On Evaluating BGP Routing Stress Attack

Wenping Deng[1,2], Peidong Zhu[1], Xicheng Lu[1]
[1]School of Computer Science, National University of Defense Technology, Changsha, China
Email: wpdeng.nudt@gmail.com, {pdzhu, xchlu}@nudt.edu.cn

Bernhard Plattner[2]
[2]Computer Engineering and Networks Laboratory, ETH Zurich, Switzerland
Email: plattner@tik.ee.ethz.ch

Abstract—**The routing system is playing a critical role in the Internet. Numerous routing security events reveal that the Internet is not so dependable yet. Some hackers even boasted that they could bring down the whole Internet in a short time. This paper investigates a new attack on BGP routing system inspired from synchronization and resonance in complex system. The attack applies routing stress by periodically injecting and propagating excessive BGP routing advertisements which are beyond the processing ability and the storage capacity of the BGP routers in the routing system. Our contributions are twofold. First, we describe a BGP routing stress attack method inspired from synchronization and resonance of complex network. Second, we devise a cascading failure model to evaluate the robustness under BGP routing stress attack on the real Internet AS-level topology. We measure the dependability and the connectivity under cascading failures with three metrics: the proportion of failed ASes, the proportion of failed links, and the proportion of disconnected AS-AS pairs. Our experimental results show that BGP routing stress attack can eventually lead to a high proportion of failures and bring about serious impacts on the connectivity of the Internet routing system.**

Index Terms—**BGP, inter-domain routing, BGP routing stress, cascading failure, synchronization**

I. Introduction

The Internet is composed of tens of thousands of Autonomous Systems (AS), where an autonomous system is a domain operating a network or a group of networks under the same administration and with common routing policies. Each AS is identified by a unique numerical ID obtained from RIRs. The Border Gateway Protocol (BGP) [1] is the standard inter-domain routing protocol, on which the Internet relies to convey routing information. BGP routers in different ASes run eBGP to exchange routing information, while BGP routers in the same AS run iBGP to synchronize their routes learned from the outside. When a BGP session is established for the first time, the peers exchange all the routes in their routing tables. After that, only the new changes are exchanged by BGP update messages.

The frequently happened events have told us that the Internet is confronted with many problems in security and robustness. As illustrated by the AS 7007 incident: A small ISP with AS number 7007, which accidentally announced routes learned from its provider (SprintLink), to most of the Internet, crashed many routers and disrupted most Internet connectivity for over two hours. In 2001, the spread of Code Red II and Nimda, led to route instabilities in the global Internet. In [2], a small misconfiguration caused many BGP routers to become overloaded and crash repeatedly, and additionally destabilized the surrounding network. Similarly, on December 24, 2004, AS 9121 incorrectly originated routes to more than 100 000 prefixes, and then much traffic to these prefixes was forwarded to AS 9121, which then essentially dropped the packets, affecting thousands of organizations [3]. As Schneier pointed out in one of his essays [4]: "Hackers from the L0pht boasted that they could bring down the Internet in 30 minutes!". But L0pht did not give any detailed description about the method.

In this paper we demonstrate a new method that could bring about cascading failures in the whole Internet by BGP routing stress attack. Here BGP routing stress refers to excessively propagating BGP routing updates beyond the processing ability (due to excessive updates) and the storage capacity of the BGP routers (due to routing table overload) in the inter-domain routing system. These excessive BGP updates may cause a large number of routers' BGP routing tables to be overloaded and exhaust the resources such as memory, CPU, and buffers of the BGP routers. In the worst case, the routers will restart. These restarting routers may generate new update messages and potentially cause cascading failures in larger scales. Hence the Internet's connectivity will be seriously perturbed due to a high proportion of failed ASes.

The rest of the paper is organized as follows. Section II presents the limitations of current routing security mechanisms. Section III describes the attack method of BGP routing stress. In Section IV, we present a cascading failure model and evaluate the propagating of poisoned BGP updates upon a topology with AS commercial relationships. Section V gives a simulation on the real Internet AS-level topology. Section VI reviews some

Manuscript received April 19, 2009; revised June 30, 2009; accepted September 17, 2009.

Corresponding author: Wenping Deng, School of Computer Science, National University of Defense Technology, Changsha 410073, China; Email: wpdeng.nudt@gmail.com.

doi:10.4304/jcm.5.1.13-22

related works. Finally, in section VII, we conclude the paper and discuss our future work.

II. Background

A good alternative way to prevent malicious actions and misconfigurations would be a global registry of prefix ownership and routing policies, coupled with verification of the contents of BGP update messages and filtering the inconsistent advertisements. The current global registry of prefix ownership and routing policies are composed of regional Internet registries (RIR) and Internet routing registries (IRR). Unfortunately, due to various reasons, they have been shown to be outdated and incomplete. For instance, RIPE is considered to be the best maintained RIR, however, only 73% of its prefix-AS registry information can be strongly validated in 2004 [5]. First, prefix ownerships change frequently accompanied by commercial contracts between organizations. Creation, reorganization, and bankruptcy of companies and organizations usually lead to changes of prefix ownerships. Second, ideally, the RIRs would be notified when the ownership changes, but most ISPs care more about their privacy issues and want to keep their commercial contract and policy confidential.

Meanwhile, there is another obstacle for this kind of approaches. The size of router filter lists is in a dilemma for wider deployment: if the filter lists are too big, they will consume too much memory and CPU resources; otherwise, they may be incomplete, so that they may have limited applications. Hence, most ISPs use other non-registry-based mechanisms, such as route-flap damping [6], prefix limiting (PL) [7], and graceful restart [8]. Some ISPs also filter small prefixes such as /26 and smaller ones.

Route-flap damping is used to identify and to suppress unstable routes for better reliability within and outside the autonomous system. However, it doesn't work on a newly announced route. Prefix limiting is a mechanism that places a configured limit on the number of prefixes that a router will accept from a given BGP neighbor. When the number of prefixes announced by a particular peer reaches the warning threshold, the router generates a warning message to its logging facility. If the number exceeds a given threshold, the router tears down the BGP peering session with its neighbor. Clearly, this feature prevents router memory overrun caused by a single large routing table infusion from a peer, but can not protect the router from multiple overloaded neighbors. Graceful restart suggests keeping forwarding state across TCP resets. It allows the router to continue forwarding traffic while it re-establishes BGP peering sessions. This mechanism is somewhat complementary to the impact of large routing loads that we consider. It can possibly preserve routing forwarding capability across some of the kinds of failures, but does not avoid the BGP session reestablishment. MRAI (Minimal Route Advertisement Interval) is enabled by default on some routers. After receiving an update for a certain prefix, a router waits a certain time if it receives further updates for the same prefix before propagating out updates to its neighbors. This means the overall number of updates can be somewhat restricted by MRAI.

Almost all of the large providers do not filter their peers, and only some coarse grain filtering mechanisms as the above-mentioned have been limitedly used to filter customers and small ISPs [9]. It is because there are some other dilemmas for filtering in the whole Internet routing system. Firstly, most of these mechanisms need a manual configuration by network managers. Secondly, diversity of router vendors also leads to various software releases of routers so that not all the routers support all those mechanisms. Thirdly, too complex configurations may have negative impacts on routes' reachability, thus the practical networks prefer simpler configurations. Due to some defects of the filtering mechanisms and human's carelessness, they do not play a strong safeguard in protecting the Internet from such an attack in fact.

III. An Attack Method of BGP Routing Stress

A. *Analysis of BGP Failure under Routing Stress*

Due to the flaws of IRR/RIR, it is still difficult to validate whether an AS is authorized to announce a certain prefix or not. An AS can advertise any route to its neighbors since BGP has several security problems. Murphy *et al* [10] lists three primary security related limitations of the current BGP: (1) BGP does not protect the integrity, freshness and source authentication of messages; (2) BGP does not validate an AS's authority to announce reachability information; (3) BGP does not ensure the authenticity of the path attributes announced by an AS.

The primary causes of BGP link and router failures are as follows [11]: (1) out of memory, (2) CPU overload, (3) queue overflows, and (4) router software bugs. When an AS maliciously advertises a large number of forged BGP routes, if its neighbors can not filter all the illegal routes, some of them will be propagated to the rest of Internet. These excessive BGP updates may exhaust resources such as memory, CPU, and buffers, or even cause the routes' BGP processes and BGP routing tables overloaded.

To demonstrate the impact of BGP routing stress, we design an experiment like Chang's experiments [12] as shown in Figure 1. We denote the advertising BGP speaker as A (Attacker), the router under test as T (Targeted Router, which doesn't announce its learned routes to its neighbors), the reference BGP speaker as R (Reference Router), and the monitoring terminal as M (Monitor). Here R is used as a reference point to judge whether T has a link failure or a router failure. It neither announces any route to T nor receives routes from T. R establishes its BGP session with T through an exchanger. A and T establish their BGP session over their Ethernet interface directly. In our experiment, T and R are two Cisco routers with 40MHz CPU, 64MB memory, and IOS 12.2 as their OS. The default input queue size and output queue size are 75 and 40 respectively. T's BGP KeepAlive timer and hold timer is 60s and 180s. A is a PC running Quagga [13] in Linux to act as a real BGP router. Compared to T, A has greater capacity and higher performance.

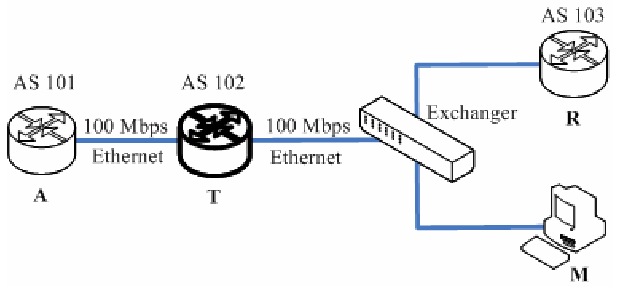

Figure 1. Network topology

To examine the router's response under BGP routing table overload, we increasingly inject new routes into T's BGP routing table until it falls into an abnormal state. The detailed process of our experiment is shown in Figure 2.

1: *Inject 10k new routes to A's routing table, A then advertises the new routes to T;*
2: *Telnet to T via M's terminal;*
3: *issue "show ip bgp summary" in M's terminal;*
4: *if all T's peering sessions keep established and T's routing table size equals to A's then do step 1;*
5: *else if A and R are both torn down then T fails;*
6: *else if only A is torn down then there is only a link failure in T;*
7: *else there isn't any failure in T, but T's table is not an accurate reflection of A's table.*

Figure 2. The procedure of the stress experiment

We have observed that the routing table size of router T repeatedly oscillates between 0 and the maximum value permitted by its capacity (about 39500). Meanwhile, all T's BGP session are reset when its BGP routing table is overloaded. However, we should confess that Cisco 2600 is not as high-performance as the routers used in Internet backbone. It is true that different series of routers may have quite different reactions when their routing tables are overloaded.

B. Synchronization Inspired from other Complex System

Synchronization and resonance in complex system are manifested in many branches of natural sciences, engineering and social life, for instance, the synchronous clapping in theaters [14], and the synchronization of flashing among fireflies [15]. Thousands of fireflies gathered in certain swarm trees begin flashing soon after sunset and synchrony builds up slowly through the night. This phenomenon has been modeled with the theory of pulse-coupled oscillators [16]. Each firefly is modeled as an oscillator, and each emitted light flash is treated as an infinitely short pulse. The oscillator transmits pulses periodically, and upon reception of a pulse from another oscillator adjusts its clock. At last, synchronization will emerge and pulses of different oscillators are transmitted simultaneously.

There is another example somewhat related to resonance, the dramatic Tacoma Narrows Bridge disaster of 1940. Technical experts still disagree on the exact cause of the bridge's destruction [17], but most of them agree the collapse had something to do with a complex phenomenon called resonance.

Inspired from such phenomena, if we can manipulate several compromised ASes and stimulate thousands of ASes to resonate, then our attack method can probably carry out a tremendous attack which may lead to a cascading failure on the whole Internet, while targeted attacks usually can only bring down local networks.

On AS level, the Internet can be treated as a complex network, which is composed of tens of thousands of AS components interconnected by various coupled relations. We model each AS as an oscillator, and each BGP update message as an impulse. No matter whether the coupling is strong or weak, the network is probable to be synchronized. Given there are *N* ASes in the Internet, the discrete coupled network state equation can be described as follows [17].

$$x_i(t+1) = f(x_i(t)) + c \cdot \sum_{j=1}^{N} (a_{ij} \cdot h(x_j(t))) \tag{1}$$

where $x_i(t)$ is the state of AS *i* (*i*=1, 2, ⋯, *N*) at time *t* (it can be either the BGP routing table size of AS *i* or the total BGP updating messages of AS *i*); *f* describes the relationship between $x_i(t)$ and $x_i(t+1)$; $c \geq 0$ is a constant; $h: R^n \to R^n$ is the coupling function between any given pair of ASes; $A = (a_{ij}) \in R^{N \times N}$ reflects the topology of the network, if AS *j* is AS *i*'s neighbor, then $a_{ij} = 1$, else $a_{ij} = 0$. The principle of synchronization will be used in the attack method and the cascading-failure model in the following sections.

C. Description of the Attack Method

In our attack method, several compromised ASes are employed to inject large number of poisoned BGP routes into the routing system periodically. These periodically generated routes give pulse to the Internet routing system when they are propagated to the rest of the Internet, which leads to resonance and synchronization among BGP routers and ASes. As more and more updating messages are propagated, many ASes become overloaded and failed. These failed ASes even spawn great deal of new BGP updating information which gives more heavy burdens to the routing system in return, which go beyond most ASes' abilities and capacities, will finally cause a cascading failure in the Internet.

Although all IP prefixes can be used as poisoned routes announced by the compromised routers, the adversaries may choose proper addresses to make the attack more efficient. Special addresses like private addresses are easy to be discovered and filtered as bogus routes, while advertisement of non-special address blocks are very difficult to be detected and filtered. A malicious attack might use a violated BGP speaking router to start advertising large ranges of non-special space to overload BGP and forwarding tables in routers.

To carry out such an attack, we implement a poisoned route injector which is based on Quagga [13]. The injectors work as follows. First, the injectors establish BGP sessions with legal BGP routers. Second, the injectors periodically generate large numbers of poisoned routes and announce these routes to their BGP neighbors. These poisoned routes will be propagated in the scope of the whole Internet (although they are probable to be filtered by some router). As shown in Figure 3, AS 1 and AS 4 are two compromised ASes which we call injecting points. The injectors are placed within AS 1 and AS 4. They announce the poisoned routes to their neighbors as their own routes, and their neighbors propagate the poisoned routes to the rest of the Internet.

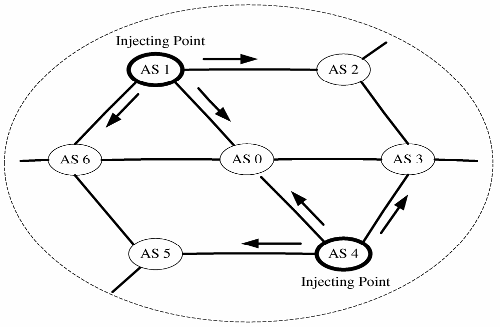

Figure 3. The injecting points

To perform a powerful stress attack, several principles are utilized for the deployment of injecting points. First, all the injectors issue the routes periodically in synchronization, therefore, all the newly injected routes play as a strong impulsion to stimulate all the ASes to be synchronized. Second, try to deploy as many injecting points as possible. Third, it is better to deploy the injecting points in the compromised ASes that have large number of AS neighbors. Finally, if the locations of the injecting points distribute dispersedly, it is more probable for poisoned routes to be propagated in the whole Internet. Meanwhile, the injecting points are more difficult to be discovered.

IV. Evaluating the Cascading Failures

Cascading failure [19][20] is a standard cause of catastrophic failure in interconnected complex systems. It is initiated when a heavily loaded node is failed for some reason, and it imposes its stress on other nodes in the network. This may cause other nodes to exceed their capacity causing them also to fail. Even if an overloaded node doesn't actually fail, protection mechanisms designed into the network may cause it to shut down anyway, in an attempt to prevent node failure. Hence the number of failed or stressed nodes increases, propagating throughout the network. In particularly serious cases the entire network is affected.

Reference [19] defines the cascading failure model with some general features: (1) Multiple identical components, each of which has an initial load and an initial disturbance; (2) When a component is overloaded, it fails and transfers some load to the other components. In BGP routing system, each AS has its own initial static routes, and it will exchange its route information with its neighbors after the BGP sessions being established. Because of the commercial relationships among different ASes, when one BGP router in an AS boots or fails, its neighbors have to announce or withdraw all the routes related to it, and this will trigger a cascading effect in the whole Internet. In this section we will explain the Internet cascading failures under BGP routing stress attacks.

A. Evaluation of Route Update Events Propagation

BGP is an incremental distance vector protocol. A BGP route describes an AS path to a given destination network from the current autonomous system. When a BGP router in an AS receives a new advertised BGP route from its neighbors, the router will compare this route with its existing routes destined to the same network, only if it becomes the best route, the router need to advertise it to its other neighbors according to their commercial agreements. Given an AS *X*, it has *K* neighbors including its providers, its peers, its siblings, and its customers. For any prefix in *X*'s BGP table, there must be one best route, probably another *K*-1 backup routes, so there may be at most *K* routes destined to the same network for each *X*'s neighbor may announce a route to the same destination.

To discuss stresses spawned by excessive BGP updates, we only care about whether and how a received route can be propagated by the current AS (transfer the stress to its neighbors), which we call stress propagating. Reference [21] roughly classified AS relationships into three categories: the provider-customer relationship, the peer-to-peer relationship, and the sibling-to-sibling relationship. The AS relationships govern the following BGP export policy rules [21]:

RULE-1: While exchanging routing information with a provider or a peer, an AS can export its routes and its customer routes, but usually doesn't export its provider or peer routes.

RULE-2: While exchanging routing information with a customer or a sibling, an AS can export its routes, its customer routes, its provider routes, as well as its peer routes.

Each update message may simultaneously announce a feasible route and withdraw multiple infeasible routes. That is, one update message may contain multiple different updating events indicating route announcement or withdrawal. To be more detailed, we decompose a BGP update message into 4 kinds of atomic BGP updating events according to route announcements and withdrawals. The BGP route export rules are listed as follows:

1) Original announcing: If the newly announced route is directly advertised by one of AS *X*'s neighbors AS *Y*, which implies the destination network of the new route originates from AS *Y*, there must be no route destined to the newly announced network in *X*'s routing table (because the attackers tend to use un-allocated addresses). There is no question that the new route will become the best because its AS-Path is the shortest (only one), hence

AS *X* will advertise this new route to its peers at a probability of 1.

2) Non-original announcing: If the newly announced route is indirectly advertised by one of AS *X*'s neighbors AS *Y*, that is, AS *Y* learns the route from its neighbors. AS *X*'s could also have learned the route from its other neighbors, hence there might be at most *K* routes in *X*'s routing table. The probability for the new route to become the best route to the destination is only 1/*K* (the probability is no less than 1/*K*, here we simply use 1/*K*), hence AS *X* will advertise this route to its other neighbors at a probability of 1/*K*.

3) Original withdrawing: If the newly withdrawn route is directly advertised by one of AS *X*'s neighbors AS *Y*, which implies the withdrawn route originates from AS *Y*. AS *X* will advertise this withdrawal to its neighbors at a probability of 1.

4) Non-original withdrawing: If the newly withdrawn route is indirectly advertised by one of AS *X*'s neighbors AS *Y*, which implies the withdrawn network does not belong to AS *Y*. AS *X* will advertise this new route to its peers at a probability of 1. AS *X* will advertise this withdrawal to its neighbors at a probability of 1/*K*.

For non-original announced routes and withdrawn routes, AS *X* performs a counteraction effect at a probability of 1/*K* on their further propagation to AS *X*'s downstream AS neighbors.

B. The Cascading-failure Model

This section gives a demonstration of a cascading-failure model. The model can be used to simulate and evaluate the attack method in a real Internet topology. In our cascading-failure model, we ignore the detailed structures of ASes. For the sake of simplicity, each AS is modeled by a single BGP router.

Table I lists the items of an AS object in our cascading-failure model. Each AS has an identifier *i*, a status number *S*, six static items, and two dynamic items.

TABLE I.
ITEMS OF A SINGLE AS OBJECT

Items	Descriptions
i	current AS number, as an identifier
S	current state, if *S*<0, failed; *S=0*, initial; else normal
Static items:	
C	if a router fails, it will takes *C* cycles to recover
R_{static}	total number of static routes, as the initial table size of the current AS
R_{max}	threshold for AS's routing table size
U_{max}	threshold for AS's BGP updating events
TS	TRUE if immune against routing table overload failure, otherwise FALSE
EU	TRUE if immune against excessive updates failure, otherwise FALSE
Dynamic items:	
R_{total}	current table size
U_{total}	the sum of announced and withdrawn events

For any AS (giving its AS number *i*), Figure 4 shows its state machine. It has one of the following four kinds of running state: (1) $S(i) = 0$, the initial state, AS *i* only needs to load its static routes; (2) $S(i) = 1$, it learns routes from its neighbors' routing tables, and then calculates its received updating events and updates its routing table size. If $R_{total} > R_{max}$ and $TS = FALSE$ or $U_{total} > U_{max}$ and $EU = FALSE$, then $S(i) = -C(i)$, AS *i* is broken down by BGP routing stress and it will spend $C(i)$ cycles to recover; (3) $S(i) > 1$, it calculates its received updating events and updates its routing table size. If $R_{total} > R_{max}$ and $TS = FALSE$, or $U_{total} > U_{max}$ and $EU = FALSE$, then $S(i) = -C(i)$, AS *i* fails, and it will spend $C(i)$ cycles to recover; and (4) $S(i) < 0$, it sets all its dynamic items to 0.

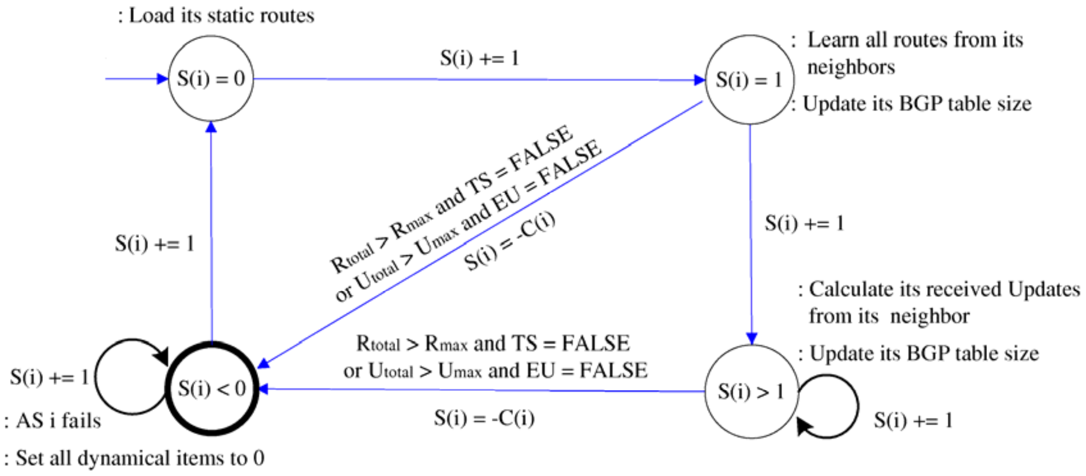
: Load its static routes
S(i) = 0
S(i) += 1
S(i) = 1
: Learn all routes from its neighbors
: Update its BGP table size
Rtotal > Rmax and TS = FALSE or Utotal > Umax and EU = FALSE
S(i) = -C(i)
S(i) += 1
S(i) += 1
: Calculate its received Updates from its neighbor
: Update its BGP table size
Rtotal > Rmax and TS = FALSE or Utotal > Umax and EU = FALSE
S(i) += 1
S(i) < 0
S(i) > 1
S(i) += 1
: AS i fails
S(i) = -C(i)
: Set all dynamical items to 0

Figure 4. A Finite State Machine of AS *i*

Assume that the current time is *t*. Based on the BGP route export rules and the discrete coupled network state equation (1), we demonstrate how AS *i* updates its received updating events number and its routing table size as shown in Figure 5.

```
1: Generate a real Internet AS-level topology G from multiple large BGP routing tables collected from different ISPs. G=(V, E), where V is the AS set and E is the edge set on V. Each v in V is modeled by an AS object;
2: Infer their commercial relationships. For each e in E, Given e=(AS1, AS2, relationship), their relationship is -1 if AS1 is a customer of AS2, 0 if AS1 and AS2 are peers, 1 if AS1 is a provider of AS2, and 2 if AS1 and AS2 are siblings which belong to the same organization;
3: All ASes are initially unfailed and their initial states S=0, set all their dynamic items to 0;
4: For each AS $i \in V$, update AS i's dynamic items and its state S(i) according to the finite state machine designed above;
5: Assume n denotes the total number of ASes satisfying S(i) < 0, then the current failure rate $p = n/|V|$.
```

Figure 5. The simulation procedure

When an AS (AS *i*) is failed, all of its neighboring ASes have to withdraw all the routes learned from AS *i*. Consequently, these withdrawals will bring in significant routing stress and lead to a cascading effect in the whole routing system.

C. Connectivity and Dependability Assessment

There are already several important statistical metrics such as the average path length, the clustering coefficient, and the degree distribution to measure the performance of a communication network. These metrics are effective in evaluating the performances and properties of network, however, none of them can be used to straightforwardly measure the status of the network connectivity under failures.

We measure the connectivity of the network with three metrics: the proportion of failed ASes (the failure rate of AS nodes), the proportion of failed links, and the proportion of disconnected AS-AS pairs.

Given an unweighted and undirected graph *G*=(*V*, *E*), *V* is the set of the ASes and *E* is the set of the AS links. We shall focus on the following three basic robust metrics:

(1) The proportion of failed ASes α (the failure rate): Given the sum of failed ASes (whose $S < 0$) is N_f, then $\alpha = N_f / |V|$.

(2) The proportion of failed links β: For any $e \in E$, *e* is the link between AS v_i and AS v_j, if v_i or v_j is failed, then *e* is a failed link. Given the sum of failed links is E_f, then $\beta = E_f / |E|$.

(3) The proportion of disconnected AS-AS pairs *γ*: Given two ASes v_i and v_j, we call the AS-AS pair (v_i, v_j) is a connected AS-AS pair if and only if there is a path from v_i to v_j (hence there is also a path from v_i to v_j because *G* is undirected), otherwise, (v_i, v_j) is a disconnected AS-AS pair. There are totally |*V*| ASes, hence there are $\frac{|V|(|V|-1)}{2}$ AS-AS pairs. Given the sum of disconnected AS-AS pairs is E_p, then $\gamma = E_p / \frac{|V|\cdot(|V|-1)}{2}$.

We measure the connectivity of the network in Figure 6. $V = \{1,2,3,4,5,6\}$, *E*={<1, 4>, <2, 4>, <3, 4>, <4, 5>, <5, 6>, <5, 7>}. Each AS is connected to the rest ASes, e.g., AS 1 is connected to AS 6 in 3 hops via AS 4 and AS 5, $\alpha = \beta = \gamma = 0$. Given AS 5 is failed as shown in Figure 7, $\alpha = 1/7 = 0.143$ because there is one AS failed (AS 5), $\beta = 3/6 = 0.5$ because there are 3 failed links <4, 5>, <5, 6>, and <5, 7>, and $\gamma = (21-3)/21 = 0.857$ because there are totally 21 AS-AS pairs but only 3 connected AS-AS pairs left (1-4, 2-4, and 3-4).

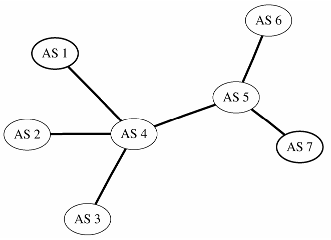

Figure 6. The original network topology

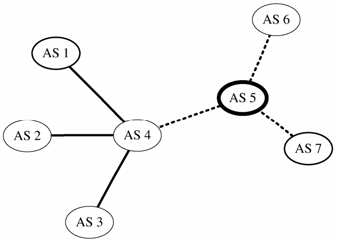

Figure 7. The network topology with AS 5 failed

V. Simulation and Result

This section performs several network simulations with our cascading-failure model under a real Internet topology. These simulations aim to evaluate the effectiveness of the attack method.

A. Simulation Data and Setup

Although many tools can be used to model a BGP router or a BGP real-time environment, including SSFNet [22] for BGP-4, C-BGP [23], and other networking simulators [24][25], there are two obstacles for us to use them in our simulation. First, because the current Internet AS-level topology consists of tens of thousands of AS nodes and connections, it is almost impossible to simulate such a large-scale BGP topology on the currently existing BGP simulators efficiently except C-BGP. Second, ASes' commercial relationships are hard to simulate on current BGP simulators. Therefore, we implement a BGP simulator based on our cascading-failure model with JDK 1.60 under Java Eclipse environment.

TABLE II.
THE INITIAL DEFAULT CONFIGURATION (30610 ASES)

rank	degree (K)	# of ASes	C	R_{max}	U_{max}	R_{static}
1	$K \geq 400$	30	5	5 000 000	50 000	100
2	$50 \leq K < 400$	210	5	1 000 000	10 000	50
3	$2 \leq K < 50$	19 328	5	200 000	5 000	20
4	$K = 1$	11 042	5	100 000	2 000	10

There are some publicly available sources of raw and processed BGP data. The data used for this paper mainly comes from RouteViews [26]. We also adopt the ASes' commercial relationships data from CAIDA [27]. The topology in our simulation comes from a file of [27] corresponding to 20090105, in which 30610 ASes and 68559 edges are given (see Figure 8 for a visualization).

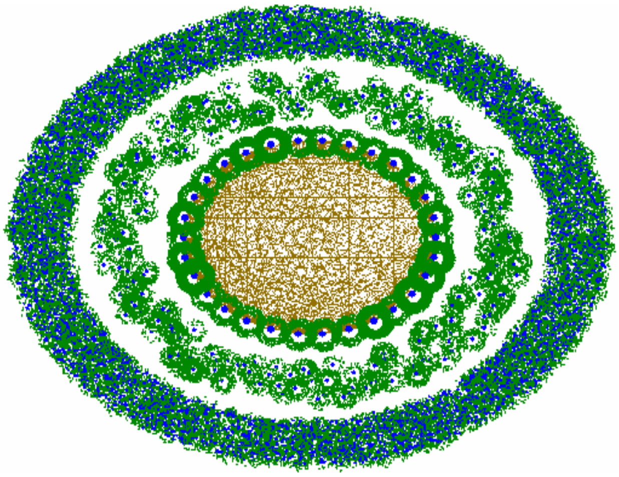

Figure 8. The initial topology with 30610 ASes (only links between rank-1 ASes visible)

To present a good visualization, we divide the ASes into four hierarchies according to their degrees like Govindan's method [28]. Because their connections are very complex, we only draw the edges of rank-1 ASes. Rank-4 ASes are placed around their neighbors. Usually, BGP routers in large-scale ASes may have higher capacity than those in small ASes, hence they can generally bear heavier burdens. Therefore, we diversify the initial default values for them according to Table I, as shown in Table II.

B. Relationships of (α, β) and (α, γ)

Here we give analysis on the relationship between the connectivity quality (the proportion of failed links and the proportion of disconnected AS-AS pairs) and the failure rate of AS nodes (the proportion of failed AS). We evaluate the connectivity of the network under random failures of AS nodes. We increasingly select α (from 0% to 100%) of the AS nodes at random and set them to be failed (S<0), then we calculate the corresponding proportion of failed links β and disconnected AS-AS pairs γ. The results are shown in Figure 9 and Figure 10.

1) The relationship of (α, β): We know from Figure 9 that the proportion of failed links β grows faster than the proportion of failed ASes α. When the proportion of failed ASes reaches 20%, 40% of the links are failed, and when 60% of the ASes are failed, more than 85% of the links get failed. It indicates that AS links are seriously infected by the failed ASes.

2) The relationship of (α, γ): We know from Figure 10 that the proportion of disconnected AS-AS pairs γ grows faster than the proportion of failed ASes α. When α reaches 20%, 50% of the AS-AS pairs are disconnected, and when 60% of the ASes are failed, almost 95% of the AS-AS pairs get disconnected. It indicates that AS-AS pairs are also very sensitive to the failed ASes.

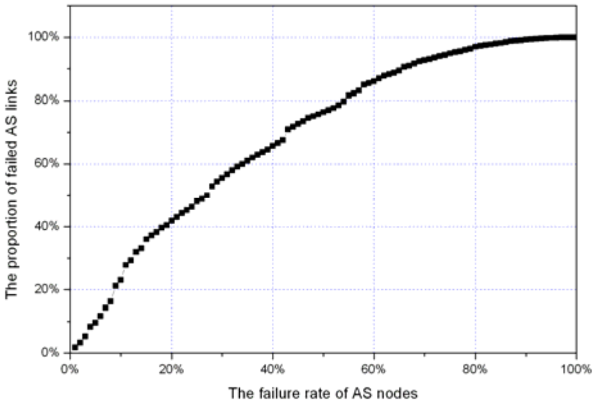

Figure 9. The proportion of failed AS links according to the failure rate of AS nodes

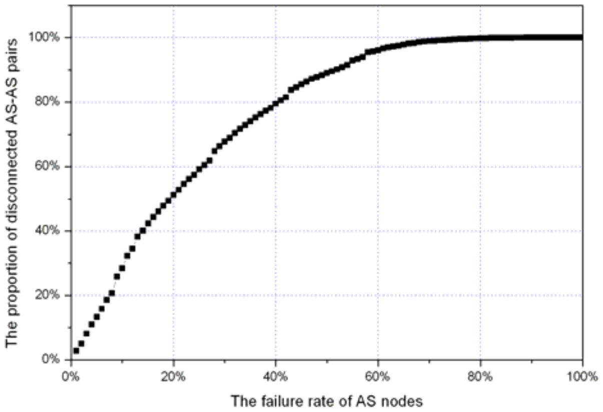

Figure 10. The proportion of disconnected AS-AS pairs according to the failure rate of AS nodes

We conclude from (1) and (2) that the connectivity of the network would be seriously destroyed when high proportion of AS nodes get failed. Even for random failures of AS nodes, a failure rate of 20% can lead to high proportions of failed links (40%) and disconnected AS-AS pairs (50%).

C. Robustness Assessment

Considering the impact of prefix limiting, MRAI timer, graceful restart, and good filter, some ASes might be immune against such a BGP routing stress. Therefore, if the *TS* and *EU* of an AS object are set to be TRUE, it can always hold on in our simulation. Given a ratio p $(0 \le p \le 1)$, p of the ASes have immunity against the BGP routing stress. We perform our simulation under two different values of $p = 0\%$ and $p = 50\%$ both in 400 cycles.

In our simulations, we select 100 ASes at random as the routing stress injecting points. The interval between two iterations is 100ms. In every cycle, each of the selected injecting points injects 1000 updates into its neighboring ASes. All ASes' *C* items are set to be 5, thus it will take 5 cycles for a failed AS to recover. If AS *i* fails at *t*, then *C*(*i*) is set to be -5, and will be increased in each iteration. Therefore, it will take 5 cycle for *C*(*i*) to get back to 0.

1) $p = 0\%$: In this simulation, *TS* and *EU* of each AS object are set to be FALSE, thus, all the AS objects in the simulator have no immunity against BGP routing stress so that they will be down for the sake of overloaded routing table or excessive BGP updating events. We observe a simulation of 400 cycles. The cascading failure rate curve is shown in Figure 11 and Figure 12. Figure 11 shows that the proportion of failed AS nodes oscillates in the simulation, and Figure 12 shows that the proportion of disconnected AS-AS pairs fluctuates in the simulation. In the front 150 cycles, the two failure rates grow gradually, after that, both of them get into oscillation. From *t*=150 to *t*=400, the average proportion of failed AS nodes is about 4% and the highest proportion is 11.6% (*t*=251), the average proportion of disconnected AS-AS pairs is about 24% and the highest proportion is 68.2% (*t*=344). The proportion of disconnected AS-AS pairs is higher than that is expected from the statistics of Figure 10, the reason would be that the simulation is different from random failures.

2) $p = 50\%$: In this simulation, 50% of the AS objects chosen at random are set to have the immunity against BGP routing stress in the simulator so that they will not be down for the sake of overloaded routing table or excessive BGP updating events, *TS* and *EU* of these AS objects are set to be TRUE. As shown in Figure 11 and Figure 12, in the front 300 cycles, the two failure rates grow gradually, after that, both of them get into oscillation. From *t*=300 to *t*=400, the average proportion of failed AS nodes is about 1.8% and the highest proportion is 2.4% (*t*=335), the average proportion of disconnected AS-AS pairs is about 12% and the highest proportion is 18.8% (*t*=342).

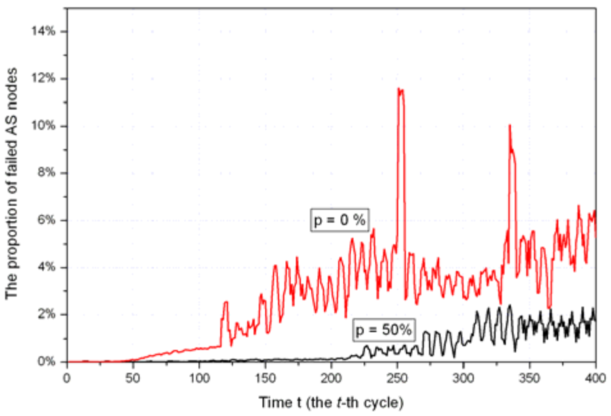

Figure 11. The proportion of failed ASes in the simulation (*p* is the ratio of the ASes that is immunity from BGP routing stress Attack)

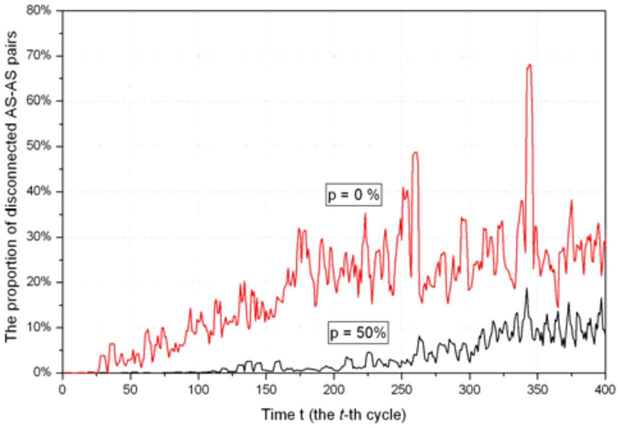

Figure 12. The proportion of disconnected AS-AS pairs in the 400-cycle simulation

We conclude from the two simulations that the routing system can be well protected from BGP routing stress when a high ratio of ASes have corresponding security mechanisms. We also validate that if we increase the injecting points, more BGP routing stress will be injected into the Internet so that a higher failure proportion will appear in the simulation. Further, the recovering cycle *C* also has significant influence on the failure rate. As we increase the recovering cycle item, failed ASes will need longer time to recover, therefore, a higher failure proportion will appear in the simulation.

Now we demonstrate why the failed ratios can not go as high as we expect (exactly, even no higher than 12%). In fact, we should not neglect that the connectivity of the Internet may be ravaged seriously when the failed ratio reaches to a high level. In this situation, the Internet is decomposed into pieces of clusters and isolated ASes, hence the propagation of the routing stress will be slowed down or held up by the failed ASes. It also gives us strong supports that the BGP routing stress attack can damage the Internet's connectivity and degrade the service quality of the Internet.

In summary, the BGP routing stress attacks can eventually lead to cascading failures in the Internet BGP routing system with an AS failure rate up to 11.6% in our simulations. Thousands of ASes synchronously fail and reboot in turn, hence the connectivity of the routing system would be seriously disturbed and damaged.

VI. Related Work

Although there are many works giving a detailed analysis on BGP behaviors during worm stress, our work gives a first step on BGP routing stress attacks. Reference [29] studied the BGP behavior under the Slammer worm outbreak. Reference [11] showed strong correlation between instabilities in BGP routing tables and the propagation phases of well known worms. In [30], the authors examined BGP's behavior during one stressful period of the Code Red/Nimda attack on September 18th, 2001. Yet, there are some significant differences between worm stress and BGP routing stress. The former happens on the data plane of the routings system. It brings about a large number of traffic which leads to network congestion, and it is easy to be detected and filtered; while the latter happens on the control plane of the routing system, it is hard to detect and prevent.

Di-Fa Chang and his partners [12] provided convincing evidences that BGP routers and BGP sessions do fail under several stress conditions (such as excessive memory demands). They did a series of experiments to investigate the detailed mechanics of routers' response to large BGP routing table load. In their experiments, they chose three commercial routers, including Cisco 7000 (IOS 11.1), Cisco 12008 GSR (IOS 12.0), and Juniper M20 (JUNOS 4.3). Their experiment showed that the responses of the three routers varied significantly. Cisco 7000 was reset while its routing table was overloaded, and it finally fell into a table-oscillation state; Cisco 12008 GSR permanently stopped responding to the interface where it peered with the advertisers when receiving excess routing announcements. This caused a link failure; similarly, Juniper M20 either reset the connections so that the connections oscillate, or frozen the sessions. However, this work does not face towards the Internet global system.

We get some key points to prevent the Internet from such a kind of BGP routing stress attacks. One point is to protect the BGP sessions so as to ensure the integrity, freshness and source authentication of messages, such as IPSec [31]. The second point is to construct a more trustable and secure Internet BGP routing system. BGP security mechanisms such as S-BGP [32], soBGP [33], and psBGP [34] can be used to validate an AS's authority to announce reachability information or to ensure the authenticity of the path attributes announced by an AS. Unfortunately, none of them have been deployed in the running Internet today for several reasons. The final point is to make the BGP routing system more resilient and robust. Fast recovery mechanisms such as graceful restart [8] are useful to alleviate the affection of node failures. All these methods and mechanisms are helpful against the BGP routing stress attack, but it is hard to deploy in the whole Internet for too many reasons and limitations.

VII. Conclusion

In this paper, we have described an attack method to destroy the Internet's connectivity. It might lead to cascading failures with high proportions of failed AS and disconnected AS-AS pairs, and perturb the connectivity of the whole Internet. There are also some weak points and limitations in our paper. We model each AS with a single router, since there will be complex interactions between routers within the same large AS, this ignores the detailed of the inner structure of an AS. Since we do not go further on BGP policies, the probability of advertising the updates to the peers is simplified.

Acknowledgment

The author would like to thank Jinyao Yan, Yanzhen Wang, Kai Xu, and Jie Li for their suggestions, and Jianfeng Zhang, for his assistance in the experiment testing the routers under BGP routing stress attacks. This research is supported by National Natural Science Foundation of China (Grant No. 60873214), National Natural Science Foundation of China (Grant No. 60873214), and China Scholarship Council (CSC).

References

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

[2] R. Mahajan, D. Wetherall, T. Anderson, "Understanding BGP Misconfiguration," in *Proceedings of the ACM SIGCOMM*, 2002.

[3] A. Popescu, B. Premore, and T. Underwood, "Anatomy of a leak: AS9121," http://www.nanog.org/mtg-0505/underwood.html.

[4] B. Schneier, "Click here to bring down the Internet," http://www.schneier.com/essay-003.html. 1998.

[5] G. Siganos, M. Faloutsos, "Neighborhood Watch for Internet Routing: Can we improve the Robustness of Internet Routing Today?" in *Proceedings of IEEE INFOCOM*, 2007.

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

[7] S. Chavali, V. Radoaca, M. Miri, L. Fang, S. Hares, "Peer Prefix Limits Exchange in BGP," *IETF draft*, April 2004.

[8] S. Sangli, Y. Rekhter, R. Fernando, J. Scudder, E. Chen, "Graceful Restart Mechanism for BGP," *RFC 4724*, January 2007.

[9] J. Deleskie, A. Popescu, T. Scholl, T. Underwood, "BGP Filtering-Myths Legends and Reality: Peer Filtering in the Modern Backbone," *NANOG 35*, October 2005

[10] A. Barbir, S. Murphy, Y. Yang, "Generic Threats to Routing Protocols," *RFC 4593*, October 2006.

[11] J. Cowie, A. Ogielski, B. Premore, Y. Yuan, "Global Routing Instabilities Triggered by Code Red II and Nimda Worm Attacks," *Tech. Rep., Renesys Corporation*, December 2001.

[12] D. Chang, R. Govindan, J. Heidemann, "An Empirical Study of Router Response to Large BGP Routing Table Load," *Tech. Rep. ISI-TR-2001-552, USC/Information Sciences Institute*, December 2001.

[13] Quagga Routing Suite, http://www.quagga.net/.

[14] Z. Néda, E. Ravasz, Y. Brechet, T. Vicsek, A. Barabási, "The Sound of Many Hands Clapping," *Nature*, 403: 849-850, 2000.

[15] A. Moiseff, J. Copeland, "A New Type of Synchronized Flashing in a North American Firefly," *J. Insect Behav.*, 13(4): 597-612, 2000.

[16] A. Winfree, "Biological Rhythms and the Behavior of Populations of Coupled Oscillators," *Journal of Theoretical Biology*, 16:15-42, July 1967.

[17] K. Billah and R. Scanlan, "Resonance, Tacoma Narrows Bridge Failure, and Undergraduate Physics, Textbooks," *American Journal of Physics*, 1991.

[18] C. Li, G. Chen, "Synchronization in General Complex Dynamical Networks with Coupling Delays," *Physical A*, 343:236-278, 2004.

[19] I. Dobson, B. Carreras, D. Newman, "A Probabilistic Loading-dependent Model of Cascading Failure and Possible Implications for Blackouts," in *Proceedings of 36th Hawaii International Conference on System Sciences*, Hawaii, 2003.

[20] A. Motter, T. Nishicawa, Y. Lai, "Cascade-based Attacks on Complex Networks," *Physical Review E*, 66: 065102(R), 2002.

[21] L. Gao, "On Inferring Autonomous System Relationships in the Internet," in *Proceedings of IEEE Global Internet*, November 2000.

[22] SSF Implementation of BGP-4 v1.5.0, http://www.ssfnet.org/bgp/doc/.

[23] C-BGP, http://cbgp.info.ucl.ac.be/.

[24] N. Feamster, J. Winick, J. Rexford, "A Model of BGP Routing for Network Engineering," in *Proceedings of ACM SIGMETRICS*, June 2004.

[25] The Network Simulator - ns-2, http://www.isi.edu/nsnam/ns/.

[26] University of Oregon Route Views Project, http://www.routeviews.org/.

[27] AS ranking, http://as-rank.caida.org/data/.

[28] R. Govindan, A. Reddy, "An Analysis of Internet Inter-domain Topology and Route Stability," in *Proceedings of INFOCOM*, 1997.

[29] M. Lad, X. Zhao, B. Zhang, D. Massey, L. Zhang, "Analysis of BGP Update Surge during Slammer Worm Attack," in *Proceedings of 6th International Workshop on Distributed Computing (IWDC)*, December 2004.

[30] L. Wang, X. Zhao, D. Pei, R. Bush, D. Massey, A. Mankin, S. Wu, L. Zhang, "Observation and Analysis of BGP Behavior under Stress," in *Proceedings of IMW 2002*, November 2002.

[31] S. Kent, R. Atkinson, "Security Architecture for the Internet Protocol," *RFC 2401*, November 1998.

[32] S. Kent, C. Lynn, K. Seo, "Secure Border Gateway Protocol (S-BGP)," *IEEE Journal on Selected Areas in Communications Special Issue on Network Security*, 18(4): 582-592, 2000.

[33] R. White, "Securing BGP Through Secure Origin BGP," *Internet Protocol Journal*, 6(3): 15-22, 2003.

[34] T. Wan, E. Kranakis, P. Oorschot, "Pretty Secure BGP (psBGP)," *ISOC*, 2005.

Wenping Deng was born in China in 1981. He received his BS and MS degrees in Computer Science from School of Computer Science of the National University of Defense Technology (NUDT), Changsha, Hunan, China, in 2004 and 2006 respectively. He started his Ph.D study in NUDT from the spring of 2007. Currently he is a visiting scholar to the Communication Systems Research Group (CSG) of ETH Zurich, Switzerland, from November 2008 to November 2009. He was one of the main contributors of several Internet security projects of the Chinese government. His research interests include Internet routing, routing security, and resilient network.

Peidong Zhu is a professor with School of Computer Science of National University of Defense Technology (NUDT), China. He received his Ph.D. degree in computer science from NUDT in 1999. In 2008，he was the James visiting chair professor at St Francis Xavier University, Canada. His research interests include network routing, network security and architecture design of the Internet and various wireless networks. He has published more than 120 papers, authored one monograph independently and co-authored four books. He holds 5 authorized patents, 2 pending patents and 4 software copyright registered certificates.

Xicheng Lu received his BS degree in computer science from Harbin Engineering Institute, Harbin, China, in 1970. He was a visiting scholar at the University of Massachusetts from 1982 to 1984. He is currently a professor in the College of Computer, National University of Defense Technology, China. His research interests include distributed computing, computer networks, and parallel computing. He has served as a member of editorial boards of several journals and has co-chaired many professional conferences. He is a joint recipient of more than a dozen academic awards, including four First Class National Scientific and Technological Progress Prizes of China. He is an academician of the Chinese Academy of Engineering.

Bernhard Plattner is a Professor of Computer Engineering at ETH Zurich, where he leads the Communication Systems Group. His research currently focuses on self-organizing networks and systems-oriented aspects of information security. He has been the principal investigator or Co-PI of numerous national and international projects. In 1996-98, Dr. Plattner served as the Head of Faculty of Electrical Engineering at ETH Zurich. He currently is the Vice-Rector for Bachelor/Master Studies at ETH Zurich. Dr. Plattner is a member of the IEEE, ACM and the Internet Society. He served as the program or general chair of various international conferences, such as ACM SIGCOMM 1991, INET 1994, and IWAN 2002.