1-1-2010

AS-TRUST: A Trust Characterization Scheme for Autonomous Systems in BGP

Jian Chang
University of Pennsylvania

Krishna K. Venkatasubramanian
University of Pennsylvania

Andrew G. West
University of Pennsylvania

Sampath Kannan
University of Pennsylvania, kannan@cis.upenn.edu

Boon Thau Loo
University of Pennsylvania, boonloo@cis.upenn.edu

See next page for additional authors

University of Pennsylvania Department of Computer and Information Science Technical Report No. MS-CIS-10-25.

AS-TRUST: A Trust Characterization Scheme for Autonomous Systems in BGP

Abstract

Border Gateway Protocol (BGP) works by frequently exchanging updates which, disseminate *reachability information* (RI) about IP prefixes (*i.e.*, address blocks) between Autonomous Systems (ASes) on the Internet. The current operation of BGP implicitly trusts the ASes to disseminate *valid*—accurate, stable and routing policy compliant — RI. This assumption is problematic as demonstrated by the recent documented instances of invalid RI dissemination. This paper presents *AS-TRUST*, a scheme which comprehensively characterizes the trustworthiness of ASes, with respect to disseminating valid RI. AS-TRUST quantifies trust using the notion of *reputation*. To compute reputation, AS-TRUST evaluates the past RI received for validity, based on a set of well-defined properties. It then classifies the resulting observations into multiple types of *feedback*. The feedback is used by a *reputation function* to compute a probabilistic view of AS trustworthiness. The *contributions* of the paper are: (1) a comprehensive trust characterization of ASes; (2) a set of well-defined properties for evaluating the validity of RI provided by ASes; and (3) a novel and theoretically sound reputation computation mechanism. Our implementation of AS-TRUST scheme using publicly available BGP traces demonstrates: the number of ASes involved in violating the BGP operational trust assumption is significant, dissemination of invalid RI is consistently present, and the proposed reputation mechanism is sensitive enough to capture even rare instances of an AS' deviation from trustworthy behavior.

Comments

University of Pennsylvania Department of Computer and Information Science Technical Report No. MS-CIS-10-25.

Author(s)

Jian Chang, Krishna K. Venkatasubramanian, Andrew G. West, Sampath Kannan, Boon Thau Loo, Oleg Sokolsky, and Insup Lee

AS-TRUST: A Trust Characterization Scheme for Autonomous Systems in BGP

Jian Chang, Krishna K. Venkatasubramanian, Andrew G. West,
Sampath Kannan, Boon Thau Loo, Oleg Sokolsky, and Insup Lee
Department of Computer and Information Science,
University of Pennsylvania, Philadelphia, PA, 19104
{jianchan, vkris, westand, kannan, boonloo, sokolsky, lee}@cis.upenn.edu

***Abstract*—Border Gateway Protocol (BGP) works by frequently exchanging updates which, disseminate *reachability information* (RI) about IP prefixes (*i.e.*, address blocks) between Autonomous Systems (ASes) on the Internet. The current operation of BGP implicitly trusts the ASes to disseminate *valid* — accurate, stable and routing policy compliant — RI. This assumption is problematic as demonstrated by the recent documented instances of invalid RI dissemination. This paper presents *AS-TRUST*, a scheme which comprehensively characterizes the trustworthiness of ASes, with respect to disseminating valid RI. AS-TRUST quantifies trust using the notion of *reputation*. To compute reputation, AS-TRUST evaluates the past RI received for validity, based on a set of well-defined properties. It then classifies the resulting observations into multiple types of *feedback*. The feedback is used by a *reputation function* to compute a probabilistic view of AS trustworthiness. The *contributions* of the paper are: (1) a comprehensive trust characterization of ASes; (2) a set of well-defined properties for evaluating the validity of RI provided by ASes; and (3) a novel and theoretically sound reputation computation mechanism. Our implementation of AS-TRUST scheme using publicly available BGP traces demonstrates: the number of ASes involved in violating the BGP operational trust assumption is significant, dissemination of invalid RI is consistently present, and the proposed reputation mechanism is sensitive enough to capture even rare instances of an AS' deviation from trustworthy behavior.**

I. Introduction

Large IP domains, called *Autonomous Systems* (ASes) use the Border Gateway Protocol (BGP) as the standard communication protocol. BGP enables ASes to exchange IP prefix (*i.e.*, address blocks) reachability information with each other, through periodic propagation of *update* messages. The *reachability information* (RI) within a BGP update consists of: an IP prefix, and an ordered list of ASes, called *AS_PATH*, through which the prefix is reachable. One of the major operational assumptions of BGP is that the RI provided by the ASes is *valid*. We define RI validity as: (1) the information in the updates are legal and correct, (2) the ASes in the *AS_PATH* provide a stable route to the prefix, and (3) no routing policies are violated in the process of propagating the updates. However, over the past decade, it has been seen that this assumption is not entirely true. Documented evidence — of *prefix hijacking*, where an AS claims to reach a prefix, contrary to its actual capability [5], [10]; *routing policy violation*, which might prevent BGP convergence in the long run [2]; and *unstable or potentially spoofed* link insertion in the *AS_PATH* to make it more attractive [23] — demonstrate as such.

It can be seen that ASes cannot be completely trusted to disseminate valid RI at all times. Therefore, it is essential to quantify the extent to which individual ASes can be trusted. Trust is defined as the competence of an entity to exhibit a specific behavior(s) [15]. In the context of BGP, the entity is an AS and the expected behavior is disseminating valid RI. Trust value for ASes should be adaptive to the behavioral changes in ASes over time. The quantification of AS' trustworthiness has two main *advantages*: (1) it provides a global view of the current state of inter-domain routing and the extent to which it is plagued by the aforementioned hijacking, stability and policy violation issues, and (2) it can potentially minimize the venues available for spammers and hackers to exploit if the AS trustworthiness information is made available to the entire BGP community.

In this paper, we present *AS-TRUST*, a novel scheme for quantifying the level of trust an one can have on ASes in terms of disseminating valid RI. To the best of our knowledge, this is the first attempt to re-examine the operational trust assumption of BGP in a quantitative manner. In AS-TRUST, trust is represented using a metric called *reputation*. To compute the reputation of an AS, AS-TRUST evaluates past RI received, for exhibition of specific behaviors, based on well-defined properties. The behavior the evaluation provides feedback to a reputation function to generate a probabilistic view of the trustworthiness of all the observable ASes in the Internet. The AS-TRUST scheme can be implemented by anyone with access to BGP updates such as, individual Autonomous Systems or third parties with access to BGP traces collector, *e.g.,* RouteViews [11].

The principal **contributions** of this paper are: (1) consideration of a comprehensive set of behaviors in computing the trustworthiness of an AS; (2) identification of a set of metrics for accurately detecting each of the AS behaviors of interest; and (3) a reputation computation scheme which provides a probabilistic view of AS trustworthiness based on Bayesian statistics. Our implementation of AS-TRUST demonstrates: (1) the incidents of poor behaviors (*i.e.,* disseminating invalid RI) is consistently present, (2) a considerable percentage of ASes (5-6%) are involved in some form of poor behaviors with

This research was supported in part by ONR MURI N00014-07-1-0907. POC: Insup Lee, lee@cis.upenn.edu

a handful exhibiting poor behavior exclusively, and (3) the proposed reputation mechanism is sensitive enough to capture even rare instances of deviation from trustworthy behavior.

The paper is organized as follows. Section II presents background on BGP and the problem statement. Section III presents details of AS-TRUST including the notion of BGP service, feedback mechanism employed and the reputation model. Section IV presents the properties for evaluating the BGP services. Section V presents the AS reputation computation and analysis. Section VI presents the result summary. Section VII presents the related work followed by Section VIII, which concludes the paper.

II. Preliminaries

A. The Border Gateway Protocol

The Border Gateway Protocol is a path-vector routing protocol for exchanging information about reaching IP prefixes. Using BGP, each AS informs its neighbors about the best available route to a prefix it *owns* — is directly reachable from the AS. In this regard, AS sends out a BGP update message *announcing* the prefix. Similarly, an AS can *withdraw* a prefix that it has previously announced. Each AS through which the update passes adds its AS number to the message. This ordered list of ASes called the *AS_PATH* informs an AS receiving the update, the path through which the prefix can be reached. When an update is received by an AS containing a prefix announcement, it has to determine whether it should be accepted or not. Acceptance means that the AS is willing to add the route to its routing information base. Each AS has its set of custom policies that determine whether it should accept an update. Routing policies serve the additional purpose at the ASes of selecting proper neighbors to export the accepted route. One of the most common export policies is adherence to the principle of *Valley Free Routing* (VFR). An *AS_PATH* is said to adhere to valley-free routing if it meets the following three conditions: (1) no AS forwards a route received from its provider[1] to another provider, (2) a path does not have more than one peer-to-peer links, and (3) a path may not contain two peers separated by one or more non-peer AS. Adherence to VFR is important as it has been shown to ensure the eventual convergence of BGP [14].

B. Problem Statement and Approach

The current version of BGP [1] was designed with only effectiveness in mind. It implicitly assumes ASes can be trusted to provide valid prefix reachability information. RI in an update is considered *valid*, if it exhibits five behaviors:

- *Legality:* The values of the AS number and the prefix in the RI are legal.
- *Accuracy:* The information regarding the prefix and the ASes in the *AS_PATH* of the RI is accurate.
- *Unwavering:* An AS announcing prefixes it owns, should do so in a reasonably sustained manner.

[1]ASes and their neighbors usually have one of the four relationships: provider-to-customer (Pv2C), customer-to-provider (C2Pv), sibling-to-sibling (S2S), and (P2P) peer-to-peer.

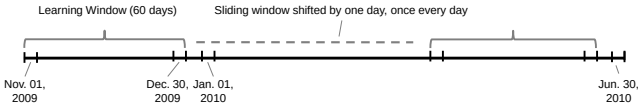

Fig. 1. **Data Source Time Windows**

- *Valley Free Path:* The *AS_PATH* of a RI is valley free.
- *Stable AS-Links:* The AS-links (*i.e.,* each hop between individual ASes) in the *AS_PATH* of a RI persist in a reasonably sustained manner.

The principal problem this paper tries to address is to develop a behavior model for ASes. In other words, AS-TRUST tries to answer the question - *what is the probability with which an AS disseminates valid reachability information*? This is done by computing the trust one can have in ASes satisfying the aforementioned five behaviors. In this regard, we use the notion of reputation. *Reputation* is a quantitative measure of an entity's likelihood to perform a specific task based on its past performance [18]. The idea is to compute the reputation for all the ASes in the Internet based on the RI received in the past. This is done in four steps following the traditional manner of reputation computation [18]: (1) collecting BGP updates in a database; (2) evaluating the data in the database, over a well-defined duration called the *learning window*, for the exhibition of the aforementioned five behaviors; (3) recording the results of the analysis as feedback; and (4) using feedback to compute reputation for the ASes. Reputation is a dynamic value which changes as the AS behavior changes, over time. This is accomplished by repeating the evaluation process over a sliding window and generating updated feedback.

C. Experiment Setup

We implemented the proposed scheme and conducted a six month long experiment measuring the evolving trustworthiness of ASes, on an Internet-scale. To receive the latest BGP updates, we use the RouteViews BGP trace collector, maintained by University of Oregon [11]. The RouteViews trace collector is a group of BGP routers which peer with a large number of ISPs via BGP sessions. At the time of writing, the RouteViews received BGP updates from 46 ASes. It has been shown in [25] that RouteViews receives RI from almost all the ASes currently active within the Internet and is therefore a good source for computing reputation of ASes.

In this work, we use BGP update data from Nov. 1, 2009 - Jun. 28, 2010 (see Figure 1). We take BGP updates received over a 60 day period called the *learning window*, evaluate the AS behavior, and compute reputation for the ASes on the 61st day. For example, data from Nov. 1, 2009 to Dec. 30, 2009 is analyzed to compute AS reputation on Jan. 1, 2010. The learning window is then slid forward by one day and the whole process is repeated. In order to be fair to ASes, we did not consider RI announced within 24 hours of the end of the learning window in computing the reputation of the ASes as they have not had enough time to prove themselves. There are over 180 learning windows between Nov. 1, 2009 and Jun. 28,

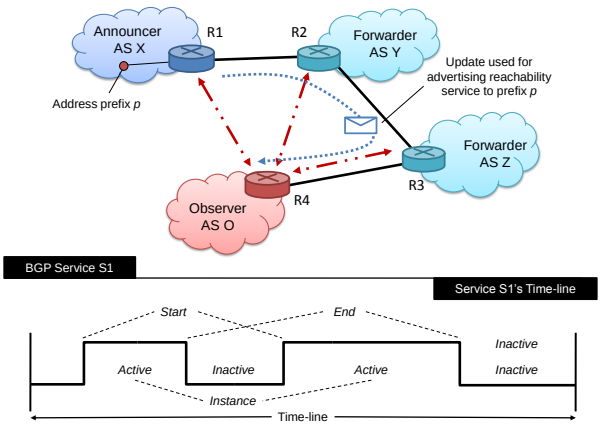

Fig. 2. **BGP Service and Timeline**

2010. The 60 day learning window was chosen as it was long enough to prevent the behavior evaluation from being biased by transient AS behavior.

III. Reputation Computation for Autonomous Systems

This section provides an overview of the principal aspects of computing trustworthiness of ASes. We begin by formalizing the notion of BGP service which forms the basis of the whole process. We then present the mechanism for obtaining feedbacks. In the subsequent sections we describe the evaluation process and reputation computation, respectively.

A. BGP Service

The principal task of BGP is to facilitate the dissemination of routing information. We model the dissemination of routing information using a novel notion called *BGP service*. A BGP service is a formal way of viewing RI provided collectively by the ASes in the *AS_PATH*, called *providers*[2], to an *observer* AS receiving a BGP update. It is defined as:

$$S_i = \{p_i, AS\_PATH = [AS_0, \ldots, AS_N]\}$$

Here S_i is the service identifier indexed by i, p_i is the prefix being announced by AS_0 as a part of the service S_i, and $AS_0, \ldots, AS_N \in AS\_PATH$ are the provider ASes which forward the reachability information as a part of the service S_i. Figure 2 illustrates the principal concepts and entities of a BGP service. A service said to have *started* when a provider AS announces a particular prefix and *ended* when the prefix is withdrawn. A service can therefore be in two modes: *active* and *inactive*. A service is said to be active, if it has started but not ended; and inactive if it has ended. An inactive service has to have been active, at least once, in the past. Each time a service is active, it is called an *instance* of that service. The bottom half of the Figure 2 illustrates some of these concepts over a *time-line* of a service.

[2]These are different from notion used in the context of VFR. Here, the term provider ASes mean provider of a service. In the rest of the paper, unless otherwise specified, the term provider refers to provider ASes.

A BGP service can be decomposed into three orthogonal *service elements*, each of which are provided by a subset of provider: (1) **AS-prefix binding:** a tuple of the form (AS_0, p), which is established when an AS_0 announces a prefix p and is broken when the prefix is withdrawn. Each BGP service has one AS-prefix binding in it. This service element is provided by AS_0. (2) **AS-path vector:** is synonymous with the the *AS_PATH* in the service. It is said to be *provided collectively by all the providers*; and (3) **AS-link binding:** a tuple of the form (AS_i, AS_j), which is established when AS_i forwards an update to AS_j. The AS-link binding is broken when no service uses it. A service has $N-1$ AS-link bindings; one between each of the N ASes in the *AS_PATH*. This service element is said to be *provided individually by the all providers* to the observer. In the rest of the paper, we use the term AS-link bindings and AS-links, interchangeably.

Upon observing a BGP service, the observer decomposes it into its constituent service elements, each of which is then evaluated on its validity. The results of the evaluation act as a feedback on the providers of the service element. The next sub-section describes the behaviors used for evaluation of the service elements, followed by the feedback mechanism used. *Note that, all behavior evaluation and feedback generation are performed locally at the observer.*

B. Behavior Evaluation

We propose three behavior sets, one corresponding to each service element, for behavior evaluation. The three behavior sets comprehensively cover the principal aspects of BGP operation. They are:

- *Behavior Set 1 (B_p)*: Requires that an AS announce only those prefixes which it *owns*, and that the prefixes announced are not bogons (*i.e.*, unallocated IP prefixes). The observance of B_p is important as it ensures that there are no prefix hijacking — announcement of prefixes not owned by an AS — being mounted, among other things.
- *Behavior Set 2 (B_o)*: Requires that no AS in the *AS_PATH* violates the *valley free routing* requirement, and that the AS numbers on the *AS_PATH* vector are legal[3]. Detecting valley routes are importance because VFR has been shown to be a sufficient condition to ensure that BGP converges [14], and is thus essential for the sustenance of BGP over the long run.
- *Behavior Set 3 (B_l)*: Requires that AS-links in the *AS_PATH* last for a substantial duration of time. Detecting such unstable AS-link bindings is important since ASes which chose a path with one or more unstable AS-links faces a high probability that the path will not be adhered to during data communication. Such a path choice may also have unintended consequences such as increased latency. Moreover, unstable AS-links can also indicate the potential of them being artificially introduced (*i.e.,* spoofed) [21], [22].

[3]A 16-bit AS number is *illegal* if its value is in the range of 64496-64511 which is reserved for use in documentation and sample code, 64512-65534 designated for private use) or 65535 which is reserved [6].

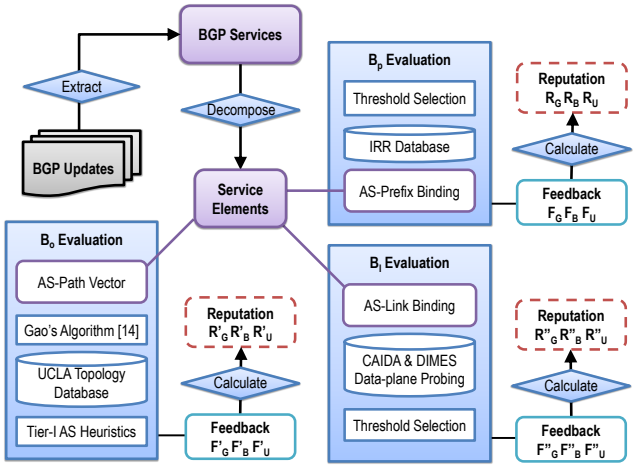

Fig. 3. **ASCRED+ Service Analysis Workflow**

Behavior Evaluation Results from Jan. 1, 2010 and Jun. 30, 2010

(For each day, the analysis considers: I. BGP updates from the past *60* days for B_p and B_l, and II. AS relationship annotated topologies of the past 24 hours for B_o)

Analysis of B_p*

Property	Value
Avg. # of ASPB** Observed	421704.1
Avg. # of ASPB Classified as Ugly	6955.61***
Avg. # of ASPB Classified as Bad	29256.6
Avg. # unique AS Observed	35448.2
Avg. # of AS announcing Bad ASPB	1132.8
Avg. # of AS announcing Ugly ASPB	605.5
Avg. # of AS *exclusively* announcing Bad ASPB	17.8
Avg. # of AS *exclusively* announcing Ugly ASPB	54.3

* No bogons were observed during the behavior analysis periods
** ASPB: AS-prefix bindings
*** The actual value was a higher than usual 16882.1 due to the Internet scale prefix hijacking mounted by AS23724 on April 8th, 2010.

Analysis of B_o

Property	Value
Avg. # of Paths	661395.7
Avg. # of Valley Routes	3447.8
Avg. # of AS creating Valley Routes	89.2
Avg. # of BGP services containing illegal AS number	44.1

Analysis of B_l

Property	Value
Avg. # of AS-links Observed	94754.2
Avg. # of Stable AS-links	91143.6
Avg. # of Unstable AS-links	3610.6
Avg. # of unique AS Observed	35667.2
Avg. # of AS announcing Unstable AS-links	1945.7
Avg. # of AS exclusively announcing Unstable AS-links	67.4

Fig. 4. **AS Behavior Evaluation Statistics**

It can be seen that there is a one-to-one mapping between the service elements and the behavior sets. Therefore, evaluating a service involves evaluating whether AS-prefix binding, AS-path vector, and AS-link binding service elements satisfy B_p, B_o, and B_l, respectively. But before we delve into the details of evaluation, we provide a overview of our feedback mechanism which is essential for eventual reputation computation, and forms an integral part of the evaluation process.

C. Feedback Mechanism

Evaluation of a BGP service element provides one of three mutually exclusive feedbacks about the ASes. The feedback mechanism utilized in this regard is similar to the one used in [13] and can have one of three values: (1) **Good:** This feedback is given on the providers which satisfies the requirements of the appropriate behavior set; (2) **Bad:** This feedback is given on the providers which do *not* satisfy the requirements of the appropriate behavior set, however, they do not affect the correct operation of BGP; and (3) **Ugly:** this feedback is given on the providers which does not satisfy the requirements of the appropriate behavior set, and can potentially subvert intended BGP operation.

In the rest of the paper, we use the term *GBU feedbacks* to refer to our feedback types. When the service element implemented by the provider(s) receives a *Good* feedback, it is referred to as *good behavior*. Conversely, a *Bad* or an *Ugly* feedback for a service element is referred to as the demonstration of *poor behavior*. In general, there exists a 3×3 feedback matrix for every provider AS_a, at the observer, of the form:

$$F_a = \begin{pmatrix} F_G & F_B & F_U \\ F'_G & F'_B & F'_U \\ F''_G & F''_B & F''_U \end{pmatrix}$$

where the element $F_a(i,j)$ stores the details of the BGP service which AS a provided when evaluated with respect to the Behavior Set i, where $1 \leq i \leq 3$. Finally, as the feedbacks are generated locally at the observer AS, we do not have to consider the case of potentially dishonest feedback affecting our reputation computation outcome.

IV. BGP Service Evaluation and Feedback

In this section, we describe the metrics used in the behavior evaluation of service elements. These metrics allow the feedback matrix to be populated, which will subsequently be used to compute reputation. As mentioned earlier, the behavior evaluation considers BGP services received during a 60 day learning window and produces feedback on the providers, on the 61st day. Figure 3 illustrates the work-flow of the evaluation process discussed in this section. The boxes with dashed outlines illustrate the output produced at the end of analyzing a service based on each of three behavior sets.

A. Evaluation of Service using B_p

The behavior set B_p is responsible for checking the presence of bogons and if an AS is announcing a prefix it owns. Evaluating bongons involves a static check, while checking for ownership is done indirectly based on the stability of the AS-prefix binding. The principle idea of evaluating stability comes from the observation that legitimate AS-prefix bindings last long periods of time [12], [19].

1) Method: Determining whether an AS-prefix binding (AS_0, p) exhibits B_p was the main thrust of our previous work [13]. Therefore, in the rest of the section we briefly summarize the metrics used and evaluation described therein. The evaluation is a three step process: (1) *Stability Analysis:* For each (AS_0, p) observed during the learning window, we compute two metrics: prevalence and persistence. *Prevalence* (Ps), is the total percentage of time an AS-prefix binding is active within the learning window. *Persistence (Pr)*, on the other hand, is the average duration of time an AS-prefix binding is active at one time, within the learning window [13]; (2) *Providing Feedback:* The value of the Ps and Pr are compared against a set of thresholds T_{pr} (1% of the learning window) and T_{Ps} (10 hours)[4] and feedback provided. Table

[4]Both the thresholds have been established empirically, based on lowest false positive rates when compared with Internet Route Registries (IRR) [13].

TABLE I
Feedback for Behavior Evaluation based on B_p

Prevalence	Persistence	Classification	Feedback Type
high	high	*Good*	F_G
high	low	*Bad* (Vacillation)	F_B
low	high	*Good*	F_G
low	low	*Ugly* (Hijack)	F_U

TABLE II
Feedback for Behavior Evaluation based on B_l

Prevalence	Persistence	Classification	Feedback Type
high	high	*Good*	F''_G
high	low	*Good*	F''_G
low	high	*Good*	F''_G
low	low	*Ugly* (Unstable)	F''_U

I shows the feedback matrix element updated for different Pr and Ps values; and (3) *Detecting Bogons:* (AS_0, p) is also checked for the presence of bogons, and their discovery results in F_U being updated in the feedback matrix of the AS announcing them.

2) Discussion: The case of Pr being high and Ps being low demonstrates a *vacillating* nature of the AS-prefix binding. Detailed analysis of such bindings demonstrate that they are usually legitimate [13]. However, the AS-prefix binding service element itself vacillates between being active and inactive at a rate which is not conducive for data communication [13]. Consequently, we classify such vacillating behavior as *Bad* because the ASes execute BGP's functionality correctly but fail to meet the requirement of the behavior set. As for bogons, we believe their announcement subverts the operation of BGP and we therefore deem them *Ugly*.

3) Results: The results of the evaluation, based on B_p, are summarized in Figure 4. On average of 421704.1 AS-prefix bindings were observed every learning window, out of which an average of 4.0% were found to be *Ugly*[5] involving 1.7% of all the ASes. Similarly, about 6.9% of AS-prefix bindings were classified as *Bad*, involving 3.1% of all the ASes. The number of ASes displaying exclusively poor behaviors is lower still. Finally, we observed zero occurrence of AS-prefix bindings with bogon prefixes during any of the learning windows. We believe this is because bogons are invariably filtered out by ASes, which encounter them, and are never forwarded. *The results demonstrate that a relatively large number of ASes (3-5%) are involved in announcing vacillating and hijacked prefixes.*

B. Evaluation of Service using B_o

Behavior set B_o checks for the exhibition of valley routes and the introduction of illegal AS numbers in the *AS_PATH*.

1) Method: To evaluate an AS-path vector based on B_o is a five step process: (1) *Generating AS Relationship Map:* Based on the BGP updates received over the past 24 hours, we generate an AS-level Internet topology and infer relationships between the ASes using Gao's algorithm [14]; (2) *Merging Topologies:* We then download that day's annotated topology from the UCLA's Internet topology site [25] and merge it with the topology we inferred previously; (3) *Introduce Peers:* We obtain the list of all tier-1 ASes from [2]. All links between tier-1 ASes are re-labeled peer-to-peer (P2P), and links between tier-1 AS and lower-tier AS are re-labeled Pv2C where the tier-1 AS is the provider (Pv); (4) *Providing Feedback:* Once the merged annotated topology has been created, the AS-path service element of all the services announced that day is evaluated for the existence of ASes which might violate VFR. If such an AS is found, then its F'_B entry in its feedback matrix is updated; and (5) *Identifying Private ASes:* The *AS_PATH* is finally examined for private AS numbers. This is done based on a static check. The first legal AS, after the set of private ones is blamed and its F'_B incremented. For all the other ASes in *AS_PATH* the F_B entry is incremented.

2) Discussion: The use of two well-known AS topology relationship inference techniques increases the confidence on our own relationship labeling. We consider the violation of VFR to be *Bad* because, though not good in the long run, it does not necessarily affect the operation of BGP in providing knowledge about routes to prefixes. In the case of private ASes, the first non-private AS after a set of private ASes is blamed because such leaking of private numbers usually happens when an AS forgets to filter out local AS numbers before forwarding the update for the service.

3) Results: The results of the evaluation, based on B_o, are summarized in Figure 4. We found that, an average of 661395.7 paths were observed per day. Out of these, 0.5% paths were found to violate VFR per day. Finally, an average of 89.2 providers out of over 35K were were seen violating VFR per day. On an average, we found only about 44.1 ASes involved in allowing private AS numbers in the *AS_PATH*, per day, during the six months of behavior analysis with respect to B_o. *In summary, the non-adherence to B_o, especially valley routes are prevalent and a recurring event in the day to day operation of BGP, but on a much smaller scale.*

C. Evaluation of Service using B_l

This behavior set is responsible for checking if an AS-link binding (AS_i, AS_j) in the *AS_PATH* is stable.

1) Method: Computing the stability of an AS-link binding (AS_i, AS_j) in the *AS_PATH* follows a similar approach to AS-prefix binding stability evaluation, and uses the *prevalence* and *persistence* metrics. The evaluation and feedback with respect to B_l is done in three steps: (1) *Identifying AS-link Bindings:* This generates a set L of all the AS-link bindings, decomposed from the services observed during the learning window; (2) *Computing Stability Metrics:* This step computes the prevalence and persistence for each of the AS-link in set L; (3) *Providing Feedback:* The computed Pr and Ps values are then compared with a threshold Tl_{Pr} and Tl_{Ps} and a feedback is provided. Table II shows the feedback matrix element updated for different Pr and Ps values.

2) Discussion: The reason we classify unstable AS-links as *Ugly* is because it is possible that poor AS-link stability is due to an attempted link spoofing which could subvert the intended BGP operation. It should be noted that it is difficult to get conclusive proof for the spoofing given a lack of ground truth,

[5]This number is unusually high due to the Internet-scale prefix hijacking attempt on April 8th, 2010 by AS_{23724}.

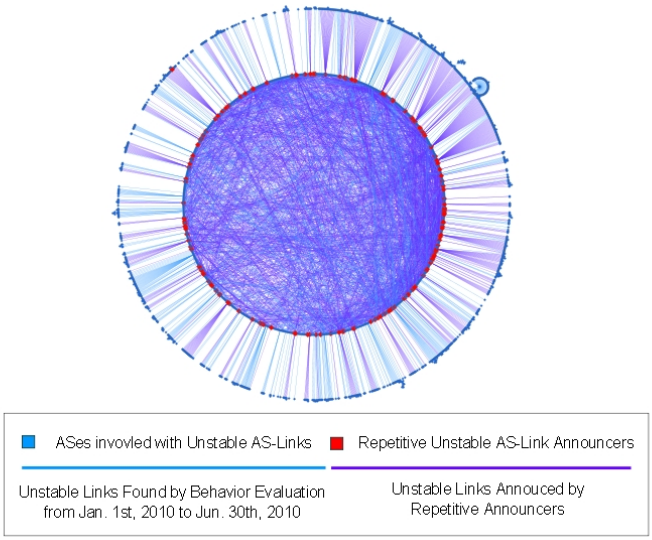

Fig. 5. **Visualization of Unstable AS-Links and the Provider ASes Involved**

though we find an interesting result which strengthens the case for their occurrence (see Section V-D). We therefore argue that the potential of spoofing merits a punitive feedback for ASes involved in unstable AS-links. The value of thresholds Tl_{Pr} and Tl_{Ps} are set to 1% of the learning window and one hour, respectively. These values are established empirically based on comparison with a set of AS-links D. The set D is obtained from data-plane probing database provided by the CAIDA [3] and DIMES [4] projects. The thresholds are the values below which, all the AS-links in the set L, with the particular Pr and Ps, have the smallest intersection with the set D. Data-plane probing is used because if a AS-link is ephemeral, it has a low probability of being found in data-plane probing. Further, it is the only form of ground-truth available, as can reliably identify AS-links which are stable enough to allow data traffic to pass through them [22].

3) Results: The results of analysis, based on B_l, reveals that an average of 95640.4 AS-links observed during the each of the learning windows. Out of these over 96.1% AS-links received *Good* feedback. From the perspective of the ASes, on average of 35667 ASes were seen every learning window, out of which 5.4% ASes announced unstable AS-links at least once. Only about an average of 0.18% of ASes announced purely unstable AS-links (see Figure 4). Figure 5 visualizes 4625 unstable AS-links seen each month over the course of the experiment involving 2305 ASes. The dots are the ASes and the lines between them are unstable AS-links. The red dots represent the 149 ASes which have established unstable AS-links at least once every month, during the course of our experiment. *In summary, these results demonstrate that unstable AS-links are a repetitive phenomena, affecting a substantial number (5-6%) of ASes.*

D. Overall Trend

In this section, we present the bigger picture that has emerged from the behavior analysis during the six months of our experiment. Figure 6 presents the extent of poor behavior seen every day between Jan. 1, 2010 and Jun. 30, 2010. The trend graphs provide an overview of the extent the poor behaviors of ASes afflicting inter-domain routing and how they have evolved over time. The numbers are raw-values and include all AS-prefix bindings, *AS_PATHS* and AS-links, decomposed from the observed BGP services. It can be seen that the number of RI with vacillating prefixes (over 10K) is an order of magnitude larger than all others (100-1K). Finally, *the problem of poor behavior is consistently present over the course of the six months and is largely stable in its intensity, with occasional spikes*. The large-scale hijacking events by China Telecom (AS_{23724}) and Cox Communication Inc's (AS_{6298}) on Apr. 8, 2010 [7] and Jun. 30th, 2010 [9], respectively, are two of the documented instances of such spikes. In the case of unstable AS-links (green line in the top graph), the number of spikes are more frequent toward the latter half of our experiment period. This indicates large instances of poor stability AS-links being observed. Due to the lack of data regarding such events, it is difficult to associate them with concrete events as in the case of prefix hijacking. However, these results do indicate the importance of monitoring AS-link stability with the same diligence as prefix hijacking. We believe AS-TRUST provides the first step in this regard.

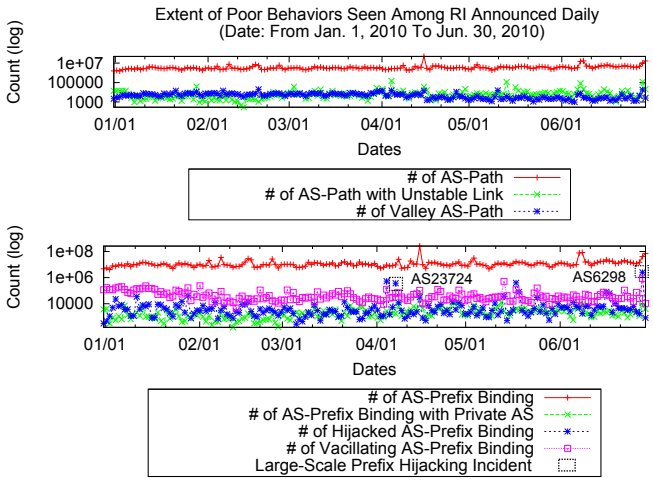

Fig. 6. **Extent of Poor Behaviors in the Internet**

V. Reputation Computation

At the end of behavior evaluation, we have a feedback matrix for each AS. This will now be used to compute reputation. The reputation will allow the observer AS to know *what is the probability of observing a service element in a BGP service provided by an AS, being Good (or Bad or Ugly)*. Given the service elements are orthogonal to each other, the reputation for an AS is computed as a 3×3 matrix R akin to the matrix F,

$$R = \begin{pmatrix} R_G & R_B & R_U \\ R'_G & R'_B & R'_U \\ R''_G & R''_B & R''_U \end{pmatrix}$$

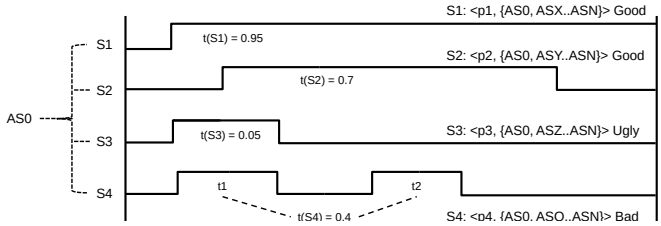

Fig. 7. **Time-line for Computing Refined Reputation**

Here, the rows correspond to the reputation of the AS with respect to an AS-prefix binding, AS-path, and AS-link binding service elements, respectively. Each row is a vector representing the probability for an AS being *Good*, *Bad* and *Ugly*. We do not arrive at a single number for reputation here, as it would not be able to describe the behavior of an AS with the same level of detail. In this section, we first present a reputation model which has probabilistic semantics. We then present our base reputation function and suggest improvements to it.

A. Reputation Model

We need reputation to provide a probabilistic view of an AS' behavior. To arrive at the reputation we use Bayesian statistics. Intuitively, if we have events with k possible outcomes, as is the case with each of our behavior evaluation which produces one of three outcomes, we can compute the posteriori probabilities, *i.e.,* the probabilities of observing these behaviors, using the Dirichlet distribution if we assume the prior to be a Dirichlet distribution as well [8]. Then, the reputation is the expected value of a posteriori probability distribution. The model presented here is a generalization of the one in [18].

Formally, suppose there is a process which can produce k outcomes $(u_1, \ldots, u_k)$. Let, $\vec{x} = [x_1, x_2, ..x_k]$ and $\vec{p} = [p_1, p_2, ..p_k]$ be two vectors where x_i and p_i represent the number of times and the probability of the occurrence of an outcome u_i, respectively. Now, Bayes' theorem states that $Pr(\vec{p}|\vec{x}) \propto Pr(\vec{x}|\vec{p})Pr(\vec{p})$. In this case, the likelihood function $Pr(\vec{x}|\vec{p})$ is nothing but a multinomial distribution. The prior $Pr(\vec{p})$ can be many different distributions; however, we choose the Dirichlet distribution as it is the conjugate prior of the multinomial distribution. That is, treating the prior as Dirichlet results in a posterior also being a Dirichlet. The Dirichlet distribution is a family of continuous, multi-variate probability distributions. It is parameterized by a vector of reals called $\vec{\alpha}$, and is given by: $Dir(\vec{p}|\vec{\alpha}) = \prod_{i=1}^{k} p_i^{\alpha_i - 1}/Z(\alpha)$. Here, $p_1, \ldots, p_k \geq 0$, $\sum_{i=1}^{k} p_i = 1$, $\alpha_1 \ldots \alpha_k > 0$ and $Z(\alpha) = \prod_{i=1}^{k} \Gamma(\alpha_i)/\Gamma(\sum_{i=1}^{k} \alpha_i)$ is the normalizing constant, which is the multinomial beta function.

The Dirichlet prior can be thought of as past experiences with random process providing us with the probability of observing $\alpha_i - 1$ number of outcome u_i. Consequently, the posterior probability distribution is a Dirichlet where the value of the hyperparameter $\alpha_i = x_i + 1$, can be thought of as a sum of the counts of each type of outcomes seen in the current experiments and in the past. As the vector $\vec{p}$ is a vector of probabilities, for a given $\vec{p}$, the probability density function $Dir(\vec{p}|\vec{\alpha})$ is a second order probability, which represents the distribution of the first-order vector taking specific values. It is only meaningful if we compute the expectation of the probability values in the vector $\vec{p}$, and is given by: $E(p_i) = \alpha_i / \sum_{i=1}^{k} \alpha_i$. This expectation forms the basis of our reputation calculation as it can be interpreted as the probability of an AS providing valid RI.

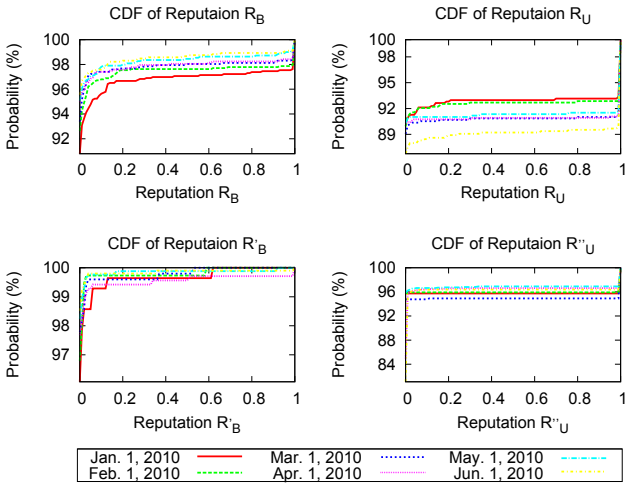

Fig. 8. **CDF of ASes with Poor Reputation Values**

B. Base Reputation

In a BGP service identified by the tuple $\{p, AS\_PATH = [AS_0, AS_1, \ldots, AS_N]\}$, AS_0 is evaluated for B_p and B_l, while AS_1 to AS_N are evaluated for B_o and B_l, resulting in one feedback for each evaluation. We view each feedback as a separate trial which produces one of the three outcomes *Good*, *Bad*, *Ugly*. Note that, all subsequent discussion assumes reputation is being computed for AS for the computation of reputation corresponding to B_p. Computing reputations for B_o and B_l is similar and not presented for brevity.

After M trials, let $|F_X|$ be the count of BGP services contained in F_X, where $X \in \{G, B, U\}$, and $|F_G| + |F_B| + |F_U| = M$. Let $\vec{p} = (p_g, p_b, p_u)$ be the probabilities of observing G, B and U, respectively, then we compute the reputation vector of AS as $R = [R_G R_B R_U]$. Using our reputation model, by setting $\alpha_1 = |F_G| + 1$, $\alpha_2 = |F_B| + 1$ and $\alpha_3 = |F_U| + 1$. Assuming that the prior probabilities are uniformly distributed, we have: $R_G = E(Pr(p_g)) = \alpha_1/\alpha_n$, $R_B = E(Pr(p_b)) = \alpha_2/\alpha_n$, and $R_U = E(Pr(p_u)) = \alpha_3/\alpha_n$, where $\alpha_n = \sum_{k=1}^{3} \alpha_k$.

C. Reputation Refinement

An important property of the Internet is that poor behaviors usually have very short duration [13][23]. Therefore, it is more interesting to use reputation to determine *the expected probability of a service element in service provided by an active AS is Good*. In this regard, we refine the reputation calculation by modifying the weight of the entries in the feedback set to a value proportional to the time the service remained active within a learning window. In other words, we re-define $|F_X| = \sum_{i=1}^{k} t(s_i)$, where k is the total number

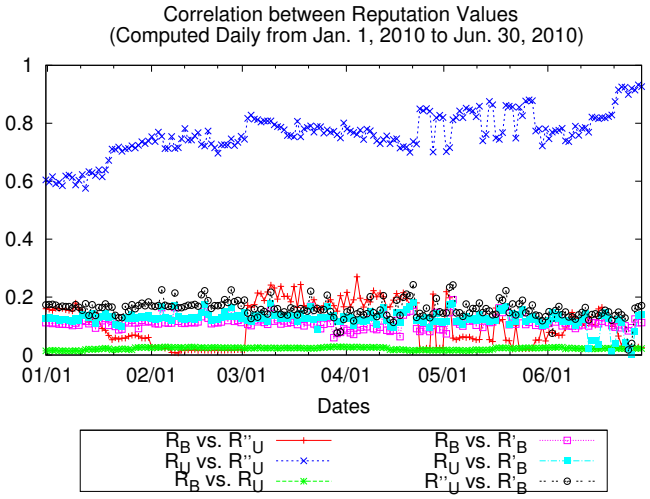

Fig. 9. **Correlation between Various Reputation Values**

of BGP services in F_X and $t(s_i)$ is the percentage of time a BGP service s_i in F_X is active within the learning window. As *Good* behaviors last a long duration compared to poor ones, at any given time within our learning window, the probability of an active service having *Good* service elements will be much higher than probability of an active service with *Bad* or *Ugly* service elements. We illustrate the difference between the base and refined reputations using an example.

Example: Let an $AS0$ provide four services $S1$, $S2$, $S3$, $S4$ where $t(S1) = 0.95$, $t(S2) = 0.70$, $t(S3) = 0.05$ and $t(S4) = 0.40$, respectively (see Figure 7). After evaluating the services based on B_p, let $S1$ and $S2$ be classified as *Good*, $S3$ as *Bad* and $S4$ as *Ugly*. As each service element in a service is independent of the others, $t(S1) + t(S2)$ may be greater than 1. In order to compute the base reputation in this scenario, we have: $|F_G| = 2$, $|F_B| = 1$, $|F_U| = 1$, therefore $\alpha_1 = 3$, $\alpha_2 = 2$, $\alpha_3 = 2$, giving $R_G = 0.42$, $R_B = R_U = 0.29$. Applying refinement, we get: $|F_G| = 0.95 + 0.70 = 1.65$, $|F_B| = 0.05$, $|F_U| = 0.4$, resulting in $R_G = 0.51$, $R_B = 0.21$, and $R_U = 0.28$.

It can be seen that the refinement redistributes the probability of poor behavior in a manner proportional to the duration the service was active. We can compute reputation values corresponding to other two behavior sets in a similar manner.

D. Reputation Analysis

In this section we analyze the reputation of ASes, generated over a period of six months from Jan. 1, 2010 - Jun. 30, 2010. As for each AS we compute six reputation values, in this section we will be focusing on presenting only the results of the reputation due to poor behaviors. The reputation due to good behavior is a complement of the results and can be easily extracted from these.

Figure 8 shows the CDF of reputations of ASes which have at least one *Bad* or *Ugly* feedback for B_p, B_o and B_l. As reputation of ASes is computed every day, we illustrate the CDFs for a sampling of six days during the six month period. These graphs demonstrate three important points: (1) among those which do demonstrate poor behaviors, over 85% of them

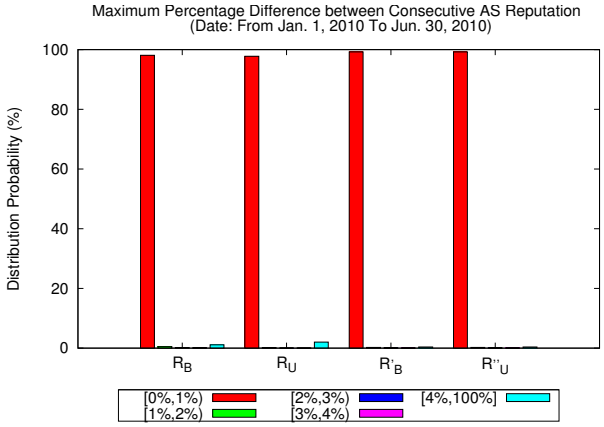

Fig. 10. **Stability of Reputations**

do so infrequently (in the case of B_o this number is over 99%), (2) about 2-8% of the ASes which demonstrate poor behavior do so exclusively (*i.e.,* the spike near the very end of the distribution) for B_p and B_l, and (3) over 99% of the ASes have a reputation close to zero for B_o, which means they are rarely involved in valley routes. The result demonstrates the *sensitivity of the reputation metric* as it is able to capture even those ASes which seldom do wrong.

The availability of a quantitative value for different aspects of AS trustworthiness in the form of reputation, allows us to mine emergent AS behavior trends which were heretofore difficult to identify. In this regard, we compute the correlation between different elements of the reputation matrix. Figure 9 illustrates the results. Interestingly, we find R_U (*i.e.,* demonstrate prefix hijacking) and R''_U (*i.e.,* are part of unstable AS-links) have a very high correlation, over the six months of our experiments. That is, ASes which have high R_U also have high R''_U . This is very intriguing as it increases the potential for low stability AS-links to be malicious (spoofed links). None of the other reputation values are strongly correlated. Particularly, the lack of correlation between R_B and R_U indicates that ASes involved in vacillating AS-prefix bindings are independent of those which commit prefix hijacking. This provides another anecdotal evidence to our earlier observation that vacillating AS-prefix bindings are largely legitimate.

Finally, we evaluate how much the reputation values change over time. In this regard, we take the daily reputation values of ASes from Jan. 1, 2010 to Jun. 30, 2010 and computed the maximum percentage difference between two consecutive values. Figure 10 shows the results in terms of R_B, R_U, R'_B and R''_U. In all four cases we find that for over 98% of the ASes, the maximum variation between two adjacent reputation values was within within 0-1%. This means that *the reputation trend remains stable and varies very smoothly over time*. As R_G, R'_G and R''_G values are complementary to ones shown here, the same stability trend holds for them as well. This limited variation indicates repetitiveness of behavior.

VI. Results Summary

The following are the principal takeaways from these results: (1) **Large Trust Violation:** Relatively large number

of ASes deviate from operational trust assumption of BGP; (2) **Emergence:** There is a strong correlation between reputations of ASes which hijack prefixes and those involved in unstable AS-links; (3) **Sensitivity:** The reputation metric is sensitive enough to capture even the smallest deviations from the expected behavior; (4) **Repetitiveness:** Both good and poor AS behavior patterns are stable over the course of our experiments; (5) **Consistent Presence:** The number of incidents of poor behavior is consistently present over the course of our experiments.

VII. Related Work

Little work has been done with respect to characterizing AS behaviors. Most of the work has focused on detecting prefix hijacking using control-plane [19], [20], [23] or data-plane probing [17], [26], [27]. In [24] the authors use the notion of reputation for accepting or rejecting updates. The idea is to form a trusted overlay network over the existing AS topology. Once such an overlay is set up, a node which wants to determine the accuracy of an update, with respect to prefix hijacking and AS path spoofing, can simply query its neighbors in the overlay network. Similarly, in [16], the authors present a reputation system for ASes, with a focus on preventing propagation of bogus routing information. However, their mechanism also depends on computing reputation based on an alliance of ASes. As AS-TRUST does not depend on feedback from other ASes to compute reputation, it does not have to compensate for any biased feedback.

In our earlier work, titled AS-CRED, [13], reputations are computed for ASes based on their tendencies to hijack or announce short-lived prefixes. AS-TRUST, on the other hand, considers many more aspects in reputation computation including valley free routing and AS-link stability. However, the principal difference between the two is in the semantics of the reputation value. For example, the *Ugly* reputation value indicated how many prefix hijacking an AS performed. This value is indifferent to the number of stable AS-prefix bindings the AS had. Reputation in this case was simply a statement of how Ugly an AS is and can be compared with the Ugly reputations of other ASes to see how they fare in comparison. With AS-TRUST, each row in the reputation matrix is a normalized value and has a probabilistic meaning. Therefore, Bad or Ugly behaviors of an AS cannot be seen independently of its Good behaviors. The reputation values of AS-TRUST thus provide a complimentary view of the AS behavior.

VIII. Conclusions

In this paper we presented AS-TRUST, a reputation-based scheme for characterizing trustworthiness of an AS with respect to disseminating valid reachability information. Reputation is computed by evaluating past RI announced by each observable AS in the Internet for the exhibition of specific behaviors. The evaluation utilizes well-defined properties for this purpose including the presence of: stable AS-prefix binding, stable AS-links and valley free *AS_PATH*. It then classifies the resulting observations into multiple types of feedback. The feedback values are input into a reputation function which provides a probabilistic view of trust. In the future, we plan to build a reputation-based alert system which predicts the validity of any new RI received.

References

[1] A Border Gateway Protocol 4 (BGP-4) RFC. http://www.rfc-editor.org/rfc/rfc4271.txt.

[2] BGP Routing Leak Detection System Routing Leak Detection System. http://puck.nether.net/bgp/leakinfo.cgi.

[3] Macroscopic Topology Measurements . http://www.caida.org/projects/macroscopic/.

[4] The DIMES project. http://www.netdimes.org/new/.

[5] 7007 Explanation and Apology. http://www.merit.edu/mail.archives/nanog/1997-04/msg00444.html/.

[6] Autonomous System (AS) Numbers. http://www.iana.org/assignments/as-numbers/.

[7] Chinese ISP hijacks the Internet. http://bgpmon.net/blog/?p=282.

[8] Dirichlet distribution. http://www.cis.hut.fi/ahonkela/dippa/node95.html.

[9] ListWare: BGP Update Report. http://www.listware.net/201007/nanog/6379-bgp-update-report.html.

[10] Pakistan hijacks YouTube. http://www.renesys.com/blog/2008/02/pakistan_hijacks_youtube_1.shtml/.

[11] RouteViews. http://www.routeviews.org/.

[12] P. Boothe, J. Hiebert, and R. Bush. Short-lived prefix hijacking on the Internet. In *In Proc. of the NANOG 36*, February 2006.

[13] J. Chang, K. Venkatasubramanian, A. G. West, S. Kannan, I. Lee, B. Loo, and O. Sokolsky. AS-CRED: Reputation service for trustworthy inter-domain routing. In *University of Pennsylvania Technical Report, MS-CIS-10-17*, April 2010.

[14] L. Gao. On inferring autonomous system relationships in the Internet. *IEEE/ACM Trans. Netw.*, 9(6):733–745, 2001.

[15] T. Grandison and M. Sloman. A survey of trust in Internet applications. *IEEE Communications Surveys and Tutorials*, 3(4), August 2000.

[16] N. Hu, P. Zhu, and P. Zou. Reputation mechanism for inter-domain routing security management. In *In Proc. of the 9th International Conference on Computer and Information Technology*, pages 98–103, October 2009.

[17] X. Hu and Z. M. Mao. Accurate real-time identification of IP prefix hijacking. In *SP '07: Proceedings of the 2007 IEEE Symposium on Security and Privacy*, pages 3–17, Washington, DC, USA, 2007. IEEE Computer Society.

[18] A. Josang and R. Ismail. The beta reputation system. In *In Proceedings of the 15th Bled Electronic Commerce Conference*, 2002.

[19] J. Karlin, S. Forrest, and J. Rexford. Autonomous security for autonomous systems. *Comput. Netw.*, 52(15):2908–2923, 2008.

[20] M. Lad, D. Massey, D. Pei, Y. Wu, B. Zhang, and L. Zhang. PHAS: a prefix hijack alert system. In *In Proc. of the 15th conference on USENIX Security Symposium*, Berkeley, CA, USA, 2006. USENIX Association.

[21] R. Mahajan, D. Wetherall, and T. Anderson. Understanding BGP misconfiguration. In *In Proc. of the 2002 conference on Applications, technologies, architectures, and protocols for computer communications*, pages 3–16, 2002.

[22] M. Nicholes and B. Mukherjee. A survey of security techniques for the Border Gateway Protocol (BGP). *IEEE Communications Surveys and Tutorials*, 11(1), First Quarter 2009.

[23] J. Qiu, L. Gao, S. Ranjan, and A. Nucci. Detecting bogus BGP route information: Going beyond prefix hijacking. In *Security and Privacy in Communications Networks and the Workshops, 2007. SecureComm 2007. Third International Conference on*, pages 381–390, Sept. 2007.

[24] H. Yu, J. Rexford, and E. Felten. A distributed reputation approach to cooperative Internet routing protection. In *Secure Network Protocols, 2005. (NPSec). 1st IEEE ICNP Workshop on*, pages 73–78, Nov. 2005.

[25] B. Zhang, R. Liu, D. Massey, and L. Zhang. Collecting the Internet AS-level topology. *SIGCOMM Comput. Commun. Rev.*, 35(1):53–61, 2005.

[26] Z. Zhang, Y. Zhang, Y. C. Hu, Z. M. Mao, and R. Bush. iSPY: detecting IP prefix hijacking on my own. *SIGCOMM Comput. Commun. Rev.*, 38(4):327–338, 2008.

[27] C. Zheng, L. Ji, D. Pei, J. Wang, and P. Francis. A light-weight distributed scheme for detecting IP prefix hijacks in real-time. *SIGCOMM Comput. Commun. Rev.*, 37(4):277–288, 2007.