An Analysis on Selective Dropping Attack in BGP

Ke Zhang
Department of Computer Science
University of California, Davis
Email: kezhang@ucdavis.edu

Xiaoliang Zhao
USC/ISI
Email: xzhao@isi.edu

S.Felix Wu
Department of Computer Science
University of California, Davis
Email: sfwu@ucdavis.edu

Abstract

Previous studies show that current inter-domain routing protocol, Border Gateway Protocol (BGP), is vulnerable to various attacks. Previously, the major concern about BGP security is that malicious BGP routers can arbitrarily falsify BGP routing messages and spread out incorrect routing information. However, one type of attack, which we term as the selective dropping attack, has been largely neglected in literatures. A selective dropping attack occurs when a malicious router intentionally drops incoming and outgoing UPDATE messages, which results in data traffic being blackholed or trapped in a loop. In this paper, we conduct a thorough analysis on this type of attack and advocate that new security countermeasures should be developed to detect and prevent such attack.

I. Introduction

Border Gateway Protocol (BGP) is the de facto inter-domain routing protocol [1]. Current Internet can be viewed as a mesh of a number of Autonomous Systems(ASes) connected by inter-domain (inter-AS) links. AS is a set of routers having a single routing policy within a single administrative domain. BGP is responsible for discovery and maintenance of paths between distant ASes in the Internet. It provides reachability information to ASes and distributes external reachability internally within an AS. Due to its wide deployment and significant role in connecting various networks, BGP has become one of the most critical Internet infrastructures today.

BGP routers exchange routing information via UPDATE messages. UPDATE messages can be classified into two types: route withdrawal and route announcement. When a BGP router receives an UPDATE from its neighboring BGP router, this message will be processed, stored, and re-distributed in accordance with both BGP specification [1] and with the routing policies of the local AS.

The previous major concern about BGP security is the integrity and authenticity of BGP UPDATEs, especially route origin information and AS path information stored in AS_PATH attribute. Incorrect UPDATEs, due to either BGP router misconfiguration or malicious attack, may cause serious problems to the global Internet. For example, in April 1997, a small ISP incorrectly announced all the prefixes learned from its upstream ISP as its own prefixes. These fault routes spread to the global Internet. Many routers were affected and even crashed, and the whole Internet was unstable for hours [2].

Some countermeasures have been proposed to mitigate BGP vulnerabilities. TCP MD5 signature [3] uses shared secret key between two BGP routers to protect BGP session from spoofed BGP UPDATEs sent by outsiders. S-BGP [4] and SoBGP [5] apply cryptography to prevent an attacker (either insider or outsider) from advertising faulty BGP messages or tampering normal messages.

However, as noted by Bellovin et al [6], [7], traditional cryptography-based security mechanisms, cannot protect routing protocols against some kind of attacks. In this paper, we also describe one kind of such attacks against BGP, termed as *selective dropping attack*, which aims to disrupt the availability of the Internet. We demonstrate via formal analysis and experiments that this attack can cause data traffic blackhole and persistent traffic loop. Further, we examine current approaches proposed to secure BGP and we find that by design, most of them do not intend to prevent selective dropping attacks. Thus, further research on new solutions are needed.

Currently, BGP threat analysis and attack model have become active research areas at IETF. However, little attention has been paid to the dropping attack discussed in this paper. In this work, we will provide a thorough analysis on such kind of attack and its security implications.

The rest of paper is organized as follows. Section II presents the definition of selective dropping attack and analyzes its possible damages. Section III shows the attack scenario in experiments. Section IV discusses the current proposed security countermeasures and points out that all of them fail to prevent selective dropping attacks. Conclusions are drawn in section VI.

II. Selective Dropping Attack

A. Definition

This section defines selective dropping attack. First, we discuss two relevant properties of BGP protocol.

BGP is a policy routing protocol. According to inbound and outbound policy, BGP router may legitimately suppress some UPDATEs. To distinguish this valid dropping from malicious dropping, we model a simple BGP system, define malicious dropping and claim that malicious dropping can create traffic blackhole and persistent traffic loop.

In the simple BGP system, the network is represented as a simple undirected connected graph $G = (V, E)$, where $V = \{0, 1, \ldots, n\}$ is a set of nodes connected by edges from E. Each node represents a BGP router and each edge $(u, v) \in E$ represents a BGP session. For any node u, its *set of peers* is $peers(u) = \{w|(u, w) \in E\}$. It is reasonable to select one special node as the destination because route computations for multiple destinations are not interfered. In our study, we select node 0 as the destination node to which all other nodes attempt to establish paths.

A path in G is defined as a sequence of nodes, $(k, k-1, \ldots, 1, 0)$ or an empty path, denoted as ϵ. For all non-empty paths $p = (k, k-1, \ldots, 1, 0)$, we assume that the direction of the path is from the first node k to the last node 0 (the origin). The path P is a simple path. If P and Q are non-empty paths such that the first node in Q is the same as the last node in P , then PQ denotes the path formed by the concatenation of these paths. For example, the concatenation of the path $P = (4, 3, 2)$ and $Q = (2, 1, 0)$ is $PQ = (4, 3, 2, 1, 0)$. In addition, if P is a non-empty path, and node u is not in the path P, $u \circ P$ denotes a new path in which u is appended to the path P. For example, $P = (3, 2, 0)$ and $u = 4$, then $u \circ P = (4, 3, 2, 0)$.

$m(u, v) = path_u$ denotes the message that node u sends to v. If the $path_u \neq \epsilon$, $m(u, v)$ is route announcement, otherwise, it is route withdrawal.

In real BGP, each BGP router has three Routing Information Bases (RIBs), the Adj-RIBs-In, the Loc-RIB, and Adj-RIBs-Out. The Adj-RIBs-In store the routes learned from inbound UPDATE messages. The Loc-RIB contains the routes that the BGP router has selected from the routes contained in the Adj-RIBs-In. The Adj-RIBs-Out store the routes that the local BGP router has selected to advertise to its peers.

In our simple model, we let rib-in($u \Leftarrow w$) denote node u's most recently received message from peer w , which is stored in the Adj-RIBs-In of u. Because we only consider the path to the destination 0, rib-in($u \Leftarrow w$) stores the path that w advertises to u. Thus, for simplicity, we also use rib-in($u \Leftarrow w$) to denote the path that w sends to u. For the same reason, rib(u) denotes the best path that u adopts and stores in the local-RIB. If the first node of rib(u) is v we define *nexthop* of u is v, i.e $u.nexthop = v$. rib-out($u \Rightarrow w$) denotes the route that u advertises to w.

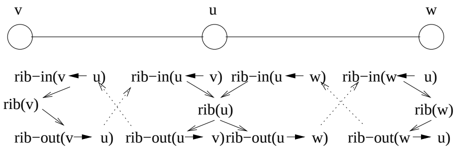

Fig. 1. Routing Information Bases Structures

Figure 1 illustrates the RIB structures for a node u. In this system, we assume that communication in an BGP session is reliable, i.e. all the messages will be delivered without being lost, reordered or tampered. Thus, at the stable state, if $u \in peers(v)$, rib-out($v \Rightarrow u$) = rib-in($u \Leftarrow v$) and rib-out($u \Rightarrow v$) = rib-in($v \Leftarrow u$).

The correct BGP operation defines the following consistency properties:

1) a) If rib(u) $\neq \epsilon$, there must $\exists v \in peers(u)$[rib-in($u \Leftarrow v$) = rib(u)].
   b) If rib(u) $= \epsilon$, rib-in($u \Leftarrow v$) can be arbitrary.
2) a) For any $w \in peers(u)$, if rib-out($u \Rightarrow w$) $\neq \epsilon$, then rib-out($u \Rightarrow w$) $= u \circ$ rib(u).
   b) It is possible that when rib(u) $\neq \epsilon$, there exists rib-out($u \Rightarrow w$) $= \epsilon$ where $w \in peers(u)$.

"Dropping" discussed in this paper indicates that a router itself drops the incoming or outgoing message. Dropping incoming message means when a router receives an incoming message form a peer, it stores the route information into the corresponding Loc-RIB, but does not apply this route information into route selection process. Dropping outgoing message means when a router selects a new route, it does not send the new route to its peers.

BGP allows a router to drop UPDATE messages according to its policy. Policy dropping is consistent with these properties. Property 1(a) implies that node u can define a policy to select a route from one peer but drop the routes it received from the others. Property 1(b) indicates a special policy, based on which although node v announces that it has a route to reach the destination, node u does not use v's route even when u has no route. Property 2(a) guarantees that no policy dropping allows node u to use one route but announce the other route to its peers. Property 2(b) indicates a policy, which authorizes node u not to transit the traffic for node v, even though node u can reach the node 0.

This paper focuses on the malicious dropping of BGP UPDATE messages.

Definition 1: Any dropping of BGP UPDATE messages which violates the property 1(a) or 2(a) is defined as a **selective dropping attack**, or **malicious dropping** [1].

We list the inconsistencies caused by four types of malicious droppings.

- **Malicious dropping outgoing route withdrawal**: For node u, when rib$(u) = \epsilon$, u drops the withdrawal message that should be sent to peer w. Thus, rib-out$(u \Rightarrow w) \neq \epsilon$. Obviously, dropping outgoing route withdrawal violates property 2(a).
- **Malicious dropping outgoing route announcement**: When node u has a new route P, i.e. rib$(u) = P$, u drops a route announcement message that should be sent to w, where, $w \in peers(u)$[rib-out$(u \Rightarrow w) \neq \epsilon$]. Thus, [rib-out$(u \Rightarrow w) \neq \epsilon \wedge$ rib-out$(u \Rightarrow w) \neq u \circ$ rib(u)]. This malicious dropping violates property 2(a).
- **Malicious dropping incoming route withdrawal**: Originally, node u sets rib$(u) =$ rib-in$(u \Leftarrow v)$. When u receives a withdrawal message from v, u drops this message. Thus, $u.nexthop = v \wedge$ rib-in$(u \Leftarrow v) = \epsilon$. It is easy to see that malicious dropping incoming withdrawal violates property 1(a).
- **Malicious dropping incoming route announcement**: Originally, node u sets rib$(u) =$ rib-in$(u \Leftarrow v)$, when u receives a new route from v, u drops this message. Thus, $u.nexthop = v \wedge$ rib-in$(u \Leftarrow v) \neq$ rib(u). Obviously, malicious dropping incoming route announcement violates property 1(a).

B. Damage Analysis

Malicious dropping can cause traffic blackhole and persistent traffic loop. We will define traffic blackhole and persistent traffic loop first.

Blackhole: There exists a node p which cannot reach the origin, however, there exists a node q whose route traverses p. p is a traffic blackhole. Formally, $\exists p \in V[\text{rib}(p) = \epsilon] \wedge \exists q \in V[p \in \text{rib}(q)]$.

Persistent Loop: $\exists p, q \in V[p \in \text{rib}(q) \wedge q \in \text{rib}(p)]$

Given the definitions of blackhole and malicious dropping, we can claim that dropping outgoing withdrawal and incoming withdrawal can form traffic blackhole.

Theorem 1: If $\exists w \in peers(u)[w.nexthop = u]$ and u maliciously drops the outgoing withdrawal message to w, u becomes a blackhole.

Proof: Suppose $w.nexthop = u$, that is $u \in$ rib(w) and rib$(w) =$ rib-in$(w \Leftarrow u)$. Since u drops the withdrawal message to w, then rib$(u) = \epsilon$ and rib-out$(u \Rightarrow w)$ does not change and still equals to rib-in$(w \Leftarrow u)$, i.e. rib(w) does not change. Thus, u is blackhole, since rib$(u) = \epsilon \wedge u \in$ rib(w). ■

Similarly, we can conclude that malicious dropping the incoming withdrawal message can form a blackhole.

Theorem 2: If a node maliciously drops outgoing route announcements and incoming route updates, it is possible for persistent traffic loop to occur in the network. We will show examples to prove this claim in the experiment section.

Theorem 3: If a node drops outgoing incoming route updates, it is possible that persistent loop would occur in the network.

Proof: Suppose u is malicious and $u.nexthop = v$, that is $v \in$ rib(u). Based on BGP policy, it is possible for v to use a new route which contains u in the middle, i.e. $u \in$ rib(v). Because, u maliciously drops the incoming update, rib(u) does not change. Thus, the loop is formed, since $v \in$ rib$(u) \wedge u \in$ rib(v). ■

From the analysis, we can learn that possible damages caused by selective dropping attacks depend on where the fault/malicious router is located. The damage is limited if the malicious router is not on the best path to victim network for the downstream ASes. That is, so long as the downstream routers never select the route previously announced by the fault/malicious router as the best path, they will never be cheated. The only trivial damage is poisoned backup routes set.

However, in order to function as a router to deliver data traffic, a router has to be selected on the best path for some prefixes. We summarize the following possible routing problems caused by malicious dropping attacks.

- As theorem 1 states, if the fault/malicious router drops the incoming withdrawal or outgoing withdrawal messages for a particular prefix, then a blackhole for that traffic may be formed. Because as long as the downstream AS is not informed that its best path is not valid any more, the downstream AS will continue to deliver data traffic along its "best" path to the malicious router. Consequently, all the packets may be dropped by the malicious router and network bandwidth has been wasted. Note that, it is possible for the downstream router to receive withdrawal messages from other routers. However, those withdrawals will not cause the downstream router to remove or replace its "best" path. An example is shown in Section III.
- If the fault/malicious router drops the outgoing route announcement, the attack may have several possible impacts. First, it may cause the sub-optimal routing. Because the routing conditions have changed, the downstream routers need to reevaluate all possible paths and select the best one for optimal routing. However, by removing such routing signals, dropping updates will cause downstream routers keep using the previous path which may not be the best path anymore. Second, as we have claimed, if the router

[1] In this paper, these two terms are used interchangeably.

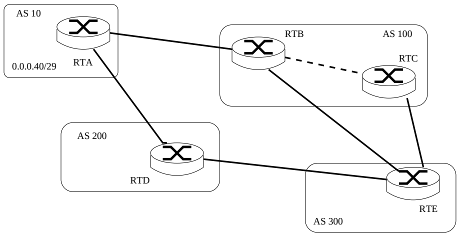

Fig. 2. experimental network for blackhole

also drops incoming routes, either announcement or withdrawal, persistent packet forwarding loop can occur. Examples are shown in Section III.

- If the fault/malicious router drops the incoming route announcement, the attack can also cause sub-optimal routing. In the worst case, it can cause persistent traffic loop as stated in theorem 3.

III. Experiments

In this section, we show some examples for dropping attack in the experimental network. The goal of these experiments is to demonstrate the correctness of our analysis.

A. Blackhole

In the experimental network shown in Figure 2, there are four ASes, five BGP routers and six BGP sessions. Solid line between routers denotes External BGP (EBGP) session and dash line denotes Internal BGP (IBGP) session.

Our target network, 0.0.0.40/29, is owned by AS10. We study the routes to this network in five BGP routers. In the initial stable state, RTE uses {AS100, AS10} as the best AS path to reach the target network. RTE uses RTB as the next hop. When we cut the link between RTA and RTB, under normal circumstances, RTE will remove {AS100, AS10} from the BGP routing table and select {AS200, AS10} as the best path. This path will be announced to AS100, which will use the path {AS300, AS200, AS10}. In the forwarding table, for the entry of network 0.0.0.40/29, RTE sets RTD as the next hop, RTB and RTC set RTE as the next hop. However, if RTB is malicious, it can hijack the normal traffic to the target network by selective dropping attack. In this experiment, we let RTB hold the withdrawal messages to RTE and only send a withdrawal message to RTC. Consequently, although RTE receives the route withdrawal from RTC, it will still use RTB as next hop to deliver traffic to the network 0.0.0.40/29. Therefore, all the traffic from AS300 will be blackholed by RTB.

B. Routing Loop

Another experimental network, shown in Figure 3, is set up to create persistent routing loop. Comparing to previous network, this network has one more AS, AS400. The target network is still 0.0.0.40/29. In Figure 3, we list the IP address of each router's interface. (Note we only list the last 8-bit of IP address, so 69 denotes 0.0.0.69) The major difference from the first experimental network is that RTB is a normal node whereas RTE is malicious. RTE selectively drops outgoing route announcements to RTF in the beginning. In the example, RTE sets {A100, AS10} as the best path, yet drops the route update to RTF. It announces {AS300, AS200, AS10} to RTF instead of announcing the current route stored in the Loc-RIB. RTC announces {AS200, AS10} to RTF. RTF sets a larger local preference value to the route learned from RTE than the route learned from RTC so that RTF uses {AS300, AS200, AS10} as the best AS path. Initially, in the stable state, every router chooses the correct next hop for the network 0.0.0.40/29. Table I shows the entry for target network in each router's local forwarding table.

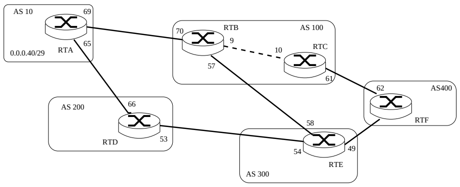

Fig. 3. experimental network for routing loop case 1

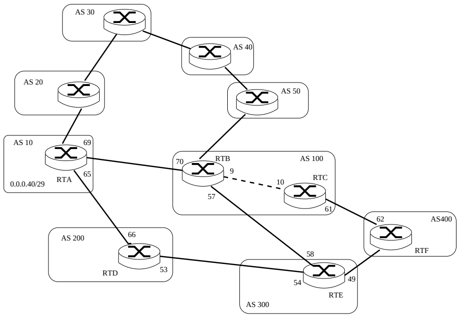

Fig. 4. experimental network for routing loop case 2

Same as the first example, we cut the link between RTA and RTB. Consequently, RTB sends route withdrawal to RTE and RTC. RTE maliciously drops this incoming message and still uses the route {AS200, AS10}. When RTC receives the withdrawal message, RTC uses the route {AS400, AS300, AS200, AS10} which was received from RTF previously, and sends this route back to RTB. Finally, RTB uses the route {AS400, AS300, AS200, AS10}. The new local forwarding table is listed in Table II.

From Table II, we can see that the loop has been formed. For the route to 0.0.0.40/29, RTB sets RTC as next hop, RTC sets RTF as next hop, RTF sets RTE as next hop, RTE sets RTB as next hop.

TABLE I

THE LOCAL FORWARDING TABLE BEFORE LINK CUT

RTB Local Forwarding Table					
Destination	NextHop	Cost	AdmDist	Src	OutgoingInterface
0.0.0.40/29	0.0.0.69/32	-1	20	EBGP	0.0.0.70/32
RTC Local Forwarding Table					
Destination	NextHop	Cost	AdmDist	Src	OutgoingInterface
0.0.0.40/29	0.0.0.9/32	-1	200	IBGP	0.0.0.10/32
RTE Local Forwarding Table					
Destination	NextHop	Cost	AdmDist	Src	OutgoingInterface
0.0.0.40/29	0.0.0.57/32	-1	20	EBGP	0.0.0.58/32
RTF Local Forwarding Table					
Destination	NextHop	Cost	AdmDist	Src	OutgoingInterface
0.0.0.40/29	0.0.0.49/32	-1	20	EBGP	0.0.0.50/32

TABLE II

THE LOCAL FORWARDING TABLE AFTER LINK CUT

RTB Local Forwarding Table					
Destination	NextHop	Cost	AdmDist	Src	OutgoingInterface
0.0.0.40/29	0.0.0.10/32	-1	200	IBGP	0.0.0.9/32
RTC Local Forwarding Table					
Destination	NextHop	Cost	AdmDist	Src	OutgoingInterface
0.0.0.40/29	0.0.0.62/32	-1	20	EBGP	0.0.0.61/32
RTE Local Forwarding Table					
Destination	NextHop	Cost	AdmDist	Src	OutgoingInterface
0.0.0.40/29	0.0.0.57/32	-1	20	EBGP	0.0.0.58/32
RTF Local Forwarding Table					
Destination	NextHop	Cost	AdmDist	Src	OutgoingInterface
0.0.0.40/29	0.0.0.49/32	-1	20	EBGP	0.0.0.50/32

In the above example, RTF drops the incoming route withdrawal from RTB. In fact, this example can be extended to the case in which RTF drops incoming route announcements. For instance, we add 4 ASes to the experimental network 2 (Fig. 4). When the link between RTA and RTB is cut, RTB should announce another AS path {AS50, AS40, AS30, AS20, AS10} to router RTC and RTE. Since RTE drops this incoming route announcement, RTE will still use the old route and set RTB as next hop. RTB, RTC, RTF and RTE will form the same loop as in the previous example.

IV. Security Challenge

In this section, we examine the effectiveness of current security approaches on selective dropping attack.

BGP, as a path vector protocol, has some inherent vulnerabilities to insider attacks, which are related to BPG trust model. BGP routers trust one another. Moreover, a BGP router has to rely on its upstream BGP routers to send it correct routing information. Current BGP protocol does not enable a downstream BGP router to verify the accuracy of received routing information. Further, there is no guarantee for an upstream BGP router to spread the necessary routing information to its downstream BGP routers.

Some countermeasures, such as SBGP, SoBGP and ASRAP [4], [5], [8] have been proposed to secure BGP. We first briefly survey these security approaches and show that by design, none of them has intended to address the selective dropping attack. Thus, such attack remains a security challenge calling for new solutions.

A. S-BGP and SoBGP

Both S-BGP and SoBGP mainly concern about the faulty routing information, especially faulty UPDATEs generated or tampered by malicious or misconfigured BGP routers. They enable BGP routers to verify the accuracy of incoming UPDATEs. Thus, an router will not accept the faulty routes.

S-BGP consists of three major components—IPsec, PKIs, and attestations. IPsec is used to protect the underlying TCP connection. Public Key infrastructures (PKIs) provide the secure identification of BGP speakers and of ASes and of address blocks [9] In such PKIs, two types of certificates are used, one for validation of entities and authorization of AS number allocation, the other for address allocation. For the integrity of BGP UPDATE messages, S-BGP introduces the concept of attestations, which are digitally signed statements used to verify the authenticity of route information. Address attestations are certificates signed by an authority that maps a prefix or prefixes to the origin AS of the address space. Route attestation is a statement signed by an AS that lists an

AS sequence that it received and the legitimate AS to which it plans to forward an UPDATE message. Since each AS makes a route attestation when it forwards the announcements to neighboring AS, the nested route attestations are created. Based on these certificates and attestations, for a given route, a BGP router is able to check the routing information from the following aspects. First, the BGP router can verify if the address space is properly allocated. Second, the BGP router can check the identification (route ID, AS number) of the peers. Third, the BGP router can check whether the origin AS is the owner of the address space or whether it is authorized to announce the route for the address space. Finally, the BGP router can verify the authenticity and integrity of the whole AS path. S-BGP proposes to modify BGP UPDATE message and store the attestations in it. Given an incoming UPDATE, BGP router accepts it only if this new UPDATE message passes all the verifications.

SoBGP [5] is as similar as S-BGP. SoBGP uses three types of certificates. Entities certificate is used so that SoBGP server can validate the entities. Policy certificate provides the policy information of an AS which originates the routes. From the policy certificate, SoBGP servers can get AS level connectivity and relationship information so that SoBGP server can construct a directed graph to represent all known valid transit paths through the Internet. Based on this directed graph, SoBGP server may be able to verify the path of each UPDATE. Authorization certificate is used to authorize ASes to advertise the blocks of address spaces. Prefixes hijack can be prevented by examining the authorization certificates. Unlike S-BGP, SoBGP requires a new BGP SECURITY message to carry these certificates. After receiving and validating these certificates, SoBGP servers or other devices put these certificates into a local database. Whenever BGP router receives UPDATEs, the new route is verified with the local certificates database. Only the UPDATE that does not violate the certificates will be accepted by a BGP router.

Both S-BGP and SoBGP focus on preventing the malicious or misconfigured routers from announcing routes for the unauthorized address space or routes with invalid AS path. Because selective dropping attack does not generate any invalid UPDATEs, SBGP and SoBGP will not detect the occurrence of such attack.

B. ASRAP

Another proposed countermeasure is Autonomous Systems Routing Authority Protocol (ASRAP) and associated architecture [8] . ASRAP allows an AS to verify the accuracy of BGP UPDATEs by cooperating with other participating ASes. Each participating AS maintains an ASRA server to answer the queries from other ASes. Users of the system query the ASRA to validate received BGP messages or to acquire additional route-relevant information. ASRAP is a receiver-driven protocol, which means that recipients of UPDATEs are responsible to make queries from the upstream ASes when new UPDATEs arrive. Similarly, ASRAP does not intend to address selective dropping attack either, because ASRAP assumes downstream AS will receive UPDATEs from a malicious BGP router.

V. Related Work

Packet dropping attack has been studied in other network protocols. X.Zhang et al [10] explored the negative impact of packet dropping attacks in TCP. In their paper, they discussed three packet dropping patterns and concluded that packet dropping can severely degrade TCP performance. They also proposed a statistics based detection mechanism to detect malicious dropping attack. K.Bradley et al [11] presented WATCHERS protocol to detect and react to routers that drop or misroute packets. The basic idea of WATCHERS is that ideally an legal router must abide by the “principle of packet flow conservation” , i.e. the number of incoming packets for a router, excluding those destined to it, should be the same as the number of outgoing packets, excluding those generated by it. Using multiple decentralized counter, WATCHERS track the traffic flow and detect the routers which violate the conservation principle. Comparing to malicious dropping of data packets, dropping routing packets may cause much more damages. Vetter [12] studied insider threats in OSPF and mentioned dropping attack in OSPF. Unlike BGP, OSPF is a link state protocol. They observed that dropping attack might not be able to cause severe problems in OSPF. However, our paper points out that dropping routing packets could be a real threat in BGP and undetectable for previous detection mechanisms.

Bellovin et al [6] studied another attack model called link-cutting. They assume that attackers have a router-level topology map and a list of already-compromised links and routers in advance. The attackers can cut/disable some key links so that the selected traffic will pass the compromised routing devices. In that paper,the attackers' objective is to divert traffic past an attacker-controlled point so that they can do anything they want on data packet streams, such as eavesdropping or connection-hijacking. Such attack, if practical, is also undetectable for all the current BGP security countermeasures. Our work share some similarity in terms of stealthiness of attacks. But we had a focus on attacking techniques, which could be used to achieve the same objective as discussed in [6].

VI. Conclusion

In this paper, we demonstrate that a particular kind of BGP routing attack, selective dropping attack, may cause severe routing problems. We define this attack and prove via formal analysis and experiments that this attack can lead to traffic blackhole and persistent routing loop. We also show that current security solutions are not sufficient to address this attack, which may call for novel approaches to detect and/or prevent such attack.

VII. Acknowledgment

We would like to thank Daniel Massey and others for their valuable comments and proofreading.

References

[1] Y. Rekhter and T. Li. Border Gateway Protocol 4. RFC 1771, SRI Network Information Center, July 1995.

[2] V. J. Bono. 7007 Explanation and Apology. http://www.merit.edu/mail.archives/nanog/1997-04/msg00444.htm.

[3] A. Heffernan. Protection of BGP Sessions via the TCP MD5 signature option. RFC 2385, SRI Network Information Center, August 1998.

[4] S. Kent, C. Lynn, and K. Seo. Secure Border Gateway Protocol (S-BGP). *IEEE JSAC Special Issue on Network Security*, 2000.

[5] J. Ng. Extensions to BGP to Support Secure Origin BGP. http://www.ietf.org/internet-drafts/draft-ng-sobgp-extensions-00.txt, October 2002.

[6] Steven M. Bellovin and Emden R. Gansner. Using Link Cuts to Attack Internet Routing. draft.

[7] Steven M. Bellovin. Routing Security. Talk at British Columiba Institute of Technology, June 2003.

[8] 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 *NDSS*, 2003.

[9] K. Seo, C. Lynn, and S. Kent. Public-key infrastructure for the secure border gateway protocolS-BGP. In *DARPA Infromation Suvivability Conference on Exposition paper*, 2001.

[10] Xiaobing Zhang, S.F. Wu, Zhi Fu, and Tsung-Li Wu. Malicious Packet Dropping : How It Might Impact the TCP Performance and How We Can Detect It. In *Proceedings of ICNP*, 2000.

[11] K. A. Bradley, S. Cheung, N. Puketza, B. Mukherjee, and R. A. Olsson. Detecting Disruptive Routers: A Distributed Network Monitoring Approach. In *Proceedings of IEEE Symposium on Security and Privacy*, 1998.

[12] B. Vetter, F. Wang, and S.F. Wu. An experimental study of insider attacks for the ospf routing protocol. In *Proceedings of the INCP 1997*, October 1997.