Network Reachability-based IP Prefix Hijacking Detection

2012

정 천

박 사 학 위 논 문

Network Reachability-based IP Prefix Hijacking Detection

홍 성 철 (洪 性 哲)

전자컴퓨터공학부 (컴퓨터공학전공)

포항공과대학교 대학원

2012

네트워크 도달성에 기반한
IP Prefix 하이재킹 탐지

Network Reachability-based IP Prefix Hijacking Detection

Network Reachability-based IP Prefix Hijacking Detection

By

Seongcheol Hong
Division of Electrical and Computer Engineering
(Computer Science and Engineering)
Pohang University of Science and Technology

A thesis submitted to the faculty of Pohang University of Science and Technology in partial fulfillment of the requirements for the degree of Doctor of Philosophy in the Division of Electrical (Computer Engineering).

Pohang, Korea
December 16, 2011

Approved by

_____

Major Advisor: Prof. James Won-Ki Hong

네트워크 도달성에 기반한
IP Prefix 하이재킹 탐지

홍 성 철

위 논문은 포항공과대학교 대학원 박사 학위논문으로
학위논문 심사위원회를 통과하였음을 인정합니다.

2011년 12월 16일

학위논문심사위원회 위원장	홍 원 기	(인)
위원	김 종	(인)
위원	서 영 주	(인)
위원	Raouf Boutaba	(인)
위원	주 홍 택	(인)

DECE 20032421 Seongcheol Hong, “Network Reachability-based IP Prefix Hijacking Detection”, Division of Electrical and Computer Engineering (Computer Science and Engineering), 2012, 102P, Advisor: James W. Hong, Text in English.

ABSTRACT

The Internet is a decentralized network comprised of many interconnected networks and designed to provide communication on the basis of trust between networks. Each network communicates reachability information using Border Gateway Protocol (BGP). The Internet was designed to provide communication on the basis of trust between networks, but has proved to be a misguided assumption, due to the various types of attacks that have taken advantage of this trust. Autonomous Systems (ASes) that exchange BGP information directly with each other are assumed to be trusted, so BGP does not implement security checks to protect against receiving bad or invalid routing information from other routers, such as checking the authenticity of origin information and path attributes. As such, the Internet routing infrastructure is vulnerable to attack.

IP prefix hijacking is the major threat to the security of the Internet routing system due to the lack of authoritative prefix ownership information. It is a BGP security attack, in which a BGP router, either with malicious purposes or simple due to misconfiguration, announces an IP prefix that the router does not own. This false announcement creates reachability problems and communication failures throughout the Internet. This problem has some common characteristics such as MOAS conflicts and invalid routes in BGP messages. Despite many efforts in designing IP prefix hijack detection schemes no existing design satisfies all the critical requirements of a truly effective system, that is, it must be real-time, deployable, as well as robust.

In this thesis, we present a novel approach that detects IP prefix hijacking in the current Internet environment. The focus of this work is keeping the BGP

routing infrastructure and not relying on mutual cooperation, to ensure ease of deployment. Also we look at fingerprinting two ASes that have the same IP prefix to distinguish IP prefix hijacking events from legitimate routing updates. This paper proposes a practical and deployable IP prefix hijacking detection algorithm with live hosts on the Internet.

Contents

1 Introduction 1

1.1 Background 1
1.2 Motivation and Problem Statements 2
1.3 Research Approach 4
1.4 Outline of this Dissertation 7

2 Related Work 9

2.1 BGP Overview 9
2.1.1 BGP Update Message 11
2.1.2 BGP Attributes 12
2.1.3 BGP Path Selection 14
2.2 BGP Attacks 16
2.3 BGP Security 17

3 IP Prefix Hijacking 18

3.1 Threats of IP Prefix Hijacking 18
3.2 IP Prefix Hijacking Incidents 21
3.3 IP Prefix Hijacking Solutions 24

4 Network Reachability Monitoring 30

4.1 Reachability Monitoring 31
4.2 Host Fingerprinting 32
4.2.1 DNS Server as Host Fingerprinting Target 33
4.2.2 DNS Server Collection Method 33
4.2.3 DNS Server Fingerprinting Method 39
4.3 Network Fingerprinting 41
4.3.1 Theoretical Background 43

4.3.2 Inferring Method 45

5 IP Prefix Hijacking Detection 49

5.1 A Probing Method using Idle Scan 49

5.1.1 IP Prefix Hijacking Detection Method using Idle Scan 50

5.1.2 Classification of Polluted and Unpolluted ASes 53

5.2 Network Reachability based IP Prefix Hijacking Detection (RBHD) 57

6 Evaluation 61

6.1 Validation of Host Fingerprinting 61

6.2 Validation of Network Fingerprinting 64

6.3 Experimental Setup and Results 68

7 Conclusion 71

7.1 Summary 71

7.2 Contributions 72

7.3 Future Work 73

References 75

List of Figures

Figure 1. BGP network example 10
Figure 2. BGP decision process 16
Figure 3. IP prefix hijacking: polluted and unpolluted ASes 20
Figure 4. DNS conversion process 34
Figure 5. The structure of zone files in DNS servers 36
Figure 6. Flowchart of DNS server collection process 37
Figure 7. The number of IP prefixes owned by each AS 39
Figure 8. Structure of DNS server fingerprinting 40
Figure 9. Example of the existing firewall policy analysis 45
Figure 10. A simple example of sweep line algorithm 47
Figure 11. Idle scan for IP prefix hijacking detection 51
Figure 12. Classification of polluted and unpolluted ASes 55
Figure 13. Overall detection algorithm 58
Figure 14. IP prefix hijacking detection system architecture 59
Figure 15. The examples of collected host fingerprints 62
Figure 16. The number of distinguishable groups in the DNS server fingerprints 63
Figure 17. Configuration of network for inferring a firewall policy 64
Figure 18. Distribution of response packet for inferring a firewall policy 65
Figure 19. Setup of hijacking testbed 69
Figure 20. Preliminarily collected and current fingerprints about the target network 70

List of Tables

Table 1. Comparison among anomaly detection system.....29
Table 2. The current state of DNS server operation in the Internet 38
Table 3. Example of firewall policy.....44
Table 4. The distribution of the type and version of DNS servers in the Internet .61
Table 5. The Internet firewall policy of network A 66
Table 6. The Internet firewall policy of network B 67

1 Introduction

This chapter provides a brief introduction to current Internet environment and IP routing. The problems in current BGP security and IP prefix hijacking are listed and the approaches to solve these problems are mentioned in this thesis.

1.1 Background

The Internet is a decentralized network comprised of many interconnected networks. These networks are composed of end hosts (who originate and/or receive IP packets, and are identified by IP addresses) and active forwarding elements (routers) whose role is to pass IP packets through the network. The routing system is responsible for propagation the relative location of addresses to each routing element, so that routers can make consistent and optimal routing decisions in order to pass a packet from its source to its destination. Routing protocols are used to perform this reachability information propagation.

Each network communicates reachability information using Border Gateway Protocol (BGP) [1]. BGP is the de-facto inter-domain routing protocol that maintains a table of IP networks or prefixes, and designates network reachability among the various Autonomous Systems (ASes) that make up the Internet. BGP was created to replace the Exterior Gateway Protocol (EGP) to allow fully decentralized routing. This allowed the Internet to become a truly decentralized system.

An AS is a collection of connected IP routing prefixes under the control of one or more network operators that presents a common, clearly defined routing policy to the Internet [2]. When exchanging exterior routing information, each AS is identified by a unique number. An AS normally uses some interior gateway protocol to exchange routing information on its internal networks. There are more

than 35,000 such systems contained within the Internet routing system [3].

Originally, IP included a concept of classes of networks. Because the Internet is a collection of networks, IP addresses can be interpreted as having two parts: a part that identifies a network on the Internet and a part that identifies a host on that network. This division is used as the basis of traffic routing between IP networks and for address allocation policies. Classful network design for IPv4 sized the network address as one or more 8-bit groups, resulting in the blocks of Class A, B, or C addresses. Classless Inter-Domain Routing (CIDR) allocates address space to Internet service providers and end users on any address bit boundary, instead of on 8-bit segments. In IPv6, however, the interface identifier has a fixed size of 64 bits by convention, and smaller subnets are never allocated to end users.

Internet Service Providers (ISPs) connect their networks to each other in order to exchange traffic between their customers and the customers of other ISPs [4]. ISP Interconnection allows traffic originating at a source connected to one ISP's network to reach a destination connected to another ISP's network, around the block or around the world. End users see the seamless, global, ubiquitous communication medium known as the Internet; behind the scenes lie many individual networks, owned and operated by many different corporate, institutional, and governmental entities, joined to each other by interconnection arrangements. Interconnection is the glue that holds the Internet together.

1.2 Motivation and Problem Statements

The Internet was designed to provide communication on the basis of trust between networks, but has proved to be a misguided assumption, due to the various types of attacks that have taken advantage of this trust. ASes that exchange BGP information directly with each other are assumed to be trusted, so BGP does not implement any security checks, such as checking the authenticity of

origin information and path attributes, to protect against receiving invalid routing information from other routers. As such, the Internet routing infrastructure is vulnerable to attack. IP prefix hijacking is a BGP security attack, in which a BGP router, either with malicious purposes or simple due to misconfiguration, announces an IP prefix that the router does not own. This false announcement creates reachability problems and communication failures throughout the Internet.

To mitigate the impact of incorrect routing information, some BGP extensions have been proposed, such as Secure BGP (S-BGP) [45] and Secure Origin BGP (soBGP) [46]. These solutions could possibly solve some of the more well-known BGP security problems, but they are difficult to deploy in practical networks because they employ the digital signature techniques that cause high processing overheads on the router. In addition, they are still vulnerable to Denial of Service (DoS) attacks. These improved protocols also require changes to the existing BGP protocol and infrastructure. Due to these reasons the above solutions are not currently deployed, thus other approaches that can detect and respond to IP prefix hijacking are still being searched for.

Many previous studies have proposed methods to detect IP prefix hijacking [49][50][51][52][53][54][55][56][57][58]. These methods are either based on passive monitoring or active probing. Some of these approaches also use routing registry information, such as IRR (Internet Routing Registry) databases, which can become outdated. The IP prefix hijacking must be distinguished from legitimate routing updates, because both cause Multiple Origin AS (MOAS) changes (i.e. a conflict caused by a particular prefix that appears to originate from different ASes). These approaches have been shown to have limitations in deployment and accuracy. Limitations in deployment come from the requirement of closer cooperation among ASes to gather necessary information and for the implementation of additional functions in routing infrastructure [50][53]. Judgment on whether IP hijacking is occurring or not must be definitive, however statistical data measured from the Internet forms the basis of judgment in previous

approaches [51][54]. Given the continuously changing nature of the Internet, the detection accuracy of previous approaches cannot be guaranteed and false positives or negatives can easily occur.

A practically deployable, simple and accurate IP prefix hijacking detection method is needed in order to respond to the increasingly serious IP prefix hijacking incidents. It requires that IP prefix hijacking can be immediately and accurately detected as soon as the hijacking occurs. Also, it should be able to detect the hijacking without constructing an infrastructure that needs mutual cooperation between ASes. An ideal IP prefix hijacking detection method should be easily adoptable and be able to detect known hijacking attacks regardless of attack type.

In this section, the research problems and goals are briefly listed. This thesis tries to answer the following key questions.

- What are the limitations of existing BGP security techniques?
- What features should be investigated in order to detect IP prefix hijacking?
- How can we efficiently collect AS-level host and network fingerprints for reachability testing?
- What methods should be developed for an accurate, deployable, and adaptive IP prefix hijacking detection system?
- What is the next research step towards more advanced anomalous update detection?

1.3 Research Approach

In this section, we explain the solutions to the problems mentioned in section 1.2 and the research methodologies. We survey and analyze IP prefix hijacking incidents that have occurred in the Internet in the past. We propose monitoring and

detection techniques for detecting malicious BGP update messages to hijack IP prefixes. We also provide experimental results that justify the design of our IP prefix hijacking detection method and evaluate our proposed scheme using analysis based on large scale Internet measurements.

Pertaining to these questions, the following items are the goals of this thesis.

- This thesis will state Routing Information Base (RIB) statistics and suspicious features.
- This thesis will propose AS-level host and network fingerprints collection methods using reachability monitoring.
- This thesis will propose an IP prefix hijacking detection method.
- This thesis will perform deployment experiences for validation of the proposed methods.
- This thesis will provide a guideline for handling update messages for routing stability.

We propose a new approach which practically and effectively detects IP prefix hijacking based on network reachability – we call this method **RBHD** (Reachability Based Hijacking Detection). Network reachability is a property that dictates whether traffic sent by someone can reach its intended network. In other words, it means that sent packets reach the intended original network that owns a specific IP prefix, not a destination with the same, hijacked prefix. The network reachability defined above should be maintained even if the network path is changed due to routing instability. Therefore, the proposed method identifies whether we can reach the network through the changed path when a routing update message arrives and the routing path is changed. If we can reach the identical network, the update is regarded as a normal routing event. If not, we decide that it is a case of IP prefix hijacking. It is significant to note that network reachability differs from IP reachability in that network reachability has a strict

requirement of reaching the target. IP reachability only measures whether a packet can reach the intended IP host. When IP hijacking occurred, IP reachability is maintained even though the target host was changed. However network reachability is corrupted due to the change of IP path to the target network.

We use a fingerprinting scheme in order to determine the network reachability of a specific network. Like a human fingerprint, this fingerprint captured from a network can be used for identification in the network reachability test. We classify the types of fingerprints for the network reachability test into host fingerprints and network fingerprints. Host fingerprints are the implementation specifications of the operating system and protocol stacks, or the current configurations of a running application in the host. The host fingerprint identifies a network. The host fingerprint is now always available from all ASes on the Internet, so it is easier to capture and use than a network fingerprint. On the other hand, Network fingerprints characterize a specific network, for example, a value representing host availability in that network. We collect the host and network fingerprints of an AS owing a specific IP prefix before any IP hijacking event, then we compare those with newly captured fingerprints of the AS whenever a suspicious routing update is received. In this thesis, we propose host fingerprinting and network fingerprinting methods for IP prefix hijacking detection.

A technical obstacle for collecting network fingerprints is that it is difficult to collect network information due to firewalls or IDS devices. For performing network fingerprinting, we propose a method that collects firewall or IDS policies of the target network in advance, and then applies those policies to the network fingerprint collection. On the other hand, in host fingerprinting, we are more concerned about how we characterize the hosts of the IP prefix. We use the Domain Name System (DNS) as the target for host fingerprinting. DNS servers are an appropriate target to collect host fingerprints from, and as such, we propose an effective host fingerprinting method using DNS servers for IP prefix hijacking

detection.

Although our approach depends on a probing technique for network reachability monitoring, it can conclusively detect an IP prefix hijacking occurrence by fingerprint comparison. Also, the proposed method does not depend on monitoring infrastructure such as distributed vantage points in the Internet because an AS receiving routing updates detects IP prefix hijacking for itself. The work to collect fingerprints is simple, effective and used popular tools such as *ping* or *traceroute*. We validated the effectiveness of the proposed method and are ready to apply it to the real network.

To detect an IP prefix hijacking event, we monitor the routing update messages that show an incorrect announcement of their IP prefix origin. When IP prefix hijacking occurs, there must be two networks with the same IP address space on the Internet. As the basic route selection process is to select routes with the shortest path, only the ASes close to the attacker's AS are likely to be polluted. We focus on fingerprinting two ASes with the same IP prefix in order to distinguish IP prefix hijacking events from legitimate routing updates. Our experience of developing the IP prefix hijacking detection system provides firm guidelines for understanding IP prefix hijacking countermeasure mechanisms. Overall, we propose an easily deployable method that detects IP prefix hijacking without the need for mutual cooperation of AS or modification to existing routing protocols and routers.

1.4 Outline of this Dissertation

The organization of this thesis is as follows. Chapter 2 describes the overviews of BGP and the attacks and security of BGP. In Chapter 3, we introduce the IP prefix hijacking problem and existing solutions for IP prefix hijacking detection. Chapter 4 investigates the network reachability monitoring and introduces the proposed host and network fingerprinting techniques. We describe

our proposed IP prefix hijacking detection methods in Chapter 5. We describe the validation of the proposed method in Section 6. It also provides the evaluation details of the proposed diagnosis techniques. Finally, Chapter 7 concludes the thesis with summary, contributions, and possible future work.

2 Related Work

In this chapter, we introduce BGP overview and the current issues of BGP attacks. The related research on BGP security is also given. By investigating previous work, we can determine the BGP security requirements for IP prefix hijacking detection.

2.1 BGP Overview

The Internet is composed of tens of thousands of ASes under separate administrative domains. BGP is the de-facto inter-domain routing protocol and a path vector protocol where the BGP update includes a list of ASes (called *AS path*) which describes the path to destination addresses (called *IP prefix*). A destination prefix is usually announced either by the prefix owner itself, if it runs BGP and has an AS number, or by its upstream provider AS(es).

BGP is an incremental protocol – a BGP-speaking router sends an announcement message when a new route is available and a withdrawal message when a route no longer exists. BGP is also a path-vector protocol, where each AS adds it's AS number to the beginning of the AS path before advertising the route to the next AS. Each router selects a single preferred BGP route for each destination prefix and may apply complex policies for selecting a route and deciding whether to advertise the route to a neighboring router in another AS.

Most Internet Service Providers (**ISP**s) must use BGP to establish routing between one another (especially if they are multi-homed). Therefore, even though most Internet users do not use it directly, BGP is one of the most important protocols of the Internet. When BGP runs between two peers in the same AS, it is referred to as Internal BGP (**IBGP** or Interior Border Gateway Protocol). When it

runs between ASes, it is called External BGP (**EBGP** or Exterior Border Gateway Protocol). Routers on the boundary of one AS exchanging information with another AS are called border or edge routers.

As shown in Figure 1, a BGP route announcement consists of a network prefix and a list of ASes: (P, [AS_k AS_{k-1} ... AS_0]). P is the block of IP addresses being announced; the list of ASes is the ordered list of ASes traffic to P would traverse. The last AS in the list, AS_0 is the origin AS, or simply the origin, of the announcement. Each AS exports the routes to its neighbors after adding itself in the front of the received AS path.

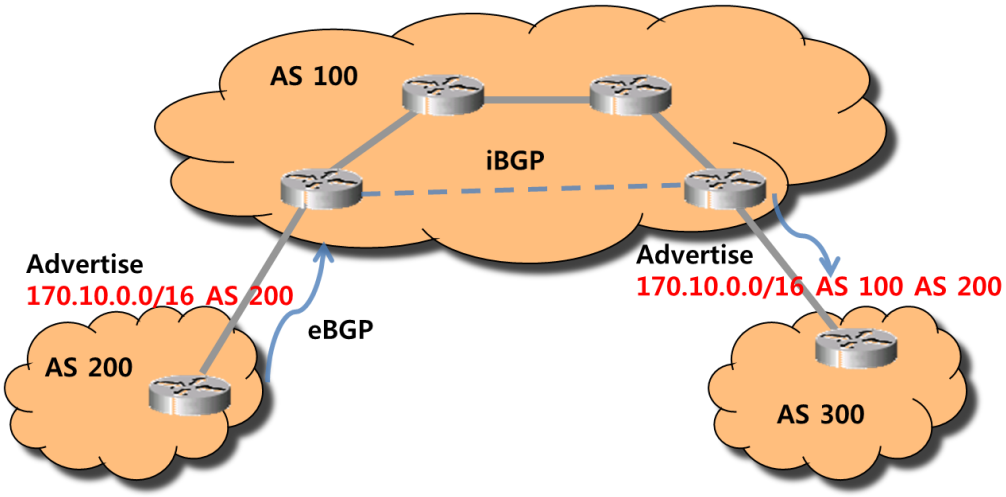


Figure 1. BGP network example

Classless Inter-Domain Routing (CIDR) is a method for allocating IP addresses and routing IP packets. The Internet Engineering Task Force (IETF) [5] introduced CIDR in 1993 to replace the previous addressing architecture of classful network design in the Internet. Their goal was to slow the growth of routing tables on routers across the Internet and to help slow the rapid exhaustion of IPv4 addresses [6][7].

2.1.1 BGP Update Message

BGP neighbors are established by manual configuration between routers to create a TCP session on port 179. Among routing protocols, BGP is unique in using TCP as its transport protocol. There are four message types in BGP. The *open message* allows BGP peers to identify their capabilities to each other, the *update message* is used to advertise/withdraw prefixes, the *notification message* is used to send errors/close the session, and the *keepalive message* serves to keep the BGP session up.

The update message is at the heart of BGP. Update messages are used to announce one or more prefixes to a BGP peer. The sender of the prefix must have a route to the prefix advertised, following the next-hop routing paradigm. Sooner or later a network failure or change will cause the sender of the prefix to lose its route the prefix it advertised. Hence, the update message must also include the ability to withdraw previously advertised prefixes. The update message specifies the following parameters:

- **Withdrawn Route Length**: The total length of the Withdrawn Routes field.
- **Withdrawn Routes:** A list of IP prefixes that the sender had announced but now wishes to withdraw. This could be a result of a change in the network topology or configuration.
- **Total Path Attributes Length:** The total length of the Path Attributes field.
- **Path Attributes:** A list of BGP attributes that apply to the prefixes described in the Network Layer Reachability Information field

- **Network Layer Reachability Information (NLRI):** A list of prefixes that the sender is advertising to its peer. Note that the path attributes listed earlier apply to all prefixes in the NLRI field.

A given BGP router may accept NLRI in update messages from multiple neighbors and advertise NLRI to the same, or a different set, of neighbors. Conceptually, BGP maintains its own “master” routing table, called the **Loc-RIB** (Local Routing Information Base), separate from the main routing table of the router. For each neighbor, the BGP process maintains a conceptual **Adj-RIB-In** (Adjacent Routing Information Base, Incoming) containing the NLRI received from the neighbor, and a conceptual **Adj-RIB-Out** (Adjacent Routing Information Base, Outgoing) for NLRI to be sent to the neighbor [8].

Conceptual, in the preceding paragraph, means that the physical storage and structure of these various tables are decided by the implementer of the BGP code. Their structure is not visible to other BGP routers, although they usually can be interrogated with management commands on the local router. It is quite common, for example, to store the two Adj-RIBs and the Loc-RIB together in the same data structure, with additional information attached to the RIB entries. The additional information tells the BGP process such things as whether individual entries belong in the Adj-RIBs for specific neighbors, whether the per-neighbor route selection process made received policies eligible for the Loc-RIB, and whether Loc-RIB entries are eligible to be submitted to the local router's routing table management process.

2.1.2 BGP Attributes

Routes learned via BGP have associated properties that are used to determine the best route to a destination when multiple paths exist to a particular destination. These properties are referred to as BGP attributes, and an understanding of how BGP attributes influence route selection is required for the design of robust

networks. The attributes that BGP uses in the route selection process are as follows [9].

- **Local Preference Attribute (LOCAL_PREF):** The local preference attribute is used to prefer an exit point from the local AS. Unlike the weight attribute, the local preference attribute is propagated throughout the local AS. If there are multiple exit points from the AS, the local preference attribute is used to select the exit point for a specific route. The route with a higher local preference will be used as the exit point.
- **Multi-Exit Discriminator Attribute (MED):** The multi-exit discriminator or metric attribute is used as a suggestion to an external AS regarding the preferred route into the AS that is advertising the metric. The term suggestion is used because the external AS that is receiving the MEDs may be using other BGP attributes for route selection. The lower value of the metric is preferred.
- **Origin Attribute (ORIGIN):** The origin attribute indicates how BGP learned about a particular route. The origin attribute is used for route selection and can have one of three possible values. '**IGP**' means that the route is interior to the originating AS. This value is set when the network router configuration command is used to inject the route into BGP. '**EGP**' is set when the route is learned via the EBGP. '**Incomplete**' means that the origin of the route is unknown or learned in some other way. An origin of incomplete occurs when a route is redistributed into BGP.
- **AS path Attribute (AS_PATH):** When a route advertisement passes through an AS, the AS number is added to an ordered list of AS numbers that the route advertisement has traversed. The smallest AS path length is preferred.
- **Next-Hop Attribute:** The EBGP next-hop attribute is the IP address that is used to reach the advertising router. For EBGP peers, the next-hop

address is the IP address of the connection between the peers. For IBGP, the EBGP next-hop address is carried into the local AS.

- **Community Attribute:** The community attribute provides a way of grouping destinations, called communities, to which routing decisions (such as acceptance, preference and redistribution) can be applied. Route maps are used to set the community attribute.

2.1.3 BGP Path Selection

BGP routers typically receive multiple paths to the same destination. The BGP best path algorithm decides which the best path is to install in the IP routing table and to use for traffic forwarding. When the path is selected, BGP puts the selected path in the IP routing table and propagates the path to its neighbors.

BGP is a Distance Vector protocol that uses the lowest metric to select the best path to a destination. However, BGP's decision process is relatively complex. This complexity is due to the number of BGP attributes; each BGP attribute has a place in the decision process. Of course, if there is only one path to a prefix, the decision process is unnecessary.

Because each entry in a routing table may specify a network, one destination address may match more than one routing table entry. The most specific table entry – the one with the highest subnet mask – is called the **longest prefix match**. It is called this because it is also the entry where the largest number of leading address bits in the table entry match those of the destination address.

The input to this decision process is a number of paths to the same prefix (with the same prefix length), known via BGP. Each path is accompanied by a set of attributes. The output of the algorithm is a single best path to the prefix. The best path is a candidate for advertisement to other BGP peers and to be placed into the routing table. The BGP decision process is as follows [10]:

- Prefer the path with the highest LOCAL_PREF value. A path without LOCAL_PREF is considered to have had the value set with the *bgp default local-preference* command, or to have a value of 100 by default.
- Prefer the path with the shortest AS_PATH attribute. This step is skipped if *bgp bestpath as-path ignore* command is used. An AS_SET counts as 1, no matter how many ASes are in the set.
- Prefer the path with the lowest origin type. IGP is lower than EGP and EGP is lower than Incomplete.
- Prefer the path with the lowest MED attribute. By default, the MED attribute is considered only when a prefix is received from neighbors in the same AS. To allow the comparison of the MED attribute when the prefix is received from neighbors in different ASes, the *bgp always-compared-med command* is used.
- Prefer an EBGP path over an IBGP path.
- Prefer the path with the lowest IGP metric to the next hop.
- Prefer the path originated by the BGP router with the lowest router ID.

This process is briefly summarized in Figure 2. Whenever a conceptual Adj-RIB-In changes, the main BGP process decides if any of the neighbor's new routes are preferred to routes already in the Loc-RIB. If so, it replaces them. If a given route is withdrawn by a neighbor, and there is no other route to that destination, the route is removed from the Loc-RIB, and no longer sent, by BGP, to the main routing table manager. If the router does not have a route to that destination from any non-BGP source, the withdrawn route will be removed from the main routing table.

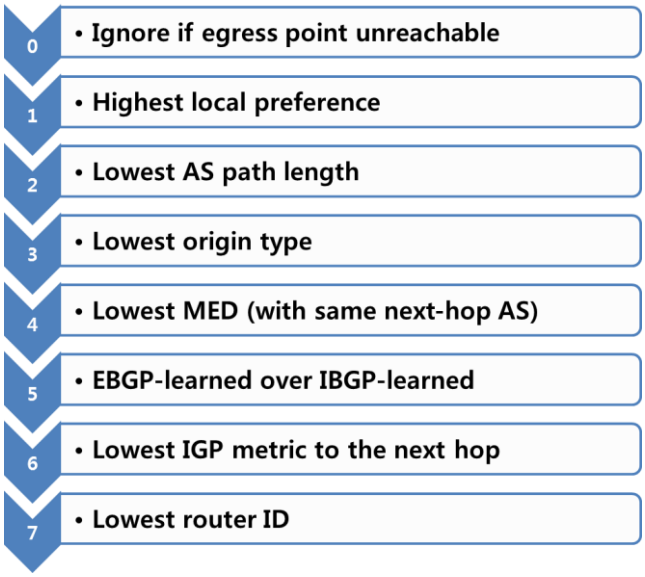


Figure 2. BGP decision process

2.2 BGP Attacks

BGP routers are critical to successful movement of data in the Internet. Attacks on BGP routers, BGP protocol, underlying protocols and network links can disrupt BGP router operation and cause loss of Internet services. Understanding the attacks that are possible will help place the various mitigation techniques in context. The following discussion of BGP attacks in based on material from several sources [11][12][13][14][15][16][17][18]. Attacks on BGP routers occur at various levels, from attacks on the underlying TCP protocol to attacks on BGP routers themselves.

Since BGP is based on TCP and IP, it is vulnerable to all threats against its underlying protocols. For example, BGP is vulnerable to a TCP reset attack which can result in significant Internet instability. In addition, BGP faces threats from both BGP speakers and BGP sessions. A BGP speaker may be compromised,

misconfigured or unauthorized. An attacker can also set up its own BGP speaker and connect it to the Internet by purchasing connection from a sloppy ISP. And a BGP session may be compromised or unauthorized.

2.3 BGP Security

Securing inter-domain routing has been a challenge for many years. The majority of defenses that have been implemented by ISPs to protect BGP have focused on solutions that can be implemented locally or require only limited interaction with parties outside the local administrative domain. In particular, protection of the underlying TCP connection and defensive filtering of BGP announcements are the most commonly implemented solutions, with some limited deployment of cryptographic protections between routers. However, these solutions are ultimately limited in the protections they can offer against more complex and sophisticated attacks that target BGP itself. Ultimately, a more complete view of which routes are valid is necessary for protecting against this latter class of attacks.

3 IP Prefix Hijacking

This chapter introduces the causes and threats of IP prefix hijacking. It also illustrates IP prefix hijacking incidents and the related research on IP prefix hijacking detection.

3.1 Threats of IP Prefix Hijacking

The Internet is a collection of ASes compromised of a set of routers and networks under the same management. ASes are connected by dedicated lines or Internet exchanges and use the BGP to exchange their routing information. The routers determine the paths to forward IP packets along according to routing information exchanged by BGP.

The Internet was designed to provide communication on the basis of trust between networks, but has proved to be a misguided assumption, due to the various types of attacks that have taken advantage of this trust. ASes that exchange BGP information directly with each other are assumed to be trusted, so BGP does not implement any security checks, such as checking the authenticity of origin information and path attributes, to protect against receiving invalid routing information from other routers. As such, the Internet routing infrastructure is vulnerable to attack. IP prefix hijacking is a BGP security attack, in which a BGP router, either with malicious purposes or simple due to misconfiguration, announces an IP prefix that the router does not own. This false announcement creates reachability problems and communication failures throughout the Internet. IP prefix hijacking incidents are often reported on the NANOG mailing list [35].

IP prefix hijacking is a well-known security threat that corrupts Internet routing tables. As there is no authentication mechanism used in BGP, a malicious

or misbehaving router can announce routes to any destination prefix on the Internet and even manipulate route attributes of the routing updates it sends to neighboring routers. Taking advantage of this weakness has become the fundamental mechanism for IP prefix hijack attacks. IP prefix hijacking is sometimes used by malicious users as a means of illegally obtaining IP addresses for spamming or launching Distributed Denial of Service (DDoS) attacks.

IP prefix hijacking can occur on purpose or by accident in one of several ways:

- **Regular prefix hijacking**: An AS announces that it originates a prefix that it does not actually originate. This attack occurs when the attack router originates a route to an existing IP prefix of the victim network. As a result, the Internet is partially polluted, depending on how preferable the bogus route is compared to the valid route from the perspective of various networks.
- **Sub-prefix hijacking:** An AS announces a more specific prefix than what may be announced by the true originating AS. This attack results from stealing a subnet of an existing prefix in the routing tables by announcing a route for the subnet originating from the attacker network. Due to longest prefix matching based forwarding, most networks are polluted.
- **AS Path Falsification:** An AS announces that it can route traffic to the hijacked AS through a shorter route than is already available, regardless of whether or not the route actually exists.

IP prefix hijacking can occur on purpose or by accident in several ways. Many previous studies have classified IP prefix hijacking in detail [51][52][54]. We briefly explain the three types of IP prefix hijacking. Regular prefix hijacking occurs when the attacker AS announces a prefix that it does not actually own. As its wrong announcement is propagated, the Internet becomes to be polluted.

Because the routers prefer the shortest AS path to forward traffic, not all of ASes in the Internet are polluted. Sub-prefix hijacking happens when the attacker AS announces a more specific prefix than what may be announced by the true origin AS. Most ASes are impacted by this announcement because the priority of more specific IP prefix is higher in route selection process. Lastly, IP prefix interception is that the attacker AS forwards the hijacked traffic to the origin AS. In this case, the victim cannot recognize the occurrence of prefix interception.

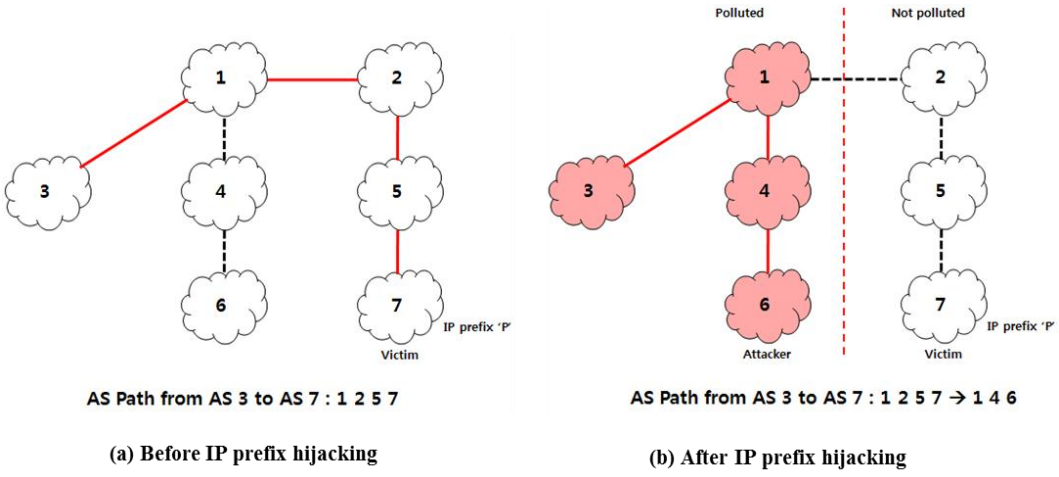


Figure 3. IP prefix hijacking: polluted and unpolluted ASes

Figure 3 shows an example of IP prefix hijacking with AS relationships. We suppose that the attacker AS is 6 and the victim AS is 7. When the attacker announces the IP prefix that the victim actually originated, this malicious routing information is propagated in the Internet. Typically, ISPs can filter the announcements containing invalid IP addresses, but previous hijacking incidents show that the filtering may not be applied (e.g. through poor configuration). With the given shortest AS path preference in routing, networks (AS 1 and AS 4 in Figure 3) close to the attacker, ASes are polluted by the malicious announcement. AS 3 also receives the announcement and must decide whether the update is applied to the routing table.

Because the routing tables of the ASes near AS 3 are polluted, AS 3 is unable to communicate directly with AS 7. However, the unpolluted ASes can still reach the victim AS. A detection system using the information from multiple BGP monitoring points can recognize a MOAS conflict caused by IP prefix hijacking. However, this requires that monitoring points are located in both polluted and unpolluted ASes. That is, appropriate probing locations must be selected so that probing packets should reach two conflicted origin ASes through the different AS paths.

3.2 IP Prefix Hijacking Incidents

So far, there have been no major BGP routing attacks (or at least, they have not been publicly documented as malicious attacks). As a result, relatively little attention from the network research community has been placed on studying the routing infrastructure's overall susceptibility to malicious users. On the other hand, it has been shown that routing misconfigurations are quite common in practice, and they can cause the same reachability and BGP convergence problems that an attack could cause [17].

- The AS 7007 incident [36] was a major disruption of the Internet on April 25, 1997 that started with a router operated by AS 7007 accidentally leaking a substantial part of its entire route table to the Internet, creating a routing black hole. Probably because of a bug in the affected router, the routes leaked were de-aggregated to /24 prefixes, which were more specific than the routes originally present on the Internet, and had the AS path rewritten to 7007, leading the BGP used by the Internet's routers to prefer the leaked routes. This was then exacerbated by other problems that prevented the routes from disappearing from other networks' routing tables, even after the original router that had sent them had been disconnected. The combination of these factors resulted in an extended

disruption of operations throughout the Internet. Analysis of this event led to major changes in ISPs' BGP operations intended to mitigate the effects of any subsequent similar events.

- Similar events occurred on April 7, 1998, when AS8584 announced about 10,000 prefixes it did not own, and on April 6, 2001 when AS15412 announced about 5,000 prefixes it did not own [37].
- On December 24, 2004, TTNet in Turkey (AS 9121) started announcing what appeared to be a full table (well over 100,000 entries) of Internet routes to all of their transit providers [38]. A misconfigured peering with an upstream AS allows these advertisements to be propagated throughout the global Internet, affecting just about everyone. Various portions of the Internet become unavailable for different organizations around the world for several hours.
- On January 22, 2006, Con Edison (AS 27506) begins originating routes for a number of prefixes which are not its own [39]. Some belong to its customers, while others are entirely unaffiliated. The invalid advertisements persist for several hours.
- On February 24, 2008, Pakistan Telecom (AS 17557) begins announcing part of YouTube's address space [40]. This was intended to be done internally as part of an effort to block access to YouTube from within Pakistan, but was propagated to the global Internet as well. Because the prefix advertised by Pakistan is more specific than the legitimate YouTube route, it is preferred by all autonomous systems which received it. Throughout much of the world, traffic destined for YouTube is routed toward Pakistan, where it is discarded. YouTube responds first by advertising its own /24 route for the affected prefix, equivalent to the route announced by Pakistan Telecom, and then by splitting the route into two /25 prefixes. It is not until PCCW Global (AS 3491), Pakistan

Telecom's upstream provider, withdraws all routes originating from AS 17557 that normal connectivity to YouTube from the rest of the world is restored.

- On April 2010, AS 23724, which belongs to China Telecom, begins announcing bad routing information for between 32,000 and 37,000 networks, redirecting them to IDC China Telecommunication instead of their rightful owners [41]. These networks included about 8,000 U.S. networks including those operated by Dell, CNN, Starbucks and Apple. More than 8,500 Chinese networks, 1,100 in Australia and 230 owned by France Telecom were also affected. That bad information was then accepted by the larger China Telecommunications, which shared the data with other major ISPs. Within minutes the bad data had spread around the globe. ISPs may have accepted the hijacked route information, but that doesn't necessarily mean that a lot of Web surfers got redirected. It's common for routers to learn several BGP routes, and then route traffic to what they consider the best route. Often they choose the shortest route available. So most routers in the U.S. would have routed traffic to Apple's servers, for example, instead of IDC China Telecommunication.
- On January 2011, in response to civilian uprising and wide-spread protest, the Egyptian government moves to block in-country access to several prominent social network sites [42]. Days later, virtually all Internet access from within the country is severed. This is believed to be the first ever example of an entire country intentionally isolating itself from the global Internet. The outage lasted from January 27th to February 2nd.
- On February 2011, spurred by uprising in neighboring Egypt, protests break out in Libya [42]. In a manner similar to Egypt, the Libyan government responds by severing Internet access. Although the initial outage lasted for only a few hours, the coming weeks would see repeated interruptions to Libyan Internet access.

3.3 IP Prefix Hijacking Solutions

The RPSEC (Routing Protocol Security Requirements) working group of IETF has released a number of Internet-Drafts regarding a scheme to improve routing protocol security, and they discuss general security threats and requirements of routing protocols [43][44]. Path attributes and Network Layer Reachability Information (NLRI) authentication is one of the requirements. This provides a means to verify peer relationships and prefix advertisements to prevent unauthorized announcements.

Previous cryptographic work provides integrity for routing information. One of the proposed BGP security architectures is S-BGP [45], which employs three security mechanisms – Public Key Infrastructure (PKI) for the secure identification of BGP speakers, ASes, and address blocks; Attestations to ensure the authenticity and integrity of data for route and address; and use of the IP security (IPsec) protocol to secure point-to-point communication between BGP speakers. S-BGP requires collaboration between Internet registries and ISPs to set up PKI. In addition, PKI causes high overheads and requires wide deployment in Internet registries, router vendors and ISPs. The other proposal is soBGP [46], which uses certificates dedicated to authenticating AS peers. It then uses two additional certificates (to prove that an AS has the authority to advertise a block of addresses and to certify the advertisement conforms with the policies of the originator). Even though S-BGP and soBGP solve the security problems of the routing protocol, they are not easy to deploy in the current Internet environment.

As mentioned above, there are some problems and limitations in the cryptography techniques to protect against known IP prefix hijacking cases. As such, many previous studies have proposed methods to detect IP prefix hijacking events. The IP prefix hijacking detection schemes have different characteristics from the viewpoint of the information used, detection subject and attack type. Either control plane or data plane information can be used in order to detect IP

prefix hijacking. The control plane is part of the router architecture, and the detection system uses information from a routing table or information collected by routing protocols. Methods using data plane information rely on active probing in real-time, and then analyze the probing results to detect whether an abnormal case occurs.

From the viewpoint of the detection subject, previous work on IP hijacking detection can be categorized into victim-centric, infrastructure-based or peer-centric approaches. The victim-centric approach detects IP hijacking for its own IP. Infrastructure-based approach relies on a centralized database or a set of vantage locations distributed over the Internet to detect IP hijacking at any point of the Internet. The peer-centric approach detects hijacking of communication peer IPs based on the analysis of routing messages. The victim-centric approach has the advantage of protecting itself from its own IP being hijacked. However this method requires a lot of probing work from the network all over the Internet. The infrastructure-based approach has a wide range of coverage and has advantages in terms of the simplicity of the detection algorithm. However it has drawbacks in deployment due to the amount of vantage points needed and the requirement for cooperation among ASes. The peer-centric approach guarantees communication with the intended party. However it has narrow range of coverage in the sense that IP hijacking will not be detected by others whose detection system is not working, including the victim.

The attack types of IP prefix hijacking can be divided into prefix hijacking and AS path falsification. Prefix hijacking means involves an attacker falsifying the NLRI field of a BGP update message. Sub-prefix hijacking clearly has a wider influence on the polluted ASes than regular prefix hijacking because of high routing priority for short IP ranges. AS path falsification attacks occurs when an attacker modifies a path attribute of an update message.

We have categorized previous work on IP prefix hijacking detection, Table 1

shows the characteristics of previous work from the three points of view described in the above three paragraphs.

Kruegel et al. proposed an IP hijacking detection algorithm based on the understanding that prefix ownership is not dynamic in nature [49]. The prefix ownership lists are computed in advance and new AS announcements are verified with pre-computed prefix ownership to remain consistent with network topology. This approach relies on routing information that triggers verification of topology changes and a peer-centric approach for detecting prefix hijacking. The main drawback is low accuracy that comes from the depending on statistical data about network topology changes.

Lad et al. proposed a method which monitors the occurrence of new origin ASes in real time and notifies the prefix owners that a suspicious update has occurred [50]. To detect a suspicious event, routing updates from RouteViews [47] and RIPE repository [48] are investigated and e-mail notifications are sent to the origin prefix owner through the PHAS server. This method relies on mutual cooperation between ASes and inevitably has no protection against false registration on the PHAS server. However it gives us an indication about how to mitigate IP prefix hijacking attacks.

Zheng et. al utilized information collected from the data plane to detect IP prefix hijacking [51]. They are motivated by two key observations: hop count instability and path disagreement during IP hijacking. Monitoring systems from multiple vantage points monitor hop counts and path to ASes and calculate their similarity before and after routing update. If the similarity crosses some threshold value, the monitor raises an alarm to indicate IP prefix hijacking.

Hu and Mao classify IP prefix hijacking attacks in various types and suggested detection methods for each attack type [52]. Their methods are largely based on fingerprint comparison performed by monitoring systems from multiple

vantage points. When the monitoring systems sense suspicious routing updates, they collect fingerprints from the prefix owner's AS. They conclude IP hijacking has occurred if the fingerprints are in conflict with each other. However, Hu and Mao do not provide an appropriate fingerprinting method for this IP hijacking detection. They rely on general fingerprint methods such as host property, IP identity and TCP timestamp which are used mainly for understanding the target host or network. IP hijacking detection should rely on appropriate identification technology to make an accurate decision. Also, they do not consider the limitations on fingerprint collection caused by firewalls in some networks that block external probing packets. Our proposed method is also based on fingerprint comparison in order to detect IP hijacking. However, we propose new fingerprinting methods that can be used for host and network identification. Our proposed method works well with networks that are protected by firewalls, IDS and IPS. Additionally, we make it clear that the DNS server is the live host for collecting fingerprints of the target host. However Hu and Mao do not deal with the matter of live hosts for fingerprint collection.

Karlin et al. proposed a system that automatically delays the use and propagation of suspicious routes [53]. Introducing a delay gives human operators and systems time to investigate the suspicious routes. The router identifies suspicious routes by consulting a table of trusted routing information learned from the recent history of BGP update messages. This method has some false positive cases, for example when the provider changes or the occurrence of a previously unseen provider.

Zhang et al. presented a victim-centric approach using reachability from the prefix owner's view point [54]. They observed that IP prefix hijacking is likely to result in more diverse reachability problems from other networks. If an AS is polluted by IP prefix hijacking, the reply packet sent by the polluted network will not correctly reach the victim network. A victim network should perform a

continuous round of probing from its prefix to all transit ASes. When the number of unreachable network exceeds a predetermined threshold value, the detection system reports the occurrence of the hijacking.

Existing proposals used control plane or data plane information to detect anomalous behavior. The techniques using control plane information perform analyzing routing tables and passive monitoring of BGP routing updates [49][50][52][53]. Such routing information implies abnormal activities such as origin AS changes and false AS edges. Data probing schemes provide a way to check the reachability of destination prefix at the time of monitoring [51][52][54]. It is important to properly select monitors that are able to collect host properties or measure hop count to an IP prefix.

There are two types of infrastructure-based approaches. One uses a centralized database such as RouteViews and RIPE repository that collect real-time information about the global routing system from several different backbones and locations around the Internet [50][52]. The other uses a set of vantage points in order to collect several sets of active probing results and compare those with past data or each other for detecting IP prefix hijacking [51][52]. The detection systems using peer-centric schemes analyze routing messages through their own network and apply a detection algorithm to the message in order to identify whether a suspicious event occurred [49][53].

Compared to other approaches, RBHD uses both routing information and data probing, covers known attack types and focuses on third-party schemes. The existing proposals using one of the two types of used data have some limitations such as not covering known attack types. The approaches using both control and data plane information can increase detection accuracy. Also, the proposed method uses a peer-centric scheme in order to detect hijacking incidents without constructing any additional infrastructure. Therefore, our proposed method can be easily adoptable and detect known kinds of hijacking attack regardless of attack

types without mutual cooperation between ASes.

Table 1. Comparison among anomaly detection system

Research work	Type of used data		Detection approach			Attack type	
	control plane	data plane	victim-centric	infrastructure-based	peer-centric	prefix hijacking	AS path falsification
Topology	V				V	V	
PHAS	V			V		V	
Distance		V		V		V	
Fingerprint	V	V		V		V	V
pgBGP	V				V	V	
iSPY		V	V			V	
RBHD	V	V			V	V	V

4 Network Reachability Monitoring

We propose a new approach which practically and effectively detects IP prefix hijacking based on network reachability – we call this method RBHD (Reachability Based Hijacking Detection). Network reachability is a property that dictates whether traffic sent by someone can reach its intended network. In other words, it means that sent packets reach the intended original network that owns a specific IP prefix, not a destination with the same, hijacked prefix. The network reachability defined above should be maintained even if the network path is changed due to routing instability. Therefore, the proposed method identifies whether we can reach the network through the changed path when a routing update message arrives and the routing path is changed. If we can reach the identical network, the update is regarded as a normal routing event. If not, we decide that it is a case of IP prefix hijacking. It is significant to note that network reachability differs from IP reachability in that network reachability has a strict requirement of reaching the target. IP reachability only measures whether a packet can reach the intended IP host. When IP hijacking occurred, IP reachability is maintained even though the target host was changed. However network reachability is corrupted due to the change of IP path to the target network.

We use a fingerprinting scheme in order to determine the network reachability of a specific network. Like a human fingerprint, this fingerprint captured from a network can be used for identification in the network reachability test. We classify the types of fingerprints for the network reachability test into host fingerprints and network fingerprints. Host fingerprints are the implementation specifications of the operating system and protocol stacks, or the current configurations of a running application in the host. The host fingerprint identifies a network. The host fingerprint is now always available from all ASes on the Internet, so it is easier to capture and use than a network fingerprint. On the other

hand, Network fingerprints characterize a specific network, for example, a value representing host availability in that network. We collect the host and network fingerprints of an AS owing a specific IP prefix before any IP hijacking event, then we compare those with newly captured fingerprints of the AS whenever a suspicious routing update is received. In this paper, we propose host fingerprinting and network fingerprinting methods for IP prefix hijacking detection.

4.1 Reachability Monitoring

The Internet is a huge and decentralized network comprised of ASes that exchange routing information using BGP. The goal of routing is to select a path that can send network traffic between hosts that are not directly connected. However, network communication is affected by changes of the routing path, network topology and device configuration.

Network reachability is a decision about whether traffic sent by someone can reach the intended network through a path announced by the routing protocol. Network reachability cannot be guaranteed by BGP routing because of physical failure or a software error in network devices. In other words, we cannot be sure of the network reachability of a destination just by knowing the routing information. Therefore, we need to identify network reachability by performing an actual verification test such as active probing.

IP prefix hijacking is an attack that influences network reachability. When IP prefix hijacking occurs, network traffic is delivered to the attacker network, not the destination network. That is, network reachability to the destination network is not guaranteed. If we know some preliminary information about the destination network and analyze responses by active probing based on that information, then we can identify whether network reachability is guaranteed. In this thesis, we propose an IP prefix hijacking detection method using the approach mentioned

above.

When we monitor network reachability, it is hard to determine how to collect characteristics about a specific network. Network reachability monitoring usually depends on active monitoring techniques because there is a bit limitation to data collected by passive monitoring and it is difficult to analyze this data in real-time. Existing reachability monitoring among active monitoring techniques uses a direct method with ICMP that is used in tools such as *ping* or *traceroute*. However, ICMP packets suffer from improved network security policies when they pass through a firewall or an Internet junction in an enterprise network. Also, an end-host can filter ICMP packets or not respond to active probing packets. We must make an appropriate probing packet and select a host which always responds to it in order to collect network characteristics through a firewall. In other words, various and smart probing methods should be applied because there is limitation to data collected by using existing simple tools.

In this thesis, we propose two approaches in order to collect network characteristics. One is to perform host fingerprinting by selecting some hosts that are representative servers in the target network, for examples, web or DNS servers. A probing packet to the services provided on these servers can easily pass through a firewall and the servers should properly respond to the received packet. Therefore, without a deep regard to network security we can collect host fingerprint information about a host running as a server at any time. The other is to perform network fingerprinting by inferring a network's firewall policy and using it as a network characteristic. We design an effective method to accurately infer a firewall policy by performing active probing of a target network.

4.2 Host Fingerprinting

For network reachability monitoring, we need a method to confirm whether a

traffic sent from a source reaches the intended destination network. It is possible to check reachability by performing active probing to the destination network and analyzing response packets. If we already know the characteristics of the target network, such as the server's information in the network or the usage pattern of the domain names, then we can easily verify the network reachability by comparing between previously collected and current characteristics.

4.2.1 DNS Server as Host Fingerprinting Target

To collect unique features of a network, we utilize live hosts in the target network, which are used as a target of active probing. Some requirements of live hosts in terms of reachability monitoring are as follows: They must be 1) operated in most ASes, 2) have easy to obtain IP addresses, 3) always provide services for its AS, and 4) allow external connection and respond to active probing. Enforcing these requirements provide us certain benefits such as ensuring the live host's suitability in terms of network characteristics, allowing a simple preprocess for active probing, helping with the consistency of host fingerprints and ease of collecting fingerprints.

DNS servers that provide a conversion service between domain names and IP addresses satisfy all of these requirements. DNS servers are essential components of the Internet, and are operated in most network domains. If a domain does not operate its own DNS server, we have to use an IP address composed of dotted numbers, which is unpopular and inconvenient. As such, we use DNS servers as the target of host fingerprinting because they are part of the core infrastructure of the Internet, which always provides service and allows external connections from any host.

4.2.2 DNS Server Collection Method

The domain name is a character address that easily connects users to the

Internet. The domain is the logical area of the domain name and more than one DNS server typically operates in the domain. The DNS server has the responsibility of converting from domain name to IP address information including AS domain IPs. An IP prefix follows a similar concept to the domain, which has a network ID which is essentially a logical bunch of IPs. An AS applies equal routing policy within the network and typically has more than one prefix.

DNS server's main behavior is to convert from IP to domain name and vice versa. Converting is of two types, first one is reverse lookup and second one is forward lookup. Two types of operations are equal in behavior.

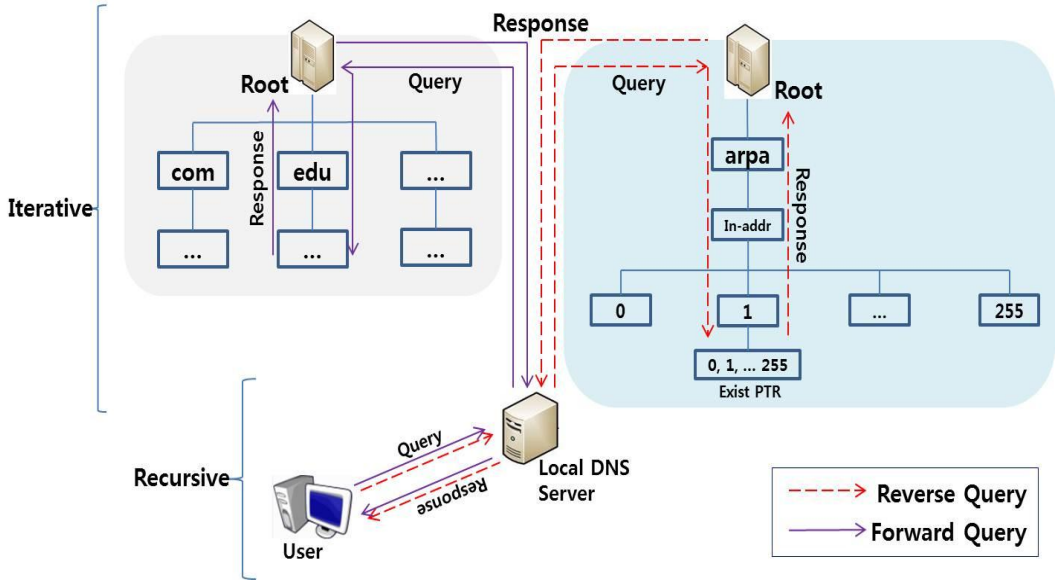


Figure 4. DNS conversion process

Figure 4 shows DNS server operation for user's request. The user in Figure 4 performs the conversion of IP or domain name at local DNS server. If local DNS server cache exists, then it answers the user's queries. If local DNS server cache does not exist, it sends queries to root DNS server. Once received by root server, it

sends user queries to responsible lower level DNS server. If lower DNS servers have answers to the queries received, then responses to the queries, and sends its answers to local DNS server.

If the lower level DNS server cannot answer the query received by the user, then it sends to their lower level DNS server. The preceding process continues that DNS server can answer user query. As a result, local DNS server receives the response by lower level DNS server for the sent query.

Operational behaviors of Reverse lookup and forward lookup are same. But, two have different responses to their queries. Reverse lookup requests for reverse point about IP and Forward lookup requests for IP about domain name. As a result, we receive reverse point by reverse lookup and IP by forward lookup.

DNS server maintains database which is called as zone file. Figure 5 show the structure of zone file. DNS server operates to search zone file by received question. Reverse zone file searches reverse lookup and forward zone file searches forward lookup.

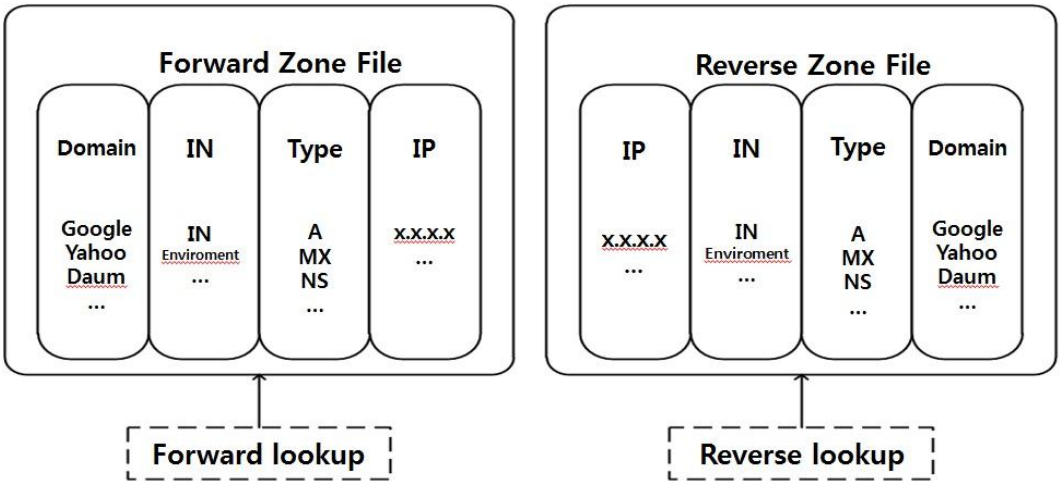


Figure 5. The structure of zone files in DNS servers

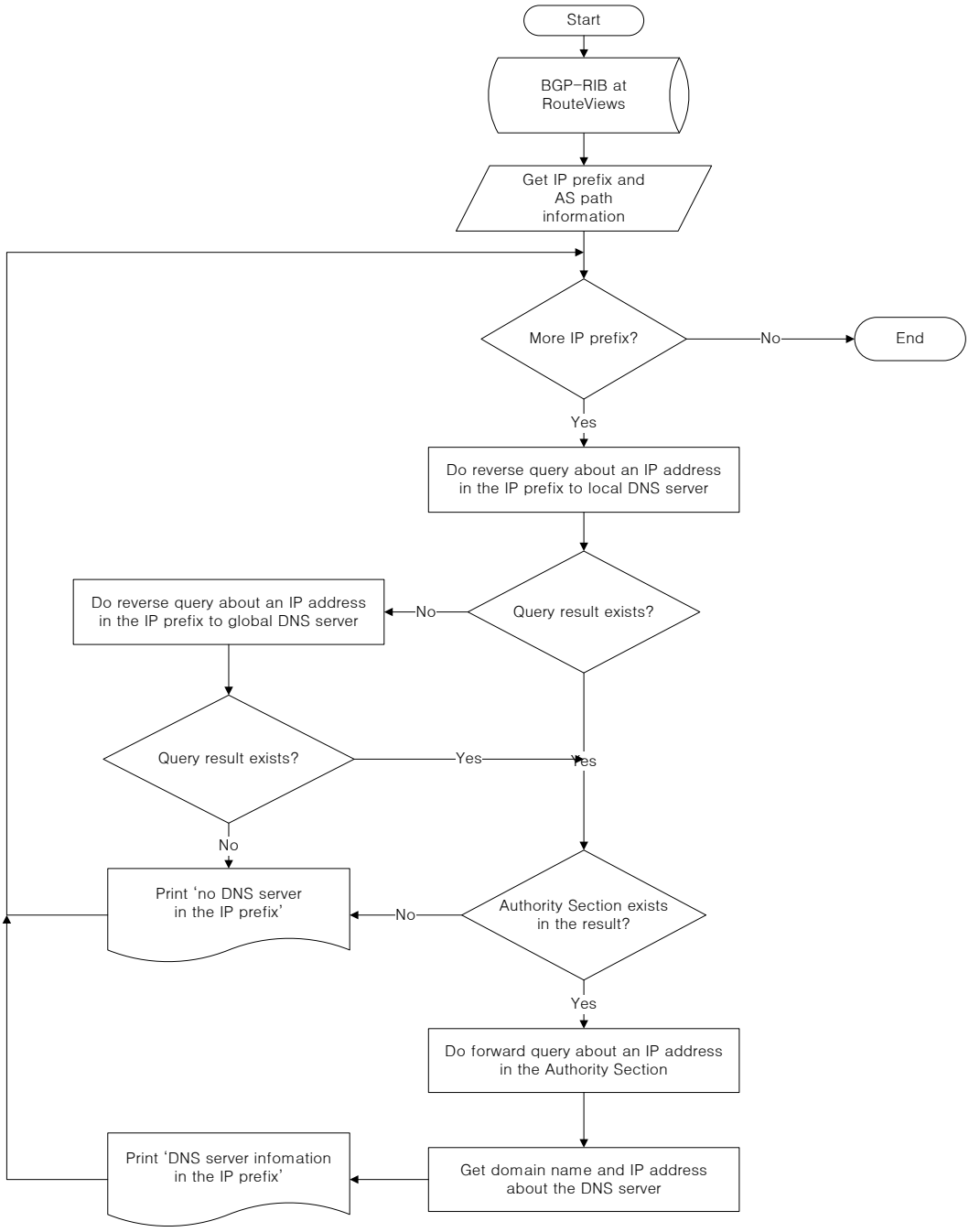


Figure 6. Flowchart of DNS server collection process

We use the BGP-RIB (Routing Information Base) of RouteViews which collects global routing information. RIB consists of IP prefixes and the routing path to those (AS path). To collect live hosts for all IP prefixes, we use a reverse DNS lookup querying local DNS and global DNS servers. Reverse DNS lookup is a query that returns domain names for an IP address. We perform that all prefixes in RouteViews divide into /24 prefix because reverse lookup does not support CIDR (Classless Inter-Domain Routing). Next, we perform a reverse lookup on the /24 prefixes. The first query is to find the authority server with authority over a particular IP prefix. The second query is to obtain an IP address through the domain name of a DNS server. As a result, we can finally obtain the IP addresses of the DNS servers on the Internet. Figure 6 is a flowchart of our proposed DNS server collection process.

Table 2 shows the current state of DNS server operation in the Internet. There were 314,106 IP prefixes consistently appeared in BGP-RIB at RouteViews over three months. We divided these prefixes into 8,414,294 /24 prefixes and collected total 102,843 DNS server's information using DNS forward/reverse query to /24 prefixes. The number of IP prefixes in which was operating a DNS server is 279,384 (88.94%) and the number of ASes in which was operating a DNS server is 32,182 (98.30%).

Table 2. The current state of DNS server operation in the Internet

	IP Prefixes		ASes	
	operated	total	operated	total
The number of DNS servers	279,384	314,106	32,182	33,756

Figure 7 shows how many IP prefixes each AS has in the Internet. The

average number of IP prefixes which one AS should have is 9.3. However, many IP prefixes are concentrated in specific ASes, shown in Figure 7. ASes around the world have at least one prefix and up to 4,378 prefixes. The most of ASes which have many prefixes are large ISPs. For examples, Korea Telecom (KT) [59], Level 3 Communications [60] and etc. belong to this category.

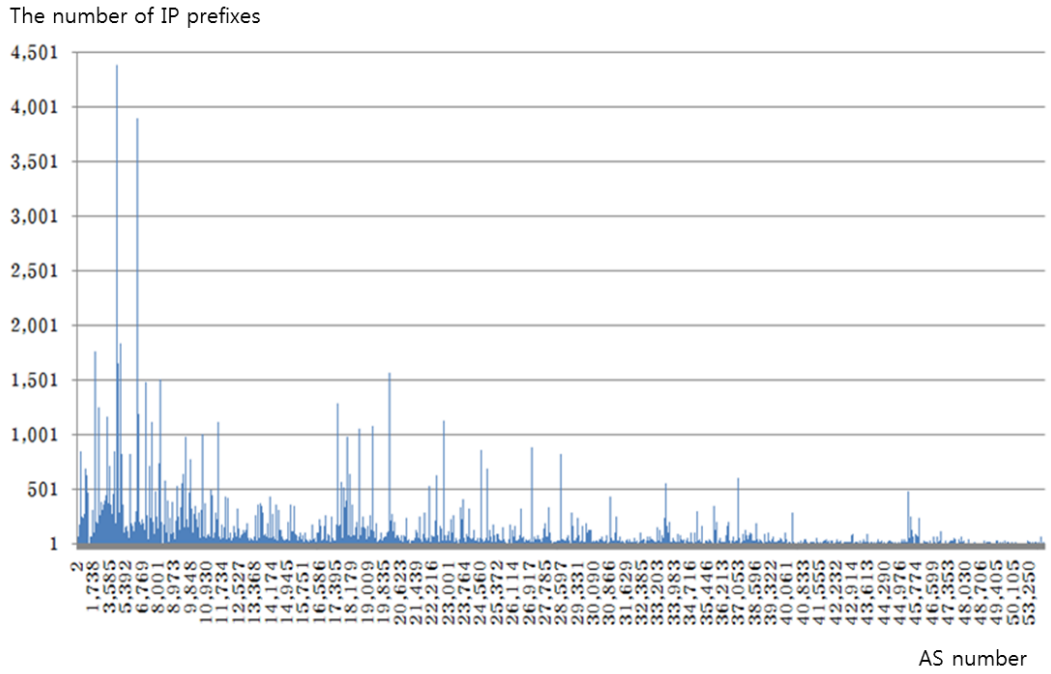


Figure 7. The number of IP prefixes owned by each AS

4.2.3 DNS Server Fingerprinting Method

DNS server fingerprinting uses DNS protocol information, DNS domain name information and DNS server configuration information. The proposed DNS server fingerprints are as follows:

- **DNS protocol information**: the type and version of DNS software [61] obtained through DNS query/response, and the uptime of DNS server obtained by using TCP timestamp option
- **DNS domain name information**: AA (Authoritative Answer) flag, authority section and additional section in DNS query/response message
- **DNS server configuration information**: the application of DNSSEC (Domain Name System Security Extension) [62][63] and the state of TCP port 53

For the generation of distinguishable host fingerprints about DNS servers, we use all the proposed DNS server fingerprints.

Figure 8 shows how to find the DNS server fingerprinting, organization of system for collecting of DNS protocol, DNS query, domain name information, DNS server configuration. The DNS server list shown in Figure 8 was obtained from RIB that appeared in RouteViews from May to July 2011 over the three months. We collected preliminary results about distribution of DNS servers on the Internet.

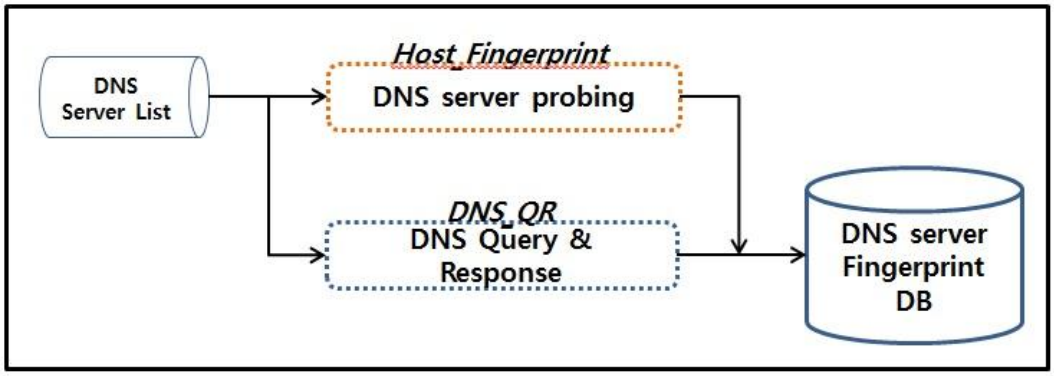


Figure 8. Structure of DNS server fingerprinting

Then, we used the collected of DNS server list to perform host fingerprinting

with DNS query and reply. Throughout the host fingerprinting, we obtained DNS server fingerprints such as the DNS software type and version, state of the TCP 53 port, uptime of the device. The responses of the DNS queries include authority and additional section of DNS message, AA flag bit status, and application of DNSSEC. The results are stored in the DNS server fingerprint database. The fingerprints stored in the DB are combined with zone information that can distinguish the information of each DNS server. The structure of the DNS server fingerprint (***HF***) is as follows:

HF {

aa_flag; // whether AA flag is used

auth_c; // the number of record in authority section

addi_c; // the number of record in additional section

dnssec; // whether DNSSEC is applied

software; // the type and version of DNS software

tcp_53; // the state of TCP port 53 (open | closed | filtered)

uptime; // system uptime

}

4.3 Network Fingerprinting

In addition to host fingerprinting for network reachability monitoring, we can use network fingerprinting as a method of collecting network features. Host fingerprinting has a shortcoming in that it is impossible to always find a live host in the target network, or sometimes although a live host can be found it is difficult

to perform active probing due to security issues. Therefore, we collect the network's characteristics themselves as a fingerprint, not the fingerprints of an individual host in the network.

There are many features that can be used as network characteristics, such as network topology, host availability and etc. However, it is nearly impossible to know the inner topology of a faraway network, and host availability in almost all networks frequently change. As such, we need a gatherable and consistent property in order to perform network fingerprinting.

In this thesis, we use the firewall policy of the destination network as a network fingerprint. The Internet is exposed to many threats and the attackers often make the target network unable to provide normal service by sending bad traffic. Therefore, most organizations secure their internal hosts by a firewall that blocks traffic from outside to inward and vice versa. The firewall policy, that is which traffic is accepted or denied, is extremely varied for each network according to the objective of an organization, security degree and the application of network infrastructure. As such, we can use firewall policy as a feature to distinguish between different networks.

Inferring firewall policy is semantically equivalent to the policy of a firewall deployed at the Internet access point of the target network. To discover the firewall policy of the target network, we generate probing packets, send them and infer the policy by analyzing response packets. First, we assume that default deny rules exists and add them to the allow rules sequentially. If we receive a response packet to the probing packet, we add to the allow rule regarding response packets header information. To add an allow rule effectively, we determine the next probing packet based on the firewall response. We need to select probing packets intelligently to allow us to effectively infer the firewall policy. It is very important to select appropriate probing packets. The firewall policy of the target network should be constructed by repeatedly sending and receiving packets. Eventually,

the constructed firewall becomes semantically equivalent to the policy of the firewall deployed in the target network. If many packets are sent to the target network during this process, there is a concern that these packets may be recognized as an Internet attack like IP address or port scanning. Therefore, packets need to be selected intelligently.

Our approach is similar to Tagrid Samak's approach [65][66] which uses space searching, but we are concerned with searching 1~*n*-1 dimensional shapes in *n*-dimensional space, instead of *n*-dimensional shapes. For instance, we search 1-dimensional shapes such as lines, 2-dimensional shapes such as rectangles in 2-dimensional space. By decreasing 1-dimension, Searching space was permitted efficiently.

4.3.1 Theoretical Background

As already mentioned, we changed the inferring a firewall policy problem that converts polygon from *n* dimension to 1 ~ *n*-1 dimension. For example, we search a line from 2-dimension to 1-dimension. Previous work searched rectangle from 2-dimension to 2-dimension. Validity of our method is proved a thought of previous work. In addition, this thesis suggests effective method of proposed space search. In this section, we explain inference results are more useful than previous work.

Direction fields of firewall policy are only two types for inbound and outbound, so search space is small. In the case of protocol, we consider to restrict protocol about TCP, UDP, ICMP, IP, SNMP, RTP and etc. As a result, search space is small. The primary fields of search space are source IP, source port, destination IP and destination port. When a firewall policy is constructed using four fields, permission rules are comprised of operating server which allows access from internal to external.

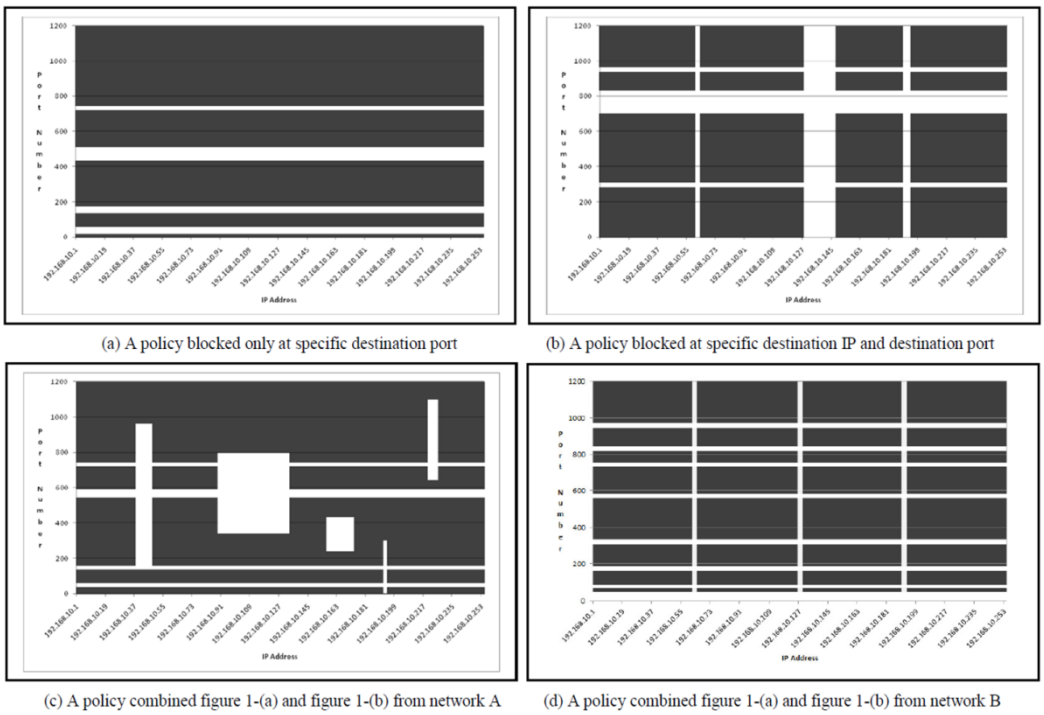
The characteristics of the firewall policy should be constructed so that, it's should be smaller than dimensional shapes of the search space. In other words, external access which has specific IP, replaces restriction and effect of restriction decrease on one dimension on. As per the discussion made on the server restriction, same effect is appeared by specific server restriction on connection, which allows the access from internal to external. For example, Table 3 shows the effect of restriction of rules R1 and R2 on destination port source port numbered 80 respectively.

Table 3. Example of firewall policy

Rule	**Direction**	**Protocol**	**Src IP**	**Src Port**	**Dst IP**	**Dst Port**	**Action**
R1	Outbound	TCP	192.168.10.*	1024:65535	Any	80	Permit
R2	Inbound	TCP	Any	80	192.168.10.*	1024:65535	Permit
R3	Inbound	TCP	Any	1024:65535	192.168.10.10	80	Permit
R4	Outbound	TCP	192.168.10.10	80	Any	1024:65535	Permit
Default	Any	Any	Any	Any	Any	Any	Deny

Analysis of experimental results is made on validity of existing firewall policy, where method searches 1 to *n*-1 dimensional shapes in *n*-dimensional space. Figure 9-(a) has a result which drew for the firewall policy where destination port and destination IP address are of local network of class C using firewalk method. This consists of sent packet which combines all IP address and well-known port. Therefore, this is same as the actual firewall policy. In this example, the policy was applied to blocked policy only at specific destination port.

Figure 9-(b) is results which drew firewall policy for C class. In this example, this policy is applied to blocked policy at specific destination port and specific destination IP address. Figure 9-(c) and Figure 9-(d) is results which are drew for firewall policy for firewall of actual network. The Figure 9-(c) and Figure 9-(d) is a combination of Figure 9-(a) and Figure 9-(b). We investigated the firewall policy through all sent probing packets in a local network of C class on twenty. As shown in Figure 9, the firewall policies were presented to four types.



(a) A policy blocked only at specific destination port

(b) A policy blocked at specific destination IP and destination port

(c) A policy combined figure 1-(a) and figure 1-(b) from network A

(d) A policy combined figure 1-(a) and figure 1-(b) from network B

Figure 9. Example of the existing firewall policy analysis

4.3.2 Inferring Method

To infer firewall policy is a suitable method for searching for shapes of *n*-1 dimensions in a *n*-dimensional space through the analysis of these firewall policies. We describe a method for effectively searching for 1-dimensional straight

lines in a 2-dimensional plane. We search for straight lines that are not diagonal lines, but are vertical or horizontal. The method for this search uses a sweep line algorithm. The idea behind the algorithms is to imagine that a line (often a vertical line) is swept or moved across the plane, stopping at some points in Euclidean space. The algorithm infers firewall policy at specific points. Once the line search is completed results are calculated. A typical example of sweep algorithm application is in 2-dimensional space, search for the crossing points where lines exist. We apply the sweep algorithm to search start and end points of crossing lines. The sweep line is inclined an angle of 45 degree because lines to be searched are horizontal or vertical lines. Also this algorithm can search both horizontal and vertical lines simultaneously. Search intervals of the sweep line are modified by changing the crossing points. If the lines before and after sweeping are the same line, then search interval is increased. If it is not the same line, then the search interval is decreased.

The sweep line algorithm executes as follows:

1) Generate packets covering the sweep line which has an angle of 45 degree between the start and end points

2) Randomly probe for the generated packets

3) Searching the cross point for correspondent permission rules

4) Move the sweep line with search interval and generate packets covering the sweep line

5) Compare line before and after sweeping

6) Repeat until the total space has been searched

Figure 10 shows an example of the sweep line algorithm on a 2-dimensional

space. The sweep line is indicated by a gray dotted line, and the sweep lines of previous steps are black line. According to step 2 to step 4 in Figure 10, the search interval is doubled because the sweep line and the cross line are same. If the sweep line and cross line are differ, the interval would be half. We search for the start and end points of the cross line and apply the searched cross line to the firewall rules. The sweep line is implemented by sending probing packets. The sweep line algorithm has the advantage of discovering points on both horizontal and vertical lines easily. Also, it overcomes the problem where probing packets are classified as port scan packets by the firewall.

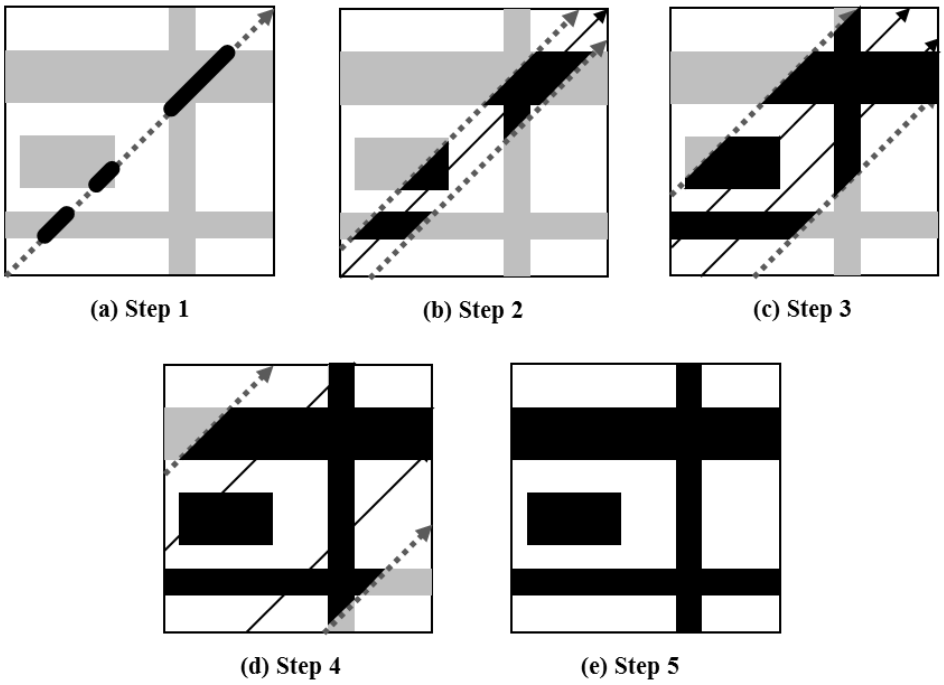


Figure 10. A simple example of sweep line algorithm

If n defines the length of the x-axis and m defines the length of the y-axis in the proposed method, the time complexity is O(*n***m*) in the worst case. The average time complexity of Region Growing method [66] in previous work was

O(k*log n*log m), when the number of search shapes was defined by k. We propose a method that is not impacted by the number of search shape. The Region Growing method has been reduced time by exponentially increasing the search interval. If the search shape is close to the line, time complexity is O(k*n*m). Therefore, our proposed method is better than previous work in the worst case.

5 IP Prefix Hijacking Detection

Our motivation is based on the fact that, existing BGP security solutions are not capable of dealing with known IP prefix hijacking cases. Filtering is used mostly to defend against invalid updates announced by ‘stub ASes’, but it cannot cover ‘transit ASes.’ It is impractical to deploy cryptographic techniques for all the routing devices on the Internet. Also, traditional probing methods cannot detect IP prefix hijacking effectively due to enhanced Internet security. So a novel approach is required that can cover known IP prefix hijacking threats.

In this section, an attempt is made to propose a novel IP prefix hijacking detection method that can be applied in the current Internet environment. Two important approaches are as follows:

1) Develop a better probing method by responding more to the probing packets in hosts for reachability monitoring.

2) Develop a practical IP prefix hijacking detection algorithm based on BGP threat analysis.

To detect IP prefix hijacking, we need a network reachability test for all ASes using active probing. Network reachability means determining whether data can be delivered along a configured path by routing. The current routing table does not guarantee reachability of all destination ASes, so we need to confirm network reachability by real-time testing to detect IP prefix hijacking.

5.1 A Probing Method using Idle Scan

Hu and Mao [52] propose a probing technique called reflect-scan for fingerprinting the victim network. This method is derived from the TCP idle scan

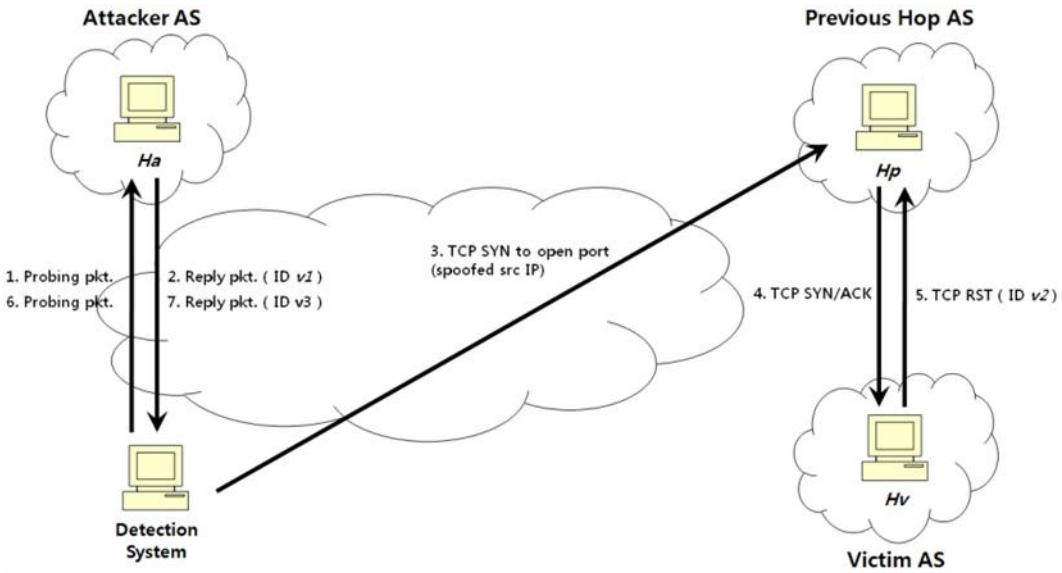
technique described in [67]. The reflect-scan focuses on detecting sub-prefix hijacking cases, but it is applied to regular prefix hijacking in our approach.

The idle scan technique is used for completely blind port scanning where attackers scan a target by sending a packet to a dummy host instead of the target. We utilize this technique to reach the victim AS because we cannot directly arrive at the victim AS when an IP prefix hijacking occurs as in section 3. The key idea is to use the sequential IP ID increment property in an IP packet and allow the unpolluted AS to forward the traffic to the victim AS.

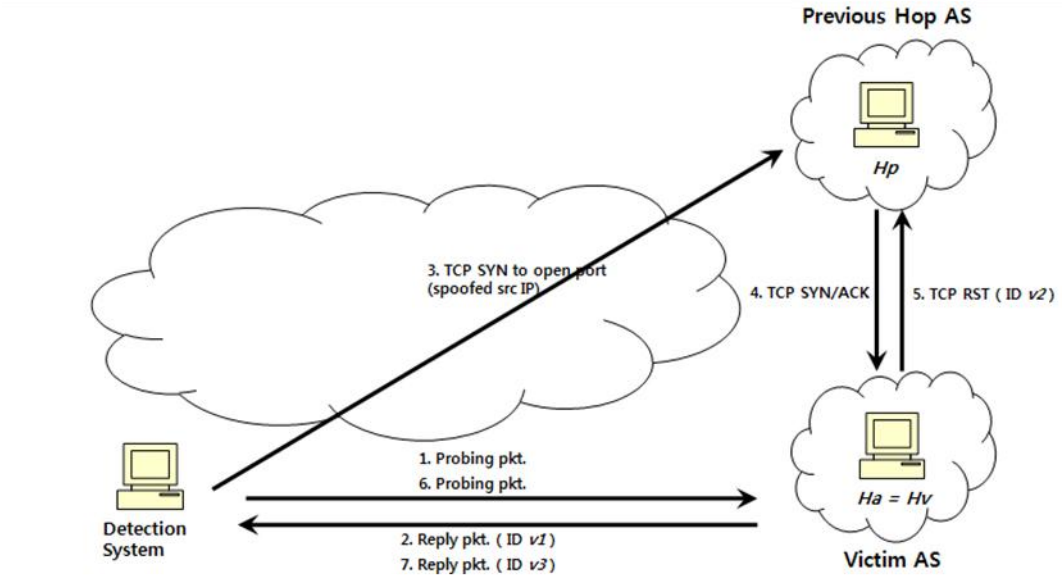
5.1.1 IP Prefix Hijacking Detection Method using Idle Scan

When IP prefix hijacking occurs, the BGP speakers subsequently propagate its false announcement according to selecting its shortest path as the best path. As its false announcement is propagated, the Internet becomes polluted. However, not all of the ASes in the Internet are polluted. With the given shortest AS path preference in routing, networks close to the attacker AS are polluted by the malicious announcement.

There are unpolluted ASes between the detection system and the victim AS when IP prefix hijacking occurs. To identify the unpolluted ASes, we use the idle scan technique. The unpolluted ASes are a crucial key in our proposed method; the detection system belonging to polluted ASes cannot directly reach the victim AS, but the unpolluted ASes still can.



(a) IP prefix hijacking case



(b) Legitimate case

Figure 11. Idle scan for IP prefix hijacking detection

The proposed technique using idle scan is demonstrated in Figure 11. First, the detection system selects a host (*Ha*) in the suspicious IP prefix, which satisfies the property to assign IP ID packets incrementally. Also, Ha should be idle because other traffic except for IP ID probing packets can interfere with the scan logic. And, the detection system should select a host (*Hp*) in the previous hop AS which is the previous AS in the path of the AS_PATH to the target IP prefix. For example, if the AS_PATH is ‘a b c d’, the previous hop AS is ‘c’ to the target AS ‘d’. This host in the previous AS should be alive and in service with an open TCP port. A web server that always opens the HTTP port is a good candidate for *Hp*.

After selecting the hosts, the detection system starts IP ID probing. In figure 3(a), the system sends a probing packet to *Ha* and records an IP ID value in the reply packet. If a spoofed TCP SYN packet in which the source IP is same with *Ha*’s IP address is sent to *Hp*, then *Hp* would response with a TCP SYN/ACK packet to *Ha*’s IP address. When IP prefix hijacking occurs, *Ha* and *Hv* should be different. (*Hv* is a host in the attacker AS, with *Ha*’s IP address) An *Hv* that receives an unsolicited SYN/ACK packet will respond with a TCP RST. Therefore, one more probing of *Ha* can verify whether the received BGP update is an IP prefix hijacking event, because the proposed method uses the sequential IP ID increment property in an IP packet and the IP ID difference between step 2 and step 7 is only one (that is, v3 = v1 + 1). In the case of legitimate updates, that is, in Figure 11.(b), *Ha* and *Hv* are the same host, and the IP ID difference is likely to be two or more (that is, v3 = v2 + 1 = v1 + 2).

The probing packets used in step 1 and 6 do not need to be only TCP SYN/ACK packets like in the TCP idle scan technique. The proposed method requires the target hosts to have predictable IP ID numbers for outgoing IP packets. To satisfy this requirement, we can select the protocol of probing packets that are expected to reply with incremental IP ID generation.

The target hosts should be idle to reduce the false detection rate. To increase the detection accuracy, we can try to send multiple probing packets at steps 1, 3, and 6. If the target is not as busy as a well-known web server, we can confirm the occurrence of an IP prefix hijacking by sending the large number of probing packets.

5.1.2 Classification of Polluted and Unpolluted ASes

We try to classify polluted and unpolluted ASes from the routing table information. For example, a polluted AS has two AS paths to reach the victim prefix P, namely {AS_1, AS_2, … AS_m, AS_{m+1}, … AS_{n-1}, AS_n} and {AS_1, AS_2, … AS_m, $AS_{m+1`}$, … $AS_{n-1`}$, $AS_{n`}$}. In this case, {$AS_1$, $AS_2$, … $AS_m$} is a *common path* in two AS paths, and {$AS_{m+1}$, … $AS_{n-1}$} and {$AS_{m+1`}$, … $AS_{n-1`}$} are *different paths*. Practically, there may be no common ASes in two AS paths. As mentioned above, not all of the ASes in the Internet will be polluted. Therefore we will classify ASes in *common path* and *different paths* into polluted and unpolluted ASes. Consequently, the detection system identifies the occurrence of IP prefix hijacking and decides to accept or reject the related update message.

Figure 12 shows the overall procedure to detect IP prefix hijacking events. The detection system connected to the BGP router in observer AS is monitoring BGP update messages and its routing table. When a suspicious BGP update causing a MOAS conflict is received, the detection system starts the proposed detection process.

First, we recognize the victim AS (*vAS*), *common path* and *different paths* (both are represented as *sfASes*). Then we have to find the target host (*vH*) in the victim AS, which satisfies the property to assign IP ID packets incrementally and be idle. The ‘Find Active Host (*vAS*)’ and ‘Active Probe (*vH*)’ processes select a target host vH and send the TCP SYN/ACK packet to it until the proper target host is found. The target host vH must reply with a TCP RST packet that has an

arbitrary IP ID number, not zero. This IP ID number should be recorded and used in the next steps.

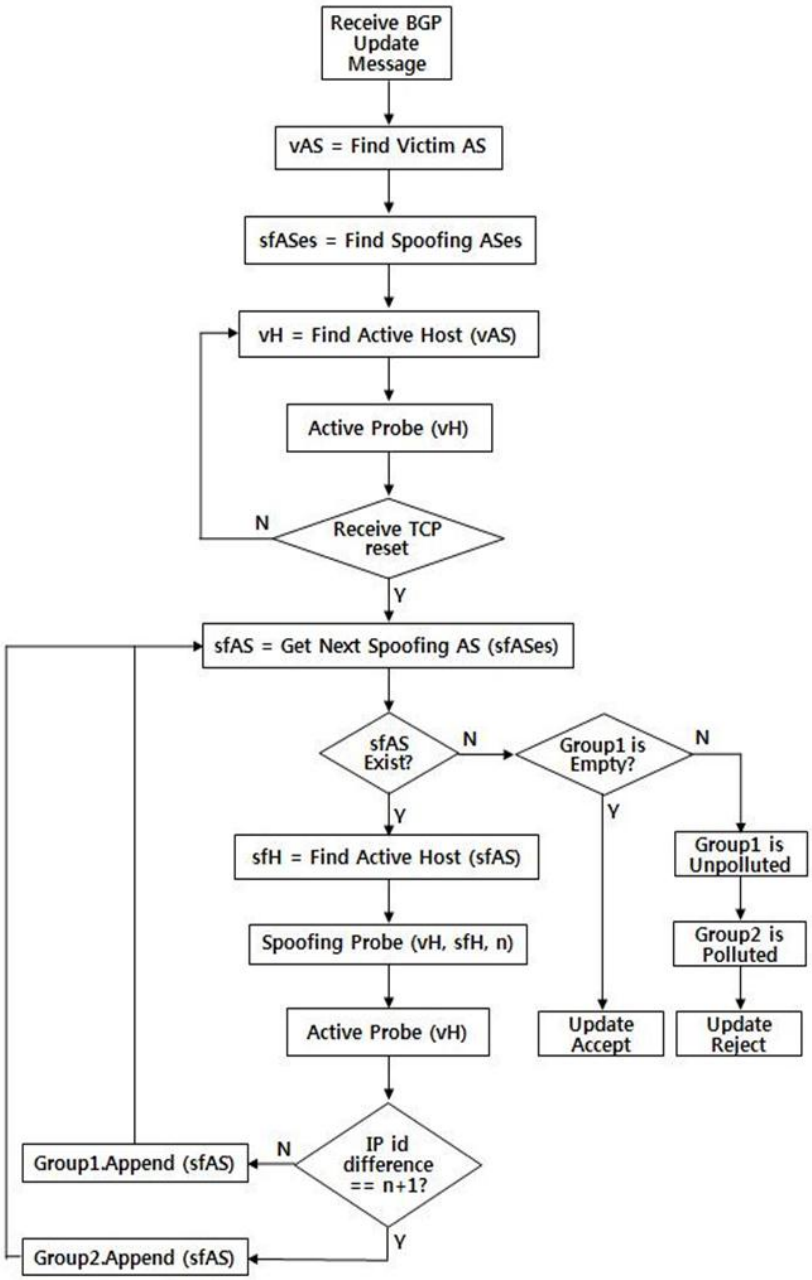


Figure 12. Classification of polluted and unpolluted ASes

The *sfASes* contains the ASes in *common path* and *different paths*. We iterate over all of the ASes in *sfASes* and find the spoofing target host (*sfH*) for each AS (*sfAS*) using the 'Find Active Host (*vAS*)' process. The *sfH* should be alive and in service with an open TCP port. The '*Spoofing Probe* (*vH, sfH, n*)' sends spoofed TCP SYN packets, in which the source IP address is the same as the *vH*'s IP address, to *sfH*. (We suppose n = 1 now.) Next, one more probe to *vH* with the TCP SYN/ACK packet is executed during the '*Active Probe* (*vH*)' process. The IP ID number in the reply packet is recorded and compared with the previously recorded IP ID. If the IP ID difference is 1, the *sfAS* in this iteration belongs to *Group1*. On the other hand, *sfAS* belongs to *Group2* if the IP ID difference is 2. The iteration continues until all of the ASes in the *sfASes* are probed.

Finally, we should examine which ASes belong to *Group1* and *Group2*. We already recorded the IP ID number of reply packet during the '*Active Probe* (*vH*) process. If an IP prefix hijacking event occurs, networks in the Internet divide into polluted and unpolluted ASes. When the *sfAS* is one of the polluted ASes, the *sfH* during '*Spoofing Probe* (*vH, sfH, n*)' process responds TCP SYN/ACK to *vH* in the attacker AS and *vH* would send TCP RST packet with an incremented IP ID number. In the next process ('*Active Probe* (*vH*)'), the *vH* resends a reply packet with an incremented IP ID number. Therefore, the difference between the current and previously recorded IP ID numbers is 2 and this *sfAS* belongs to Group2. On the other hand, when the *sfAS* is one of the unpolluted ASes, *vH*es between '*Active Probe* (*vH*)' and '*Spoofing Probe* (*vH, sfH, n*)' are different hosts. In this case, the IP ID difference between two reply packets is 1 and the *sfAS* belongs to *Group1* because the '*Spoofing Probe* (*vH, sfH, n*)' process does not induce the reply packet of *vH* in the attacker AS. We conclude the ASes in *Group1* are unpolluted and those in *Group2* are polluted.

If IP prefix hijacking does not occur and the update message is legitimate, *vH*es between '*Active Probe* (*vH*)' and '*Spoofing Probe* (*vH, sfH, n*)' are always

the same hosts in each probe. In this case, the IP ID difference between the two reply packets is 2 and all of the *sfAS* belong to *Group2*. Therefore, we conclude that IP prefix hijacking does not occur when *Group1* is empty and the detection system applies the update message in the routing table.

The *vH* should be idle to reduce the false detection rate. To increase the detection accuracy, we can try to send multiple probing packets during the '*Spoofing Probe* (*vH, sfH, n*)' process. If the process sends *n* packets and *sfAS* belongs to *Group2*, the IP ID difference would then be *n*+1.

5.2 Network Reachability based IP Prefix Hijacking Detection (RBHD)

When IP prefix hijacking occurs, the BGP speakers subsequently propagate a false announcement by selecting the shortest path as the best path. As the false announcement is propagated, the Internet becomes polluted. Polluted ASes cannot reach the victim network due to corrupted routing tables. However, not all suspicious updates are IP prefix hijacking events. So we should distinguish legitimate cases from IP prefix hijacking.

In this thesis, we propose a network reachability based IP prefix hijacking detection method, RBHD (Reachability Based IP prefix Hijacking Detection), using host fingerprinting and network fingerprinting as described in section IV. Before performing fingerprinting and comparison analysis, we obtain preliminary data which is used to detect MOAS occurrence and AS path falsification. MOAS conflict means that multiple origin ASes announce the same IP prefix. Originally, a unique AS number was allocated to each AS for use in BGP routing [37]. However, the use of static routing and private AS numbers, for example, can cause MOAS conflicts on the Internet [52]. When only looking at BGP update messages we cannot find a difference between legitimate MOAS conflicts, and those caused

by IP prefix hijacking. The detection system needs to analyze the routing table in the BGP router to detect these anomalous events. Also, host and network fingerprints used in the detection system should be collected about each AS on the Internet. If the BGP router receives an update message, then the detection system is able to detect an IP prefix hijacking incident by applying RBHD.

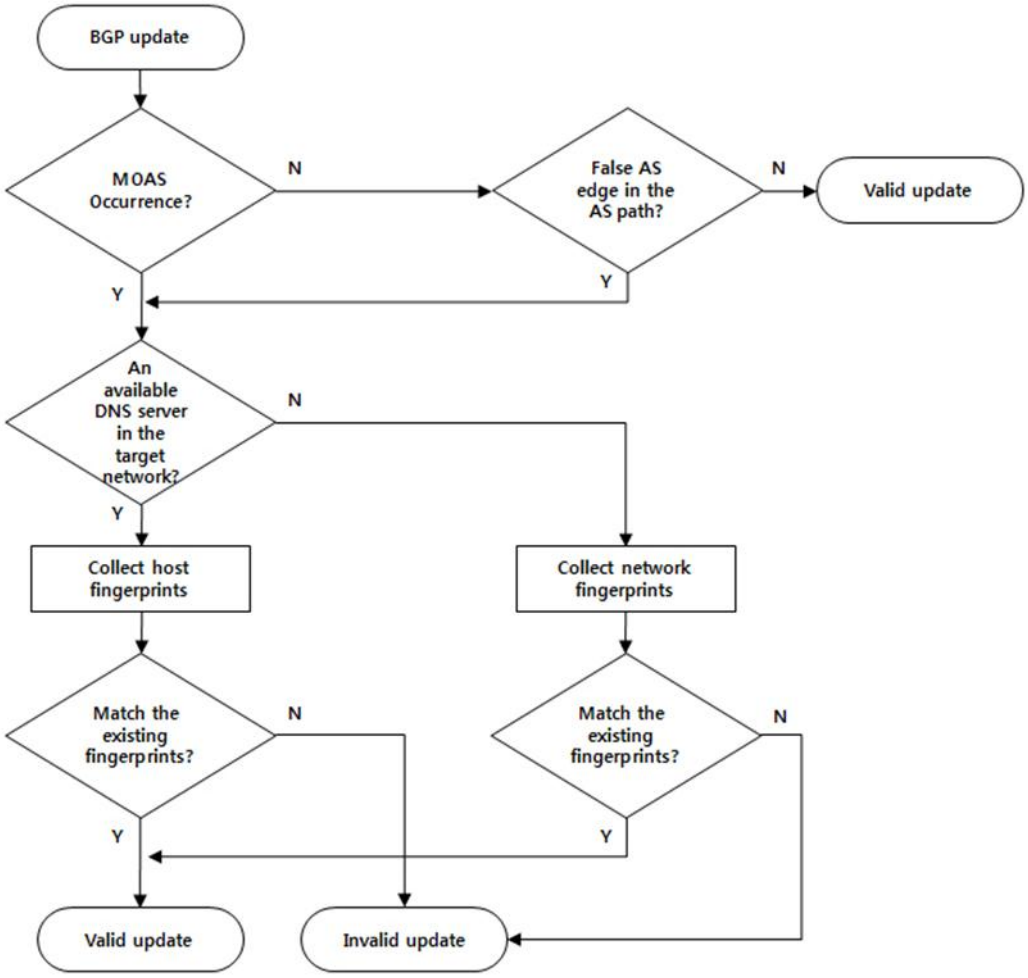


Figure 13. Overall detection algorithm

Figure 13 presents the detection process of RBHD. Suspicious BGP update

messages are divided into two cases according to the attack type. One is prefix hijacking where an attacker makes an attack by falsifying the NLRI field of a BGP update message, so MOAS (including subMOAS) occurs in BGP routing. The other is AS path falsification attack which occurs when an attacker modifies a path attribute of the update message, so an unseen AS edge that does not exist in the routing table is 'discovered'. When a suspicious event such as a MOAS or false AS edge is found, RBHD checks the possibility of host fingerprinting on the target network. In other words, among the preliminarily collected data RBHD searches available DNS servers which are used for a reachability test of the target network. If there is a DNS server inside the target network, RBHD performs host fingerprinting to it and analyzes the probing responses. Network fingerprinting is used when we cannot perform host fingerprinting on the DNS server.

Figure 14 shows an overview of the detection system in deployment. The detection system is connected with the BGP router in an observer AS in order to monitor the routing table and BGP update messages. A BGP update message consists of withdrawn routes, reachability information in the NLRI field and the AS_PATH attribute. The NLRI field indicates the IP address space of the destination AS, and the AS_PATH attribute has the AS level path to reach the announced address space.

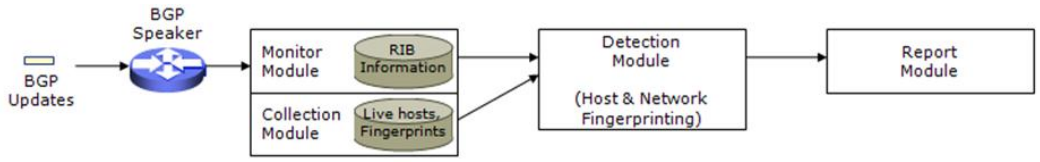


Figure 14. IP prefix hijacking detection system architecture

By comparing update messages and the routing table, we can observe suspicious events, especially a MOAS conflict. When a suspicious BGP update such as a MOAS conflict is received, the system starts the detection process to

distinguish IP prefix hijacking from a legitimate routing update. As the detection system performs active probing by itself and the router uses existing BGP, it does not need to modify the router's software or routing protocol. Therefore, the proposed method is a deployable approach that uses existing Internet infrastructure.

6 Evaluation

In this chapter, we describe the evaluation of the proposed IP prefix hijacking detection methods.

6.1 Validation of Host Fingerprinting

We collected the fingerprint information of 102,843 DNS servers and analyzed them. First, we collected the type and version of DNS software and the uptime of DNS server. According to our collected fingerprints, there are 65 kinds of DNS software in the Internet and Table 4shows the top DNS software type and version that over 1% of all the DNS servers are using. In Table 4, 'TIMEOUT' means that we received no response to queries.

Table 4. The distribution of the type and version of DNS servers in the Internet

The type and version of DNS software	The number of DNS servers	Percentage
ISC BIND 9.2.3rc1 — 9.4.0a0	45,971	44.70
ISC BIND 9.2.3rc1 — 9.4.0a0 [recursion enabled]	16,662	16.20
TIMEOUT	15,079	14.66
No match found	14,152	13.76
DJ Bernstein TinyDNS 1.05	2,235	2.17
Microsoft Windows DNS 2000	1,334	1.30
ISC BIND 9.2.0rc7 – 9.2.2-P3 [recursion enabled]	1,161	1.13
ISC BIND 8.3.0-RC1 – 8.4.4 [recursion enabled]	973	0.95

‘No match found’ means that we received responses to queries, but we could not identify the DNS software. Table 4 shows that most of the DNS servers use ‘ISC BIND,’ as shown in Table 4.

When collecting the fingerprints of a DNS server, we need only a few DNS queries which are less than or equal to five, so it does not take much time to collect the fingerprints of all servers in the DNS server list. Figure 15 shows the examples of host fingerprints that we collected for DNS servers.

```
xxx.16.210.13#1#3#3#0#ISC BIND 9.2.3rc1 -- 9.4.0a0#open#35.178
xxx.68.52.12#1#2#2#1#ISC BIND 9.2.3rc1 -- 9.4.0a0#open#
xxx.0.23.165#0#0#1#1#ISC BIND 9.2.3rc1 -- 9.4.0a0#open#
xxx.0.23.165#1#2#1#1#ISC BIND 9.2.3rc1 -- 9.4.0a0#open#
xxx.0.23.166#0#0#1#1#ISC BIND 9.2.3rc1 -- 9.4.0a0#open#
xxx.105.130.11#0#14#22#1#ISC BIND 9.2.3rc1 -- 9.4.0a0#open#
xxx.106.1.2#1#2#2#1#ISC BIND 9.2.3rc1 -- 9.4.0a0#open#35.015
xxx.106.1.3#1#2#2#1#ISC BIND 9.2.3rc1 -- 9.4.0a0#open#31.401
xxx.106.160.15#null#null#null#null#TIMEOUT#filtered#
xxx.106.192.3#1#2#2#1#ISC BIND 9.2.3rc1 -- 9.4.0a0#filtered#
xxx.106.204.19#1#2#3#1#TIMEOUT#closed#
xxx.106.204.19#1#2#4#1#TIMEOUT#closed#
xxx.106.204.3#1#2#2#1#ISC BIND 9.2.3rc1 -- 9.4.0a0#open#
xxx.106.216.1#1#1#1#1#No match found#open#
xxx.106.96.12#0#4#3#1#ISC BIND 9.2.3rc1 -- 9.4.0a0 [recursion enabled]#open#
xxx.107.160.11#1#4#2#1#ISC BIND 9.2.3rc1 -- 9.4.0a0 [recursion enabled]#open#1.099
xxx.107.26.1#1#3#4#1#ISC BIND 9.2.3rc1 -- 9.4.0a0 [recursion enabled]#open#35.506
xxx.107.28.1#1#2#2#1#ISC BIND 9.2.3rc1 -- 9.4.0a0#filtered#
xxx.108.108.226#1#3#5#1#ISC BIND 9.2.3rc1 -- 9.4.0a0#open#6.988
xxx.108.208.130#1#2#2#1#ISC BIND 9.2.3rc1 -- 9.4.0a0#open#144.448
xxx.109.239.202#null#null#null#null#ISC BIND 9.2.3rc1 -- 9.4.0a0#open#12.775
xxx.110.128.35#1#2#1#1#ISC BIND 9.2.3rc1 -- 9.4.0a0#open#
xxx.110.160.170#1#2#4#1#ISC BIND 9.2.3rc1 -- 9.4.0a0#open#26.907
xxx.111.32.3#1#0#1#1#ISC BIND 9.2.3rc1 -- 9.4.0a0 [recursion enabled]#open#
xxx.120.140.171#null#null#null#null#TIMEOUT#closed#
xxx.120.143.110#1#3#3#1#ISC BIND 9.2.3rc1 -- 9.4.0a0#open#
xxx.121.255.3#1#2#2#1#ISC BIND 9.2.0rc7 -- 9.2.2-P3 [recursion enabled]#open#383.391
xxx.121.255.4#1#2#2#1#ISC BIND 9.2.0rc7 -- 9.2.2-P3 [recursion enabled]#open#21.261
```

Figure 15. The examples of collected host fingerprints

The x-axis in Figure 16 is the number of groups which have a distinguishable fingerprint. We analyzed the collected fingerprints and the total number of distinguishable groups was calculated as 73,781. The y-axis represents the number of DNS servers that belong to the group.

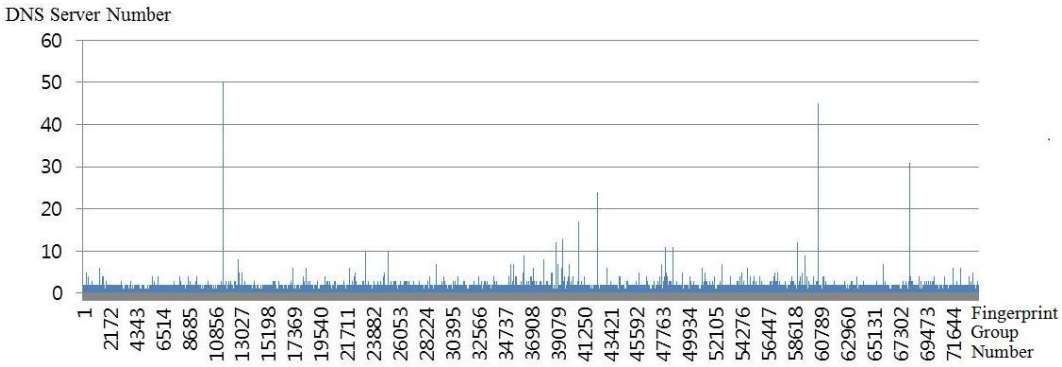


Figure 16. The number of distinguishable groups in the DNS server fingerprints

We discuss the possibility that two host fingerprints are identical when two DNS servers are randomly selected. We define that N is the total number of collected DNS servers and G is the total number of mutually exclusive fingerprints. For each group, n_i is defined as the number of DNS servers that belong to i-th fingerprint group N_i. Then the collision possibility P_C can be calculated as:

$$P_C = \sum_{i=1}^{G} \left(\frac{n_i}{N_i} \times \frac{n_{i-1}}{N_{i-1}} \right) \quad (1)$$

According to the fingerprint information that we collected, N is 77,530 and G is 73,781. The collision possibility P_C in our experiment is 2.6921 x 10^{-6}. This value means that the possibility that two fingerprints are identical is very low. Therefore, we conclude that the sufficient level of distinction can be applied in our proposed host fingerprinting method.

6.2 Validation of Network Fingerprinting

To infer policies of the Internet firewall, policies should be applied on actual network presented in the previous chapter. To infer policies of the Internet firewall for configuration of network in Figure 17. Shows the Firewall infer system to detection packet of send and receive by used Internet connection, where security system such as firewall or IDS does not exists. Tool packet of detection used for sending and receiving is *Hping*. *Hping* is used as a tool for the detection packet for sending and receiving, also offer variety of options and provide information to enough for analyzed packet of receiving.

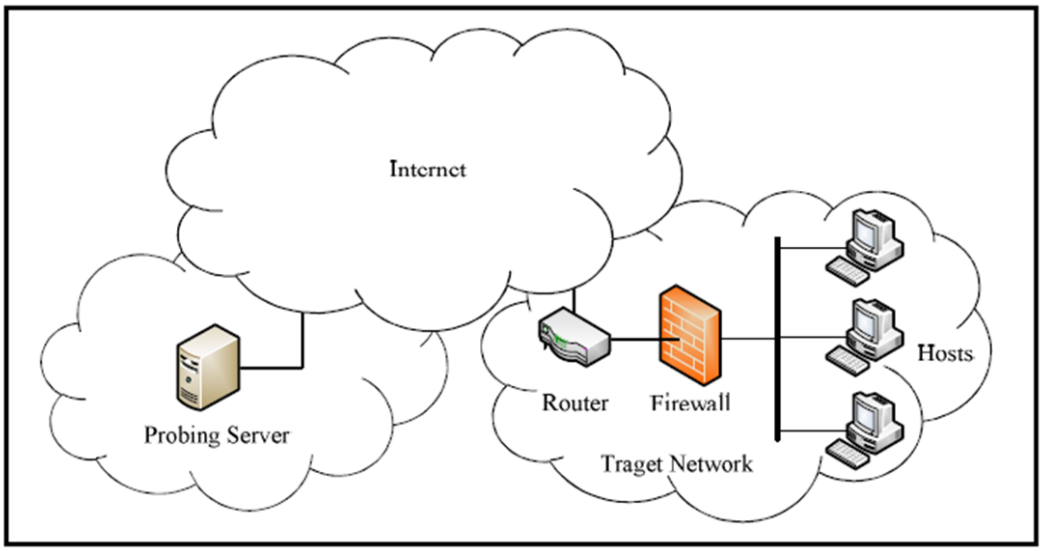
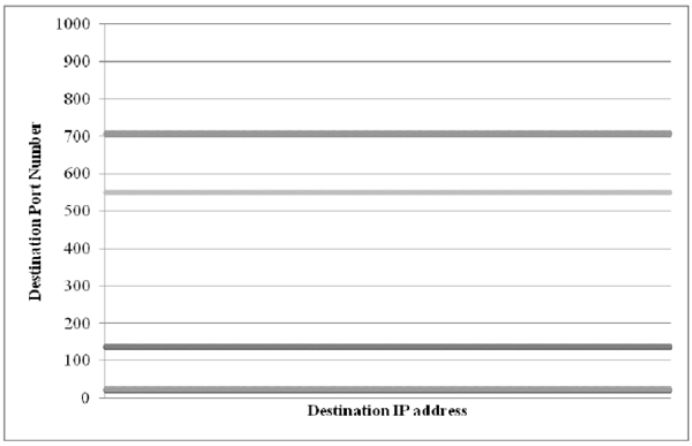


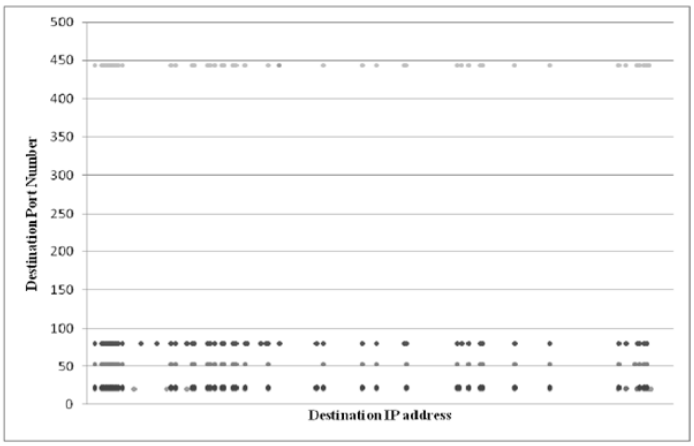
Figure 17. Configuration of network for inferring a firewall policy

To infer firewall policy, probing packets are generated with TTL values set to the length of the path to firewall plus one, and are sent to the destination address. If probing packets are not filtered by the firewall, we receive the ICMP Time Exceed message. If probing packets are filtered, we do not receive any response packets. For validating our proposed fingerprinting method, we sent probing

packets to more than 100 networks which were randomly chosen among all ASes in the BGP-RIB. The chosen networks are of diverse sizes.



(a) Distribution of packets denied from network A of C class



(b) Distribution of packets permitted from network B of C class

Figure 18. Distribution of response packet for inferring a firewall policy

Among these networks, we select two C class networks and analyze them. The distributions of packets permitted or denied are shown in Figure 18. Network A is a university network consisting of a variety of servers, personal computers, printers and network devices. Network B is a data center consisting of server

farms with a high level of security. From a fingerprinting point of view, two networks have quite different characteristics according to their default policy (accept or deny) and have various specific rules. Also, we observed that once a firewall policy to a specific network is inferred it hardly changes over time.

Table 5. The Internet firewall policy of network A

Rule	Protocol	Src IP	Src Port	Dst IP	Dst Port	Action
R1	ICMP	Any	-	192.168.10.*	-	Permit
R2	TCP	Any	Any	192.168.10.*	22:23	Deny
R3	TCP	Any	Any	192.168.10.*	135	Deny
R4	TCP	Any	Any	192.168.10.*	137	Deny
R5	TCP	Any	Any	192.168.10.*	139	Deny
R6	TCP	Any	Any	192.168.10.*	550	Deny
R7	TCP	Any	Any	192.168.10.*	707	Deny
R8	TCP	Any	Any	192.168.10.*	1:103	Permit
Default	Any	Any	Any	Any	Any	Deny

Table 6. The Internet firewall policy of network B

Rule	Protocol	Src IP	Src Port	Dst IP	Dst Port	Action
R1	ICMP	Any	-	192.168.10.*	-	Deny
R2	TCP	Any	Any	192.168.10.*	21:23	Permit
R3	TCP	Any	Any	192.168.10.*	53	Permit
R4	TCP	Any	Any	192.168.10.*	80	Permit
R5	TCP	Any	Any	192.168.10.*	443	Permit
Default	Any	Any	Any	Any	Any	Deny

Table 5 shows a firewall policy result. The proposed method was compared with FireCracker. When detection was made by FireCracker, run-time was found to be 25 minutes and the total number of sent packets was 3,056. For verifying the accuracy of inference policy in all cases, sent packets must completely draw the firewall policy. At this point of time, firewall classified as port scan to detect the packets which are been blocked or not. Therefore, proper time-interval to detect the sent packets the accuracy of the inferred firewall policy can be calculated by number of packets in the denominator and matching the two policy number of packets in the numerator. In our work, the accuracy of the proposed detection method 98%, and the accuracy of FireCracker is 98%. These results are primarily firewall policy of the network A because to block is based on port, difference between accuracy and number of sent packet are interpreted.

In the network B, a strong firewall policy is been applied, control of the TTL

value used to infer policy of the Internet firewall is impossible. Therefore, to detect host firewall and it should be to infer Internet firewall. Figure 18-(b) shows method proposed in this work to detect of packet for response result. In these cases Firecracker could not detect a firewall policy and cannot be compared. Figure 18-(b) draw firewall policy is shown Table 6. In this case, sent packet draw every field of the complete firewall policy and result of accuracy of derive firewall policy is 99%.

6.3 Experimental Setup and Results

In this section, we validate our proposed system in real environment. However, it is difficult to witness any hijacking events over a short time and the generation of a real hijacking attack can cause a serious problem in the Internet. To overcome this challenge, we set up a controlled hijacking testbed which simulates a hijacking attack. We deployed RBHD in this testbed and observed its live action during the hijacking event.

Figure 19 shows this testbed setup which allows us to simulate a hijacking event. Our prefix hijacking testbed selects two target networks for reachability test and RBHD connected to an upstream provider ISP. Because RBHD belongs to the peer-centric detection, we need an AS to monitor the routing updates, which is AS M in Figure 19. The target networks for reachability monitoring are AS A and B in which we anonymize IP addresses. There is a NAT machine in front of RBHD in order to emulate a hijacking scenario. We implemented additional functions at the NAT machine, which announce a routing update to RBHD and translate the IP addresses of packets communicating with the victim network into the attacker network’s addresses for reflecting reachability change about the update.

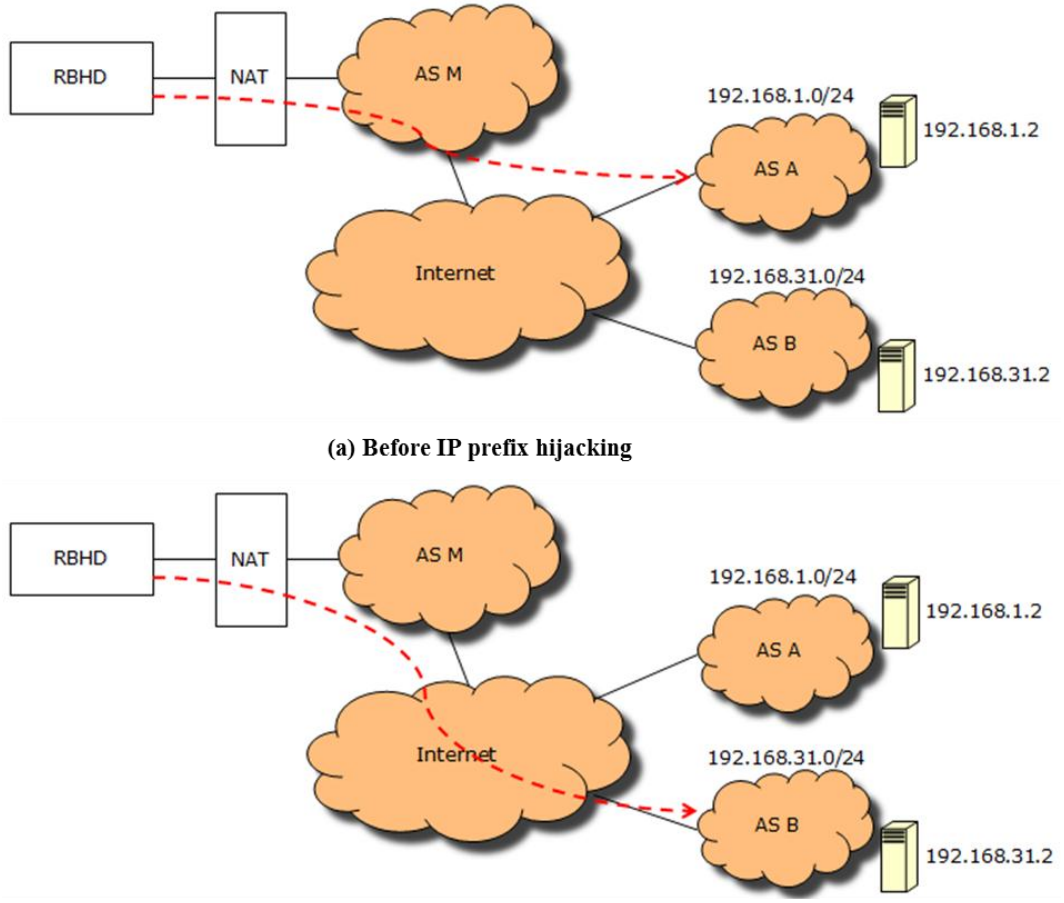


Figure 19. Setup of hijacking testbed

For a hijacking event, we picked one network as the victim and another as the attacker. Initially, RBHD monitors the AS A which has one prefix (192.168.1.0/24) and a DNS server (192.168.1.2). To simulate a hijacking event, NAT sent a false update message to RBHD as AS B announces a prefix 192.168.1.0/24 owned by AS A. Then RBHD started the detection process that probed the target network and compared current fingerprints with the preliminarily collected fingerprints. At this time, NAT translated the IP spaces of 192.168.1.0/24 into 192.168.31.0/24 in

order to change the destination network from the victim to the attacker as the hijacking event occurs.

The fingerprint of the DNS server in AS A (192.168.1.2) which RBHD preliminarily collected is described at (1) in Figure 20. After the hijacking event is simulated, RBHD performed host fingerprinting process in order to collect the current fingerprint of DNS server in the target network. In this process, NAT changed the destination IP address of probing packets from 192.168.1.2 to 192.168.31.2. The current fingerprint after the hijacking is described at (2) in Figure 20. Because the difference between two fingerprints (HF_192.168.1.2 and HF_target) is clearly evident, RBHD concludes that the false update message received from NAT is a hijacking attempt.

(1) HF_192.168.1.2 = { 1#2#2#1#ISC BIND 9.2.3rc1 -- 9.4.0a0#open#15.269 }

(2) HF_target = { 1#2#2#1#ISC BIND 9.2.3rc1 -- 9.4.0a0 [recursion enabled]#open#309.366 }

Figure 20. Preliminarily collected and current fingerprints about the target network

7 Conclusion

This section summarizes the overall contents of the thesis and lists a set of contributions. Suggested areas for future work are also discussed.

7.1 Summary

The Internet is a decentralized network comprised of many interconnected networks. Each network communicates reachability information using BGP (Border Gateway Protocol). BGP is the de-facto inter-domain routing protocol that maintains a table of Internet Protocol networks or prefixes, and designates network reachability among the various Autonomous Systems that make up the Internet. The routers maintain and update their own routing table according to routing information exchanged via BGP.

The Internet was designed to provide communication on the basis of trust between networks, but has proved to be a misguided assumption, due to the various types of attacks that have taken advantage of this trust. ASes that exchange BGP information directly with each other are assumed to be trusted, so BGP does not implement any security checks, such as checking the authenticity of origin information and path attributes, to protect against receiving invalid routing information from other routers. As such, the Internet routing infrastructure is vulnerable to attack. IP prefix hijacking is a BGP security attack, in which a BGP router, either with malicious purposes or simple due to misconfiguration, announces an IP prefix that the router does not own. This false announcement creates reachability problems and communication failures throughout the Internet.

This thesis proposed a new approach that practically and effectively detects IP prefix hijacking based on network reachability. Network reachability differs

from IP reachability in the strict sense that it defines whether the actual target is reachable, and IP prefix hijacking is an attack that corrupts network reachability due to changes of the IP path to the target network. We use a fingerprinting scheme in order to determine the network reachability of a specific network. The two types of fingerprints used for the network reachability test are host fingerprints and network fingerprints. We also proposed host and network fingerprinting methods for IP prefix hijacking detection. Our approach can conclusively detect an IP prefix hijacking occurrence through a fingerprint comparison. In particular, the proposed method does not depend on monitoring infrastructure and the work to collect fingerprints is a simple, effective and makes use of popular tools. We validated the effectiveness of the proposed method and are ready to apply it on real networks.

The followings are summary of this thesis.

- This thesis stated Routing Information Base (RIB) statistics and suspicious features.
- This thesis proposed AS-level host and network fingerprints collection methods using reachability monitoring.
- This thesis proposed an IP prefix hijacking detection method.
- This thesis performed deployment experiences for validation of the proposed methods.
- This thesis provided a guideline for handling update messages for routing stability.

7.2 Contributions

The followings are the expected findings of the thesis.

- The inability of existing BGP security to provide solutions for known IP prefix hijacking cases is described. The absence of a practical IP prefix hijacking detection method is also highlighted.
- We present how to apply network reachability monitoring techniques in the current Internet environment.
- A detailed design of the proposed IP prefix hijacking detection method is described.
- Through validation of the proposed method, we present the novelty and smartness of our approach from a practical point of view.

The followings are key contributions of the thesis.

- The problems of existing IP prefix hijacking detection techniques are addressed. The absence of detection techniques which deal with known IP prefix hijacking cases leads to the development of new methodologies which are suitable for the current Internet.
- Our approach provides accurate and practical probing techniques for the reachability test of all ASes. We present a method that collects real live hosts in the Internet and their fingerprint information.
- Novel and real-time IP prefix hijacking detection methods are described and validated with the real network data.

7.3 Future Work

Collecting DNS server information will be under future enhancement. That will be studied as another fingerprinting method; along with it new zone information collection method will also be studied. New zone information collection method improves the speed and accuracy of the information gathering than the proposed method. Finally, it is also necessary how to solve malicious

attack on the Internet using proposed fingerprint method.

If firewall regard probing as malicious and drop the packet, it should be impossible to make consecutive probing. That is a really important but practically insoluble problem. Therefore, to avoid the problem, we sent only 2 probing packets in one transmission interval and confirmed the response. Then, we performed the next behavior after enough transmission intervals. The following research will reflect the problem, and distinction between inbound and outbound in the firewall policy also will be a future work. In addition, it should be expanded to protocol, source Port, destination IP and destination Port. In this paper, though we only infer the stateless based policy, the state-based policy also ought to be inferred in the future.

Analyzing the performance and feasibility of our approach by collecting host and network fingerprinting on the Internet is planned for the future. We also plan to implement a hijacking detection system based on our proposed scheme and apply it to a real research network.

References

[1] Y. Rekhter, T. Li and S. Hares, “A Border Gateway Protocol 4 (BGP-4),” RFC 4271, January 2006.

[2] J. Hawkinson and T. Bates, “Guidelines for Creation, Selection, and Registration of an Autonomous System (AS),” RFC 1930, March 1996.

[3] BGP Routing Table Analysis Reports, http://bgp.potaroo.net/.

[4] L. Chapin and C. Owens, “Interconnection and Peering among Internet Service Providers,” An Interisle White Paper.

[5] Internet Engineering Task Force, http://www.ietf.org/.

[6] Y. Rekhter and T. Li, “An Architecture for IP Address Allocation with CIDR,” RFC 1518, September 1993.

[7] V. Fuller, T. Li, J. Yu and K. Varadhan, “Classless Inter-Domain Routing (CIDR): an Address Assignment and Aggregation Strategy,” RFC 1519, September 1993.

[8] Border Gateway Protocol, http://en.wikipedia.org/wiki/BGP/.

[9] Border Gateway Protocol, http://docwiki.cisco.com/wiki/Border_Gateway_Protocol/.

[10] BGP Best Path Selection Algorithm, http://www.cisco.com/en/US/tech/tk365/technologies_tech_note09186a0080094431.shtml/.

[11] K. Butler, T. Farley, P. McDaniel and J. Rexford, “A Survey of BGP Security,” Technical Report TD-5UGJ33, AT&T Labs-Research, Florham Park, NJ, April 2005.

[12] O. Nordstrom and C. Dovrolis, “Beware of BGP Attacks,” ACM SIGCOMM Computer Communications Review, Volume 34, Issue 2, April 2004, pp. 1-8.

[13] M. O. Nicholes and B. Mukherjee "A Survey of Security Techniques for the Border Gateway Protocol (BGP)", IEEE Communications Surveys & Tutorials, Volume 11, Issue 1, First Quater 2009, pp. 52-65.

[14] K. Butler, T. Farley, P. McDaniel and J. Rexford, "A Survey of BGP Security Issues and Solutions", in the Proceedings of the IEEE, Volume 98, Issue 1, Janurary 2010, pp. 100-122.

[15] T. Wan, P.C. van Oorschot and E. Kranakis, "A Selective Introduction to Border Gateway Protocol (BGP) Security Issues", In Proceedings of the NATO Advanced Studies Institute on Network Security and Intrusion Detection, Nork, Yerevan, Armenia. IOS Press.

[16] E. Kranakis, P. C. van Oorschot and T. Wan, "Security Issues in the Border Gateway Protocol (BGP)," Technical Report TR-05-07, School of Computer Science, Carleton University, Ottawa, Canada, August 1, 2005.

[17] R. Mahajan, D. Wetherall, and T. Anderson, "Understanding BGP misconfiguration," ACM SIGCOMM Conference, Pittsburgh, PA, USA, August 19-23, 2002, pp. 3-16.

[18] O. Bonaventure, "Interdomain Routing with BGP: Issues and Challenges," IEEE Symposium on Communications and Vehicular Technology (SCVT) 2002, Louvain-la-Neuve, Belgium, October 2002.

[19] R. Rivest, "The MD5 Message-Digest Algorithm," RFC 1321, April 1992.

[20] H. Krawczyk, M. Bellare and R. Canetti, "HMAC: Keyed-Hashing for Message Authentication, RFC 2104, April 1997.

[21] A. Heffernan, "Protection of BGP Sessions via the TCP MD5 Signature Option," RFC 2385, August 1998.

[22] S. M. Bellovin, "Security Problems in the TCP/IP Protocol Suite," ACM SIGCOMM Computer Communication Review, Volume 19, Issue 2, April 1, 1989, pp. 32–48.

[23] B. Green, "BGP Security Update: Is the Sky Falling?" NANOG 25, June 2002.

[24] V. Gill, J. Heasley and D. Meyer, "The Generalized TTL Security Mechanism (GTSM)," RFC 3682, February 2004.

[25] T. Bates, E. Gerich, L. Joncheray, J.-M. Jouanigot, D. Karrenberg, M. Terpstra and J. Yu, "Representation of IP Routing Policies in a Routing Registry," RFC 1786, March 1995.

[26] L. Blunk, J. Damas, F. Parent and A. Robachevsky, "Routing Policy Specification Language Next Generation (RPSLng)," RFC 4012, March 2005.

[27] N. Spring, R. Mahajan and D. Wetherall, "Measuring ISP topologies with Rocketfuel," IEEE/ACM Transaction Networking, Volume 12, Issue 1, February 2004, pp. 2–16.

[28] L. Gao, "On inferring autonomous system relationships in the Internet," IEEE/ACM Transaction Networking, Volume 9, Issue 6, December 2001, pp. 733–745.

[29] L. Subramanian, S. Agarwal, J. Rexford and R. Katz, "Characterizing the Internet hierarchy from multiple vantage points," IEEE INFOCOM 2002, New York, Junuary 2002.

[30] X. Dimitropoulos, D. Krioukov, M. Fomenkov, B. Huffaker, Y. Hyun, K. C. Claffy and G. Riley, "AS relationships: Inference and validation," ACM SIGCOMM Computer Communication Review, Volume 37, Number 1, January 2007, pp. 29–40.

[31] T. Griffin, Personal Communication, Jun. 2003.

[32] T. Bates, P. Smith and G. Huston, "CIDR Report for 30 January 08," March 2008.

[33] J. Stewart, T. Bates, R. Chandra and E. Chen, "Using a Dedicated AS for Sites Homed to a Single Provider," RFC 2270, January 1998.

[34] D. Chang, R. Govindan and J. Heidemann, "An Empirical Study of Router Response to Large BGP Routing Table Load," ACM SIGCOMM Internet Measurement Workshop (IMW), November 2002.

[35] NANOG Mailing Lists, http://www.nanog.org/mailinglist/.

[36] 7007 Explanation and Apology, http://www.merit.edu/mail.archives/nanog/1997-04/msg00444.html/.

[37] X. Zhao, D. Pei, L. Wang, D. Massey, A. Mankin, S. F. Wu, and L. Zhang, "An Analysis of BGP Multiple Origin AS (MOAS) Conflicts," In Proceedings of the 1st ACM SIGCOMM workshop on Internet Measurement, San Francisco, USA, November 2001, pp. 31-35.

[38] Internet-Wide Catastrophe-Last Year, http://www.renesys.com/blog/2005/12/internetwide_nearcatastrophela.shtml/ .

[39] Con-Ed Steals the Net, http://www.renesys.com/blog/2006/01/coned-steals-the-net.shtml/.

[40] YouTube Hijacking : A RIPE NCC RIS Case Study, http://www.ripe.net/internet-coordination/news/industry-developments/youtube-hijacking-a-ripe-ncc-ris-case-study/.

[41] A Chinese ISP Momentarily Hijacks the Internet, http://www.networkworld.com/news/2010/040810-a-chinese-isp-momentarily-hijacks.html/.

[42] A Brief History of Notable Internet Disruptions, http://packetlife.net/blog/2011/mar/17/brief-history-notable-internet-disruptions/.

[43] A. Barbir, S. Murphy, and Y. Yang, "Generic Threats to Routing Protocols," IETF Draft: draft-ietf-rpsec-routing-threats-07, October 2006.

[44] B. Christian and T. Tauber, "BGP Security Requirements," IETF Draft: draft-ietf-rpsec-bgpsecrec-04, March 2006.

[45] C. Lynn, J. Mikkelson, and K. Seo, "Secure BGP (S-BGP)," IETF Draft: draft-clynn-s-bgp-protocol-01.txt, June 2003.

[46] B. Weis, "Secure Origin BGP (soBGP) Certificates," IETF Draft: draft-weis-sobgp-certificates-02.txt., July, 2004.

[47] Oregon Route Views Project, http://www.routeviews.org/.

[48] RIPE RIS, http://www.ripe.net/ris/.

[49] C. Kruegel, D. Mutz, W. Robertson, and F. Valeur, "Topology-based Detection of Anomalous BGP Messages," In Proceedings of the 6th

Symposium on Recent Advances in Intrusion Detection (RAID), LNCS 2820, Pittsburgh, PA, USA, September 2003, pp. 17-35.

[50] M. Lad, D. Massey and D. Pei, "PHAS: A Prefix Hijacking Alert System," In Proceedings of the 15th USENIX Security Symposium, Vancouver, B.C., Canada, August 2006, pp. 153-166.

[51] C. Zheng, L. Ji, D. Pei, J. Wang, and P. Francis, "A Light-Weight Distributed Scheme for Detecting IP Prefix Hijacks in Real-time," ACM SIGCOMM Computer Communication Review, Vol37, Issue4, October 2007.

[52] X. Hu and Z. M. Mao, "Accurate Real-time Identification of IP Prefix Hijacking," In Proceedings of the IEEE Security and Privacy, Oakland, California, USA, May 2007, pp. 3-17.

[53] J. Karlin, S. Forrest and J. Rexford, "Pretty Good BGP: Improving BGP by Cautiously Adopting Routes," In Proceedings of the 14th IEEE International Conference on Network Protocols, Santa Barbara, California, USA, November 2006, pp. 290-299.

[54] Z. Zhang, Y. Zhang, Y. C. Hu, Z. M. Mao, and R. Bush, "iSPY: Detecting IP Prefix Hijacking on My Own," In Proceedings of the ACM SIGCOMM 2008 conference on Data Communication, Seattle, USA, 2008, pp. 327-338.

[55] M. Tahara, N. Tateishi, T. Oimatsu, S. Majima, "A Method to Detect Prefix Hijacking by Using Ping Tests," In Proceedings of the 11th Asia-Pacific Symposium on Network Operations and Management (APNOMS 2008), LNCS 5297, Beijing, China, October 2008, pp. 390-398.

[56] James W. Mickens, John R. Douceur, William J. Bolosky and Brain D. Noble. "StrobeLight: Lightweight Availability Mapping and Anomaly Detection."USENIX'09 Proceedings of the 2009 conference on USENIX Annual technical conference, CA, USA, 2009.

[57] G. Goodell, W. Aiello, T. Griffin, J. Ioannidis, P. McDaniel and A. Rubin, "Working around BGP: An Incremental Approach to Improving Security and Accuracy of Interdomain Routing," in Proc. ISOC NDSS'03, San Diego, CA, Feb. 2003, pp. 75-85.

[58] L. Subramanian, V. Roth, I. Stoica, S. Shenker and R. H. Katz, "Listen and Whisper: Security Mechanisms for BGP", In Proc. Symposium on

Networked Systems Design and Implementation (NSDI'04), San Francisco, CA, March 2004.

[59] Korea Telecom (KT), http://www.kt.com/.

[60] Level 3 Communications, http://www.level3.com/.

[61] fpdns, http://code.google.com/p/fpdns/.

[62] D. Eastlake, "Domain Name System Security Extensions," RFC 2535, March 1999.

[63] R. Arends and R. Austein, "DNS Security Introduction and Requirements," RFC 4033, March 2005.

[64] F. Veysset, O. Courtay and O. Heen. "New Tool and Technique for Remote Operating System Fingerprinting," Intranode Software Technologies, April 2002.

[65] T. Samak, A. El-Atawy, E. Al-Shaer and H. Li, "Firewall Policy Reconstruction by Active Probing: An Attacker's View." In The Second Workshop on Secure Network Protocols (NPSec 2006), 2006.

[66] T. Samak, A. El-Atawy, and E. Al-Shaer, "FireCracker: A Framework for Inferring Firewall Policy using Smart Probing," 2007 IEEE International Conference on Network Protocols, 2007, pp. 294-303.

[67] Insecure.Org, "TCP Idle Scan (-sI)," http://nmap.org/book/idlescan.html.

Biography

Birthday:	Aug. 30, 1980
Address:	RIST Building 4, Rm. 4405
	POSTECH, Pohang, Korea, 790-784
E-mail:	pluto80@postech.ac.kr
Tel:	+82-54-279-5654
Cell:	+82-10-5348-6588
Fax:	+82-54-279-5699

Education

September 2003 – February 2012:

Ph.D. in Computer Science and Engineering

Pohang University of Science and Technology (POSTECH), Korea

March 1999 – August 2003 :

B.S. in Computer Science and Engineering

Pohang University of Science and Technology (POSTECH), Korea

Publications: International Journal Papers

1. Seongcheol Hong, Hongtaek Ju, and James Won-Ki Hong, "Network Reachability-based IP Prefix Hijacking Detection", International Journal of Network Management (IJNM) **(SCIE)**. (Accepted to appear)

Publications: International Conference/Workshop Papers

2. Seong-Cheol Hong, Hong-Taek Ju, and James Won-Ki Hong, "IP Prefix Hijacking Detection Using the Collection of AS Characteristics," 13th Asia-Pacific Network Operations and Management Symposium (APNOMS 2011), Taipei, Taiwan, Sep. 21-23, 2011.

3. Seong-Cheol Hong, Jin Kim, Byungchul Park, Young J. Won, and James W. Hong, “Traffic Growth Analysis over Three Years in Enterprise Networks,” 15th Asia-Pacific Conference on Communications, Shanghai, China, Oct. 8-10, 2009.

4. Seong-Cheol Hong, Hong-Taek Ju, and James W. Hong, “IP Prefix Hijacking Detection Using Idle Scan,” 12th Asia-Pacific Network Operations and Management Symposium, Jeju, Korea, Sept. 23-25, 2009.

5. Young J. Won, B.C. Park, S.C. Hong, K.B. Jung, H.T. Ju, and James W. Hong, “Measurement Analysis of Mobile Data Networks,” Passive and Active Measurement Conference (PAM 2007), Louvain-la-neuve, Belgium, Apr. 5-6, 2007, pp. 223-227.

6. Long-Quan Zhao, Seong-Chul Hong, Hong-Taek Ju, and James Won-Ki Hong, “A Real-Time Network Traffic Based Worm Detection System for Enterprise Networks,” Proc. of APNOMS 2005 Conference, Okinawa, Japan, Sep. 27-30, 2005, pp.446-457.

7. Seung-Hwa Chung, Young J. Won, Deepali Agrawal, Seong-Cheol Hong, and James Won-Ki Hong, “Detection and Analysis of Packet Loss on Underutilized Enterprise Network Links,” 2005 E2EMON, Nice, France, May 15, 2005, pp. 164-176.

8. Seong-Cheol Hong, Long-Quan Zhao, Hong-Tack Ju, and James Won-Ki Hong, “A Worm Traffic Detection Algorithm for Enterprise Networks,” 9th IFIP/IEEE International Symposium on Integrated Network Management (IM 2005), Nice, France, May 15, 2005.

9. Myung-Sup Kim, Hun-Jeong Kang, Seong-Cheol Hong, Seung-Hwa Chung, and James W. Hong, “A Flow-based Method for Abnormal Network Traffic Detection,” Proc. of the IEEE/IFIP Network Operations and Management Symposium (NOMS 2004), Seoul, Korea, Apr. 19-23, 2004, pp. 599-612.

10. Hun-Jeong Kang, Seung-Hwa Chung, Seong-Cheol Hong, Myung-Sup Kim, and James W. Hong, “Towards Flow-based Abnormal Network Traffic Detection,” Proc. of 2003 Asia-Pacific Network Operations and Management Symposium (APNOMS 2003), Fukuoka, Japan, Oct. 1-3, 2003, pp. 369-380.

Publications: Domestic Journal Papers

11. 홍성철, 조룡권, 주홍택, 홍원기, "엔터프라이즈 네트워크에서의 인터넷 웜 탐지를 위한 방법", KNOM Review, Vol. 7, No. 2, Dec. 2004, pp. 11-20.

Publications: Domestic Conference Papers

12. 홍성철, 주홍택, 홍원기, "AS별 특징 수집을 통한 IP Hijacking 탐지 방법", KNOM Conference 2011, Pohang, Korea, Apr. 21-22, 2011.
13. 홍성철, 김태영, 권동우, 김현우, 주홍택, 홍원기 " 네트워크 도달성 모니터링을 위한 DNS 서버 수집 방법 ", 한국통신학회 하계학술대회, 제주, 2010년 6월 23-25, 2010.
14. 김진, 박병철, 홍성철, 홍원기, "엔터프라이즈 네트워크의 트래픽 경향 변화에 대한 분석", 한국통신학회 하계종합학술발표회, Jeju, Korea, Jun. 22-24, 2009.
15. 홍성철, 서신석, 홍원기, "IP Hijacking 유형 분석 및 방지 방안 연구", 한국통신학회 추계종합학술발표회, Seoul, Korea, Nov. 15, 2008.
16. 서신석, 홍성철, 홍원기, "BGP 보안 위협 요소와 대처 방안", 한국통신학회 추계종합학술발표회, Seoul, Korea, Nov. 15, 2008.