Performing BGP Experiments on a Semi-Realistic Internet Testbed Environment

Ke Zhang, Soon-Tee Teoh, Shih-Ming Tseng,
Rattapon Limprasittipom, Kwan-Liu Ma, S.Felix Wu
Computer Science Department
The University of California, Davis
{kezhang, steoh, smtseng, rlim, klma, sfwu}@ucdavis.edu

Chen-Nee Chuah
Electrical & Computer Engineering Department
The University of California, Davis
chuah@ucdavis.edu

Abstract

We have built a router testbed that is connected to the Deter/Emist experimental infrastructure. Our goal is to create a semi-realistic testbed to conduct BGP experiments, measure and visualize their impact on network performance and stability. Such testbed is also useful for evaluating different security countermeasures. Our testbed architecture includes four components: routing topology, background traffic, data analysis and visualization. This paper describes how we launch two specific BGP attacks, (a)Multiple Origin AS and (b)route flap damping attacks, and the lessons learned.

1 Introduction

The current Internet can be viewed as a mesh of Autonomous Systems(ASes) connected by inter-domain (inter-AS) links. An AS is a set of routers with a single routing policy, running under a single technical administration. As the de facto inter-domain routing protocol, Border Gateway Protocol (BGP) [11] 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 of connecting various networks, BGP has become one of the most critical components of the Internet infrastructure today. For the same reason, BGP security has attracted a lot of interests from researchers.

Accidents or attacks in BGP may cause world-wide connectivity loss. For example, in April 1997, a small ISP incorrectly announced all the prefixes learned from its upstream ISP as its own prefixes. As this fault information spread through the global Internet, many routers were affected and even crashed, and the whole Internet was unstable for hours [2].

To improve BGP security, some mechanisms have been proposed. S-BGP [8], SoBGP [10], Listen and Whisper [16] apply cryptography to prevent an attacker (either insider or outsider) from advertising faulty BGP messages or tampering with the normal BGP messages.

Although researchers are aware of the vulnerabilities in BGP and have proposed various security countermeasures, unfortunately, these solutions are not widely deployed yet. One major obstacle is the lack of experimental infrastructure and rigorous methodologies to evaluate these security mechanisms.

The DETER/EMIST project [1] aims to fill these gaps. As the first step, DETER/EMIST group have built a 72-node experimental network and emulated DDOS, worm and routing attacks. As the routing security subgroup, we are responsible for creating routing experiment testbed, where routing attack traffic and their effects can be studied and visualized in a contained environment. The routing testbed is a heterogeneous network with five commercial routers and dozens of a zebra routers [19]. We inject the real routing data collected from the Internet into the testbed as background traffic. Specially, we launch Multiple Origin AS (MOAS) attack and damping attack [1] in the testbed, collect

---

[1] MOAS was originally developed by Xiaoliang Zhao et al. Damping attack was initially proposed by Z.Morley Mao during an earlier EMIST-Routing teleconference among Morley Mao, Vern Paxon and Felix Wu.

the network traffic and visualize the effect caused by these attacks. This paper describes the routing testbed architecture and two attack experiments that we have conducted.

Many researchers focus on modeling BGP behaviors in the simulator or testing their proposed mechanism in simulation [6, 9]. SSFNet [14] is the most popular simulator in BGP research domain. Another simulator, BGP++ [4] is developed by Dimitropoulos et al, which basically ports GNU Zebra bgpd into ns-2 simulation environment.

Our work explores BGP emulations. We conduct our experiments on a semi-realistic emulated routing environment. We also run the same experiment in the SSFNet. We observe the different attack effects, which proves that our work is valuable.

The rest of the paper is organized as follows. Section 2 introduces the architecture of our testbed. Section 3 and Section 4 describes the MOAS and damping attacks. Section 5 summarizes the paper and the future work.

2   Routing Testbed Architectures

We build the routing testbed in the DeterLab [3], which is based on Emulab [5]. DeterLab is a software system that provides a time- and space-shared platform for experiment in distributed systems and networks. It controls a cluster of nodes and allocates resources to individual users over particular time slots. In this paper, we refer to these nodes as DeterLab nodes. Currently, DeterLab has 72 nodes which are located in ISI/USC. To build a routing testbed, we access these DeterLab nodes by specifying a virtual topology via an ns scripts. We install the GNU Zebra software [19] in the allocated nodes and run the BGP daemon — bgpd. We call these nodes zebra routers.

To generate a realistic experimental environment, we introduce five commercial routers into the testbed. These commercial routers support BGP, but they are from different vendors. This commercial router testbed is located in UC-Davis, CA. We connected this network with DeterLab nodes via an IPsec/VPN connection. Although both zebra routing software and these commercial routing software comply with the BGP RFCs, some subtle differences still exist. For example, we find that BGP route flap damping implementations in two platforms are slightly different. These differences are enough to produce different observations in our experiments.

The routing testbed should comply with the following basic requirements:

1. The experiments should be conducted in the networks with the emulated realistic topology.

2. Real world routing background traffic must be injected into the experiment networks.

This paper focuses on BGP experiments rather than developing new attacks

2.1   Testbed Topology

To emulate a realistic inter-AS topology in the experimental network is challenging, given that there are over 17000 ASes in the world. Some large ASes may have hundreds of BGP routers. However, in our experimental testbed, we only have up to 72 nodes to emulate the BGP routers. Thus, we have to first simplify the topology while preserving the following important characteristics of AS-level connectivity observed in the current Internet [15].

1. Paper [15] classifies the ASes into a 5-layer hierarchical structure. Tier-1 ASes are the major ISPs, including Sprint, AT&T, UUNet etc, which form the backbone of the Internet. Tier-2 to Tier-4 ASes are the regional ISPs or transit ASes which provide transit service for smaller or customer networks. Tier-5 ASes are stub ASes, which are usually campus networks or company networks.

2. Tier-1 ASes are fully meshed. In real life, each Tier-1 AS covers a large geographical area and owns hundreds of BGP routers. To connect with the other Tier-1 AS, one usually puts its BGP routers in the Internet exchange points, where it is connected to other routers to exchange routing information.

3. The number of multi-homing ASes is increasing. A multi-homed AS is connected to at least two provider networks.

4. The incoming and outgoing data traffic and routing information should follow the BGP policies at different ASes. There are three basic relationships: provider-customer, peer-peer, sibling-sibling. In short, provider-customer relationship means that the provider forwards all its learned routes to the customer. The customer only tells the provider the route to reach the customer's own network. Peer-peer relationship means each peer AS only exchanges the route to reach its own network. Thus, one peer does not forward the traffic for the other peer. Sibling-sibling relationship means the two ASes exchange all the routing information they learned.

Since we only have limited resources (nodes), we emulate a network with three-level hierarchical topology in the DeterLab. There are three Tier-1 ASes, four Tier-2 ASes and seven Tier-3 ASes. Each Tier-1 AS has three fully-connected Zebra routers. The three Tier-1 ASes are full meshed. In 4 Tier-2 ASes, two of them multi-home to two Tier-1 ASes, the others only connect to one Tier-1 AS. Tier-3 ASes emulate stub ASes. They are either single-homed or multi-homed to Tier-1 or Tier-2 ASes. Besides the zebra

routers in DeterLab, the commercial router network in UC-Davis is also connected to one Tier-1 AS via the IPsec/VPN.

To enforce the AS relationships, we configure the filters in every zebra router. The following is an example to explain how the filter works. A Tier-2 AS may be connected to two Tier-1 ASes. We set a filter in the zebra routers of the Tier-2 AS, such that, only routes to the Tier-2 AS network and its customers' network are announced to external peers. This filter assures that this Tier-2 AS is the customer of the two Tier-1 ASes.

2.2 Testbed Traffic Generation and Collection

To create a realistic BGP testbed, we must inject the real world BGP traffic. Real world BGP traffic are collected from several "observation points", such as RouteViews [13] and RIPE-RIS [12]. These observation points maintain multi-hop BGP sessions with many ASes, including Tier-1 ASes, such as AT&T and Sprint. They only receive the BGP updates message from the peer ASes without any route announcement.

For our experiment, we use the BGP traffic from Sprint (AS1239) collected by RouteViews. We apply the BGP replayer [2] to regenerate all the BGP traffic observed from AS1239 into DeterLab. BGP replayer basically reads the BGP data files, opens a BGP session and sends BGP updates to zebra routers in DeterLab.

In the routing testbed, we have a near-real-time data collection module. Like RouteViews [13], we set up some special zebra routers as collection points in the testbed. These observation points are responsible for dumping and recording raw BGP data every T_i seconds. We are able to choose a higher recording rate to increase the accuracy. Assuming one event E happened at time T and the recording interval is T_i seconds, We will have E recorded in the next dump file at time $T + x$ (where $0 <= x <= T_i$). Hence, in the worst case, event E can be recorded at time $T + T_i$. In the experiments, the recording interval T_i is set to be 60 seconds.

2.3 Traffic Analysis and Visualization

The recorded data are sent to a dedicated anomaly detection engine and visualization client. The anomaly detection engine apply both signature-based and statistical-based detection techniques to examine the BGP traffic. The anomalous BGP updates are marked and statistical information is attached. For details about anomaly detection engine, please refer to [20].

The examined data are sent to an interactive visualization engine to display. There are two major visualization engines. One is to display MOAS event (Fig 1). The other
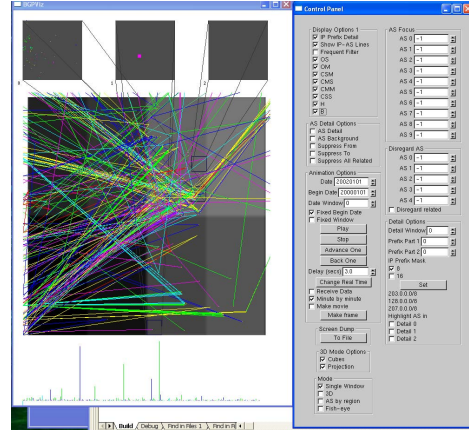

Figure 1. Visualization interface for MOAS event

is to visualize the BGP traffic for some specific prefixes. MOAS display engine focuses on origin AS changes. It helps users to detect and analyze the anomalous event in which origin AS changes. BGP dynamics engine focuses on some pre-selected prefixes. It displays the detailed BGP updates information in history and runtime. In addition, it can also display the statistical information provided by anomaly detection engine. Please refer to [18, 17] for further information about the visualization engine.

3 MOAS Attack Experiment

A BGP prefix is normally announced by a single AS [7]. We call that AS the origin AS of that prefix. However, in real BGP dumped data, we always find that some prefixes are fully or partially originated by multiple ASes. Although most of these events are due to multi-homing, it is possible that some events are caused by fault or attack. More importantly, the current BGP does not deploy any authority mechanism to prevent this origin AS conflict. Thus, a malicious BGP router can easily hijack some network by originating false BGP announcements.

There are two simple ways to launch the MOAS attack.

- An attacker originates the same prefix as the victim AS with shorter AS path. Since the shorter AS path is prefered in BGP route selection process, some ASes may choose the fake routes.

- An attacker originates the prefix that is the subnetwork of the victim AS network. Because BGP always chooses the more specific route, the traffic destined to the subnetwork will go to the attacker.

We test the two attack scenarios in the testbed. We randomly select 500 hundred prefixes, attack these prefixes by

announcing either the shorter AS path or more specific subnet prefixes. These MOAS events are immediately captured by the visualization engine. This not only confirms that MOAS attack can disrupt BGP easily but also demonstrates that visualization engine can effectively detect this type of attack.

4 Differential Damping Penalty Attack Experiment

4.1 BGP Route Flap Damping

Route Flap Damping (RFD) is a mechanism to reduce the amount of update messages in the Internet caused by instability.

Each BGP router that employs route flap damping maintains a list of penalty values, one value per peer per prefix. The penalty value for a pair of peer i and prefix j will be increased when there is an unstable event (i.e., an update) from peer i regarding to prefix j. In other words, each penalty value represents the instability of one particular route; higher penalty value indicates the higher instability.

Each router configures two thresholds locally: suppression and reuse. If the penalty value is increased to be greater than the suppression threshold, the route is suppressed. After the route is suppressed, the penalty value still increases when there is an update from the suppressed route. The penalty value also decreases with time. When the route is stable (i.e., no updates arrive), the penalty value decays exponentially with the configured half-life value. If the current penalty value is p_0, and half-life time is H, then after time t without any update comes in, the penalty decays as the following:.

$$p_t = \frac{p_0}{2^{t/H}}$$

After the route is suppressed, if the route is stable for some time and the penalty value decays to be lower than the reuse threshold, the route is reused again.

There are three main unstable events that a peer can generate: path withdrawal, attribute change, and path re-advertisement. Usually, the router increases the penalty value differently depending on the type of update. For example, withdrawal should indicate higher instability than the attribute change and thus should incur higher penalty. In practice, different routers can use different configuration of damping parameters. Table 1 illustrates the default parameters of Cisco router.

Figure 2 illustrates an example of the penalty value of a particular route in the damping process configured with the Cisco recommended parameters. When there are one withdrawal and three updates at the beginning stage, the penalty

Table 1. Damping Parameters of Cisco Router

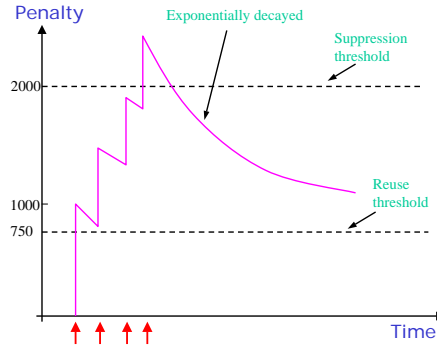| RFD parameters	Cisco
Withdrawal penalty	1000
Re-advertisement penalty	0
Attributes change penalty	500
Suppression threshold	2000
Reuse threshold	750
Half-life (min)	15

Figure 2. RFD penalty function with the Cisco default parameters

value is increased past the suppression threshold; as a result, the route is suppressed. If there is no further update, then the penalty value decays exponentially and, after some time, the penalty value decreases below than the reused threshold, thus the route can be used again.

4.2 Attack Scenario in Routing Testbed

Route flap damping (RFD) attempts to stabilize routing by suppressing the unstable routes. Taking advantage of RFD, an attacker may be able to intentionally generate a few route updates such that the original stable route between two communication ends is suppressed. In our routing testbed, we conducted the following damping attack experiment.

Assume that the topology of the network is exactly the same as that illustrated in Figure 3, and router A and D employ damping by the Cisco recommended parameter. S is the prefix originator, D is the router of the victim network, M is an attacker, and the best path from D to S is D-A-M-S. The route we discussed is the route to reach S. For simplicity, we use $P(A, M)$ to denote A's damping penalty for the route heard from M. Similarly, $P(D, A)$ denotes D's damping penalty for the route heard from A.

The attacker M can do the following steps to prevent D to reach some particular prefix originated by S.

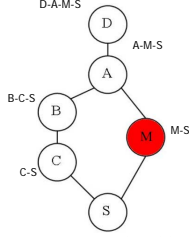1. M sends withdrawal message to A. At this time,

Figure 3. network topology in differential damping attack

$P(A, M)$ will increase by the withdrawal penalty. With Cisco recommended parameter, this will increase $P(A, M)$ by 1000.

2. M waits until the previous $P(A, M)$ decays to a small value. With the Cisco recommended parameter, the penalty value is decayed from 1000 to 15 within 90 minutes. During this time, the path D-A-B-C-S is used as a backup path.

3. M waits until S sends out the attribute change update; then M suppresses this update by not propagating it to A. In this way, this update will propagate through the path D-A-B-C-S, but not D-A-M-S. As a result, this update increases $P(D, A)$ (but does not increase the penalty $P(A, M)$) by the attribute change penalty. With the Cisco recommended parameter, this increases the penalty value by 500. At this point, the penalty value at D to peer A is greater than the penalty value at A to peer M by almost 500, that is, $P(D, A) - P(A, M) \approx 500$

4. M sends the re-announcement to A. This will not increase the penalty $P(A, M)$. However, A will send an update to D to change its best path from A-B-C-S to A-M-S, increasing the penalty $P(D, A)$ by 500 more. At this point, $P(D, A) - P(A, M) \approx 1000$.

5. By prepending its own AS number in the AS-PATH attribute, A sends the new path A-M-M-M-S to A. Because of the longer AS path, A selects the path A-B-C-S to update the old A-M-S and send it to D. M's update increases both $P(A, M)$ and $P(D, A)$ by 500. After 30 seconds, M can send the previous route M-S to A, and cause A send A-M-S to D. This increases $P(A, M)$ and $P(D, A)$ by another 500. Repeating this AS prepending and de-prepending, M causes penalty value $P(D, A) \approx 1500$ and $P(D, A) \approx 2500$. At this moment, A does not suppress the route learned from M, but D suppresses the route learned from peer A, including the path A-B-C-S. Thus, M successfully isolates D from S.

6. To maintain D's route suppression, M repeats the AS prepending and de-prepending route updates every 400 seconds. This maintains the $P(D, A)$ above the reuse threshold and $P(A, M)$ below the suppression threshold.

This attack can be separated into two phases: attack phase and maintain phase. In the attack phase (step 1-5), an attacker raises the penalty value till D suppress the A's route and creates the penalty difference for the second phase. In the maintain phase (step 6), the attacker attempts to maintain D's penalty value above reuse threshold.

The essential point in this attack is that attacker has the ability to create the penalty difference between $P(A, M)$ and $P(D, A)$. Otherwise, the attacker (M) cannot generate updates between A and D, because A will first suppress M. The penalty difference between $P(A, M)$ and $P(D, A)$ also exponentially decays. It is easy to prove.

Suppose the initial difference is $\triangle = P(D, A) - P(A, M)$, after time t,

$$P(D, A, t) = \frac{P(D, A)}{2^{t/H}}$$

$$P(A, M, t) = \frac{P(A, M)}{2^{t/H}}$$

$$\triangle(t) = P(D, A, t) - P(A, M, t) = \frac{\triangle}{2^{t/H}}$$

$$\lim_{t \to +\infty} \triangle(t) = 0$$

Thus, theoretically $P(D, A)$ will equal to $P(A, M)$ eventually. That means if the attacker maintains the $P(A, M)$ above reuse threshold, D will suppress the route forever.

4.3 Simulation and Emulation

4.3.1 Simulation in SSFNet

This damping attack is implemented in the SSFNet, the most popular BGP simulator. As we expected, attacker M successfully maintains the P(D,A) above reuse threshold. The penalty value is plotted in the Figure 4 on the left. In the simulation, attacker continuously sends updates every 500 seconds, as described in step 6. The P(D,A) are maintained between 1385 and 1885. Thus, D are isolated from S forever.

Although the SSFNET-based simulation demonstrates that this damping attack is feasible, the emulation in our testbed reveals a few differences. We will first describe how we emulate the attack in both zebra router testbed and commercial router testbed. Then, we compare three different results.
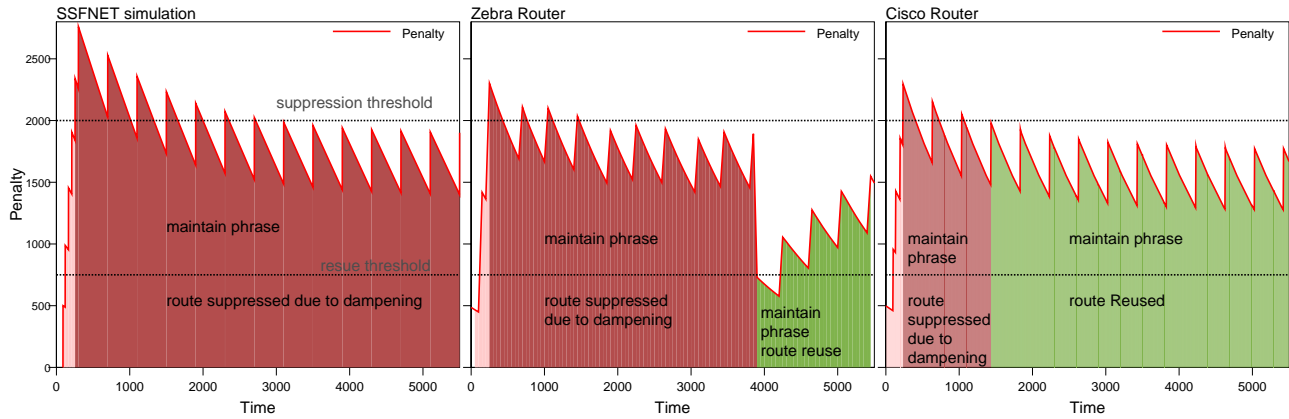
Figure 4. RFD difference in damping attack

4.3.2 Emulation in Zebra Router Testbed

To fulfill the emulation, we must find the similar topology as the above scenario. The qualified topology must meet two requirements:

1. Attacker must exists in the best route. From the scenario, it is easy to see if the attacker is not in the best route, the route updates caused by the attacker will not be propagated to the victim, since the victim's provider will always announces another route (best route) to the victim. Thus, the victim will not raise the penalty value for the provider.

2. The provider of the victim must have at least two neighbor ASes. In the attack scenario, the attacker will first withdrawal the best routes and re-announce the best route to boost up the penalty difference in victim and victim's provider, the victim's provider have to peer with two neighbors, otherwise, no penalty difference can be created.

It is not difficult to identify the qualified topology. For example, in our zebra router testbed, AS B and C are two Tier-1 ASes, AS A and S are two Tier-2 ASes, AS D and M are two Tier-3 ASes. M connects two providers AS A and S by multi-homing. D is A's customer network. We assume that no filters are deployed in AS A, AS A learns the route updates from M. Thus, AS M can be a transit AS. From D's point of view, the AS route A-M-S is the best path since it is shorter than the other path A-B-C-S.

Having identified the feasible topology, we launch the attack in the zebra router testbed. $P(D, A)$ is plotted in the middle of Figure 4. In the attack phase, following the step 1-5, the attacker raises $P(D, A)$ above 2300. In the maintain phase, attacker M sends route updates every 500 seconds. $P(D, A)$ is maintained above 1400 for one hour. However,

after one hour, $P(D, A)$ suddenly drops to 750, the previously suppressed route is reused. The zebra router fails to maintain the suppression state is because zebra router applies 60 minutes maximum suppression time. Therefore, the maintain phase can last only one hour.

4.3.3 Emulation in Commercial Router Testbed

Since we only have five commercial routers, we set up the same topology as we described. In the attack phase, $P(D, A)$ is raised as we expected. In the maintain phase, when a new update arrives, $P(A, D)$ should raise by 500. However, if the penalty for the new route is less than the suppression threshold, the route is reused (Figure 4 on the right). This is totally different from zebra router RFD implementation. In zebra router, if the route is suppressed, it can be reused only when penalty is less than reuse threshold or the total suppression time is longer than the maximum suppression time. In the Cisco router, when the previously suppressed route is replaced by a new route, the Cisco router recalculates the penalty and decides if the new route should be suppressed only by current penalty value, no matter whether the previous route is suppressed or not. If current penalty is greater than suppression threshold, the new route is suppressed. If not, the new route will be used.

From the damping attack simulation and emulation, we reveal the subtle difference between simulator and real router. This confirms that routing testbed is valuable and indispensable for the evaluation of new security mechanisms. Although some new mechanisms may work properly in theory and simulation, it may fail in real life network environments.

In addition, from the damping attack emulation, we discover the subtle differences between zebra software routers and Cisco routers. This also suggests that the heterogeneous

routing testbed is necessary. Homogeneous zebra router testbed may not reveal the properties of real life network. Thus, the routing testbed must incorporate both commercial routers and zebra routers.

5 Summary and Future Work

This paper describes the design and implementation of a BGP routing testbed. This testbed is composed of dozens of zebra routers and five commercial routers. To emulate the semi-realistic routing environment, we carefully design topology with three levels of AS hierarchy and insert real BGP trace into the testbed as background traffic. We also implement the BGP data analysis engine and visualization engine to analyze and display BGP traffic. Since this testbed aims to evaluate the BGP security mechanisms, we conduct two BGP attacks in the testbed – MOAS attack and the differential damping penalty attack. Our experiments confirm that these two attacks can effectively disrupt the BGP routing. In addition, in the damping attack experiment, we discover the subtle implementation difference between zebra router and Cisco router, which yield different attack effects. This finding suggests that the realistic routing testbed should include different route implementations.

Currently, we only have one testbed and virtually one attachment point to "the real Internet", where real BGP traffic is replayed. In other words, our BGP traffic replay is "one way only", i.e., from the real Internet into our routing testbed, but not vice versa. However, we will consider a much more powerful paradigm in future to perform BGP experiments (we call it BGP plug and play) such the dynamics being produced in the DETER testbed will be used to update/change the replay behavior/sequences. Virtually, we want to allow a two-way communication between the emulated real Internet and the experiments running on the testbed.

A very simple example is to simultaneously have two different ASes (and they are NOT connecting to each other directly) being emulated. Under the situation, a change happening in the first AS should be "observable" by another AS. Therefore, the BGP plug-and-play mechanism needs to propagate the changes consistently from one experiment to other related experiments.

References

[1] R. Bazjscy, T. Benzel, M. Bishop, B. Braden, C. Brodley, S. Fahmy, S. Floyd, W. Hardaker, G. Kesidis, K. Levitt, B. Lindell, P. Liu, D. Miller, R. Mundy, C. Neuman, R. Ostrenga, V. Paxson, P. Porras, C. Rosenberg, S. Sastry, D. Sterne, and S. Wu. Cyber Defense Technology Networking and Evaluation. In *Communication of the ACM*, March 2004.

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

[3] Network Security Testbed based on Emulab. http://www.isi.deterlab.net.

[4] X. A. Dimitropoulos and G. F. Riley. Creating realistic bgp models. In *11th IEEE/ACM International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems*, 2003.

[5] Network Emulation Testbed. http://www.emulab.net.

[6] T. Griffin and B. Premore. An Experimental Analysis of BGP Convergence Time. In *Proceedings of ICNP*, November 2001.

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

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

[9] Z. Mao, R. Govindan, G. Varghese, and R. Katz. Route Flap Damping Exacerbates Internet Routing Convergence. In *Proceedings of ACM SIGCOMM*, August 2002.

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

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

[12] The RIPE Routing Information Services. http://www.ris.ripe.net.

[13] The Route Views Project. http://www.antc.uoregon.edu/route-views/.

[14] The SSFNET Project. http://www.ssfnet.org.

[15] L. Subramanian, S. Agarwal, J. Rexford, and R. H. Katz. Characterizing the internet hierarchy from multiple vantage points. In *Proceedings of the IEEE INFOCOM*, June 2002.

[16] L. Subramanian, V. Roth, I. Stoica, S. Shenker, and R. Katz. Listen and whisper: Security mechanisms for bgp. In *First symposium on Networked Systems Design and Implementation (NSDI'04)*, March 2004.

[17] S. T. Teoh, K.-L. Ma, and S. F. Wu. Visual exploration process for the analysis of internet routing data. In *Proceedings of the IEEE Conference on Visualization 2003*, pages 523–530, 2003.

[18] S. T. Teoh, K.-L. Ma, S. F. Wu, and X. Zhao. Case study: Interactive visualization for internet security. In *Proceedings of the IEEE Visualization Conference 2002*, pages 505–508, 2002.

[19] Gnu zebra, free routing software. http://www.zebra.org.

[20] K. Zhang, A. Yen, X. Zhao, D. Massey, S. Wu, and L. Zhang. On Detection of Anomalous Routing Dynamics in BGP. In *Proceedings of Networking*, 2004.