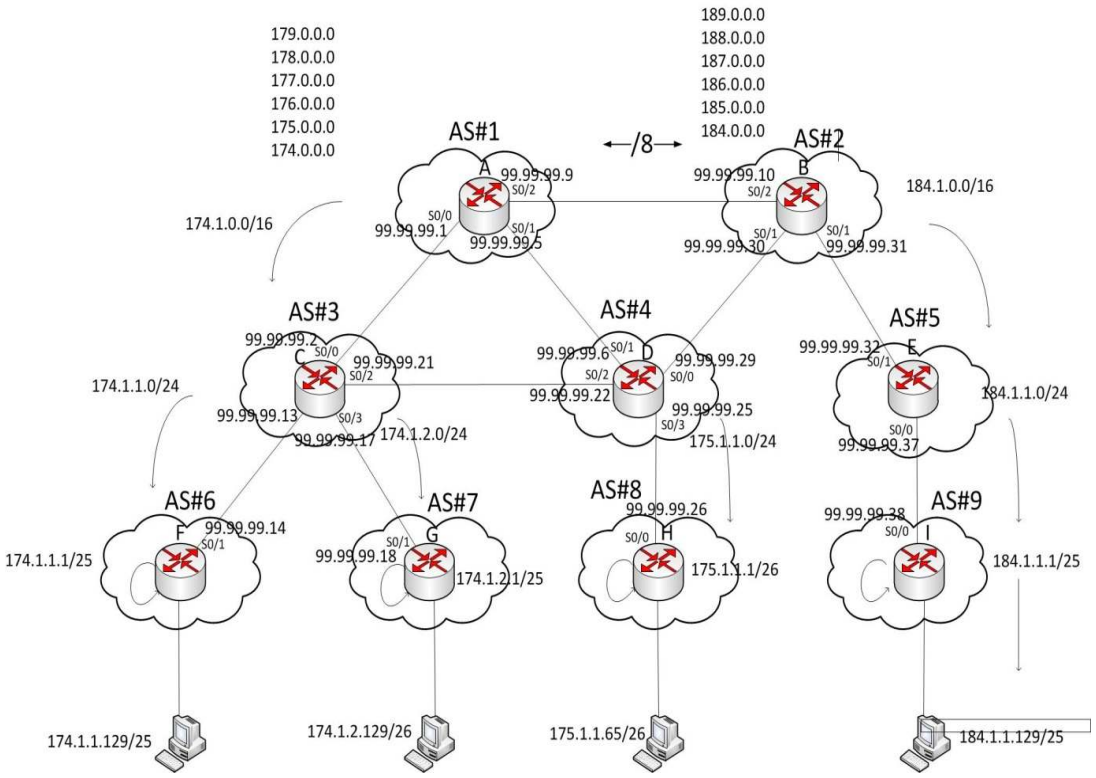

Figure 5-24 CASE STUDY 4 topology diagram

We have implemented these practical security measures on network configured in Case Study 3 and determine the security protection provided by these functions. Security functions implemented in this case study include:

- Route dampening to avoid problems associated with flapping routes.
- MD-5 based password protection between peers.
- Time to live based DoS attack protection from remote hosts.
- Limiting number of prefixes received.
- Limiting length of AS_Path.
- Implementing proper prefix filtering.

5.7.1 Prefix Filtering

According to our topology we have determined the following filtering rules:

AS number 1 to AS number 2- only 172.0.0.0/6 and block all bogons (defined in prefix a2b)

AS number 1 to AS number 3- only 172.0.0.0/6 and 184.0.0.0/5

AS number 1 from AS number 4 – only 172.0.0.0/6 and 184.0.0.0/5

AS number 2 from AS number 1 – only 184.0.0.0/5

AS number 2 from AS number 4 – only 172.0.0.0/6 and 184.0.0.0/5

AS number 2 from AS number 5 – only 172.0.0.0/6 and 184.0.0.0/5

AS number 3 from AS number 1 – only 174.1.0.0/16 and 175.1.0.0/16

AS number 3 from AS number 4 – only 174.1.0.0/16 and 172.0.0.0/6

AS number 3 from AS number 6 – only 172.0.0.0/6 and 184.0.0.0/5

AS number 3 from AS number 7 – only 172.0.0.0/6 and 184.0.0.0/5

AS number 4 from AS number 1 – only 175.1.0.0/6 and 184.0.0.0/5

AS number 4 from AS number 3 – only 175.1.0.0/6 and 184.0.0.0/5

AS number 4 from AS number 2 – only 175.1.0.0/16, 174.1.0.0/16 and 172.0.0.0/6

AS number 4 from AS number 8 – only 172.0.0.0/6 and 184.0.0.0/5

AS number 5 from AS number 2 – only 184.1.0.0/16

AS number 5 from AS number 9 – only 172.0.0.0/6 and 184.0.0.0/5

AS number 6 from AS number 3 – only 174.1.1.0/24

AS number 7 from AS number 3 – only 174.1.2.0/24

AS number 8 from AS number 4 – only 175.1.1.0/24

AS number 9 from AS number 5 – only 184.1.1.0/24

BGP routing protocol is configured on this topology with the above mentioned security measures and prefix filter rules. Now routers F, G and H are generating false information about 184.1.1.0/24 and router I is generating false information about 174.1.1.0/24. The routing table of router F is shown in figure 5-25 below.

```
F#sh ip  bgp
BGP table version is 8, local router ID is 174.1.1.1
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     Weight     Path
*> 172.0.0.0/6      99.99.99.43                         0     3 1 i
s> 174.1.1.0/25     0.0.0.0                  0      32768     i
```

```
r> 174.1.1.0/24     0.0.0.0                  32768      i
*> 184.0.0.0/5      99.99.99.43                  0      3 1 2 i
*> 184.1.1.0/24     0.0.0.0          0       32768      i
F#
```

Figure 5-25 Router F originating false information

Routing table of router F has entry for false route 184.1.1.0/24 and it advertises this incorrect route reachability information to router C, which may propagate it further to router A and router D.

Similarly router G and H are also generating false information about this prefix and advertising it to upwards routers for advertising this information in routing domain. Figure 5-26 below displays the routing table of router G.

```
G#sh ip bgp
BGP table version is 8, local router ID is 174.1.2.1
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     Weight     Path
*> 172.0.0.0/6      99.99.99.43                        0      3 1 i
s> 174.1.2.0/25     0.0.0.0                  0       32768      i
r> 174.1.2.0/24     0.0.0.0                           32768      i
*> 184.0.0.0/5      99.99.99.43                        0      3 1 2 i
*> 184.1.1.0/24     0.0.0.0                  0       32768      i

G#ping 174.1.2.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 174.1.2.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/4 ms
G#ping 184.1.1.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 184.1.1.1, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)
```

Figure 5-26 Router G advertising unreachable route

Its evident from above output that router G has incorrect information to 184.1.1.0/24 network (ping being failed).

Routing table of router H and its connectivity tests are shown in Figure 5-27 below.

```
H#sh ip bgp
BGP table version is 7, local router ID is 175.1.1.1
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     Weight     Path
*> 172.0.0.0/6      99.99.99.44                       0       4 1 i
s> 175.1.1.0/25     0.0.0.0                0          32768       i
*> 175.1.1.0/24     0.0.0.0                           32768       i
*> 184.0.0.0/5      99.99.99.44                       0       4 2 i
*> 184.1.1.0/24     0.0.0.0                0          32768       i
H# ping 184.1.1.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 184.1.1.1, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)
H#ping 174.1.2.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 174.1.2.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 44/64/104 ms
H#
```

Figure 5-27 Router H advertising unreachable route

Similarly routing table of router I is shown in Figure 5-28 below.

```
I#sh ip bgp
BGP table version is 9, local router ID is 184.1.1.1
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     Weight     Path
*> 172.0.0.0/6      99.99.99.45                       0       5 2 1 i
*> 174.1.2.0/25     0.0.0.0                0          32768       i
*> 184.0.0.0/5      99.99.99.45                       0       5 2 i
s> 184.1.1.0/25     0.0.0.0                0          32768       i
*> 184.1.1.0/24     0.0.0.0                           32768       i
I# ping 184.1.1.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 184.1.1.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms
I#ping 174.1.2.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 174.1.2.1, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)
I#
```

Figure 5-28 Router I avoided false routes

But due to implementation of proper security filtering at router C, D and E, they will not accept and propagate this false information to upwards routers and hence avoid disconnectivity that happened in case study 3.

Routing table of C and ping tests are shown in Figure 5-29 below.

```
C#sh ip bgp
BGP table version is 7, local router ID is 99.99.99.43
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     Weight     Path
*> 172.0.0.0/6      99.99.99.41              0          0      1 i
*  174.1.0.0        0.0.0.0                             32768  i
*>                  0.0.0.0                  0          32768  i
s> 174.1.2.0/24     99.99.99.47              0          0      7 i
*> 175.1.0.0        99.99.99.44              0          0      4 i
*  184.0.0.0/5      99.99.99.44                         0      4 2 i
*>                  99.99.99.41                         0      1 2 i
C#
C#ping 174.1.2.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 174.1.2.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 4/27/68 ms
C#ping 184.1.1.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 184.1.1.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 28/61/132 ms
C#
```

Figure 5-29 Router C routing table and successful connectivity test

Correct routing table at router E and ping tests from router E are shown in Figure 5-30 below.

```
E#
E#sh ip bgp
BGP table version is 7, local router ID is 99.99.99.45
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     Weight     Path
*> 172.0.0.0/6      99.99.99.42                         0      2 1 i
*> 184.0.0.0/5      99.99.99.42              0          0      2 i
*> 184.1.0.0        0.0.0.0                  0          32768  i
*> 184.1.1.0/24     99.99.99.49              0          0      9 i
E#
E#
E#
E#ping 184.1.1.1
```

```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 184.1.1.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 8/27/52 ms
E#ping 174.1.2.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 174.1.2.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 28/42/72 ms
```

Figure 5-30 Router E still have connectivity

Core router A remains undisturbed by this false information as shown in its routing table below in Figure 5-31.

```
A#
A#sh ip bgp
BGP table version is 7, local router ID is 99.99.99.41
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     Weight     Path
*  172.0.0.0/6      0.0.0.0                        32768      i
*>                  0.0.0.0                 0      32768      i
s> 174.1.0.0        99.99.99.43             0          0      3i
s> 175.1.0.0        99.99.99.44             0          0      4 i
*  184.0.0.0/5      99.99.99.44                        0      4 2 i
*>                  99.99.99.42             0          0      2 i
A#
A#ping 174.1.2.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 174.1.2.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 16/32/60 ms
A#ping 184.1.1.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 184.1.1.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 32/48/72 ms
A#
```

Figure 5-31 Routing table of router A is also valid

Similarly another core router, router B has also learned only correct information by blocking false information as shown in Figure 5-32 below.

```
B#sh ip
*Mar  1 00:29:10.707: %SYS-5-CONFIG_I: Configured from console by console bgp
BGP table version is 5, local router ID is 189.0.0.1
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     Weight     Path
*  172.0.0.0/6      99.99.99.44          0                     4 1 i
*>                  99.99.99.41          0          0          1 i
*  184.0.0.0/5      0.0.0.0                         32768        i
*>                  0.0.0.0              0          32768        i
s> 184.1.0.0        99.99.99.45          0          0          5 i

B#ping 184.1.1.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 184.1.1.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 16/32/48 ms
B#ping 174.1.2.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 174.1.2.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 24/49/124 ms
```

Figure 5-32 Successful Ping result of router B

Similarly other protection measures are in effect to protect against other well-known practical attacks that have happened in history to core routers of the Internet. For example another attack happened in recent years is related to AS_path length, which we have implemented protection for it but our topology contains all Cisco routers which do not allow to even originate this attack; therefore we cannot show its effectiveness here.

Chapter 6 Conclusion and Future Work

BGP routing protocol being used to convey routing information in Internet has many weaknesses that can be exploited to disrupt Internet connectivity. There are vulnerabilities in the routing protocol itself and the protocols used for its functioning. BGP is complex and has large domain, minor misconfigurations are possible and they have huge effect which is on the Internet scale. It is also relatively easy to find weak router or host in this large domain to exploit trust relationship. BGP has not built-in security because there is no peer authentication and no verification of update messages.

Internet services have been disrupted in the past due to these vulnerabilities in BGP routing protocol like 'AS7007 Internet blockage incident' and 'YouTube blockage by PTA'. BGP speaking routers use TCP as transport layer protocol to establish connections for information exchange and therefore vulnerabilities associated with TCP like SYN Flooding and RST can be used to temper the normal functioning of BGP. Furthermore, it is possible to inject false information intentionally or due to some mis-configuraiton. Harmful peer can send route advertisements with more specific routes (de-aggregated route) or route with modified path attributes, in order to influence path selection which may be resulting a blackhole, eavesdrop, congestion delay or loops.

Many security enhancements like TCP MD5 based peer authentication are added to get protection from these vulnerabilities but MD5 provides only peer authentication and not update message authentication. Many new secure forms of BGP are proposed like SBGP and SoBGP but they are not implemented due to cost associated with their heavy processing. Routers running BGP are already very heavily loaded and could not bear the load of these heavy protocols.

Therefore solution lies in utilizing the already known best practices security measures available to make BGP as secure as possible. Making the router secure and enabling only control access to it; is a first step towards BGP security. There are many other mechanisms like generalized TTL security mechanism, route dampening, maximum prefix limiting, limiting AS_Path length and prefix filtering.

In the first case study, we studied how path attributes can effect path selection decisions. We designed the scenario having alternate paths towards Internet and demonstrated how manipulation of BGP path attribute can result in the path selection. We have revealed how both links can be utilized so that no link bandwidth is wasted and load balancing is being done. Fault tolerance is also present such that if one path fails traffic will be routed through the other path. Similarly there are multiple incoming paths towards one organization, and ISPs will route traffic to these internal networks according to the organisation's policy and doing load balancing between two available paths. This is the main feature of BGP routing protocol; to support policy based routing.

In the second case study, we studied problems associated with iBGP routing that is used in transit ISP. There are certain iBGP rules that may result in black hole problems and routing loops. For instance next-hop is not changed when advertising to internal peers and it may become unreachable. Another issue is the rule of split horizon in iBGP routing, which prevent

router from forwarding advertisements recieved from one peer to another peer. This may result in missing information which may result in black holing or routing loop.

We have also shown how we can overcome these problems by using synchronization rule or by distributing complete IGP information within domain and by making full-mesh iBGP connection. But redistributing all information in IGP and creating full-mesh iBGP connectivity is difficult to maintain. Route reflectors can be used to make iBGP full-mesh solution scalable.

In the third case study we demonstrated how false BGP path advertisements due to some misconfiguration or from trusted peer can disrupt overall routing decisions. False information originated from one router is forwarded to others who will consider this information as correct due to lack of verification mechanism and propagate these advertisements further.

In the fourth and final case study we have implemented possible practical security measures to protect against BGP misconfiguration and attacks mentioned in case study 3 and we have checked their effectiveness. Cisco IOS configuration scripts for all routers in the topology of this case study are given in appendixes of this report. Route dampening is configured to avoid problems associated with flapping routes and MD5-based password protection between peers is configured for peer authentication. This topology also has TTL based DDoS attack prevention, proper prefix filtering, limiting AS_Path length and limiting number of prefixes received.

Major outages that happened in history were due to false route reachability advertisements either because of an attack or misconfiguration. This type of attacks can be prevented, with implementation of properly designed policy filters at every ISP router along with well-known best practices of protection measures proposed by industry and vendors. Moreover, we could identify that most problems happened in BGP routing are due to unintentional mistakes since BGP is difficult to configure and the CLI-based system is vulnerable to mistakes. Even a slight typographical mistake can launch an attack on the whole Internet [11].

Therefore, in order to prevent such problems in future, there should be some graphical based utility to configure BGP which facilitates the configuration of BGP and validates the input provided by user before applying these changes to the router's actual configuration, so that unintentional or intentional mistakes are blocked in the first place. Therefore in future work, research or master thesis can be done for designing this GUI based BGP checker and its detection mechanism to identify and rectify the problems in BGP scripts while achieving minimum false alarms.

References

[1] K. Hubbard, M. Kosters, Internet registry ip allocation guidelines, RFC 2050, November 1996

[2] V. Fuller, T. LI, Classless Inter-domain Routing (CIDR): The Internet Address Assignment and Aggregation, August 2006

[3] Y. Rekhter, T. Li, and S. Hares, A Border Gateway Protocol 4 (BGP-4), RFC 4271, Jan. 2006.

[4] Document ID: 13753, BGP Best Path Selection Algorithim. [online]. Available at http://www.cisco.com/en/US/tech/tk365/technologies_tech_note09186a0080094431.shtml

[5] Merike Kaeo, Routing Security , Double Shot Security,[online]. Available at http://doubleshotsecurity.com/pdf/ISSA-2004-RouterSec.pdf

[6] R. Barrett, S. Haar, and R. Whitestone, Routing causes Internet outage,[Interactive Week, Apr. 25, 1997.

[7] P. Boothe, J. Hiebert, and R. Bush, How prevalent is prefix hijacking on the Internet?'' in Proc. NANOG 36, Feb. 2006

[Online]. Available: http://www.nanog.org/mtg-0602/boothe.html

[8] S. Murphy, BGP Security Vulnerabilities Analysis, RFC 4272 January 2006

[9] Vincent J. Bono, 7007 Explanation and Apology [online]. Available at http://www.merit.edu/mail.archives/nanog/1997-04/msg00444.html

[10] Rensys Blog, Con-Ed Steals the 'Net.[Online]. Available: http://www.renesys.com/blog/2006/01/coned_steals_the_net.shtml

[11] Rensys Blog, Pakistan Hijacks YouTube. [Online]. Available: http://www.renesys.com/blog/2008/02/pakistan_hijacks_youtube_1.shtml%

[12] Rensys Blog, Longer is not better . [Online]. Available: http://www.renesys.com/blog/2009/02/longer-is-not-better.shtml#more

[13] W. Eddy, TCP SYN Flooding Attacks and Common Mitigations, RFC 4987, Aug. 2007.

[14] O. Nordstro¨m and C. Dovrolis, Beware of BGP attacks,[Comput. Communication Rev., vol. 34, no. 2, pp. 1–8, Apr. 2004

[15] R. Mahajan, D. Wetherall, and T. Anderson, Understanding BGP Miscon_guration," in Proceedings of ACM Sigcomm, Aug. 2002, pp. 3-16.

[16] R. Rivest, The MD5 Message-Digest Algorithm, RFC 1321, Apr. 1992.

[17] J. Touch, A. Mankin, and R. Bonica. The TCP Authentication Option, Internet Draft, Jul. 2009.

[18] S. Kent, C. Lynn, and K. Seo, Secure Border Gateway Protocol (S-BGP),[IEEE J. Sel. Areas Commun., vol. 18, no. 4, Apr. 2000.

[19] S. Kent, C. Lynn, J. Mikkelson, and K. Seo, Secure Border Gateway Protocol (S-BGP) real world performance and deployment issues,[in Proc. ISOC Symp. Network and Distributed System Security (NDSS), San Diego, CA, Feb. 2000.

[20] http://bgp.potaroo.net/index-as.html

[21] R. White, \Securing BGP: soBGP," ftp-eng.cisco.com/sobgp/index.html, Sept. 2003.

[22] V. Gill, J. Heasley, D. Meyer ,P. Savola, Ed.,C. Pignataro, The Generalized TTL Security Mechanism (GTSM), RFC 5082 October 2007 [online]. Available at http://tools.ietf.org/html/rfc5082

[23] C. Villamizar , R. Chandra,R. Govindan,BGP Route Flap Damping, RFC 2439 November 1998

[24] configuring maximum prefix limits [online]. Available at http://www.cisco.com/en/US/tech/tk365/technologies_configuration_example09186a008010a28a.shtml

[25] Protecting Border Gateway Protocol for the Enterprise [online].

Available at http://www.cisco.com/web/about/security/intelligence/protecting_bgp.html

[26] O. Bonaventure, Interdomain routing with BGP: Issues and challenges,[in Proceedings of the IEEE Symposium on Communications and Vehicular Technology (SCVT), Louvain-la-Neuve, Belgium, Oct. 2002.

[27] L. Gao and J. Rexford, BStable Internet routing without global coordination,[IEEE/ACM Trans. Networking, Dec. 2001.

[28] The Bogons Reference [online]. Available at http://www.team-cymru.org/Services/Bogons/

[29] Y. Rekhter, B. Moskowitz, Chrysler Corp, D. Karrenberg, E. Lear,

Address Allocation for Private Internets, RFC 1918 February 1996.

[30] Portecting your network edge with TTL security [online]. Available at

http://www.networkworld.com/community/node/18760

Appendix A

```
version 12.3
no service pad
service tcp-keepalives-in
service tcp-keepalives-out
service timestamps debug datetime msec localtime show-timezone
service timestamps log datetime msec localtime show-timezone
service password-encryption
service sequence-numbers
!
hostname A
!
boot-start-marker
boot-end-marker
!
security authentication failure rate 10 log
security passwords min-length 6
logging buffered 4096 debugging
logging console critical
enable secret 5 $1$Gujl$J67S8UDmTblS9rKYqH5/i0
enable password 7 051B0704285F5A0817
!
username user1 password 7 08314D5D1A0E0A05165A
aaa new-model
!
!
aaa authentication login local_auth local
aaa session-id common
ip subnet-zero
no ip source-route
no ip gratuitous-arps
!
!
ip cef
!
no ip bootp server
!
!
!
!
interface Loopback0
 ip address 99.99.99.41 255.255.255.255
!
interface FastEthernet0/0
 no ip address
 no ip redirects
 no ip unreachables
```

```
 no ip proxy-arp
 shutdown
 duplex auto
 speed auto
!
interface Serial0/0
 ip address 99.99.99.1 255.255.255.252
 no ip redirects
 no ip unreachables
 no ip proxy-arp
 clockrate 2000000
!
interface FastEthernet0/1
 no ip address
 no ip redirects
 no ip unreachables
 no ip proxy-arp
 shutdown
 duplex auto
 speed auto
!
interface Serial0/1
 ip address 99.99.99.5 255.255.255.252
 no ip redirects
 no ip unreachables
 no ip proxy-arp
 clockrate 2000000
!
interface Serial0/2
 ip address 99.99.99.9 255.255.255.252
 no ip redirects
 no ip unreachables
 no ip proxy-arp
 clockrate 2000000
!
interface Serial0/3
 no ip address
 no ip redirects
 no ip proxy-arp
 shutdown
 duplex auto
 speed auto
!
interface Serial0/1
 ip address 99.99.99.5 255.255.255.252
 no ip redirects
 no ip unreachables
 no ip proxy-arp
 clockrate 2000000
!
```

```
interface Serial0/2
 ip address 99.99.99.9 255.255.255.252
 no ip redirects
 no ip unreachables
 no ip proxy-arp
 clockrate 2000000
!
interface Serial0/3
 no ip address
 no ip redirects
 no ip unreachables
 no ip proxy-arp
 shutdown
 clockrate 2000000
!
router eigrp 1
 network 99.0.0.0
 no auto-summary
!
router bgp 1
 no synchronization
 bgp log-neighbor-changes
 bgp maxas-limit 20
 network 172.0.0.0 mask 252.0.0.0
 aggregate-address 172.0.0.0 252.0.0.0 summary-only
 neighbor 99.99.99.42 remote-as 2
 neighbor 99.99.99.42 ebgp-multihop 2
 neighbor 99.99.99.42 update-source Loopback0
 neighbor 99.99.99.42 prefix-list b2a in
 neighbor 99.99.99.42 prefix-list a2b out
 neighbor 99.99.99.42 password 7 070C285F4D06485744
 neighbor 99.99.99.42 maximum-prefix 100
 neighbor 99.99.99.43 remote-as 3
 neighbor 99.99.99.43 ebgp-multihop 2
 neighbor 99.99.99.43 update-source Loopback0
 neighbor 99.99.99.43 prefix-list c2a in
 neighbor 99.99.99.43 prefix-list a2c out
 neighbor 99.99.99.43 password 7 110A1016141D5A5E57
 neighbor 99.99.99.43 maximum-prefix 100
 neighbor 99.99.99.44 remote-as 4
 neighbor 99.99.99.44 ebgp-multihop 2
 neighbor 99.99.99.44 update-source Loopback0
 neighbor 99.99.99.44 prefix-list d2a in
 neighbor 99.99.99.44 prefix-list a2d out
 neighbor 99.99.99.44 password 7 045802150C2E1D1C5A
 neighbor 99.99.99.44 maximum-prefix 100
 no auto-summary
!
no ip http server
ip classless
```

```
ip route 172.0.0.0 252.0.0.0 Null0
!
!
!
ip prefix-list a2b seq 5 permit 172.0.0.0/6
ip prefix-list a2b seq 10 deny 5.0.0.0/8
ip prefix-list a2b seq 15 deny 10.0.0.0/8
ip prefix-list a2b seq 20 deny 23.0.0.0/8
ip prefix-list a2b seq 25 deny 37.0.0.0/8
ip prefix-list a2b seq 30 deny 39.0.0.0/8
ip prefix-list a2b seq 35 deny 100.0.0.0/8
ip prefix-list a2b seq 40 deny 102.0.0.0/8
ip prefix-list a2b seq 45 deny 105.0.0.0/8
ip prefix-list a2b seq 50 deny 107.0.0.0/8
ip prefix-list a2b seq 55 deny 127.0.0.0/8
ip prefix-list a2b seq 60 deny 169.254.0.0/16
ip prefix-list a2b seq 65 deny 172.16.0.0/12
ip prefix-list a2b seq 70 deny 179.0.0.0/8
ip prefix-list a2b seq 75 deny 192.0.0.0/24
ip prefix-list a2b seq 80 deny 192.0.2.0/24
ip prefix-list a2b seq 85 deny 198.18.0.0/15
ip prefix-list a2b seq 90 deny 192.168.0.0/16
ip prefix-list a2b seq 95 deny 203.0.113.0/24
ip prefix-list a2b seq 100 deny 198.51.100.0/24
ip prefix-list a2b seq 105 deny 0.0.0.0/0 le 32
!
ip prefix-list a2c seq 5 permit 172.0.0.0/6
ip prefix-list a2c seq 10 permit 184.0.0.0/5
ip prefix-list a2c seq 15 deny 5.0.0.0/8
ip prefix-list a2c seq 20 deny 10.0.0.0/8
ip prefix-list a2c seq 25 deny 23.0.0.0/8
ip prefix-list a2c seq 30 deny 37.0.0.0/8
ip prefix-list a2c seq 35 deny 39.0.0.0/8
ip prefix-list a2c seq 40 deny 100.0.0.0/8
ip prefix-list a2c seq 45 deny 102.0.0.0/8
ip prefix-list a2c seq 50 deny 105.0.0.0/8
ip prefix-list a2c seq 55 deny 107.0.0.0/8
ip prefix-list a2c seq 60 deny 127.0.0.0/8
ip prefix-list a2c seq 65 deny 169.254.0.0/16
ip prefix-list a2c seq 70 deny 172.16.0.0/12
ip prefix-list a2c seq 75 deny 179.0.0.0/8
ip prefix-list a2c seq 80 deny 192.0.0.0/24
ip prefix-list a2c seq 85 deny 192.0.2.0/24
ip prefix-list a2c seq 90 deny 198.18.0.0/15
ip prefix-list a2c seq 95 deny 192.168.0.0/16
ip prefix-list a2c seq 100 deny 203.0.113.0/24
ip prefix-list a2c seq 105 deny 198.51.100.0/24
ip prefix-list a2c seq 110 deny 0.0.0.0/0 le 32
!
ip prefix-list a2d seq 5 permit 172.0.0.0/6
```

```
ip prefix-list a2d seq 10 permit 184.0.0.0/5
ip prefix-list a2d seq 15 deny 5.0.0.0/8
ip prefix-list a2d seq 20 deny 10.0.0.0/8
ip prefix-list a2d seq 25 deny 23.0.0.0/8
ip prefix-list a2d seq 30 deny 37.0.0.0/8
ip prefix-list a2d seq 35 deny 39.0.0.0/8
ip prefix-list a2d seq 40 deny 100.0.0.0/8
ip prefix-list a2d seq 45 deny 102.0.0.0/8
ip prefix-list a2d seq 50 deny 105.0.0.0/8
ip prefix-list a2d seq 55 deny 107.0.0.0/8
ip prefix-list a2d seq 60 deny 127.0.0.0/8
ip prefix-list a2d seq 65 deny 169.254.0.0/16
ip prefix-list a2d seq 70 deny 172.16.0.0/12
ip prefix-list a2d seq 75 deny 179.0.0.0/8
ip prefix-list a2d seq 80 deny 192.0.0.0/24
ip prefix-list a2d seq 85 deny 192.0.2.0/24
ip prefix-list a2d seq 90 deny 198.18.0.0/15
ip prefix-list a2d seq 95 deny 192.168.0.0/16
ip prefix-list a2d seq 100 deny 203.0.113.0/24
ip prefix-list a2d seq 105 deny 198.51.100.0/24
ip prefix-list a2d seq 110 deny 0.0.0.0/0 le 32
!
ip prefix-list b2a seq 5 permit 184.0.0.0/5
ip prefix-list b2a seq 10 deny 5.0.0.0/8
ip prefix-list b2a seq 15 deny 10.0.0.0/8
ip prefix-list b2a seq 20 deny 23.0.0.0/8
ip prefix-list b2a seq 25 deny 37.0.0.0/8
ip prefix-list b2a seq 30 deny 39.0.0.0/8
ip prefix-list b2a seq 35 deny 100.0.0.0/8
ip prefix-list b2a seq 40 deny 102.0.0.0/8
ip prefix-list b2a seq 45 deny 105.0.0.0/8
ip prefix-list b2a seq 50 deny 107.0.0.0/8
ip prefix-list b2a seq 55 deny 127.0.0.0/8
ip prefix-list b2a seq 60 deny 169.254.0.0/16
ip prefix-list b2a seq 65 deny 172.16.0.0/12
ip prefix-list b2a seq 70 deny 179.0.0.0/8
ip prefix-list b2a seq 75 deny 192.0.0.0/24
ip prefix-list b2a seq 80 deny 192.0.2.0/24
ip prefix-list b2a seq 85 deny 198.18.0.0/15
ip prefix-list b2a seq 90 deny 192.168.0.0/16
ip prefix-list b2a seq 95 deny 203.0.113.0/24
ip prefix-list b2a seq 100 deny 198.51.100.0/24
ip prefix-list b2a seq 105 deny 0.0.0.0/0 le 32
!
ip prefix-list c2a seq 5 permit 174.1.0.0/16
ip prefix-list c2a seq 10 deny 5.0.0.0/8
ip prefix-list c2a seq 15 deny 10.0.0.0/8
ip prefix-list c2a seq 20 deny 23.0.0.0/8
ip prefix-list c2a seq 25 deny 37.0.0.0/8
ip prefix-list c2a seq 30 deny 39.0.0.0/8
```

```
ip prefix-list c2a seq 35 deny 100.0.0.0/8
ip prefix-list c2a seq 40 deny 102.0.0.0/8
ip prefix-list c2a seq 45 deny 105.0.0.0/8
ip prefix-list c2a seq 50 deny 107.0.0.0/8
ip prefix-list c2a seq 55 deny 127.0.0.0/8
ip prefix-list c2a seq 60 deny 169.254.0.0/16
ip prefix-list c2a seq 65 deny 172.16.0.0/12
ip prefix-list c2a seq 70 deny 179.0.0.0/8
ip prefix-list c2a seq 75 deny 192.0.0.0/24
ip prefix-list c2a seq 80 deny 192.0.2.0/24
ip prefix-list c2a seq 85 deny 198.18.0.0/15
ip prefix-list c2a seq 90 deny 192.168.0.0/16
ip prefix-list c2a seq 95 deny 203.0.113.0/24
ip prefix-list c2a seq 100 deny 198.51.100.0/24
ip prefix-list c2a seq 105 deny 0.0.0.0/0 le 32
!
ip prefix-list d2a seq 5 permit 175.1.0.0/16
ip prefix-list d2a seq 10 permit 184.0.0.0/5
ip prefix-list d2a seq 15 deny 5.0.0.0/8
ip prefix-list d2a seq 20 deny 10.0.0.0/8
ip prefix-list d2a seq 25 deny 23.0.0.0/8
ip prefix-list d2a seq 30 deny 37.0.0.0/8
ip prefix-list d2a seq 35 deny 39.0.0.0/8
ip prefix-list d2a seq 40 deny 100.0.0.0/8
ip prefix-list d2a seq 45 deny 102.0.0.0/8
ip prefix-list d2a seq 50 deny 105.0.0.0/8
ip prefix-list d2a seq 55 deny 107.0.0.0/8
ip prefix-list d2a seq 60 deny 127.0.0.0/8
ip prefix-list d2a seq 65 deny 169.254.0.0/16
ip prefix-list d2a seq 70 deny 172.16.0.0/12
ip prefix-list d2a seq 75 deny 179.0.0.0/8
ip prefix-list d2a seq 80 deny 192.0.0.0/24
ip prefix-list d2a seq 85 deny 192.0.2.0/24
ip prefix-list d2a seq 90 deny 198.18.0.0/15
ip prefix-list d2a seq 95 deny 192.168.0.0/16
ip prefix-list d2a seq 100 deny 203.0.113.0/24
ip prefix-list d2a seq 105 deny 198.51.100.0/24
ip prefix-list d2a seq 110 deny 0.0.0.0/0 le 32
logging trap debugging
logging facility local2
no cdp run
banner motd ^C unauthorized access to this router is not permitted, so ^C
!
line con 0
 exec-timeout 5 0
 login authentication local_auth
 transport preferred all
 transport output telnet
line aux 0
 login authentication local_auth
```

```
 transport preferred all
 transport output telnet
line vty 0 4
 login authentication local_auth
 transport preferred all
 transport input telnet
 transport output all
!
end

A#
```

Appendix B

```
version 12.3
no service pad
service tcp-keepalives-in
service tcp-keepalives-out
service timestamps debug datetime msec localtime show-timezone
service timestamps log datetime msec localtime show-timezone
service password-encryption
service sequence-numbers
!
hostname B
!
boot-start-marker
boot-end-marker
!
security authentication failure rate 10 log
security passwords min-length 6
logging buffered 4096 debugging
logging console critical
enable secret 5 $1$62hl$Vdh7NBQcmBiVuBXIR4h32.
enable password 7 15020A070D393F2526
!
username user1 password 7 00141215174C04140B70
aaa new-model
!
!
aaa authentication login local_auth local
aaa session-id common
ip subnet-zero
no ip source-route
no ip gratuitous-arps
!
!
ip cef
!
no ip bootp server
!
!
!
!
interface Loopback0
 ip address 99.99.99.42 255.255.255.255
!
interface Loopback1
 ip address 185.0.0.1 255.0.0.0
!
```

```
interface Loopback2
 ip address 186.0.0.1 255.0.0.0
!
interface Loopback3
 ip address 187.0.0.1 255.0.0.0
!
interface Loopback4
 ip address 188.0.0.1 255.0.0.0
!
interface Loopback5
 ip address 189.0.0.1 255.0.0.0
!
interface FastEthernet0/0
 no ip address

no ip redirects
 no ip unreachables
 no ip proxy-arp
 shutdown
 duplex auto
 speed auto
!
interface Serial0/1
 ip address 99.99.99.33 255.255.255.252
 no ip redirects
 no ip unreachables
 no ip proxy-arp
 clockrate 2000000
!
interface Serial0/2
 ip address 99.99.99.10 255.255.255.252
 no ip redirects
 no ip unreachables
 no ip proxy-arp
 clockrate 2000000
!
interface Serial0/3
 no ip address
 no ip redirects
 no ip unreachables
 no ip proxy-arp
 shutdown
 clockrate 2000000
!
router eigrp 1
 network 99.0.0.0
 no auto-summary
!
router bgp 2
 no synchronization
```

```
 bgp log-neighbor-changes
 bgp maxas-limit 20
 network 184.0.0.0 mask 248.0.0.0
 aggregate-address 184.0.0.0 248.0.0.0 summary-only
 neighbor 99.99.99.41 remote-as 1
 neighbor 99.99.99.41 ebgp-multihop 2
 neighbor 99.99.99.41 update-source Loopback0
 neighbor 99.99.99.41 prefix-list a2b in
 neighbor 99.99.99.41 prefix-list b2a out
 neighbor 99.99.99.41 password 7 045802150C2E1D1C5A
 neighbor 99.99.99.41 maximum-prefix 100
 neighbor 99.99.99.44 remote-as 4
 neighbor 99.99.99.44 ebgp-multihop 2
 neighbor 99.99.99.44 update-source Loopback0
 neighbor 99.99.99.44 prefix-list d2b in
 neighbor 99.99.99.44 prefix-list b2d out
 neighbor 99.99.99.44 password 7 0822455D0A16544541
 neighbor 99.99.99.44 maximum-prefix 100
 neighbor 99.99.99.45 remote-as 5
 neighbor 99.99.99.45 ebgp-multihop 2
 neighbor 99.99.99.45 update-source Loopback0
 neighbor 99.99.99.45 prefix-list e2b in
 neighbor 99.99.99.45 prefix-list b2e out
 neighbor 99.99.99.45 password 7 030752180500701E1D
 neighbor 99.99.99.45 maximum-prefix 100
 no auto-summary
!
no ip http server
ip classless
ip route 184.0.0.0 248.0.0.0 Null0
!
!
!
ip prefix-list a2b seq 5 permit 172.0.0.0/6
ip prefix-list a2b seq 10 deny 5.0.0.0/8
ip prefix-list a2b seq 15 deny 10.0.0.0/8
ip prefix-list a2b seq 20 deny 23.0.0.0/8
ip prefix-list a2b seq 25 deny 37.0.0.0/8
ip prefix-list a2b seq 30 deny 39.0.0.0/8
ip prefix-list a2b seq 35 deny 100.0.0.0/8
ip prefix-list a2b seq 40 deny 102.0.0.0/8
ip prefix-list a2b seq 45 deny 105.0.0.0/8
ip prefix-list a2b seq 50 deny 107.0.0.0/8
ip prefix-list a2b seq 55 deny 127.0.0.0/8
ip prefix-list a2b seq 60 deny 169.254.0.0/16
ip prefix-list a2b seq 65 deny 172.16.0.0/12
ip prefix-list a2b seq 70 deny 179.0.0.0/8
ip prefix-list a2b seq 75 deny 192.0.0.0/24
ip prefix-list a2b seq 80 deny 192.0.2.0/24
ip prefix-list a2b seq 85 deny 198.18.0.0/15
```

```
ip prefix-list a2b seq 90 deny 192.168.0.0/16
ip prefix-list a2b seq 95 deny 203.0.113.0/24
ip prefix-list a2b seq 100 deny 198.51.100.0/24
ip prefix-list a2b seq 105 deny 0.0.0.0/0 le 32
!
ip prefix-list b2a seq 5 permit 184.0.0.0/5
ip prefix-list b2a seq 10 deny 5.0.0.0/8
ip prefix-list b2a seq 15 deny 10.0.0.0/8
ip prefix-list b2a seq 20 deny 23.0.0.0/8
ip prefix-list b2a seq 25 deny 37.0.0.0/8
ip prefix-list b2a seq 30 deny 39.0.0.0/8
ip prefix-list b2a seq 35 deny 100.0.0.0/8
ip prefix-list b2a seq 40 deny 102.0.0.0/8
ip prefix-list b2a seq 45 deny 105.0.0.0/8
ip prefix-list b2a seq 50 deny 107.0.0.0/8
ip prefix-list b2a seq 55 deny 127.0.0.0/8
ip prefix-list b2a seq 60 deny 169.254.0.0/16
ip prefix-list b2a seq 65 deny 172.16.0.0/12
ip prefix-list b2a seq 70 deny 179.0.0.0/8
ip prefix-list b2a seq 75 deny 192.0.0.0/24
ip prefix-list b2a seq 80 deny 192.0.2.0/24
ip prefix-list b2a seq 85 deny 198.18.0.0/15
ip prefix-list b2a seq 90 deny 192.168.0.0/16
ip prefix-list b2a seq 95 deny 203.0.113.0/24
ip prefix-list b2a seq 100 deny 198.51.100.0/24
ip prefix-list b2a seq 105 deny 0.0.0.0/0 le 32
!
ip prefix-list b2d seq 5 permit 184.0.0.0/5
ip prefix-list b2d seq 10 permit 172.0.0.0/6
ip prefix-list b2d seq 15 deny 5.0.0.0/8
ip prefix-list b2d seq 20 deny 10.0.0.0/8
ip prefix-list b2d seq 25 deny 23.0.0.0/8
ip prefix-list b2d seq 30 deny 37.0.0.0/8
ip prefix-list b2d seq 35 deny 39.0.0.0/8
ip prefix-list b2d seq 40 deny 100.0.0.0/8
ip prefix-list b2d seq 45 deny 102.0.0.0/8
ip prefix-list b2d seq 50 deny 105.0.0.0/8
ip prefix-list b2d seq 55 deny 107.0.0.0/8
ip prefix-list b2d seq 60 deny 127.0.0.0/8
ip prefix-list b2d seq 65 deny 169.254.0.0/16
ip prefix-list b2d seq 70 deny 172.16.0.0/12
ip prefix-list b2d seq 75 deny 179.0.0.0/8
ip prefix-list b2d seq 80 deny 192.0.0.0/24
ip prefix-list b2d seq 85 deny 192.0.2.0/24
ip prefix-list b2d seq 90 deny 198.18.0.0/15
ip prefix-list b2d seq 95 deny 192.168.0.0/16
ip prefix-list b2d seq 100 deny 203.0.113.0/24
ip prefix-list b2d seq 105 deny 198.51.100.0/24
ip prefix-list b2d seq 110 deny 0.0.0.0/0 le 32
!
```

```
ip prefix-list b2e seq 5 permit 184.0.0.0/5
ip prefix-list b2e seq 10 permit 172.0.0.0/6
ip prefix-list b2e seq 15 deny 5.0.0.0/8
ip prefix-list b2e seq 20 deny 10.0.0.0/8
ip prefix-list b2e seq 25 deny 23.0.0.0/8
ip prefix-list b2e seq 30 deny 37.0.0.0/8
ip prefix-list b2e seq 35 deny 39.0.0.0/8
ip prefix-list b2e seq 40 deny 100.0.0.0/8
ip prefix-list b2e seq 45 deny 102.0.0.0/8
ip prefix-list b2e seq 50 deny 105.0.0.0/8
ip prefix-list b2e seq 55 deny 107.0.0.0/8
ip prefix-list b2e seq 60 deny 127.0.0.0/8
ip prefix-list b2e seq 65 deny 169.254.0.0/16
ip prefix-list b2e seq 70 deny 172.16.0.0/12
ip prefix-list b2e seq 75 deny 179.0.0.0/8
ip prefix-list b2e seq 80 deny 192.0.0.0/24
ip prefix-list b2e seq 85 deny 192.0.2.0/24
ip prefix-list b2e seq 90 deny 198.18.0.0/15
ip prefix-list b2e seq 95 deny 192.168.0.0/16
ip prefix-list b2e seq 100 deny 203.0.113.0/24
ip prefix-list b2e seq 105 deny 198.51.100.0/24
ip prefix-list b2e seq 110 deny 0.0.0.0/0 le 32
!
ip prefix-list d2b seq 5 permit 172.0.0.0/6
ip prefix-list d2b seq 10 deny 5.0.0.0/8
ip prefix-list d2b seq 15 deny 10.0.0.0/8
ip prefix-list d2b seq 20 deny 23.0.0.0/8
ip prefix-list d2b seq 25 deny 37.0.0.0/8
ip prefix-list d2b seq 30 deny 39.0.0.0/8
ip prefix-list d2b seq 35 deny 100.0.0.0/8
ip prefix-list d2b seq 40 deny 102.0.0.0/8
ip prefix-list d2b seq 45 deny 105.0.0.0/8
ip prefix-list d2b seq 50 deny 107.0.0.0/8
ip prefix-list d2b seq 55 deny 127.0.0.0/8
ip prefix-list d2b seq 60 deny 169.254.0.0/16
ip prefix-list d2b seq 65 deny 172.16.0.0/12
ip prefix-list d2b seq 70 deny 179.0.0.0/8
ip prefix-list d2b seq 75 deny 192.0.0.0/24
ip prefix-list d2b seq 80 deny 192.0.2.0/24
ip prefix-list d2b seq 85 deny 198.18.0.0/15
ip prefix-list d2b seq 90 deny 192.168.0.0/16
ip prefix-list d2b seq 95 deny 203.0.113.0/24
ip prefix-list d2b seq 100 deny 198.51.100.0/24
ip prefix-list d2b seq 105 deny 0.0.0.0/0 le 32
!
ip prefix-list e2b seq 5 permit 184.1.0.0/16
ip prefix-list e2b seq 10 deny 5.0.0.0/8
ip prefix-list e2b seq 15 deny 10.0.0.0/8
ip prefix-list e2b seq 20 deny 23.0.0.0/8
ip prefix-list e2b seq 25 deny 37.0.0.0/8
```

```
ip prefix-list e2b seq 30 deny 39.0.0.0/8
ip prefix-list e2b seq 35 deny 100.0.0.0/8
ip prefix-list e2b seq 40 deny 102.0.0.0/8
ip prefix-list e2b seq 45 deny 105.0.0.0/8
ip prefix-list e2b seq 50 deny 107.0.0.0/8
ip prefix-list e2b seq 55 deny 127.0.0.0/8
ip prefix-list e2b seq 60 deny 169.254.0.0/16
ip prefix-list e2b seq 65 deny 172.16.0.0/12
ip prefix-list e2b seq 70 deny 179.0.0.0/8
ip prefix-list e2b seq 75 deny 192.0.0.0/24
ip prefix-list e2b seq 80 deny 192.0.2.0/24
ip prefix-list e2b seq 85 deny 198.18.0.0/15
ip prefix-list e2b seq 90 deny 192.168.0.0/16
ip prefix-list e2b seq 95 deny 203.0.113.0/24
ip prefix-list e2b seq 100 deny 198.51.100.0/24
ip prefix-list e2b seq 105 deny 0.0.0.0/0 le 32
logging trap debugging
logging facility local2
no cdp run
banner motd ^C this router is protected, please stay away ^C
!
line con 0
 exec-timeout 5 0
 login authentication local_auth
 transport preferred all
 transport output telnet
line aux 0
 login authentication local_auth
 transport preferred all
 transport output telnet
line vty 0 4
 login authentication local_auth
 transport preferred all
 transport input telnet
 transport output all
!
end
```

Appendix C

En
Config t
Hostname C

neighbor 99.99.99.46 update-source Loopback0
neighbor 99.99.99.46 prefix-list f2c in
neighbor 99.99.99.46 prefix-list c2f out
neighbor 99.99.99.46 password 7 05080F1C22431F5B4A
neighbor 99.99.99.46 maximum-prefix 100
neighbor 99.99.99.47 remote-as 7
neighbor 99.99.99.47 ebgp-multihop 2
neighbor 99.99.99.47 update-source Loopback0
neighbor 99.99.99.47 prefix-list g2c in
neighbor 99.99.99.47 prefix-list c2g out
neighbor 99.99.99.47 password 7 060506324F41584B56
neighbor 99.99.99.47 maximum-prefix 100
no auto-summary
!
no ip http server
ip classless
ip route 174.1.0.0 255.255.0.0 Null0
!
!
!
ip prefix-list a2c seq 5 permit 172.0.0.0/6
ip prefix-list a2c seq 10 permit 184.0.0.0/5
ip prefix-list a2c seq 15 deny 5.0.0.0/8
ip prefix-list a2c seq 20 deny 10.0.0.0/8
ip prefix-list a2c seq 25 deny 23.0.0.0/8
ip prefix-list a2c seq 30 deny 37.0.0.0/8
ip prefix-list a2c seq 35 deny 39.0.0.0/8
ip prefix-list a2c seq 40 deny 100.0.0.0/8
ip prefix-list a2c seq 45 deny 102.0.0.0/8
ip prefix-list a2c seq 50 deny 105.0.0.0/8
ip prefix-list a2c seq 55 deny 107.0.0.0/8
ip prefix-list a2c seq 60 deny 127.0.0.0/8
ip prefix-list a2c seq 65 deny 169.254.0.0/16
ip prefix-list a2c seq 70 deny 172.16.0.0/12
ip prefix-list a2c seq 75 deny 179.0.0.0/8
ip prefix-list a2c seq 80 deny 192.0.0.0/24
ip prefix-list a2c seq 85 deny 192.0.2.0/24
ip prefix-list a2c seq 90 deny 198.18.0.0/15
ip prefix-list a2c seq 95 deny 192.168.0.0/16
ip prefix-list a2c seq 100 deny 203.0.113.0/24

```
ip prefix-list a2c seq 105 deny 198.51.100.0/24
ip prefix-list a2c seq 110 deny 0.0.0.0/0 le 32
!
ip prefix-list c2a seq 5 permit 174.1.0.0/16
ip prefix-list c2a seq 10 permit 175.1.0.0/16
ip prefix-list c2a seq 15 deny 5.0.0.0/8
ip prefix-list c2a seq 20 deny 10.0.0.0/8
ip prefix-list c2a seq 25 deny 23.0.0.0/8
ip prefix-list c2a seq 30 deny 37.0.0.0/8
ip prefix-list c2a seq 35 deny 39.0.0.0/8
ip prefix-list c2a seq 40 deny 100.0.0.0/8
ip prefix-list c2a seq 45 deny 102.0.0.0/8
ip prefix-list c2a seq 50 deny 105.0.0.0/8
ip prefix-list c2a seq 55 deny 107.0.0.0/8
ip prefix-list c2a seq 60 deny 127.0.0.0/8
ip prefix-list c2a seq 65 deny 169.254.0.0/16
ip prefix-list c2a seq 70 deny 172.16.0.0/12
ip prefix-list c2a seq 75 deny 179.0.0.0/8
ip prefix-list c2a seq 80 deny 192.0.0.0/24
ip prefix-list c2a seq 85 deny 192.0.2.0/24
ip prefix-list c2a seq 90 deny 198.18.0.0/15
ip prefix-list c2a seq 95 deny 192.168.0.0/16
ip prefix-list c2a seq 100 deny 203.0.113.0/24
ip prefix-list c2a seq 105 deny 198.51.100.0/24
ip prefix-list c2a seq 110 deny 0.0.0.0/0 le 32
!
ip prefix-list c2d seq 5 permit 174.1.0.0/16
ip prefix-list c2d seq 10 permit 172.0.0.0/6
ip prefix-list c2d seq 15 deny 5.0.0.0/8
ip prefix-list c2d seq 20 deny 10.0.0.0/8
ip prefix-list c2d seq 25 deny 23.0.0.0/8
ip prefix-list c2d seq 30 deny 37.0.0.0/8
ip prefix-list c2d seq 35 deny 39.0.0.0/8
ip prefix-list c2d seq 40 deny 100.0.0.0/8
ip prefix-list c2d seq 45 deny 102.0.0.0/8
ip prefix-list c2d seq 50 deny 105.0.0.0/8
ip prefix-list c2d seq 55 deny 107.0.0.0/8
ip prefix-list c2d seq 60 deny 127.0.0.0/8
ip prefix-list c2d seq 65 deny 169.254.0.0/16
ip prefix-list c2d seq 70 deny 172.16.0.0/12
ip prefix-list c2d seq 75 deny 179.0.0.0/8
ip prefix-list c2d seq 80 deny 192.0.0.0/24
ip prefix-list c2d seq 85 deny 192.0.2.0/24
ip prefix-list c2d seq 90 deny 198.18.0.0/15
ip prefix-list c2d seq 95 deny 192.168.0.0/16
ip prefix-list c2d seq 100 deny 203.0.113.0/24
ip prefix-list c2d seq 105 deny 198.51.100.0/24
ip prefix-list c2d seq 110 deny 0.0.0.0/0 le 32
!
ip prefix-list c2f seq 5 permit 172.0.0.0/6
```

```
ip prefix-list c2f seq 10 permit 184.0.0.0/5
ip prefix-list c2f seq 15 deny 5.0.0.0/8
ip prefix-list c2f seq 20 deny 10.0.0.0/8
ip prefix-list c2f seq 25 deny 23.0.0.0/8
ip prefix-list c2f seq 30 deny 37.0.0.0/8
ip prefix-list c2f seq 35 deny 39.0.0.0/8
ip prefix-list c2f seq 40 deny 100.0.0.0/8
ip prefix-list c2f seq 45 deny 102.0.0.0/8
ip prefix-list c2f seq 50 deny 105.0.0.0/8
ip prefix-list c2f seq 55 deny 107.0.0.0/8
ip prefix-list c2f seq 60 deny 127.0.0.0/8
ip prefix-list c2f seq 65 deny 169.254.0.0/16
ip prefix-list c2f seq 70 deny 172.16.0.0/12
ip prefix-list c2f seq 75 deny 179.0.0.0/8
ip prefix-list c2f seq 80 deny 192.0.0.0/24
ip prefix-list c2f seq 85 deny 192.0.2.0/24
ip prefix-list c2f seq 90 deny 198.18.0.0/15
ip prefix-list c2f seq 95 deny 192.168.0.0/16
ip prefix-list c2f seq 100 deny 203.0.113.0/24
ip prefix-list c2f seq 105 deny 198.51.100.0/24
ip prefix-list c2f seq 110 deny 0.0.0.0/0 le 32
!
ip prefix-list c2g seq 5 permit 172.0.0.0/6
ip prefix-list c2g seq 10 permit 184.0.0.0/5
ip prefix-list c2g seq 15 deny 5.0.0.0/8
ip prefix-list c2g seq 20 deny 10.0.0.0/8
ip prefix-list c2g seq 25 deny 23.0.0.0/8
ip prefix-list c2g seq 30 deny 37.0.0.0/8
ip prefix-list c2g seq 35 deny 39.0.0.0/8
ip prefix-list c2g seq 40 deny 100.0.0.0/8
ip prefix-list c2g seq 45 deny 102.0.0.0/8
ip prefix-list c2g seq 50 deny 105.0.0.0/8
ip prefix-list c2g seq 55 deny 107.0.0.0/8
ip prefix-list c2g seq 60 deny 127.0.0.0/8
ip prefix-list c2g seq 65 deny 169.254.0.0/16
ip prefix-list c2g seq 70 deny 172.16.0.0/12
ip prefix-list c2g seq 75 deny 179.0.0.0/8
ip prefix-list c2g seq 80 deny 192.0.0.0/24
ip prefix-list c2g seq 85 deny 192.0.2.0/24
ip prefix-list c2g seq 90 deny 198.18.0.0/15
ip prefix-list c2g seq 95 deny 192.168.0.0/16
ip prefix-list c2g seq 100 deny 203.0.113.0/24
ip prefix-list c2g seq 105 deny 198.51.100.0/24
ip prefix-list c2g seq 110 deny 0.0.0.0/0 le 32
!
ip prefix-list d2c seq 5 permit 175.1.0.0/16
ip prefix-list d2c seq 10 permit 184.0.0.0/5
ip prefix-list d2c seq 15 deny 5.0.0.0/8
ip prefix-list d2c seq 20 deny 10.0.0.0/8
ip prefix-list d2c seq 25 deny 23.0.0.0/8
```

```
ip prefix-list d2c seq 30 deny 37.0.0.0/8
ip prefix-list d2c seq 35 deny 39.0.0.0/8
ip prefix-list d2c seq 40 deny 100.0.0.0/8
ip prefix-list d2c seq 45 deny 102.0.0.0/8
ip prefix-list d2c seq 50 deny 105.0.0.0/8
ip prefix-list d2c seq 55 deny 107.0.0.0/8
ip prefix-list d2c seq 60 deny 127.0.0.0/8
ip prefix-list d2c seq 65 deny 169.254.0.0/16
ip prefix-list d2c seq 70 deny 172.16.0.0/12
ip prefix-list d2c seq 75 deny 179.0.0.0/8
ip prefix-list d2c seq 80 deny 192.0.0.0/24
ip prefix-list d2c seq 85 deny 192.0.2.0/24
ip prefix-list d2c seq 90 deny 198.18.0.0/15
ip prefix-list d2c seq 95 deny 192.168.0.0/16
ip prefix-list d2c seq 100 deny 203.0.113.0/24
ip prefix-list d2c seq 105 deny 198.51.100.0/24
ip prefix-list d2c seq 110 deny 0.0.0.0/0 le 32
!
ip prefix-list f2c seq 5 permit 174.1.1.0/24
ip prefix-list f2c seq 10 deny 5.0.0.0/8
ip prefix-list f2c seq 15 deny 10.0.0.0/8
ip prefix-list f2c seq 20 deny 23.0.0.0/8
ip prefix-list f2c seq 25 deny 37.0.0.0/8
ip prefix-list f2c seq 30 deny 39.0.0.0/8
ip prefix-list f2c seq 35 deny 100.0.0.0/8
ip prefix-list f2c seq 40 deny 102.0.0.0/8
ip prefix-list f2c seq 45 deny 105.0.0.0/8
ip prefix-list f2c seq 50 deny 107.0.0.0/8
ip prefix-list f2c seq 55 deny 127.0.0.0/8
ip prefix-list f2c seq 60 deny 169.254.0.0/16
ip prefix-list f2c seq 65 deny 172.16.0.0/12
ip prefix-list f2c seq 70 deny 179.0.0.0/8
ip prefix-list f2c seq 75 deny 192.0.0.0/24
ip prefix-list f2c seq 80 deny 192.0.2.0/24
ip prefix-list f2c seq 85 deny 198.18.0.0/15
ip prefix-list f2c seq 90 deny 192.168.0.0/16
ip prefix-list f2c seq 95 deny 203.0.113.0/24
ip prefix-list f2c seq 100 deny 198.51.100.0/24
ip prefix-list f2c seq 105 deny 0.0.0.0/0 le 32
!
ip prefix-list g2c seq 5 permit 174.1.2.0/24
ip prefix-list g2c seq 10 deny 5.0.0.0/8
ip prefix-list g2c seq 15 deny 10.0.0.0/8
ip prefix-list g2c seq 20 deny 23.0.0.0/8
ip prefix-list g2c seq 25 deny 37.0.0.0/8
ip prefix-list g2c seq 30 deny 39.0.0.0/8
ip prefix-list g2c seq 35 deny 100.0.0.0/8
ip prefix-list g2c seq 40 deny 102.0.0.0/8
ip prefix-list g2c seq 45 deny 105.0.0.0/8
ip prefix-list g2c seq 50 deny 107.0.0.0/8
```

```
ip prefix-list g2c seq 55 deny 127.0.0.0/8
ip prefix-list g2c seq 60 deny 169.254.0.0/16
ip prefix-list g2c seq 65 deny 172.16.0.0/12
ip prefix-list g2c seq 70 deny 179.0.0.0/8
ip prefix-list g2c seq 75 deny 192.0.0.0/24
ip prefix-list g2c seq 80 deny 192.0.2.0/24
ip prefix-list g2c seq 85 deny 198.18.0.0/15
ip prefix-list g2c seq 90 deny 192.168.0.0/16
ip prefix-list g2c seq 95 deny 203.0.113.0/24
ip prefix-list g2c seq 100 deny 198.51.100.0/24
ip prefix-list g2c seq 105 deny 0.0.0.0/0 le 32
logging trap debugging
logging facility local2
no cdp run
banner motd ^C this router is protectred, so please stay away ^C
!
line con 0
 exec-timeout 5 0
 login authentication local_auth
 transport preferred all
 transport output telnet
line aux 0
 login authentication local_auth
 transport preferred all
 transport output telnet
line vty 0 4
 login authentication local_auth
 transport preferred all
 transport input telnet
 transport output all
!
end
```

Appendix D

```
neighbor 99.99.99.43 update-source Loopback0
 neighbor 99.99.99.43 prefix-list c2d in
 neighbor 99.99.99.43 prefix-list d2c out
 neighbor 99.99.99.43 password 7 05080F1C22431F5B4A
 neighbor 99.99.99.43 maximum-prefix 100
 neighbor 99.99.99.48 remote-as 8
 neighbor 99.99.99.48 ebgp-multihop 2
 neighbor 99.99.99.48 update-source Loopback0
 neighbor 99.99.99.48 prefix-list h2d in
 neighbor 99.99.99.48 prefix-list d2h out
 neighbor 99.99.99.48 password 7 110A1016141D5A5E57
 neighbor 99.99.99.48 maximum-prefix 100
 no auto-summary
!
no ip http server
ip classless
ip route 175.1.0.0 255.255.0.0 Null0
!
!
!
ip prefix-list a2d seq 5 permit 172.0.0.0/6
ip prefix-list a2d seq 10 permit 184.0.0.0/5
ip prefix-list a2d seq 15 deny 5.0.0.0/8
ip prefix-list a2d seq 20 deny 10.0.0.0/8
ip prefix-list a2d seq 25 deny 23.0.0.0/8
ip prefix-list a2d seq 30 deny 37.0.0.0/8
ip prefix-list a2d seq 35 deny 39.0.0.0/8
ip prefix-list a2d seq 40 deny 100.0.0.0/8
ip prefix-list a2d seq 45 deny 102.0.0.0/8
ip prefix-list a2d seq 50 deny 105.0.0.0/8
ip prefix-list a2d seq 55 deny 107.0.0.0/8
ip prefix-list a2d seq 60 deny 127.0.0.0/8
ip prefix-list a2d seq 65 deny 169.254.0.0/16
ip prefix-list a2d seq 70 deny 172.16.0.0/12
ip prefix-list a2d seq 75 deny 179.0.0.0/8
ip prefix-list a2d seq 80 deny 192.0.0.0/24
ip prefix-list a2d seq 85 deny 192.0.2.0/24
ip prefix-list a2d seq 90 deny 198.18.0.0/15
ip prefix-list a2d seq 95 deny 192.168.0.0/16
ip prefix-list a2d seq 100 deny 203.0.113.0/24
ip prefix-list a2d seq 105 deny 198.51.100.0/24
ip prefix-list a2d seq 110 deny 0.0.0.0/0 le 32
!
ip prefix-list b2d seq 5 permit 184.0.0.0/5
ip prefix-list b2d seq 10 permit 172.0.0.0/6
ip prefix-list b2d seq 15 deny 5.0.0.0/8
```

```
ip prefix-list b2d seq 20 deny 10.0.0.0/8
ip prefix-list b2d seq 25 deny 23.0.0.0/8
ip prefix-list b2d seq 30 deny 37.0.0.0/8
ip prefix-list b2d seq 35 deny 39.0.0.0/8
ip prefix-list b2d seq 40 deny 100.0.0.0/8
ip prefix-list b2d seq 45 deny 102.0.0.0/8
ip prefix-list b2d seq 50 deny 105.0.0.0/8
ip prefix-list b2d seq 55 deny 107.0.0.0/8
ip prefix-list b2d seq 60 deny 127.0.0.0/8
ip prefix-list b2d seq 65 deny 169.254.0.0/16
ip prefix-list b2d seq 70 deny 172.16.0.0/12
ip prefix-list b2d seq 75 deny 179.0.0.0/8
ip prefix-list b2d seq 80 deny 192.0.0.0/24
ip prefix-list b2d seq 85 deny 192.0.2.0/24
ip prefix-list b2d seq 90 deny 198.18.0.0/15
ip prefix-list b2d seq 95 deny 192.168.0.0/16
ip prefix-list b2d seq 100 deny 203.0.113.0/24
ip prefix-list b2d seq 105 deny 198.51.100.0/24
ip prefix-list b2d seq 110 deny 0.0.0.0/0 le 32
!
ip prefix-list c2d seq 5 permit 174.1.0.0/16
ip prefix-list c2d seq 10 permit 172.0.0.0/6
ip prefix-list c2d seq 15 deny 5.0.0.0/8
ip prefix-list c2d seq 20 deny 10.0.0.0/8
ip prefix-list c2d seq 25 deny 23.0.0.0/8
ip prefix-list c2d seq 30 deny 37.0.0.0/8
ip prefix-list c2d seq 35 deny 39.0.0.0/8
ip prefix-list c2d seq 40 deny 100.0.0.0/8
ip prefix-list c2d seq 45 deny 102.0.0.0/8
ip prefix-list c2d seq 50 deny 105.0.0.0/8
ip prefix-list c2d seq 55 deny 107.0.0.0/8
ip prefix-list c2d seq 60 deny 127.0.0.0/8
ip prefix-list c2d seq 65 deny 169.254.0.0/16
ip prefix-list c2d seq 70 deny 172.16.0.0/12
ip prefix-list c2d seq 75 deny 179.0.0.0/8
ip prefix-list c2d seq 80 deny 192.0.0.0/24
ip prefix-list c2d seq 85 deny 192.0.2.0/24
ip prefix-list c2d seq 90 deny 198.18.0.0/15
ip prefix-list c2d seq 95 deny 192.168.0.0/16
ip prefix-list c2d seq 100 deny 203.0.113.0/24
ip prefix-list c2d seq 105 deny 198.51.100.0/24
ip prefix-list c2d seq 110 deny 0.0.0.0/0 le 32
!
ip prefix-list d2a seq 5 permit 175.1.0.0/16
ip prefix-list d2a seq 10 permit 184.0.0.0/5
ip prefix-list d2a seq 15 deny 5.0.0.0/8
ip prefix-list d2a seq 20 deny 10.0.0.0/8
ip prefix-list d2a seq 25 deny 23.0.0.0/8
ip prefix-list d2a seq 30 deny 37.0.0.0/8
ip prefix-list d2a seq 35 deny 39.0.0.0/8
```

```
ip prefix-list d2a seq 40 deny 100.0.0.0/8
ip prefix-list d2a seq 45 deny 102.0.0.0/8
ip prefix-list d2a seq 50 deny 105.0.0.0/8
ip prefix-list d2a seq 55 deny 107.0.0.0/8
ip prefix-list d2a seq 60 deny 127.0.0.0/8
ip prefix-list d2a seq 65 deny 169.254.0.0/16
ip prefix-list d2a seq 70 deny 172.16.0.0/12
ip prefix-list d2a seq 75 deny 179.0.0.0/8
ip prefix-list d2a seq 80 deny 192.0.0.0/24
ip prefix-list d2a seq 85 deny 192.0.2.0/24
ip prefix-list d2a seq 90 deny 198.18.0.0/15
ip prefix-list d2a seq 95 deny 192.168.0.0/16
ip prefix-list d2a seq 100 deny 203.0.113.0/24
ip prefix-list d2a seq 105 deny 198.51.100.0/24
ip prefix-list d2a seq 110 deny 0.0.0.0/0 le 32
!
ip prefix-list d2b seq 5 permit 172.0.0.0/6
ip prefix-list d2b seq 10 deny 5.0.0.0/8
ip prefix-list d2b seq 15 deny 10.0.0.0/8
ip prefix-list d2b seq 20 deny 23.0.0.0/8
ip prefix-list d2b seq 25 deny 37.0.0.0/8
ip prefix-list d2b seq 30 deny 39.0.0.0/8
ip prefix-list d2b seq 35 deny 100.0.0.0/8
ip prefix-list d2b seq 40 deny 102.0.0.0/8
ip prefix-list d2b seq 45 deny 105.0.0.0/8
ip prefix-list d2b seq 50 deny 107.0.0.0/8
ip prefix-list d2b seq 55 deny 127.0.0.0/8
ip prefix-list d2b seq 60 deny 169.254.0.0/16
ip prefix-list d2b seq 65 deny 172.16.0.0/12
ip prefix-list d2b seq 70 deny 179.0.0.0/8
ip prefix-list d2b seq 75 deny 192.0.0.0/24
ip prefix-list d2b seq 80 deny 192.0.2.0/24
ip prefix-list d2b seq 85 deny 198.18.0.0/15
ip prefix-list d2b seq 90 deny 192.168.0.0/16
ip prefix-list d2b seq 95 deny 203.0.113.0/24
ip prefix-list d2b seq 100 deny 198.51.100.0/24
ip prefix-list d2b seq 105 deny 0.0.0.0/0 le 32
!
ip prefix-list d2c seq 5 permit 175.1.0.0/16
ip prefix-list d2c seq 10 permit 184.0.0.0/5
ip prefix-list d2c seq 15 deny 5.0.0.0/8
ip prefix-list d2c seq 20 deny 10.0.0.0/8
ip prefix-list d2c seq 25 deny 23.0.0.0/8
ip prefix-list d2c seq 30 deny 37.0.0.0/8
ip prefix-list d2c seq 35 deny 39.0.0.0/8
ip prefix-list d2c seq 40 deny 100.0.0.0/8
ip prefix-list d2c seq 45 deny 102.0.0.0/8
ip prefix-list d2c seq 50 deny 105.0.0.0/8
ip prefix-list d2c seq 55 deny 107.0.0.0/8
ip prefix-list d2c seq 60 deny 127.0.0.0/8
```

```
ip prefix-list d2c seq 65 deny 169.254.0.0/16
ip prefix-list d2c seq 70 deny 172.16.0.0/12
ip prefix-list d2c seq 75 deny 179.0.0.0/8
ip prefix-list d2c seq 80 deny 192.0.0.0/24
ip prefix-list d2c seq 85 deny 192.0.2.0/24
ip prefix-list d2c seq 90 deny 198.18.0.0/15
ip prefix-list d2c seq 95 deny 192.168.0.0/16
ip prefix-list d2c seq 100 deny 203.0.113.0/24
ip prefix-list d2c seq 105 deny 198.51.100.0/24
ip prefix-list d2c seq 110 deny 0.0.0.0/0 le 32
!
ip prefix-list d2h seq 5 permit 172.0.0.0/6
ip prefix-list d2h seq 10 permit 184.0.0.0/5
ip prefix-list d2h seq 15 deny 5.0.0.0/8
ip prefix-list d2h seq 20 deny 10.0.0.0/8
ip prefix-list d2h seq 25 deny 23.0.0.0/8
ip prefix-list d2h seq 30 deny 37.0.0.0/8
ip prefix-list d2h seq 35 deny 39.0.0.0/8
ip prefix-list d2h seq 40 deny 100.0.0.0/8
ip prefix-list d2h seq 45 deny 102.0.0.0/8
ip prefix-list d2h seq 50 deny 105.0.0.0/8
ip prefix-list d2h seq 55 deny 107.0.0.0/8
ip prefix-list d2h seq 60 deny 127.0.0.0/8
ip prefix-list d2h seq 65 deny 169.254.0.0/16
ip prefix-list d2h seq 70 deny 172.16.0.0/12
ip prefix-list d2h seq 75 deny 179.0.0.0/8
ip prefix-list d2h seq 80 deny 192.0.0.0/24
ip prefix-list d2h seq 85 deny 192.0.2.0/24
ip prefix-list d2h seq 90 deny 198.18.0.0/15
ip prefix-list d2h seq 95 deny 192.168.0.0/16
ip prefix-list d2h seq 100 deny 203.0.113.0/24
ip prefix-list d2h seq 105 deny 198.51.100.0/24
ip prefix-list d2h seq 110 deny 0.0.0.0/0 le 32
!
ip prefix-list h2d seq 5 permit 175.1.1.0/24
ip prefix-list h2d seq 10 deny 5.0.0.0/8
ip prefix-list h2d seq 15 deny 10.0.0.0/8
ip prefix-list h2d seq 20 deny 23.0.0.0/8
ip prefix-list h2d seq 25 deny 37.0.0.0/8
ip prefix-list h2d seq 30 deny 39.0.0.0/8
ip prefix-list h2d seq 35 deny 100.0.0.0/8
ip prefix-list h2d seq 40 deny 102.0.0.0/8
ip prefix-list h2d seq 45 deny 105.0.0.0/8
ip prefix-list h2d seq 50 deny 107.0.0.0/8
ip prefix-list h2d seq 55 deny 127.0.0.0/8
ip prefix-list h2d seq 60 deny 169.254.0.0/16
ip prefix-list h2d seq 65 deny 172.16.0.0/12
ip prefix-list h2d seq 70 deny 179.0.0.0/8
ip prefix-list h2d seq 75 deny 192.0.0.0/24
ip prefix-list h2d seq 80 deny 192.0.2.0/24
```

```
ip prefix-list h2d seq 85 deny 198.18.0.0/15
ip prefix-list h2d seq 90 deny 192.168.0.0/16
ip prefix-list h2d seq 95 deny 203.0.113.0/24
ip prefix-list h2d seq 100 deny 198.51.100.0/24
ip prefix-list h2d seq 105 deny 0.0.0.0/0 le 32
logging trap debugging
logging facility local2
no cdp run
banner motd ^C this router is protected, so please stay away ^C
!
line con 0
 exec-timeout 5 0
 login authentication local_auth
 transport preferred all
 transport output telnet
line aux 0
 login authentication local_auth
 transport preferred all
 transport output telnet
line vty 0 4
 login authentication local_auth
 transport preferred all
 transport input telnet
 transport output all
!
end

D#
```

Appendix E

```
E#sh run
Building configuration...

Current configuration : 6529 bytes
!
version 12.3
no service pad
service tcp-keepalives-in
service tcp-keepalives-out
service timestamps debug datetime msec localtime show-timezone
service timestamps log datetime msec localtime show-timezone
service password-encryption
service sequence-numbers
!
hostname E
!
boot-start-marker
boot-end-marker
!
security authentication failure rate 10 log
security passwords min-length 6
logging buffered 4096 debugging
logging console critical
enable secret 5 $1$MD9n$BJudm1Uh3bfY5e4V7vE2k.
enable password 7 06160E2A455D1D180B
!
username user1 password 7 08314D5D1A0E0A05165A
aaa new-model
!
!
aaa authentication login local_auth local
aaa session-id common
ip subnet-zero
no ip source-route
no ip gratuitous-arps
!
!
ip cef
!
no ip bootp server
!
!
!
!
interface Loopback0
 ip address 99.99.99.45 255.255.255.255
```

```
!
interface FastEthernet0/0
 no ip address
 no ip redirects
 no ip unreachables
 no ip proxy-arp
 shutdown
 duplex auto
 speed auto
!
interface Serial0/0
 ip address 99.99.99.37 255.255.255.252
 no ip redirects
 no ip unreachables
 no ip proxy-arp
 clockrate 2000000
!
interface FastEthernet0/1
 no ip address
 no ip redirects
 no ip unreachables
 no ip proxy-arp
 shutdown
 duplex auto
 speed auto
!
interface Serial0/1
 ip address 99.99.99.34 255.255.255.252
 no ip redirects
 no ip unreachables
 no ip proxy-arp
 clockrate 2000000
!
router eigrp 1
 network 99.0.0.0
 no auto-summary
!
router bgp 5
 no synchronization
 bgp log-neighbor-changes
 bgp maxas-limit 20
 network 184.1.0.0
 aggregate-address 184.1.0.0 255.255.255.0 summary-only
 neighbor 99.99.99.42 remote-as 2
 neighbor 99.99.99.42 ebgp-multihop 2
 neighbor 99.99.99.42 update-source Loopback0
 neighbor 99.99.99.42 prefix-list b2e in
 neighbor 99.99.99.42 prefix-list e2b out
 neighbor 99.99.99.42 password 7 060506324F41584B56
 neighbor 99.99.99.42 maximum-prefix 100
```

```
 neighbor 99.99.99.49 remote-as 9
 neighbor 99.99.99.49 ebgp-multihop 2
 neighbor 99.99.99.49 update-source Loopback0
 neighbor 99.99.99.49 prefix-list i2e in
 neighbor 99.99.99.49 prefix-list e2i out
 neighbor 99.99.99.49 password 7 00071A1507545A545C
 neighbor 99.99.99.49 maximum-prefix 100
 no auto-summary
!
no ip http server
ip classless
ip route 184.1.0.0 255.255.0.0 Null0
!
!
!
ip prefix-list b2e seq 5 permit 184.0.0.0/5
ip prefix-list b2e seq 10 permit 172.0.0.0/6
ip prefix-list b2e seq 15 deny 5.0.0.0/8
ip prefix-list b2e seq 20 deny 10.0.0.0/8
ip prefix-list b2e seq 25 deny 23.0.0.0/8
ip prefix-list b2e seq 30 deny 37.0.0.0/8
ip prefix-list b2e seq 35 deny 39.0.0.0/8
ip prefix-list b2e seq 40 deny 100.0.0.0/8
ip prefix-list b2e seq 45 deny 102.0.0.0/8
ip prefix-list b2e seq 50 deny 105.0.0.0/8
ip prefix-list b2e seq 55 deny 107.0.0.0/8
ip prefix-list b2e seq 60 deny 127.0.0.0/8
ip prefix-list b2e seq 65 deny 169.254.0.0/16
ip prefix-list b2e seq 70 deny 172.16.0.0/12
ip prefix-list b2e seq 75 deny 179.0.0.0/8
ip prefix-list b2e seq 80 deny 192.0.0.0/24
ip prefix-list b2e seq 85 deny 192.0.2.0/24
ip prefix-list b2e seq 90 deny 198.18.0.0/15
ip prefix-list b2e seq 95 deny 192.168.0.0/16
ip prefix-list b2e seq 100 deny 203.0.113.0/24
ip prefix-list b2e seq 105 deny 198.51.100.0/24
ip prefix-list b2e seq 110 deny 0.0.0.0/0 le 32
!
ip prefix-list e2b seq 5 permit 184.1.0.0/16
ip prefix-list e2b seq 10 deny 5.0.0.0/8
ip prefix-list e2b seq 15 deny 10.0.0.0/8
ip prefix-list e2b seq 20 deny 23.0.0.0/8
ip prefix-list e2b seq 25 deny 37.0.0.0/8
ip prefix-list e2b seq 30 deny 39.0.0.0/8
ip prefix-list e2b seq 35 deny 100.0.0.0/8
ip prefix-list e2b seq 40 deny 102.0.0.0/8
ip prefix-list e2b seq 45 deny 105.0.0.0/8
ip prefix-list e2b seq 50 deny 107.0.0.0/8
ip prefix-list e2b seq 55 deny 127.0.0.0/8
ip prefix-list e2b seq 60 deny 169.254.0.0/16
```

```
ip prefix-list e2b seq 65 deny 172.16.0.0/12
ip prefix-list e2b seq 70 deny 179.0.0.0/8
ip prefix-list e2b seq 75 deny 192.0.0.0/24
ip prefix-list e2b seq 80 deny 192.0.2.0/24
ip prefix-list e2b seq 85 deny 198.18.0.0/15
ip prefix-list e2b seq 90 deny 192.168.0.0/16
ip prefix-list e2b seq 95 deny 203.0.113.0/24
ip prefix-list e2b seq 100 deny 198.51.100.0/24
ip prefix-list e2b seq 105 deny 0.0.0.0/0 le 32
!
ip prefix-list e2i seq 5 permit 184.0.0.0/5
ip prefix-list e2i seq 10 permit 172.0.0.0/6
ip prefix-list e2i seq 15 deny 5.0.0.0/8
ip prefix-list e2i seq 20 deny 10.0.0.0/8
ip prefix-list e2i seq 25 deny 23.0.0.0/8
ip prefix-list e2i seq 30 deny 37.0.0.0/8
ip prefix-list e2i seq 35 deny 39.0.0.0/8
ip prefix-list e2i seq 40 deny 100.0.0.0/8
ip prefix-list e2i seq 45 deny 102.0.0.0/8
ip prefix-list e2i seq 50 deny 105.0.0.0/8
ip prefix-list e2i seq 55 deny 107.0.0.0/8
ip prefix-list e2i seq 60 deny 127.0.0.0/8
ip prefix-list e2i seq 65 deny 169.254.0.0/16
ip prefix-list e2i seq 70 deny 172.16.0.0/12
ip prefix-list e2i seq 75 deny 179.0.0.0/8
ip prefix-list e2i seq 80 deny 192.0.0.0/24
ip prefix-list e2i seq 85 deny 192.0.2.0/24
ip prefix-list e2i seq 90 deny 198.18.0.0/15
ip prefix-list e2i seq 95 deny 192.168.0.0/16
ip prefix-list e2i seq 100 deny 203.0.113.0/24
ip prefix-list e2i seq 105 deny 198.51.100.0/24
ip prefix-list e2i seq 110 deny 0.0.0.0/0 le 32
!
ip prefix-list i2e seq 5 permit 184.1.1.0/24
ip prefix-list i2e seq 10 permit 194.1.1.0/24
ip prefix-list i2e seq 15 deny 5.0.0.0/8
ip prefix-list i2e seq 20 deny 10.0.0.0/8
ip prefix-list i2e seq 25 deny 23.0.0.0/8
ip prefix-list i2e seq 30 deny 37.0.0.0/8
ip prefix-list i2e seq 35 deny 39.0.0.0/8
ip prefix-list i2e seq 40 deny 100.0.0.0/8
ip prefix-list i2e seq 45 deny 102.0.0.0/8
ip prefix-list i2e seq 50 deny 105.0.0.0/8
ip prefix-list i2e seq 55 deny 107.0.0.0/8
ip prefix-list i2e seq 60 deny 127.0.0.0/8
ip prefix-list i2e seq 65 deny 169.254.0.0/16
ip prefix-list i2e seq 70 deny 172.16.0.0/12
ip prefix-list i2e seq 75 deny 179.0.0.0/8
ip prefix-list i2e seq 80 deny 192.0.0.0/24
ip prefix-list i2e seq 85 deny 192.0.2.0/24
```

```
ip prefix-list i2e seq 90 deny 198.18.0.0/15
ip prefix-list i2e seq 95 deny 192.168.0.0/16
ip prefix-list i2e seq 100 deny 203.0.113.0/24
ip prefix-list i2e seq 105 deny 198.51.100.0/24
ip prefix-list i2e seq 110 deny 0.0.0.0/0 le 32
logging trap debugging
logging facility local2
no cdp run
banner motd ^C this router is prtectred, so please stay away ^C
!
line con 0
 exec-timeout 5 0
 login authentication local_auth
 transport preferred all
 transport output telnet
line aux 0
 login authentication local_auth
 transport preferred all
 transport output telnet
line vty 0 4
 login authentication local_auth
 transport preferred all
 transport input telnet
 transport output all
!
end

E#
E#
```

Appendix F

```
F#sh run
Building configuration...

Current configuration : 4523 bytes
!
version 12.3
no service pad
service tcp-keepalives-in
service tcp-keepalives-out
service timestamps debug datetime msec localtime show-timezone
service timestamps log datetime msec localtime show-timezone
service password-encryption
service sequence-numbers
!
hostname F
!
boot-start-marker
boot-end-marker
!
security authentication failure rate 10 log
security passwords min-length 6
logging buffered 4096 debugging
logging console critical
enable secret 5 $1$Avur$XU6dnu5qUEPTTh9isn2V90
enable password 7 0014120D0D481F0701
!
username user1 password 7 12090404011C03162E7A
aaa new-model
!
!
aaa authentication login local_auth local
aaa session-id common
ip subnet-zero
no ip source-route
no ip gratuitous-arps
!
!
ip cef
!
no ip bootp server
!
!
!
!
interface Loopback0
 ip address 99.99.99.46 255.255.255.255
```

```
!
interface Loopback1
 ip address 174.1.1.1 255.255.255.128
!
interface FastEthernet0/0
 no ip address
 no ip redirects
 no ip unreachables
 no ip proxy-arp
 shutdown
 duplex auto
 speed auto
!
interface Serial0/0
 no ip address
 no ip redirects
 no ip unreachables
 no ip proxy-arp
 shutdown
 clockrate 2000000
!
interface FastEthernet0/1
 no ip address
 no ip redirects
 no ip unreachables
 no ip proxy-arp
 shutdown
 duplex auto
 speed auto
!
interface Serial0/1
 ip address 99.99.99.14 255.255.255.252
 no ip redirects
 no ip unreachables
 no ip proxy-arp
 clockrate 2000000
!
router eigrp 1
 network 99.0.0.0
 no auto-summary
!
router bgp 6
 no synchronization
 bgp log-neighbor-changes
 bgp maxas-limit 20
 network 174.1.1.0 mask 255.255.255.128
 network 174.1.1.128 mask 255.255.255.128
 network 184.1.1.0 mask 255.255.255.0
 aggregate-address 174.1.1.0 255.255.255.0 summary-only
 neighbor 99.99.99.43 remote-as 3
```

```
 neighbor 99.99.99.43 ebgp-multihop 2
 neighbor 99.99.99.43 update-source Loopback0
 neighbor 99.99.99.43 prefix-list c2f in
 neighbor 99.99.99.43 prefix-list f2c out
 neighbor 99.99.99.43 route-map asextend out
 neighbor 99.99.99.43 password 7 104D000A061843595F
 neighbor 99.99.99.43 maximum-prefix 100
 no auto-summary
!
no ip http server
ip classless
ip route 174.1.1.0 255.255.255.0 Null0
ip route 184.1.1.0 255.255.255.0 Null0
!
!
!
ip prefix-list c2f seq 5 permit 172.0.0.0/6
ip prefix-list c2f seq 10 permit 184.0.0.0/5
ip prefix-list c2f seq 15 deny 5.0.0.0/8
ip prefix-list c2f seq 20 deny 10.0.0.0/8
ip prefix-list c2f seq 25 deny 23.0.0.0/8
ip prefix-list c2f seq 30 deny 37.0.0.0/8
ip prefix-list c2f seq 35 deny 39.0.0.0/8
ip prefix-list c2f seq 40 deny 100.0.0.0/8
ip prefix-list c2f seq 45 deny 102.0.0.0/8
ip prefix-list c2f seq 50 deny 105.0.0.0/8
ip prefix-list c2f seq 55 deny 107.0.0.0/8
ip prefix-list c2f seq 60 deny 127.0.0.0/8
ip prefix-list c2f seq 65 deny 169.254.0.0/16
ip prefix-list c2f seq 70 deny 172.16.0.0/12
ip prefix-list c2f seq 75 deny 179.0.0.0/8
ip prefix-list c2f seq 80 deny 192.0.0.0/24
ip prefix-list c2f seq 85 deny 192.0.2.0/24
ip prefix-list c2f seq 90 deny 198.18.0.0/15
ip prefix-list c2f seq 95 deny 192.168.0.0/16
ip prefix-list c2f seq 100 deny 203.0.113.0/24
ip prefix-list c2f seq 105 deny 198.51.100.0/24
ip prefix-list c2f seq 110 deny 0.0.0.0/0 le 32
!
ip prefix-list f2c seq 5 permit 174.1.1.0/24
ip prefix-list f2c seq 10 deny 5.0.0.0/8
ip prefix-list f2c seq 15 deny 10.0.0.0/8
ip prefix-list f2c seq 20 deny 23.0.0.0/8
ip prefix-list f2c seq 25 deny 37.0.0.0/8
ip prefix-list f2c seq 30 deny 39.0.0.0/8
ip prefix-list f2c seq 35 deny 100.0.0.0/8
ip prefix-list f2c seq 40 deny 102.0.0.0/8
ip prefix-list f2c seq 45 deny 105.0.0.0/8
ip prefix-list f2c seq 50 deny 107.0.0.0/8
ip prefix-list f2c seq 55 deny 127.0.0.0/8
```

```
ip prefix-list f2c seq 60 deny 169.254.0.0/16
ip prefix-list f2c seq 65 deny 172.16.0.0/12
ip prefix-list f2c seq 70 deny 179.0.0.0/8
ip prefix-list f2c seq 75 deny 192.0.0.0/24
ip prefix-list f2c seq 80 deny 192.0.2.0/24
ip prefix-list f2c seq 85 deny 198.18.0.0/15
ip prefix-list f2c seq 90 deny 192.168.0.0/16
ip prefix-list f2c seq 95 deny 203.0.113.0/24
ip prefix-list f2c seq 100 deny 198.51.100.0/24
ip prefix-list f2c seq 105 deny 0.0.0.0/0 le 32
logging trap debugging
logging facility local2
no cdp run
banner motd ^C this router is protected, so please stay away ^C
!
line con 0
 exec-timeout 5 0
 login authentication local_auth
 transport preferred all
 transport output telnet
line aux 0
 login authentication local_auth
 transport preferred all
 transport output telnet
line vty 0 4
 login authentication local_auth
 transport preferred all
 transport input telnet
 transport output all
!
end
```

Appendix G

Building configuration...

Current configuration : 4478 bytes
!
version 12.3
no service pad
service tcp-keepalives-in
service tcp-keepalives-out
service timestamps debug datetime msec localtime show-timezone
service timestamps log datetime msec localtime show-timezone
service password-encryption
service sequence-numbers
!
hostname G
!
boot-start-marker
boot-end-marker
!
security authentication failure rate 10 log
security passwords min-length 6
logging buffered 4096 debugging
logging console critical
enable secret 5 1YITn$fVJMgf5xpMfLBkrdc4tpb/
enable password 7 0103070F5218120E2F
!
username user1 password 7 01030717481C091D251D
aaa new-model
!
!
aaa authentication login local_auth local
aaa session-id common
ip subnet-zero
no ip source-route
no ip gratuitous-arps
!
!
ip cef
!
no ip bootp server
!
!
!
!
interface Loopback0
ip address 99.99.99.47 255.255.255.255
!

```
interface Loopback1
 ip address 174.1.2.1 255.255.255.128
!
interface FastEthernet0/0
 no ip address
 no ip redirects
 no ip unreachables
 no ip proxy-arp
 shutdown
 duplex auto
 speed auto
!
interface Serial0/0
 no ip address
 no ip redirects
 no ip unreachables
 no ip proxy-arp
 shutdown
 clockrate 2000000
!
interface FastEthernet0/1
 no ip address
 no ip redirects
 no ip unreachables
 no ip proxy-arp
 shutdown
 duplex auto
 speed auto
!
interface Serial0/1
 ip address 99.99.99.18 255.255.255.252
 no ip redirects
 no ip unreachables
 no ip proxy-arp
 clockrate 2000000
!
router eigrp 1
 network 99.0.0.0
 no auto-summary
!
router bgp 7
 no synchronization
 bgp log-neighbor-changes
 bgp maxas-limit 20
 network 174.1.2.0 mask 255.255.255.128
 network 174.1.2.128 mask 255.255.255.128
 network 184.1.1.0 mask 255.255.255.0
 aggregate-address 174.1.2.0 255.255.255.0 summary-only
 neighbor 99.99.99.43 remote-as 3
 neighbor 99.99.99.43 ebgp-multihop 2
```

```
 neighbor 99.99.99.43 update-source Loopback0
 neighbor 99.99.99.43 prefix-list c2g in
 neighbor 99.99.99.43 prefix-list g2c out
 neighbor 99.99.99.43 password 7 1511021F07257A767B
 neighbor 99.99.99.43 maximum-prefix 100
 no auto-summary
!
no ip http server
ip classless
ip route 174.1.2.0 255.255.255.0 Null0
ip route 184.1.1.0 255.255.255.0 Null0
!
!
!
ip prefix-list c2g seq 5 permit 172.0.0.0/6
ip prefix-list c2g seq 10 permit 184.0.0.0/5
ip prefix-list c2g seq 15 deny 5.0.0.0/8
ip prefix-list c2g seq 20 deny 10.0.0.0/8
ip prefix-list c2g seq 25 deny 23.0.0.0/8
ip prefix-list c2g seq 30 deny 37.0.0.0/8
ip prefix-list c2g seq 35 deny 39.0.0.0/8
ip prefix-list c2g seq 40 deny 100.0.0.0/8
ip prefix-list c2g seq 45 deny 102.0.0.0/8
ip prefix-list c2g seq 50 deny 105.0.0.0/8
ip prefix-list c2g seq 55 deny 107.0.0.0/8
ip prefix-list c2g seq 60 deny 127.0.0.0/8
ip prefix-list c2g seq 65 deny 169.254.0.0/16
ip prefix-list c2g seq 70 deny 172.16.0.0/12
ip prefix-list c2g seq 75 deny 179.0.0.0/8
ip prefix-list c2g seq 80 deny 192.0.0.0/24
ip prefix-list c2g seq 85 deny 192.0.2.0/24
ip prefix-list c2g seq 90 deny 198.18.0.0/15
ip prefix-list c2g seq 95 deny 192.168.0.0/16
ip prefix-list c2g seq 100 deny 203.0.113.0/24
ip prefix-list c2g seq 105 deny 198.51.100.0/24
ip prefix-list c2g seq 110 deny 0.0.0.0/0 le 32
!
ip prefix-list g2c seq 5 permit 174.1.2.0/24
ip prefix-list g2c seq 10 deny 5.0.0.0/8
ip prefix-list g2c seq 15 deny 10.0.0.0/8
ip prefix-list g2c seq 20 deny 23.0.0.0/8
ip prefix-list g2c seq 25 deny 37.0.0.0/8
ip prefix-list g2c seq 30 deny 39.0.0.0/8
ip prefix-list g2c seq 35 deny 100.0.0.0/8
ip prefix-list g2c seq 40 deny 102.0.0.0/8
ip prefix-list g2c seq 45 deny 105.0.0.0/8
ip prefix-list g2c seq 50 deny 107.0.0.0/8
ip prefix-list g2c seq 55 deny 127.0.0.0/8
ip prefix-list g2c seq 60 deny 169.254.0.0/16
ip prefix-list g2c seq 65 deny 172.16.0.0/12
```

```
ip prefix-list g2c seq 70 deny 179.0.0.0/8
ip prefix-list g2c seq 75 deny 192.0.0.0/24
ip prefix-list g2c seq 80 deny 192.0.2.0/24
ip prefix-list g2c seq 85 deny 198.18.0.0/15
ip prefix-list g2c seq 90 deny 192.168.0.0/16
ip prefix-list g2c seq 95 deny 203.0.113.0/24
ip prefix-list g2c seq 100 deny 198.51.100.0/24
ip prefix-list g2c seq 105 deny 0.0.0.0/0 le 32
logging trap debugging
logging facility local2
no cdp run
banner motd ^C this router is protected, so please stay away ^C
!
line con 0
 exec-timeout 5 0
 login authentication local_auth
 transport preferred all
 transport output telnet
line aux 0
 login authentication local_auth
 transport preferred all
 transport output telnet
line vty 0 4
 login authentication local_auth
 transport preferred all
 transport input telnet
 transport output all
!
end

G#
```

Appendix H

```
H#sh run
Building configuration...

Current configuration : 4477 bytes
!
version 12.3
no service pad
service tcp-keepalives-in
service tcp-keepalives-out
service timestamps debug datetime msec localtime show-timezone
service timestamps log datetime msec localtime show-timezone
service password-encryption
service sequence-numbers
!
hostname H
!
boot-start-marker
boot-end-marker
!
security authentication failure rate 10 log
security passwords min-length 6
logging buffered 4096 debugging
logging console critical
enable secret 5 $1$FoBm$DWVd7.ckiSaIG4Urcab86.
enable password 7 15020A070D393F2526
!
username user1 password 7 13151601181B0B382F75
aaa new-model

aaa authentication login local_auth local
aaa session-id common
ip subnet-zero
no ip source-route
no ip gratuitous-arps
ip cef
no ip bootp server
interface Loopback0
 ip address 99.99.99.48 255.255.255.255
interface Loopback1
 ip address 175.1.1.1 255.255.255.128
interface FastEthernet0/0
 no ip address
 no ip redirects
 no ip unreachables
 no ip proxy-arp
 shutdown
```

```
 duplex auto
 speed auto
!
interface Serial0/0
 ip address 99.99.99.26 255.255.255.252
 no ip redirects
 no ip unreachables
 no ip proxy-arp
 clockrate 2000000
!
interface FastEthernet0/1
 no ip address
 no ip redirects
 no ip unreachables
 no ip proxy-arp
 shutdown
 duplex auto
 speed auto
interface Serial0/1
 no ip address
 no ip redirects
 no ip unreachables
 no ip proxy-arp
 shutdown
 clockrate 2000000
router eigrp 1
 network 99.0.0.0
 no auto-summary
router bgp 8
 no synchronization
 bgp log-neighbor-changes
 bgp maxas-limit 20
 network 175.1.1.0 mask 255.255.255.128
 network 175.1.1.128 mask 255.255.255.128
 network 184.1.1.0 mask 255.255.255.0
 aggregate-address 175.1.1.0 255.255.255.0 summary-only
 neighbor 99.99.99.44 remote-as 4
 neighbor 99.99.99.44 ebgp-multihop 2
 neighbor 99.99.99.44 update-source Loopback0
 neighbor 99.99.99.44 prefix-list d2h in
 neighbor 99.99.99.44 prefix-list h2d out
 neighbor 99.99.99.44 password 7 13061E010803557878
 neighbor 99.99.99.44 maximum-prefix 100
 no auto-summary
no ip http server
ip classless
ip route 175.1.0.0 255.255.255.0 Null0
ip route 184.1.1.0 255.255.255.0 Null0
ip prefix-list d2h seq 5 permit 172.0.0.0/6
ip prefix-list d2h seq 10 permit 184.0.0.0/5
```

```
ip prefix-list d2h seq 15 deny 5.0.0.0/8
ip prefix-list d2h seq 20 deny 10.0.0.0/8
ip prefix-list d2h seq 25 deny 23.0.0.0/8
ip prefix-list d2h seq 30 deny 37.0.0.0/8
ip prefix-list d2h seq 35 deny 39.0.0.0/8
ip prefix-list d2h seq 40 deny 100.0.0.0/8
ip prefix-list d2h seq 45 deny 102.0.0.0/8
ip prefix-list d2h seq 50 deny 105.0.0.0/8
ip prefix-list d2h seq 55 deny 107.0.0.0/8
ip prefix-list d2h seq 60 deny 127.0.0.0/8
ip prefix-list d2h seq 65 deny 169.254.0.0/16
ip prefix-list d2h seq 70 deny 172.16.0.0/12
ip prefix-list d2h seq 75 deny 179.0.0.0/8
ip prefix-list d2h seq 80 deny 192.0.0.0/24
ip prefix-list d2h seq 85 deny 192.0.2.0/24
ip prefix-list d2h seq 90 deny 198.18.0.0/15
ip prefix-list d2h seq 95 deny 192.168.0.0/16
ip prefix-list d2h seq 100 deny 203.0.113.0/24
ip prefix-list d2h seq 105 deny 198.51.100.0/24
ip prefix-list d2h seq 110 deny 0.0.0.0/0 le 32
!
ip prefix-list h2d seq 5 permit 175.1.1.0/24
ip prefix-list h2d seq 10 deny 5.0.0.0/8
ip prefix-list h2d seq 15 deny 10.0.0.0/8
ip prefix-list h2d seq 20 deny 23.0.0.0/8
ip prefix-list h2d seq 25 deny 37.0.0.0/8
ip prefix-list h2d seq 30 deny 39.0.0.0/8
ip prefix-list h2d seq 35 deny 100.0.0.0/8
ip prefix-list h2d seq 40 deny 102.0.0.0/8
ip prefix-list h2d seq 45 deny 105.0.0.0/8
ip prefix-list h2d seq 50 deny 107.0.0.0/8
ip prefix-list h2d seq 55 deny 127.0.0.0/8
ip prefix-list h2d seq 60 deny 169.254.0.0/16
ip prefix-list h2d seq 65 deny 172.16.0.0/12
ip prefix-list h2d seq 70 deny 179.0.0.0/8
ip prefix-list h2d seq 75 deny 192.0.0.0/24
ip prefix-list h2d seq 80 deny 192.0.2.0/24
ip prefix-list h2d seq 85 deny 198.18.0.0/15
ip prefix-list h2d seq 90 deny 192.168.0.0/16
ip prefix-list h2d seq 95 deny 203.0.113.0/24
ip prefix-list h2d seq 100 deny 198.51.100.0/24
ip prefix-list h2d seq 105 deny 0.0.0.0/0 le 32
logging trap debugging
logging facility local2
no cdp run
banner motd ^C this rouer is protected, so please stay away ^C
!
line con 0
 exec-timeout 5 0
 login authentication local_auth
```

```
 transport preferred all
 transport output telnet
line aux 0
 login authentication local_auth
 transport preferred all
 transport output telnet
line vty 0 4
 login authentication local_auth
 transport preferred all
 transport input telnet
 transport output all
!
end
```